

# Oracle® Cloud

## Using Oracle Cloud Infrastructure Compute Classic



E63022-44  
May 2020



Oracle Cloud Using Oracle Cloud Infrastructure Compute Classic,

E63022-44

Copyright © 2015, 2020, Oracle and/or its affiliates.

Primary Author: Kumar Dhanagopal, Anamika Mukherjee, Sylaja Kannan, Melwyn Paul

Contributing Authors: Jeffrey Welsch, Sudipa Bhattacharya, Gururaj BS, Mirek Chocholous, Jitendra Chouhan, Bryn Divey, Vidya Gopal, Andrei Isaev, Diby Malakar, Stephen Mayer, Tim McDuff, Irina Mok, Raja Mukherjee, Octave Orgeron, Kiran Palan, Vimal Patel, Jeffrey Pleau, Gary Resnick, Modin Shaik, Vivek Sedhumadhavan, Costa Siourbas, Sundar Srinivasan, Paul Wickstrom, Chen Xie, Xiaofeng Yang, Vincent Yee

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

# Contents

## Preface

---

Audience	xvi
Documentation Accessibility	xvi
Related Resources	xvi
Conventions	xvi

## 1 Getting Started with Oracle Cloud Infrastructure Compute Classic

---

About Compute Classic	1-1
Before You Begin with Compute Classic	1-5
How to Begin with Compute Classic Subscriptions	1-5
Compute Classic Terminology	1-5
Accessing Compute Classic Using the Web Console	1-9
Accessing Compute Classic Using REST API	1-10
Accessing Compute Classic Using the Command Line Interface	1-10
About Compute Classic Roles	1-10
About Compute Classic Sites	1-11
Workflow for Using Compute Classic	1-11

## 2 Enabling Secure Access to Instances Using SSH

---

About SSH Keys	2-1
Generating an SSH Key Pair	2-3
Adding an SSH Public Key	2-4
Attaching an SSH Public Key to an Instance	2-6
Viewing an SSH Public Key	2-6
Updating an SSH Public Key	2-7
Disabling an SSH Public Key	2-9
Enabling an SSH Public Key	2-10
Deleting an SSH Public Key	2-11

## 3 Creating Instances

---

About Instances	3-1
About Shapes	3-2
Instance Life Cycle	3-6
Workflow for Creating Your First Instance	3-6
Workflow for Creating Your First Oracle Linux Instance	3-7
Workflow for Creating Your First Oracle Solaris Instance	3-7
Workflow for Creating Your First Windows Instance	3-8
Selecting a Method of Creating Instances	3-9
Creating an Instance Using QuickStarts	3-10
Creating an Instance from the Instances Page	3-13
Creating an Instance Using a Private Image	3-26
Creating Instances Using Orchestrations	3-37
Creating an Instance Using a Blank Orchestration v2	3-38
Creating Instances Using Launch Plans	3-44
Creating an Instance Using Visual Object Editor	3-51
Logging In to an Instance	3-54
Cloning an Instance by Using Instance Snapshots	3-55
About Instance Snapshots	3-55
Creating an Instance Snapshot	3-56
Registering the Image Generated by an Instance Snapshot	3-57
Creating an Instance from an Instance Snapshot	3-58
Deleting an Instance Snapshot	3-58

## 4 Managing Instances

---

Viewing Information About an Instance	4-1
Listing Instances	4-1
Monitoring Instances	4-2
Viewing Instance Metrics	4-3
Enabling Instance Metrics Collection	4-4
Viewing the Boot Log of an Instance	4-7
Generating a Screen Capture of an Instance	4-8
Updating Packages on an Oracle Solaris Instance	4-9
Updating an Instance	4-12
Attaching a Public IP Address to an Instance on the Shared Network	4-12
Attaching a Storage Volume to an Instance	4-14
Detaching a Storage Volume from an Instance	4-15
Adding an Instance to a Security List	4-16
Removing an Instance from a Security List	4-17
Resizing an Instance	4-18

Managing Instance Lifecycle Operations	4-20
Rebooting an Instance	4-21
Shutting Down and Restarting an Instance	4-21
Deleting an Instance	4-23
Deleting and Re-creating an Instance	4-26
Retrieving Instance Metadata	4-28
About Instance Metadata	4-29
Retrieving Predefined Instance Metadata	4-30
Retrieving User-Defined Instance Attributes	4-32
Sample Scenario for Specifying and Using Instance Attributes	4-32

## 5 Managing Resources Using the Visual Object Editor

---

## 6 Managing Resources Using Orchestrations v1

---

About Orchestrations v1	6-1
Orchestrations v1 Templates	6-8
Workflow for Creating Instances Using Orchestrations v1	6-21
Building Your First Orchestration v1	6-21
Attributes in Orchestrations v1	6-25
Top-Level Attributes in Orchestrations v1	6-26
Object Plan Attributes	6-27
Orchestration v1 Attributes Specific to Each Object Type	6-28
Orchestration v1 Attributes for integrations/osscontainer	6-29
Orchestration v1 Attributes for ip/reservation	6-30
Orchestration v1 Attributes for launchplan	6-31
Orchestration v1 Attributes for instances	6-32
Orchestration v1 Attributes for network/v1/acl	6-42
Orchestration v1 Attributes for network/v1/ipaddressprefixset	6-42
Orchestration v1 Attributes for network/v1/ipassociation	6-43
Orchestration v1 Attributes for network/v1/ipnetwork	6-44
Orchestration v1 Attributes for network/v1/ipnetworkexchange	6-45
Orchestration v1 Attributes for network/v1/ipreservation	6-46
Orchestration v1 Attributes for network/v1/route	6-47
Orchestration v1 Attributes for network/v1/secprotocol	6-48
Orchestration v1 Attributes for network/v1/secrule	6-49
Orchestration v1 Attributes for network/v1/vnicset	6-50
Orchestration v1 Attributes for orchestration	6-51
Orchestration v1 Attributes for secapplication	6-52
Orchestration v1 Attributes for seciplist	6-53

Orchestration v1 Attributes for seclist	6-54
Orchestration v1 Attributes for securle	6-55
Orchestration v1 Attributes for storage/volume	6-56
Uploading an Orchestration v1	6-57
Orchestrations v1 Life Cycle	6-58
Starting an Orchestration v1	6-59
Monitoring Orchestrations v1	6-61
Return Parameters Displayed in Orchestrations v1	6-62
Terminating an Orchestration v1	6-63
Downloading an Orchestration v1	6-64
Updating an Orchestration v1	6-65
Deleting an Orchestration v1	6-66

## 7 Managing Resources Using Orchestrations v2

---

About Orchestrations v2	7-1
Comparing Orchestrations v1 and Orchestrations v2	7-4
Object References and Relationships	7-6
Object Persistence in Orchestrations v2	7-8
Object Types in Orchestrations v2	7-9
Orchestration v2 Templates and Samples	7-10
Workflow for Creating Instances Using Orchestrations v2	7-22
Building Your First Orchestration v2	7-22
Attributes in Orchestrations v2	7-26
Top-Level Orchestration v2 Attributes	7-27
General Attributes for Objects in Orchestrations v2	7-28
Orchestration v2 Attributes Specific to Each Object Type	7-31
Orchestration v2 Attributes for Acl	7-32
Orchestration v2 Attributes for Backup	7-32
Orchestration v2 Attributes for BackupConfiguration	7-33
Orchestration v2 Attributes for Instance	7-34
Orchestration v2 Attributes for IpAddressAssociation	7-46
Orchestration v2 Attributes for IpAddressPrefixSet	7-46
Orchestration v2 Attributes for IpAddressReservation	7-47
Orchestration v2 Attributes for IpNetwork	7-48
Orchestration v2 Attributes for IpNetworkExchange	7-50
Orchestration v2 Attributes for IPReservation	7-50
Orchestration Attributes for OSSContainer	7-51
Orchestration v2 Attributes for Restore	7-52
Orchestration v2 Attributes for Route	7-53
Orchestration v2 Attributes for SecApplication	7-53

Orchestration v2 Attributes for SecIPList	7-55
Orchestration v2 Attributes for SecList	7-56
Orchestration v2 Attributes for SecRule	7-56
Orchestration v2 Attributes for SecurityProtocol	7-57
Orchestration v2 Attributes for SecurityRule	7-59
Orchestration v2 Attributes for SSHKey	7-60
Orchestration v2 Attributes for StorageAttachment	7-61
Orchestration v2 Attributes for StorageSnapshot	7-62
Orchestration v2 Attributes for StorageVolume	7-63
Orchestration v2 Attributes for VirtualNicSet	7-64
Orchestration v2 Life Cycle	7-65
Managing Orchestrations v2	7-66
Uploading an Orchestration v2	7-67
Starting an Orchestration v2	7-67
Monitoring Orchestrations v2	7-69
Suspending an Orchestration v2	7-70
Terminating an Orchestration v2	7-71
Downloading an Orchestration v2	7-71
Workflows for Updating Orchestrations v2	7-72
Updating an Orchestration v2	7-76
Deleting an Orchestration v2	7-79
Managing Orchestrations v2 Using the REST API	7-80
Managing Orchestrations v2 Using CLI	7-82

## 8 Managing Machine Images

---

About Oracle-Provided Linux Images	8-1
About Oracle-Provided Solaris Images	8-2
About Oracle-Provided Windows Images	8-4
Workflow for Creating Instances Using a Private Machine Image	8-4
Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic	8-5
Registering a Machine Image in Compute Classic	8-7
Listing Machine Images	8-8
Updating a Private Machine Image	8-9
Deleting a Private Machine Image	8-9
Maintaining Versions of Private Machine Images	8-10
Building Your Own Machine Images	8-11
Guidelines for Building Private Images	8-11
Building an Oracle Linux Machine Image	8-14

## 9 Managing Storage Volumes

---

About Storage Volumes	9-1
Creating a Storage Volume	9-2
Creating a Bootable Storage Volume	9-4
Attaching a Storage Volume to an Instance	9-6
Viewing Details of a Storage Volume	9-7
Mounting and Unmounting a Storage Volume	9-7
Mounting a Storage Volume on a Linux Instance	9-8
Unmounting a Storage Volume from a Linux Instance	9-9
Mounting a Storage Volume on an Oracle Solaris Instance	9-10
Unmounting a Storage Volume from an Oracle Solaris Instance	9-15
Mounting a Storage Volume on a Windows Instance	9-17
Unmounting a Storage Volume from a Windows Instance	9-19
Increasing the Size of a Storage Volume	9-20
Detaching a Storage Volume from an Instance	9-25
Deleting a Storage Volume	9-26
Backing Up and Restoring Storage Volumes Using Snapshots	9-27
About Storage Volume Snapshots	9-27
Creating a Storage Volume Snapshot	9-29
Listing Storage Volume Snapshots	9-30
Restoring a Storage Volume from a Colocated Snapshot	9-32
Restoring a Storage Volume from a Remote Snapshot	9-33
Deleting a Storage Volume Snapshot	9-35
Scheduling Backups of Storage Volumes and Restoring from Backups	9-36
Creating a Backup Schedule	9-37
Listing Backup Schedules	9-38
Updating a Backup Schedule	9-39
Deleting a Backup Schedule	9-41
Listing Backups	9-41
Restoring a Storage Volume from a Backup	9-43
Deleting a Backup	9-44

## 10 Configuring the Shared Network

---

About Network Settings	10-1
Setting Up Networking for a Sample Scenario Using the Shared Network	10-3
Managing Security Lists	10-11
About Security Lists	10-11
Creating a Security List	10-14
Updating a Security List	10-15
Adding an Instance to a Security List	10-15



Removing an Instance from a Security List	10-16
Deleting a Security List	10-16
Managing Security Rules	10-17
About Security Rules	10-17
Creating a Security Rule	10-17
Updating a Security Rule	10-19
Deleting a Security Rule	10-19
Managing Security Applications	10-20
About Security Applications	10-20
Listing Security Applications	10-21
Creating a Security Application	10-21
Deleting a Security Application	10-22
Managing Security IP Lists	10-23
About Security IP Lists	10-24
Creating a Security IP List	10-24
Updating a Security IP List	10-25
Deleting a Security IP List	10-26
Managing Public IP Addresses	10-27
About Public IP Addresses	10-28
About Private IP Addresses	10-28
Reserving a Public IP Address	10-28
Updating an IP Reservation	10-29
Attaching a Public IP Address to an Instance	10-30
Removing a Public IP Address from an Instance	10-31
Deleting an IP Reservation	10-33

## 11 Configuring IP Networks

---

About Access Control to Interfaces on IP Networks	11-1
About IP Networks	11-3
Workflows for Using IP Networks	11-7
Managing IP Networks	11-14
Creating an IP Network	11-14
Other Ways of Creating an IP Network	11-15
Listing IP Networks	11-16
Adding an Instance to an IP Network	11-16
Updating an IP Network	11-17
Deleting an IP Network	11-19
Managing IP Network Exchanges	11-20
Creating an IP Network Exchange	11-21
Listing IP Network Exchanges	11-22

Adding an IP Network to an IP Network Exchange	11-22
Deleting an IP Network Exchange	11-23
Managing vNICsets	11-24
Creating a vNICset	11-24
Other Ways of Creating a vNICset	11-25
Listing vNICsets	11-25
Adding an Instance Interface to a vNICset	11-26
Updating a vNICset	11-26
Deleting a vNICset	11-27
Managing Routes	11-28
Creating a Route	11-29
Listing Routes	11-30
Updating a Route	11-30
Deleting a Route	11-32
Managing IP Address Prefix Sets	11-32
Creating an IP Address Prefix Set	11-33
Listing IP Address Prefix Sets	11-34
Updating an IP Address Prefix Set	11-34
Deleting an IP Address Prefix Set	11-36
Managing Security Protocols for IP Networks	11-37
Creating a Security Protocol for IP Networks	11-37
Listing Security Protocols for IP Networks	11-38
Updating a Security Protocol for IP Networks	11-39
Deleting a Security Protocol for IP Networks	11-40
Managing Security Rules for IP Networks	11-41
Creating a Security Rule for IP Networks	11-41
Listing Security Rules for IP Networks	11-43
Applying a Security Rule for IP Networks	11-43
Updating a Security Rule for IP Networks	11-44
Deleting a Security Rule for IP Networks	11-45
Managing ACLs	11-46
Creating an ACL	11-46
Listing ACLs	11-47
Adding a Security Rule to an ACL	11-48
Applying an ACL to a vNICset	11-48
Updating an ACL	11-48
Deleting an ACL	11-49
Managing Public IP Addresses	11-51
Reserving a Public IP Address for IP Networks	11-51
Listing IP Reservations for IP Networks	11-52
Updating an IP Reservation for IP Networks	11-53

Associating a Public IP Address with a vNIC	11-54
Removing an IP Reservation from a vNIC	11-55
Deleting an IP Reservation for IP Networks	11-56

## 12 Accessing an Oracle Linux Instance Using SSH

---

Accessing an Instance from UNIX and UNIX-Like Systems	12-1
Accessing an Instance from Windows	12-2
Adding Users on an Oracle Linux Instance	12-4

## 13 Accessing an Oracle Solaris Instance Using SSH

---

## 14 Accessing a Windows Instance Using RDP

---

Accessing a Windows Instance on IP Network Using RDP	14-1
Accessing a Windows Instance on Shared Network Using RDP	14-4

## 15 Connecting to Instances in a Multitenant Site Using VPN

---

Setting Up VPN	15-2
About Setting Up VPN	15-2
Creating a Cloud Gateway	15-6
Registering a Third-Party VPN Device	15-8
Connecting the Cloud Gateway with the Third-Party Device	15-9
Managing VPN	15-11
Listing VPN Gateways	15-12
Modifying the Reachable Subnets for a VPN Gateway	15-13
Workflow for Adding IP Networks to an Existing VPN Connection	15-14
Deleting a VPN Gateway	15-15
Listing Third-Party VPN Devices	15-16
Updating a Third-Party Device	15-17
Deleting a Third-Party Device	15-18
Listing VPN Connections	15-18
Updating a VPN Connection	15-19
Stopping, Restarting, and Deleting a VPN Connection	15-20

## 16 Setting Up a VPN Connection Using VPNaaS

---

Setting Up VPN Using VPNaaS	16-1
Creating VPN Connections Using VPNaaS	16-6

Prerequisites	16-7
Creating a VPN Connection Using VPNaaS	16-7
Other Ways of Creating a VPN Connection Using VPNaaS	16-10
Viewing the Event Log for a VPN Connection	16-10
Listing VPNaaS Connections	16-11
Updating a VPNaaS Connection	16-11
Deleting a VPNaaS Connection	16-14
VPNaaS Connection to other Environments	16-15

## 17 Set Up VPN Connection to Oracle Cloud Infrastructure

---

Set Up VPNaaS Connection between an IP Network and Oracle Cloud Infrastructure	17-1
About Setting Up VPN Connection between Compute Classic and Oracle Cloud Infrastructure	17-1
Create a VPN Connection in Compute Classic	17-2
Update the VPNaaS Connection in Compute Classic	17-4
Set Up VPN Connection between Shared Network and Oracle Cloud Infrastructure	17-5
Before You Begin	17-5
Create a Cloud Gateway	17-6
Register the Third-Party VPN Device	17-7
Connect the Cloud Gateway with the Oracle Cloud Infrastructure VPN	17-8
Update the Timeout	17-8

## 18 Connecting to Instances Using Oracle Cloud Infrastructure FastConnect Classic

---

About FastConnect Classic	18-1
Managing Cross Connects	18-4
Creating a Cross Connect	18-4
Forwarding Letter of Authorization	18-6
Listing Cross Connects	18-7
Updating a Cross Connect	18-7
Deleting a Cross Connect	18-8
Managing Virtual Circuits	18-9
Creating a Virtual Circuit	18-9
Listing Virtual Circuits	18-11
Updating a Virtual Circuit	18-12
Deleting a Virtual Circuit	18-14
Managing Private Gateways	18-14
Creating a Private Gateway	18-14
Listing Private Gateways	18-16

Updating a Private Gateway	18-16
Deleting a Private Gateway	18-17

## 19 Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN

---

About Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic	19-1
Requesting Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic	19-4
Accessing Your Instances Using VPN	19-4
Configuring Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Gateway	19-5
Example Configuration of a VPN Gateway	19-6
Managing Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Connections	19-8
Starting a VPN Connection	19-9
Listing Your VPN Connections	19-10
Viewing Details of a VPN Connection	19-10
Updating a VPN Connection	19-11
Disabling a VPN Connection	19-11
Deleting a VPN Connection	19-12

## 20 Automating Instance Initialization Using opc-init

---

About opc-init	20-1
Prerequisites for Using opc-init	20-2
Defining Instance Initialization Attributes	20-2
User Data Attributes	20-5
Using opc-init in a Private Machine Image	20-16

## 21 Best Practices for Using Compute Classic

---

## 22 Frequently Asked Questions for Compute Classic

---

Machine Images	22-1
Interfaces	22-2
Instance Properties	22-2
Instance Usage	22-2
Windows Instances	22-4
Shared Network Settings	22-4

Storage Management	22-5
Orchestrations v1	22-6
Using SSH Keys	22-8
Connecting to Instances	22-10
Support	22-10

## 23 Troubleshooting Compute Classic

---

Web Console Problems	23-1
Can't access the web console	23-1
Can't create, update, or delete objects	23-1
Can't upload an orchestration	23-2
My orchestration hasn't created any instances	23-2
Error while starting an orchestration: Specify imagelist or bootorder	23-3
Can't attach a storage volume to an instance	23-4
Can't detach a storage volume from an instance	23-4
Can't delete a storage volume	23-5
Can't delete a storage volume snapshot	23-5
Can't remove an IP address from an instance	23-5
Can't delete a security application	23-6
Can't delete a private image	23-6
Networking Problems	23-7
Can't connect to an instance using SSH	23-7
RSA key fingerprint error while connecting to an instance	23-8
Can't get instances to communicate with each other	23-9
My instance can't connect to instances in another IP network using an IP network exchange	23-9
Can't access my instance even though it has a public IP address	23-10
Can't remove an IP address from an instance	23-11
Can't delete a security application	23-11
SSH Key Problems	23-12
My SSH public key doesn't show up in the Create Instance wizard	23-12
Can't connect to an instance using SSH	23-12
Can't access an instance as a local user over SSH	23-13
RSA key fingerprint error while connecting to an instance	23-13
Storage Volume Problems	23-15
Can't attach a storage volume to an instance	23-15
Can't access a storage volume on my instance	23-15
I can no longer access my storage volume from my instance	23-16
Can't detach a storage volume from an instance	23-16
Can't delete a storage volume	23-16
Can't delete a storage volume snapshot	23-17

Orchestration Problems	23-17
Can't upload an orchestration	23-17
Error while uploading an orchestration: User is not permitted...	23-18
My orchestration hasn't created any instances	23-18
Error while starting an orchestration: object not found	23-18
Error while starting an orchestration: Specify imagelist or bootorder	23-19
My instance was created using a wrong image	23-20
My orchestration is stuck in the stopping state	23-21
opc-init Problems	23-21
Error using opc-init: No chef attributes passed – nothing to do	23-21
Error using opc-init: location not provided, exiting...	23-22
Configuring opc-init automation in launch plan doesn't work	23-22
Launch Plan Problems	23-22
Can't create an instance using a launch plan. Error: Unable to open file	23-23
Can't create an instance using a launch plan. Error displayed: Data is invalid JSON	23-23
Configuring opc-init automation in launch plan doesn't work	23-23

# Preface

*Using Compute Classic* describes how to provision and manage Compute Classic instances, configure network and storage resources, add machine images, and manage SSH keys.

## Topics

- [Audience](#)
- [Related Resources](#)
- [Conventions](#)

## Audience

This document is intended for administrators and users of Compute Classic.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Related Resources

For more information, see these Oracle resources:

- *CLI Reference for Oracle Cloud Infrastructure Compute Classic*
- *REST API for Oracle Cloud Infrastructure Compute Classic*
- [Compute Classic tutorials](#)

## Conventions

This table describes the text conventions used in this document.



<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

# 1


## Getting Started with Oracle Cloud Infrastructure Compute Classic

### Topics

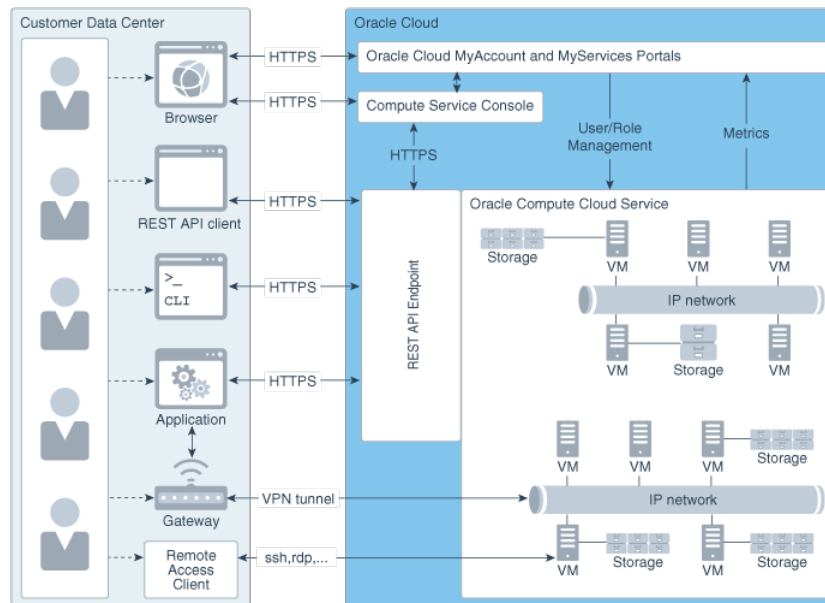
- [About Compute Classic](#)
- [Before You Begin with Compute Classic](#)
- [How to Begin with Compute Classic Subscriptions](#)
- [Compute Classic Terminology](#)
- [Accessing Compute Classic Using the Web Console](#)
- [Accessing Compute Classic Using REST API](#)
- [Accessing Compute Classic Using the Command Line Interface](#)
- [About Compute Classic Roles](#)
- [About Compute Classic Sites](#)
- [Workflow for Using Compute Classic](#)

### About Compute Classic

You can use Compute Classic to rapidly provision virtual machines on Oracle Cloud with all the necessary storage and networking resources, manage and scale your virtual machine topology in the cloud easily, and migrate your Oracle and third-party applications to Oracle Cloud.

Compute Classic is a secure, reliable, low cost, standards-based infrastructure service. For a brief introduction to the features of the service, watch this video.  [Video](#)

The following graphic provides an architectural overview of Compute Classic.



The graphic shows instances and attached block storage, along with IP networks in Compute Classic. Access to Compute Classic instances is possible in several ways. You can use a browser to access the web console, you can access the REST API directly, or you can use the command-line interface. Secure access is provided by protocols such as SSH and RDP. You can also set up a VPN tunnel to provide secure access to instances in your Compute Classic network.

Using Compute Classic, you can do the following:

- **Migrate your applications to the public cloud**
  - When you subscribe to Oracle Cloud Infrastructure Dedicated Compute Classic, you get an environment that consists of high-performance x86 servers reserved for your use. Depending on the configuration that you subscribe to, you get a certain numbers of physical cores (OCPUs) of a modern Intel Xeon processor with hyperthreading enabled. Because you're the only tenant on the site, you enjoy predictable performance in the cloud.
  - You can also subscribe for Compute Classic – Compute Capacity. In this case, no servers are reserved for your use. The instances that you provision are placed on any available server on a site that's shared with other tenants. You can opt for a metered or nonmetered subscription.

With either subscription, you can migrate your on-premises applications to the cloud and take advantage of the elastic compute, storage, and network capabilities that Compute Classic provides.

For details of the available subscription options and the prices, go to <https://cloud.oracle.com/compute-classic> and click the **Pricing** tab.

- **Use up to 2x of your allocated resources**

If you have a nonmetered subscription, you can use up to double the resources that you've subscribed to. This is referred to as bursting. For example, if you've subscribed for 4 OCPUs per month, bursting allows you to use a total of up to 8 OCPUs. The extra resources are charged per hour, and you're billed monthly in arrears, using the Pay as You Go pricing model. For pricing information, see the

current per hour price list. Go to <https://cloud.oracle.com/compute-classic> and click the **Pricing** tab.

- **Expand your account to include additional SKUs**

If you have purchased a subscription for Compute Classic, at any time you can expand your account by adding another SKU of the same service to your order. This is called bursting out. For example, if you've subscribed to nonmetered Compute Classic – Compute Capacity, bursting out allows you to add a metered subscription of Compute Classic – High I/O Compute Capacity to your account. The extra resources are charged per hour, and you're billed monthly in arrears, using the Pay as You Go pricing model. For pricing information, see the current per hour price list. Go to <https://cloud.oracle.com/compute-classic> and click the **Pricing** tab.

- **Assign processor and memory resources from a range of resource profiles**

While creating Compute Classic instances, you can assign CPU and memory resources by selecting from a wide range of resource profiles (called *shapes*), each of which is a carefully designed combination of processor and memory limits. In some sites, you can also select high I/O shapes. When you select one of these shapes, a nonpersistent NVMe SSD disk is automatically attached to your instance.

- **Automate your instance provisioning and management workflows**

You can define all the attributes for multiple, high availability (HA)-enabled virtual machines of varying shapes and machine images in an orchestration. Using the web console, you can then easily create, remove, and re-provision all of the virtual machines and associated resources as required through the orchestration.

- **Create instances using Oracle-provided and custom machine images**

You can use one of several Oracle-provided machine images to quickly provision robust virtual machines.

Images provided by Oracle partners are available in Oracle Cloud Marketplace.

You can also build custom machine images based on the operating system and disk size of your choice and use those images to create virtual machines.

 **Note:**

The operating system and software that you use to build private images must have the required licenses. You're responsible for purchasing the required licenses and ensuring support for any third-party operating systems and software that you run on Compute Classic instances.

You can automate instance initialization by using `opc-init`. The `opc-init` scripts are included in Oracle-provided Oracle Linux and Windows images. You can also install `opc-init` in the private images that you create.

- **Provide a persistent boot disk for your instance**

Instances boot from a persistent disk, ensuring that any changes that you make at the operating system-level persist when the instance is re-created.

- **Clone your instances using snapshots**

If you create and customize an instance using a nonpersistent boot disk, you can use instance snapshots to use the instance as a template to create multiple identical instances.

- **Attach high-capacity block storage to instances**

You can attach up to 20 TB of block storage to each of your instances for storing data and applications, by creating multiple persistent storage volumes and attaching them to the instances. Even after you delete instances, the data stored in the storage volumes remains intact until you delete the volumes.

- **Back up and restore storage volumes using snapshots**

You can use storage volume snapshots to create snapshots of persistent data or boot volumes. You can then use these storage volume snapshots as a form of data backup, or to create multiple, identical storage volumes.

- **Implement shared object storage in the cloud over NFSv4**

You can use Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution to provide highly scalable, low-cost, reliable shared storage in Oracle Cloud Infrastructure Object Storage Classic for your Oracle Linux instances running in Compute Classic.

- **Set up IP networks and routes**

You can create one or more IP networks and add instances to multiple networks, if required. You can also create IP network exchanges or specify IP routes to enable traffic between different IP networks. Using IP networks you can isolate your network from the shared network and ensure that you have complete control over the private IP addresses assigned to instances.

- **Exercise fine-grained control over network traffic**

You can control network traffic among individual instances and also between specific groups of instances and external hosts. You can also control traffic to and from instances over specific protocols and ports that you can define.

- **Reserve and assign fixed public IP addresses**

For an instance that requires access to the Internet, you can reserve and use a static public IP address.

- **Monitor and manage all of your resources through a unified interface**

You can access, administer, and use Compute Classic through an easy-to-use graphical web console. The console provides a single interface that you can use to monitor and manage all your Compute Classic resources.

You can also access Compute Classic and manage resources by using REST API calls.

- **Ensure secure access to instances**

You can configure your Compute Classic Linux and Solaris instances (virtual machines) to be accessed securely from remote hosts by using SSH, and you can configure your Windows instances to be accessed securely by using RDP.

- **Configure high-speed and secure connections to instances in your account**  
(Not available on Oracle Cloud at Customer)

You can use Oracle Cloud Infrastructure FastConnect Classic to access your Oracle Cloud services using a direct connection from your premises or colocation facilities. You can also configure a VPN connection to your multitenant or dedicated Compute Classic site.

## Before You Begin with Compute Classic

Before you begin using Compute Classic:

- Create and configure your account on Oracle Cloud. See *Getting an Oracle.com Account* in *Getting Started with Oracle Cloud*.
- Understand the features of the service. See [About Compute Classic](#).
- Be familiar with the Compute Classic terminology. See [Compute Classic Terminology](#).

## How to Begin with Compute Classic Subscriptions

To get started with Compute Classic, sign up for a free credit promotion, or purchase a subscription. You can then access the web console and create users and assign roles.

Here's how to get started with Compute Classic promotions and subscriptions:

1. Sign up for a free credit promotion or purchase a subscription. (Not available on Oracle Cloud at Customer) See *Requesting and Managing Free Oracle Cloud Promotions* or *Buying an Oracle Cloud Subscription* in *Getting Started with Oracle Cloud*.
2. Access the Compute Classic service. See [Accessing Compute Classic Using the Web Console](#).
3. Learn about user accounts and roles. See [About Compute Classic Roles](#).
4. Create accounts for your users and assign them appropriate privileges and roles. See *Adding Users and Assigning Roles* in *Getting Started with Oracle Cloud*.
5. Get familiar with Compute Classic terminology. See [Compute Classic Terminology](#).

## Compute Classic Terminology

The following table lists and describes the key terms used in Compute Classic, arranged in alphabetical order.

For a visual overview of the dependencies and relationships between various objects, see [Relationships Between Compute Classic Resources](#).

Term	Definition	More Information
Image List	An <b>image list</b> is a collection of Compute Classic machine images. Each machine image in an image list is identified by a unique entry number.	<a href="#">Maintaining Versions of Private Machine Images</a>
Instance	An <b>instance</b> is a virtual machine in Compute Classic, created by using a specific machine image, with CPU and memory resources defined by a shape.	<a href="#">About Instances</a>

Term	Definition	More Information
Instance Snapshot	An <b>instance snapshot</b> captures the current state of the nonpersistent boot disk of an instance and creates a corresponding machine image. You can then register this machine image with your Compute Classic account and use it to create instances.	<a href="#">Cloning an Instance by Using Instance Snapshots</a>
IP Network	An <b>IP network</b> allows you to define an IP subnet in your account. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network. These IP addresses aren't part of the common pool of Oracle-provided IP addresses used by the shared network. When you add an instance to an IP network, the instance is assigned an IP address in that subnet. You can assign IP addresses to instances either statically or dynamically, depending on your business needs. So you have complete control over the IP addresses assigned to your instances.	<a href="#">About IP Networks</a>
IP Network Exchange	An <b>IP network exchange</b> enables access between IP networks that have non-overlapping addresses, so that instances on these networks can exchange packets with each other without NAT.	<a href="#">About IP Networks</a>
IP Reservation	An <b>IP reservation</b> is a public IP address that you can attach to a Compute Classic instance that requires access to or from the Internet	<a href="#">About Public IP Addresses</a>
Launch plan	A <b>launch plan</b> is a JSON (JavaScript Object Notation)-formatted file that defines the properties of one or more instances in Compute Classic. You can use a launch plan to quickly start multiple instances in Compute Classic. The attributes in a launch plan include the instance label and name, the image and shape to be used for the instance, and so on.	<a href="#">Creating Instances Using Launch Plans</a>
Machine Image	A <b>machine image</b> is a template of a virtual hard disk of a specific size with an installed operating system. You use machine images to create virtual machine instances in Compute Classic.	<a href="#">Managing Machine Images</a>
Orchestration	An <b>orchestration</b> defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.	<a href="#">About Orchestrations v1</a> <a href="#">About Orchestrations v2</a>
Private Gateway	A <b>private gateway</b> object allows you to set up a private peering connection between subnets in your premises and IP networks in your Compute Classic account, using Oracle Cloud Infrastructure FastConnect Classic.	<a href="#">Connecting to Instances Using Oracle Cloud Infrastructure FastConnect Classic</a>
Route	A <b>route</b> specifies the IP address of the destination as well as the vNICset that provides the next hop for routing packets.	<a href="#">About IP Networks</a>

Term	Definition	More Information
Security Application	<p>A <b>security application</b> allows you to specify the protocol and port that you want to use to enable traffic between a source and a destination using security rules.</p> <p>In the API and CLI, security applications are called <i>secapplications</i>.</p>	<a href="#">About Security Applications</a>
Security IP List	<p>A <b>security IP list</b> is a list of IP subnets (in the CIDR format) or IP addresses that are external to instances in Compute Classic. You can use a security IP list as the source or the destination in security rules to control network access to or from Compute Classic instances.</p> <p>In the API and CLI, security IP lists are called <i>seciplists</i>.</p>	<a href="#">About Security IP Lists</a>
Security List	<p>A <b>security list</b> is a group of Compute Classic instances that you can specify as the source or destination in one or more security rules. The instances in a security list can communicate fully, on all ports, with other instances in the same security list using their private IP addresses.</p> <p>When you add an instance to a security list, the inbound and outbound policies defined in the security list are applicable to that instance.</p> <p>In the API and CLI, security lists are called <i>seclists</i>.</p>	<a href="#">About Security Lists</a>
Security Rule	<p>A <b>security rule</b> is a firewall rule that you can define to control network access to Compute Classic instances over a specified security application.</p> <p>You can use a security rule to control network access,</p> <ul style="list-style-type: none"><li>• between instances in two security lists, or</li><li>• from a set of external hosts (a security IP list) to instances in a security list.</li></ul> <p>In the API and CLI, security rules are called <i>secrules</i>.</p>	<a href="#">About Security Rules</a>
Shape	<p>A <b>shape</b> is a resource profile that specifies the number of OCPUs and the amount of memory to be allocated to an instance in Compute Classic. The shape determines the type of disk drive that your instance uses. If you select a general purpose or high-memory shape, a hard-disk drive is used. If you select a high I/O shape, an NVM Express SSD disk is automatically attached to your instance. For general purpose and high-memory shapes, you can select the block storage disk size, but for high I/O shapes, the size of the disk is determined by the shape.</p>	<a href="#">About Shapes</a>
Site	<p>A <b>site</b> is a set of physical servers and the associated storage and networking resources in an Oracle Cloud data center. Each site has a distinct REST API endpoint and its network is isolated from other sites in the data center. When you subscribe to Oracle Cloud Infrastructure Dedicated Compute Classic, you get a site that's dedicated for your use. When you subscribe to Compute Classic, you share sites with other tenants. Multiple sites are assigned to your subscription, and you can see the capacity available on each site at any time. This helps you pick the site that you want to provision resources in.</p>	<a href="#">About Compute Classic Sites</a>



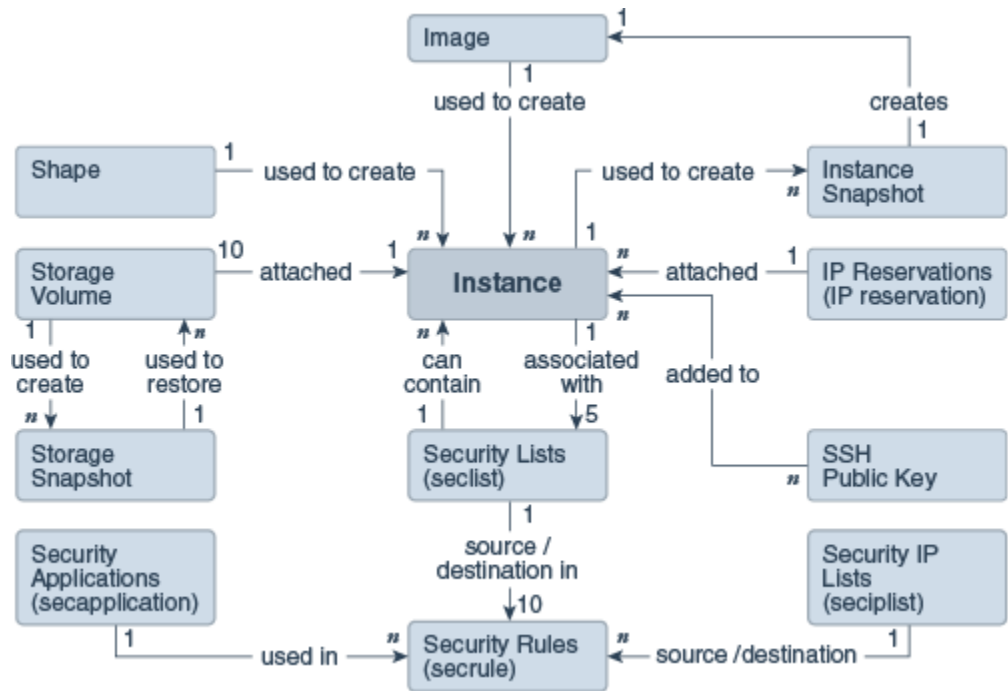
Term	Definition	More Information
Storage Volume	A <b>storage volume</b> is a virtual disk that provides persistent block storage space for instances in Compute Classic.	<a href="#">About Storage Volumes</a>
Storage Volume Snapshot	A <b>storage volume snapshot</b> is a backup of all the data currently stored on a storage volume. You can use this snapshot to make a copy of or restore a storage volume.	<a href="#">Backing Up and Restoring Storage Volumes Using Snapshots</a>
Virtual NIC	A <b>Virtual NIC</b> , or <b>vNIC</b> , is a virtual network interface card that enables an instance to be associated with a network. Instances created using Oracle-provided Oracle Linux or Windows images with the release version 16.3.6 or later support eight vNICs, enabling each instance to be associated with up to eight networks.	<a href="#">About IP Networks</a>
Virtual NIC Set	A <b>Virtual NIC Set</b> , or <b>vNICset</b> , is a collection of one or more vNICs. vNICsets are useful when you want to use multiple vNICs for the same action. For example, you use vNICsets to specify multiple vNICs as a source or a destination in a security rule. You can also use vNICsets in routes to specify multiple vNICs as the next hop destination for that route.	<a href="#">About IP Networks</a>
VPN Endpoint	A <b>VPN endpoint</b> represents a VPN tunnel between your data center and your Compute Classic site.	<a href="#">Setting Up VPN Using VPNaaS</a>  <a href="#">About Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic</a>

### Relationships Between Compute Classic Resources

The following diagram shows the relationships between the resources that you can use to create and manage instances in Compute Classic.

Each block in the diagram represents a resource in Compute Classic.

The numbers at either end of each arrow and the text label on the arrow, together, indicate the relationship between the resources that the arrow connects. For example, the number *1* at either end of the arrow between *IP reservations* and *instance* indicates that you can associate an IP reservation with only one instance and an instance with only one IP reservation. Similarly, *n* at either end of the arrow connecting *SSH public key* and *instance* indicates that you can associate any number of keys with each instance and a single key with any number of instances.




## Accessing Compute Classic Using the Web Console

You can manage and monitor your Compute Classic instances and the associated storage and networking resources through an easy-to-use graphical web console.

1. Sign in to your Cloud Account.
  - For Oracle Cloud, see *Signing in to Your Cloud Account* in *Getting Started with Oracle Cloud*.
  - For Oracle Cloud at Customer, click the Infrastructure Classic Console URL from the welcome email.

The Infrastructure Classic Console is displayed.

2. Click  in the top left corner of the Dashboard.
  3. Under **Services**, click **Compute Classic**.
- The **Compute Classic** console is displayed.
4. (Optional) This step is relevant only if your domain spans multiple sites. To change the site, click the **Site** menu near the top of the page.

See [About Compute Classic Sites](#).

### Note:

For security, the web console automatically times out after 15 minutes of inactivity. To continue using the web console, log in again.

## Accessing Compute Classic Using REST API

You can programmatically provision and manage Compute Classic instances and the associated storage and networking resources by using a REST (REpresentational State Transfer) application programming interface (API).

Each REST API call maps to an HTTP request: getting an object (`GET`), adding an object (`POST`), updating an object (`PUT`), and deleting an object (`DELETE`). The HTTP response code indicates whether the request was successful. Each object for which you can perform the `GET`, `POST`, `PUT`, and `DELETE` requests is identified uniquely by its URI.

To access Compute Classic by using the REST API you must use the REST endpoint URL that Oracle provided when your administrator subscribed to the service.

To find out the REST endpoint URL for your service, see *Send Requests in REST API for Oracle Cloud Infrastructure Compute Classic*.

## Accessing Compute Classic Using the Command Line Interface

Compute Classic provides a comprehensive command line interface (CLI) that supports all the actions you can perform using the HTTP REST API.

For information about the installing and using the CLI, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*

## About Compute Classic Roles

The following table summarizes the roles you can use to administer and use Compute Classic.

Role	Description
TenantAdminGroup (Identity Domain Administrator)	Users who are assigned this role can perform all the tasks in the Infrastructure Classic Console, including user and role management tasks. Note that Oracle assigns this role to all trial users.
<code>service-instance-name.Compute_Operations</code> (Service Administrator)	Users who are assigned this role can view, create, update, and delete Compute Classic resources. The identity domain administrator can create additional service administrators, as required, by assigning this role in Oracle Cloud Infrastructure Classic Console. For business continuity, consider creating at least two users with the <code>Compute_Operations</code> role. These users must be IT system administrators in your organization.
<code>service-instance-name.Compute_Monitor</code>	Users who are assigned this role can view Compute Classic resources. The identity domain administrator can create users with this role in Oracle Cloud Infrastructure Classic Console.

See Adding Users and Assigning Roles in *Getting Started with Oracle Cloud*.

## About Compute Classic Sites

A **site** is a set of physical servers and the associated storage and networking resources in an Oracle Cloud data center. Each site has a distinct REST API endpoint and its network is isolated from other sites in the data center. When you subscribe to Oracle Cloud Infrastructure Dedicated Compute Classic, you get a site that's dedicated for your use. When you subscribe to Compute Classic, you share sites with other tenants. Multiple sites are assigned to your subscription, and you can see the capacity available on each site at any time. This helps you pick the site that you want to provision resources in.

- If your domain spans multiple sites, then when you log in to the web console, you can select the site that you want to access.
- Remember, the web console always shows the resources in your domain in the site that you're currently logged in to. You can't see resources across all your sites.
- You also can't use resources from one site in your domain with resources in another site. For example, if you've created a storage volume in one site, you can't use this storage volume with an instance that you've created in another site.

### Note:

One exception to this rule is if you've created storage volume snapshots in another site. In this case, you can view snapshots created in another site. See [Restoring a Storage Volume from a Remote Snapshot](#).

- If your domain spans multiple sites, the web console allows you to view the aggregate resource usage by all tenants on the currently selected site. This is useful while creating instances and other resources. You can check which site has adequate capacity to create the resources you require. To view site utilization, log in to the web console and click the **Site** menu near the top of the page.
- If you have multiple entitlements of Compute Classic in your account, you can use the **Site** menu near the top of the page to switch between different entitlements.

To access the **Site** menu, log in to the web console. See [Accessing Compute Classic Using the Web Console](#).

## Workflow for Using Compute Classic

Compute Classic supports multiple workflows for creating compute, network, and storage resources.

For example, you can create the required storage volumes first and then create the instances to which the storage volumes should be attached. Alternatively, you can create instances first and then create and attach the required storage volumes to the instances. Similarly, you can create security lists first and then create instances and add them to the security lists, or you can create the instances first and then create security lists and add instances to them.

The following table provides a sample workflow to get a Compute Classic account and start creating and accessing instances. Use this workflow as a guide to get started with Compute Classic.

Task	Description	More Information
Sign up for a free credit promotion or purchase a subscription.	Provide your information, and sign up for a free credit promotion. (Not available on Oracle Cloud at Customer) Alternatively, purchase a subscription to Compute Classic.	<a href="#">How to Begin with Compute Classic Subscriptions</a>
Add and manage users and roles.	Create accounts for your users and assign them appropriate privileges. Assign the necessary Compute Classic roles.	Adding Users and Assigning Roles in <i>Getting Started with Oracle Cloud</i>
Monitor the service.	Check on the day-to-day operation of your service, monitor performance, and review important notifications.	Managing and Monitoring Oracle Cloud Services in <i>Managing and Monitoring Oracle Cloud</i>
Understand Compute Classic terminology.	Learn about instances, images, shapes, security lists, security rules, and so on.	<a href="#">Compute Classic Terminology</a>
Generate SSH key pairs.	Generate the SSH key pairs that you plan to use to access your instances. You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.	<a href="#">Generating an SSH Key Pair</a>
Access the service.	Access the service through the Compute Classic web console or RESTful API.	<a href="#">Accessing Compute Classic Using the Web Console</a>
Add and enable SSH public keys.	Add the SSH public keys that you generated, and enable the keys.	<a href="#">Adding an SSH Public Key</a>
(Optional) Build machine images and add them to Compute Classic	Build your own machine images, upload them to Oracle Cloud Infrastructure Object Storage Classic, and register them in Compute Classic.	<a href="#">Workflow for Creating Instances Using a Private Machine Image</a>
(Optional) Create boot disks.	Create storage volumes that can be used as boot disks for instances.	<a href="#">Creating a Bootable Storage Volume</a>
(Optional) Create storage volumes.	Provide storage for your instances by creating and attaching storage volumes.	<a href="#">Managing Storage Volumes</a>
Create instances.	Create instances with the required CPU, hard disk, and memory requirements according to the needs of your business.	<a href="#">Managing Instances</a>
(Optional) Configure security lists and security rules	Set up firewalls for your instances by using security lists and security rules.	<a href="#">Configuring the Shared Network</a>
Log in to the instances.	Access your instances securely.	<a href="#">Accessing an Oracle Linux Instance Using SSH</a> <a href="#">Accessing an Oracle Solaris Instance Using SSH (Not available on Oracle Cloud at Customer)</a> <a href="#">Accessing a Windows Instance Using RDP</a>

# 2

## Enabling Secure Access to Instances Using SSH

This section provides information about generating and using SSH keys to enable secure access to your instances.

 **Note:**

You can't use SSH keys to log in to a Windows instance. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

For information about using an SSH key to log in to your Oracle Linux instance, see [Accessing an Oracle Linux Instance Using SSH](#).

For information about using an SSH key to log in to your Oracle Solaris instance, see [Accessing an Oracle Solaris Instance Using SSH](#). (Not available on Oracle Cloud at Customer)

### Topics

- [About SSH Keys](#)
- [Generating an SSH Key Pair](#)
- [Adding an SSH Public Key](#)
- [Attaching an SSH Public Key to an Instance](#)
- [Viewing an SSH Public Key](#)
- [Updating an SSH Public Key](#)
- [Disabling an SSH Public Key](#)
- [Enabling an SSH Public Key](#)
- [Deleting an SSH Public Key](#)

## About SSH Keys

You can log in securely to your Compute Classic instances from a remote host by using a secure shell (SSH) connection.

 **Note:**

You can't use SSH keys to log in to a Windows instance. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

SSH is a cryptographic network protocol that uses two keys, a public key and a private key, to provide secure communication between two computers. SSH uses port 22 by default.

Before creating instances, generate at least one SSH key pair and ensure that the private key is available on each host that you'll use to access instances. You can use any SSH utility to generate SSH keys and log in to your instances. For example, if you're logging in from a Windows host, you can use PuTTY. If you're using a Linux host, you can use OpenSSH.

You can associate a single SSH public key with multiple instances. Also, if you've already created and uploaded SSH public keys to Compute Classic, then you can associate multiple SSH keys with an instance when you create the instance. If you've created your instance using an Oracle-provided image, then you can use SSH to log in to your instance as the `opc` user. You can then inject additional SSH public keys by editing the `/home/opc/.ssh/authorized_keys` file on your instance.

 **Caution:**

If you need to edit the `~/.ssh/authorized_keys` file of the `opc` user on an instance, then before you make any changes to the file, start a second `ssh` session and ensure that it remains connected while you edit the `authorized_keys` file. This second `ssh` session serves as a backup. If the `authorized_keys` file gets corrupted or you inadvertently make changes that result in the `opc` user getting locked out of the instance, then you can use the backup `ssh` session to fix or revert the changes. Before closing the backup session, test the changes you made in the `~/.ssh/authorized_keys` file by logging in with the new or updated SSH key. Remember, if you don't have any other user set up on your instance, and if any changes to the `~/.ssh/authorized_keys` file result in the `opc` user getting locked out, then you might not be left with any way to access your instance.

 **Note:**

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any SSH public keys that you added or edited manually (that is, not during instance creation) must be added or edited again. To do this, you must log in to the instance by using the original SSH private key. So retain and safeguard your original SSH private key.

To log in to an instance by using SSH, you must provide the private key that matches a public key associated with the instance.

# Generating an SSH Key Pair

To access your instances using SSH, generate an SSH key pair, associate the public key with your instances, and use the private key to log in to the instances using SSH.

## Note:

You can't use SSH keys to log in to a Windows instance. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

## Caution:

Keep your SSH keys secure. Lay down policies to ensure that the keys aren't lost or compromised when employees leave the organization or move to other departments. If you lose your private key, then you can't access your instances. For business continuity, ensure that the SSH keys of at least two IT system administrators are added to your instances.

## Topics

- [Generating an SSH Key Pair on UNIX and UNIX-like Systems](#)
- [Generating an SSH Key Pair on Windows](#)

## Generating an SSH Key Pair on UNIX and UNIX-Like Systems

Use the following procedure to generate an SSH key pair on UNIX and UNIX-like systems:

1. Run the `ssh-keygen` command.

You can use the `-t` option to specify the type of key to create.

For example, to create an RSA key, run:

```
ssh-keygen -t rsa
```

You can use the `-t` option to specify the length (bit size) of the key, as shown in the following example:

```
ssh-keygen -b 2048 -t rsa
```

2. The command prompts you to enter the path to the file in which you want to save the key.

A default path and file name are suggested in parentheses. For example: `/home/user_name/.ssh/id_rsa`. To accept the default path and file name, press **Enter**. Otherwise, enter the required path and file name, and then press **Enter**.

3. The command prompts you to enter a passphrase.



The passphrase is not mandatory if you want to log in to an instance created using an Oracle-provided image. However, it is recommended that you specify a passphrase to protect your private key against unauthorized use.

 **Note:**

With some images provided on Oracle Marketplace, the use of a passphrase might be mandatory.

4. When prompted, enter the passphrase again to confirm it.

The command generates an SSH key pair consisting of a public key and a private key, and saves them in the specified path. The file name of the public key is created automatically by appending `.pub` to the name of the private key file. For example, if the file name of the SSH private key is `id_rsa`, the file name of the public key would be `id_rsa.pub`.

Make a note of the path and file names of the private and public keys. When you create an instance, you must specify the SSH public key value. When you log in to an instance, you must provide the path to the corresponding SSH private key and you must enter the passphrase when prompted.

#### Generating an SSH Key Pair on Windows

You can generate an SSH key pair on a Microsoft Windows machine by using an application such as PuTTY. See the tutorial, [Creating SSH Keys for Use with Oracle Cloud Services](#).

## Adding an SSH Public Key

To access an instance using SSH, generate at least one SSH key pair and upload the SSH public key that should be associated with the instance to Compute Classic. You'll use this SSH key to access your instance later on, when your instance is running.

 **Note:**

You can't use SSH keys to log in to a Windows instance. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

#### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have generated an SSH key pair. See [Generating an SSH Key Pair](#).

## Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. Click **Add SSH Public Key**.
4. Enter or select the following details:

- Enter a name for the key.

Choose a name that you can use to identify the key easily.

- In the **Value** field, click **Select File**. Navigate to the path where your SSH key is saved, and select the SSH public key file that you want to add. The value of the SSH key appears in the field.

Alternatively, you can paste the value of the SSH public key that you want to add.

### ! Important:

Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks.

See the following example:

**Add SSH Public Key** ✕

Enter an SSH key name to reference this key for launching virtual machine instances. Copy your SSH public key value and paste it here. Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks. [Learn more](#).

? \* Name

? \* Value

Select File

Enabled

Add Cancel

- To enable the key, select the **Enabled** check box. Alternatively, you can deselect the check box and enable the key later.

5. Click **Add**.

After adding an SSH public key, you can attach it to an instance when you create the instance.

To add an SSH public key using the CLI, use the `opc compute ssh-key add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To add an SSH public key using the API, use the `POST /sshkey/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Attaching an SSH Public Key to an Instance

You must attach an SSH key to an instance when you create the instance. You'll use this SSH key to access your instance later on, when your instance is running.

 **Note:**

You can't use SSH keys to log in to a Windows instance. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

For more information about creating an instance, see [Creating Instances](#).

## Viewing an SSH Public Key

After you've generated an SSH key pair and added a public SSH key, you can view the SSH key name and value.

 **Note:**


You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. You can filter the list of SSH public keys according to their category or status. To list SSH keys with a specific status (such as enabled or disabled), click the **Show** menu and select the appropriate filter. To list SSH keys of a specific category (such as all or personal), click the **Category** menu and select the appropriate filter.

4. Go to the SSH key that you want to view. From the  menu, select **View**.

To view an SSH public key using the CLI, use the `opc compute ssh-key get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view an SSH public key using the API, use the `GET /sshkey/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating an SSH Public Key

After adding an SSH public key to Compute Classic, you can change the key value. The updated key value takes effect when the associated instances are re-created. You can also disable and re-enable the key.

### Caution:

When you disable a key that's associated with an instance, the instance continues to be accessible using `ssh`. But before re-creating the instance, you must either remove the disabled key from the orchestration of that instance or enable the key. Otherwise, the orchestration won't start.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. Identify the key that you want to update. From the  menu, select **Update**.

The Edit SSH Public Key dialog box is displayed.

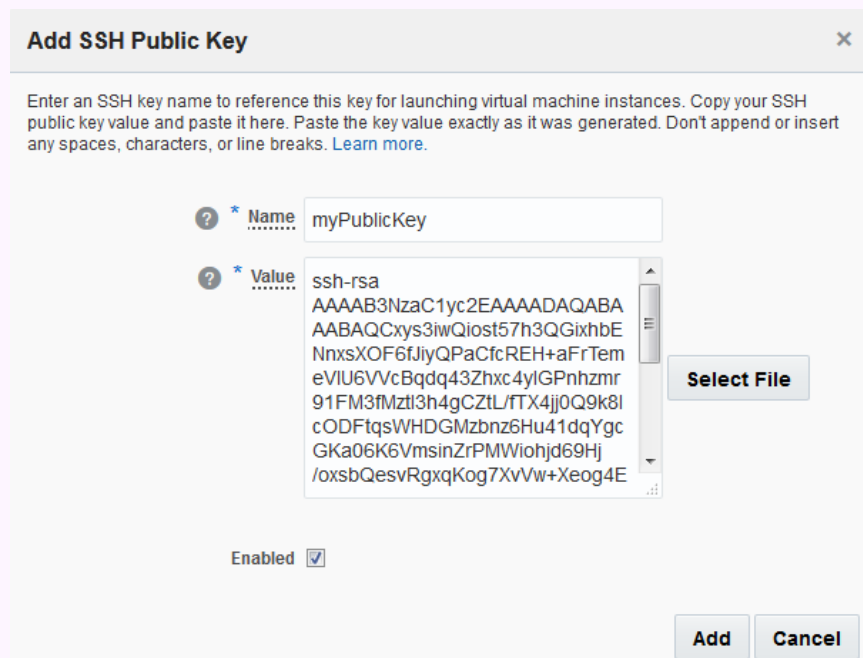
4. In the **Value** field, click **Select File**. Navigate to the path where your SSH key is saved, and select the SSH public key file that you want to add. The value of the SSH key appears in the field.

Alternatively, you can copy and paste the new value of the SSH public key in the **Value** field.

**! Important:**

Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks.

See the following example:



**Add SSH Public Key** ✕

Enter an SSH key name to reference this key for launching virtual machine instances. Copy your SSH public key value and paste it here. Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks. [Learn more](#).

? \* Name

? \* Value

**Select File**

Enabled

**Add** **Cancel**

5. Enable or disable the key, as required.

6. Click **Update**.

To change the value of the SSH public key using the CLI, use the `opc compute ssh-key update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an SSH key using the API, use the `PUT /sshkey/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

 **Tip:**

If you update the value of an SSH public key, remember to make the corresponding private key available on each of your local hosts that'll be used to access instances. The updated public key value takes effect when the instances that the key is associated with are re-created.

 **Note:**

You can also update SSH public keys associated with an instance by logging in to the instance and editing the `~/.ssh/authorized_keys` file.

If you need to edit the `~/.ssh/authorized_keys` file of a user on your instance, then before you make any changes to the file, start a second `ssh` session and ensure that it remains connected while you edit the `authorized_keys` file. This second `ssh` session serves as a backup. If the `authorized_keys` file gets corrupted or you inadvertently make changes that result in your getting locked out of the instance, then you can use the backup `ssh` session to fix or revert the changes. Before closing the backup `ssh` session, test the changes you made in the `authorized_keys` file by logging in with the new or updated SSH key.

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any SSH public keys that you added or edited manually (that is, not during instance creation) must be added or edited again. To do this, you must log in to the instance by using the original SSH private key. So retain and safeguard your original SSH private key.

## Disabling an SSH Public Key

When you add an SSH public key, by default the key is enabled. At any time, you can disable the key, and enable it again.


 **Caution:**

When you disable a key that's associated with an instance, the instance continues to be accessible using `ssh`. But before re-creating the instance, you must either remove the disabled key from the orchestration of that instance or enable the key. Otherwise, the orchestration won't start.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in [Managing and Monitoring Oracle Cloud](#)*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. Identify the SSH public key that you want to disable. From the  menu, select **Update**.
4. In the Edit SSH Public Key dialog box, deselect **Enabled** and click **Update**.

To disable an SSH key using the CLI, use the `opc compute ssh-key update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

 **Note:**

When you use the `opc compute ssh-keys update` command, you must provide the path to the SSH public key file as an argument to this command. If you no longer have the public key file on your local host, you can download the public key by using the `opc compute ssh-keys get` command, as described in *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To disable an SSH public key using the API, use the `PUT /sshkey/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Enabling an SSH Public Key

When you add an SSH public key, by default the key is enabled. If you've disabled a key, you can enable it at any time.


 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. Identify the SSH public key that you want to enable. From the  menu, select **Update**.
4. On the Edit SSH Public Key dialog box, select **Enabled** and click **Update**.

To enable an SSH key using the CLI, use the `opc compute ssh-key update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

 **Note:**

When you use the `opc compute ssh-keys update` command, you must provide the path to the SSH public key file as an argument to this command. If you no longer have the public key file on your local host, you can download the public key by using the `opc compute ssh-keys get` command, as described in *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To enable an SSH public key using the API, use the `PUT /sshkey/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting an SSH Public Key

When you no longer need an SSH public key, you can delete it.

 **Caution:**

When you delete a key that's associated with an instance, the instance continues to be accessible using `ssh`. But before re-creating the instance, you must remove the deleted key from the orchestration of that instance. Otherwise, the orchestration won't start.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.



 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.


If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **SSH Public Keys**.

The SSH Public Keys page is displayed.

3. Identify the SSH key that you want to delete. From the  menu, select **Delete**.

To delete an SSH public key using the CLI, use the `opc compute ssh-key delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an SSH public key using the API, use the `DELETE /sshkey/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

# 3

## Creating Instances

You can create Compute Classic instances in several ways.

### Topics

- [About Instances](#)
- [Workflow for Creating Your First Instance](#)
- [Selecting a Method of Creating Instances](#)
- [Creating an Instance Using QuickStarts](#)
- [Creating an Instance from the Instances Page](#)
- [Creating an Instance Using a Private Image](#)
- [Creating Instances Using Orchestrations](#)
- [Creating an Instance Using a Blank Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
- [Creating an Instance Using Visual Object Editor](#)
- [Cloning an Instance by Using Instance Snapshots](#)
- [Logging In to an Instance](#)

## About Instances

A Compute Classic instance is a virtual machine running a specific operating system and with CPU and memory resources that you specify.

### Defining Instances

An instance is defined by its machine image and shape. A **machine image** is a template of a virtual hard disk that has a specific operating system installed. See [Managing Machine Images](#). A **shape** defines the number of CPUs and RAM available to an instance. See [About Shapes](#).

### Identifying Instances

You can specify a name as well as a label to identify your instance. The instance name that you specify becomes a prefix for an ID that's generated automatically. If you've specified a label, then the label is displayed in the web console. Otherwise, the system-generated ID is displayed.

You can assign tags to your instances to make it easy to sort and find instances.

### Adding Storage

You can attach up to 20 TB of block storage to each of your instances for storing data and applications, by creating multiple persistent storage volumes and attaching them

to the instances. Even after you delete instances, the data stored in the storage volumes remains intact until you delete the volumes.

Instances boot from a persistent disk, ensuring that any changes that you make at the operating system-level persist when the instance is re-created.

See [Managing Storage Volumes](#).

### Configuring Network Settings

You can implement fine-grained control over network access to your instances, both from other Compute Classic instances as well as from external hosts.

When you create an instance, by default, it doesn't allow access from any other instance or external host. To enable unrestricted communication among some of your instances, you can create a security list and add all the instances to that security list. When you add an instance to a security list, the instance can communicate with all the other instances in the same list.

By default, the instances in a security list are isolated from hosts outside the list. You can override this default setting by creating security rules. Each security rule defines a specific communication path, which consists of a source, a destination, and a protocol-port combination over which communication is allowed.

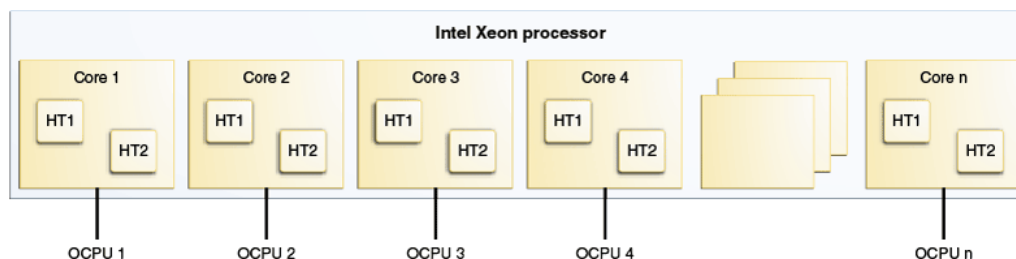
You can also add your instance to IP networks that you've created. An **IP network** allows you to define an IP subnet in your account. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network. These IP addresses aren't part of the common pool of Oracle-provided IP addresses used by the shared network. When you add an instance to an IP network, the instance is assigned an IP address in that subnet. You can assign IP addresses to instances either statically or dynamically, depending on your business needs. So you have complete control over the IP addresses assigned to your instances.

See [Configuring the Shared Network](#) and [Configuring IP Networks](#).

## About Shapes

A **shape** is a resource profile that specifies the number of OCPUs and the amount of memory to be allocated to an instance in Compute Classic. The shape determines the type of disk drive that your instance uses. If you select a general purpose or high-memory shape, a hard-disk drive is used. If you select a high I/O shape, an NVM Express SSD disk is automatically attached to your instance. For general purpose and high-memory shapes, you can select the block storage disk size, but for high I/O shapes, the size of the disk is determined by the shape.

When you select a shape, your instance is created with the corresponding number of Oracle Compute Units (OCPUs). An OCPU provides CPU capacity equivalent to one physical core of a processor with hyper threading enabled. Each OCPU corresponds to two hardware execution threads, known as vCPUs, as shown in the following figure.



A wide range of shapes is available to help you select a combination of processing power and memory for your instances that best suits your business requirement. The smallest general purpose shape provides 7.5 GB memory with a single OCPU. Larger shapes provide more OCPUs with correspondingly higher memory. If you need more memory per OCPU, select a high-memory shape.

If you require fast I/O access, select one of the high I/O shapes. An NVMe SSD disk is automatically attached to your instance with the device name `/dev/xvdz`. This is a local, nonpersistent NVMe SSD disk, which provides high I/O access rates. After your instance is created, you can mount this disk and format it as required. The size of this NVMe SSD disk is fixed depending on the selected shape.

#### Note:

Remember, when you select a high I/O shape, the NVMe SSD disk that is attached automatically is a nonpersistent disk. If you delete, shut down, or restart the instance or stop the instance orchestration, data stored on this disk is deleted.

- While selecting the shape for an instance, consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.
- Select a shape that meets the requirements of your workload with a sufficient buffer for intermittent spikes in the load. If you're not sure what shape is appropriate for an instance, then start small, experiment with a representative workload, and then settle on a shape. This approach may help you achieve an optimal trade-off between resource allocation and performance.

The following tables list the shapes that are currently available in Compute Classic.

#### General Purpose Shapes

Shape	OCPUs	vCPUs	Memory (GB)
OC3	1	2	7.5
OC4	2	4	15
OC5	4	8	30
OC6	8	16	60
OC7	16	32	120
OC8	24	48	180

Shape	OCPUs	vCPUs	Memory (GB)
OC9	32	64	240

 **Note:**

Not all Compute Classic sites include all shapes. To check which shapes are available on your site, start the Create Instance wizard and view the list of shapes on the Shape page. See [Creating an Instance from the Instances Page](#). Alternatively, to get a list of shapes using the API, use the `GET / shape / method`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

### High-Memory Shapes

Shape	OCPUs	vCPUs	Memory (GB)
OC1M	1	2	15
OC2M	2	4	30
OC3M	4	8	60
OC4M	8	16	120
OC5M	16	32	240
OC8M	24	48	360
OC9M	32	64	480

 **Note:**

Not all Compute Classic sites include all shapes. To check which shapes are available on your site, start the Create Instance wizard and view the list of shapes on the Shape page. See [Creating an Instance from the Instances Page](#). Alternatively, to get a list of shapes using the API, use the `GET / shape / method`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

### High I/O Shapes

When you select a high I/O shape, an NVM Express solid-state drive (SSD) disk is attached to your instance. The size of the disk is determined by the shape you select.

 **Note:**

Remember, when you select a high I/O shape, the NVMe SSD disk that is attached automatically is a nonpersistent disk. If you delete, shut down, or restart the instance or stop the instance orchestration, data stored on this disk is deleted.

Shape	OCPUs	vCPUs	Memory (GB)	Size of SSD Disk (GB)
<b>OCIO1M</b>	1	2	15	400
<b>OCIO2M</b>	2	4	30	800
<b>OCIO3M</b>	4	8	60	1600
<b>OCIO4M</b>	8	16	120	3200
<b>OCIO5M</b>	16	32	240	6400

 **Note:**

Not all Compute Classic sites include high I/O shapes. To check whether your site offers high I/O shapes, start the Create Instance wizard and view the list of shapes on the Shape page. See [Creating an Instance from the Instances Page](#). Alternatively, to get a list of shapes using the API, use the `GET /shape/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

**GPU Optimized Shapes**

Shape	OCPUs	vCPUs	Memory (GB)	gpuS
<b>OCSG1-K80</b>	3	6	60	1
<b>OCSG1-M80</b>	3	6	60	1
<b>OCSG2-K80</b>	6	12	120	2
<b>OCSG2-M80</b>	6	12	120	2
<b>OCSG3-K80</b>	12	24	240	4
<b>OCSG3-M80</b>	12	24	240	4

 **Note:**

Not all Compute Classic sites include all shapes. To check which shapes are available on your site, start the Create Instance wizard and view the list of shapes on the Shape page. See [Creating an Instance from the Instances Page](#). Alternatively, to get a list of shapes using the API, use the `GET /shape/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Instance Life Cycle

A Compute Classic instance can have one of the following statuses:

- When you create an instance, the initial status is **Preparing**. Compute Classic allocates resources and prepares to create the instance.
- While the specified image is being installed, the state changes to **Initializing**.
- After the image is installed and the instance is starting, the status changes to **Starting**.
- When the instance is ready, the status changes to **Running**. When an instance is in the **Running** state, you can connect to it. You can also attach or detach storage volumes and security lists.
- When an instance is running, you can shut down the instance. Its status changes to **Stopping**. When the operation is completed, its status changes to **Stopped**. When an instance is shut down or stopped, you can either start the instance, or delete it.
- When an instance is running or shut down, you can delete the instance. Its status changes to **Deleting**. When the operation is completed, the instance is deleted.
- At times, an instance can have the **Error** status.

For example, when you create or re-create an instance by starting its orchestration, if some of the resources required to create the instance aren't available, then the status of the instance changes to **Error**.

### **WARNING:**

When you shut down or reboot an instance, you might lose data on any nonpersistent boot disks, including NVMe SSD disks, that are attached automatically as part of the high I/O shapes.

## Workflow for Creating Your First Instance

Compute Classic supports several workflows for creating instances and the associated networking and storage resources.

For example, you can create the required storage volumes first and then create the instances to which the storage volumes should be attached. Alternatively, you can create instances first and then create and attach the required storage volumes to the instances. Similarly, you can create security lists first and then create instances and add them to the security lists, or you can create the instances first and then create security lists and add instances to them.

The workflow for creating an instance also varies depending on the type of instance you want to create. Use the appropriate recommended workflow for creating an Oracle Linux, Oracle Solaris, or Windows instance.

### Topics

- [Workflow for Creating Your First Oracle Linux Instance](#)

- [Workflow for Creating Your First Oracle Solaris Instance](#) (Not available on Oracle Cloud at Customer)
- [Workflow for Creating Your First Windows Instance](#) (Not available on Oracle Cloud at Customer)

## Workflow for Creating Your First Oracle Linux Instance


Here's a simple workflow that you can use to create your first instance.

1. Generate SSH key pairs. See [Generating an SSH Key Pair](#).
2. Sign in to Compute Classic. See [Accessing Compute Classic Using the Web Console](#).
3. Add the SSH public keys. See [Adding an SSH Public Key](#).
4. Create an instance using the web console. See [Creating an Instance from the Instances Page](#).

After creating the instance, you can do the following:

- Create and attach storage volumes. See [Creating a Storage Volume](#) and [Attaching a Storage Volume to an Instance](#).
- Add your instance to a security rule to control network access to the instance. See [Managing Security Rules for IP Networks](#).
- Access your instance securely by using SSH. See [Accessing an Oracle Linux Instance Using SSH](#).

### See Also:

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestrations v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Workflow for Creating Your First Oracle Solaris Instance



This topic does not apply to Oracle Cloud at Customer.

Here's a simple workflow that you can use to create your first Oracle Solaris instance.


1. Generate SSH key pairs. See [Generating an SSH Key Pair](#).
2. Sign in to Compute Classic. See [Accessing Compute Classic Using the Web Console](#).
3. Add the SSH public keys. See [Adding an SSH Public Key](#).
4. Create an instance using the web console. See [Creating an Instance from the Instances Page](#).

After creating the instance, you can do the following:



- Create and attach storage volumes. See [Creating a Storage Volume](#) and [Attaching a Storage Volume to an Instance](#).
- Add your instance to a security rule to control network access to the instance. See [Managing Security Rules for IP Networks](#).
- Access your instance securely by using SSH. See [Accessing an Oracle Solaris Instance Using SSH](#).

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestrations v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Workflow for Creating Your First Windows Instance



This topic does not apply to Oracle Cloud at Customer.

Here's a simple workflow that you can use to create your first Windows instance.

1. Sign in to Oracle Cloud Marketplace at <https://cloud.oracle.com/marketplace/faces/homePage.jspx> and select the Windows image that you want.
2. Think of a strong password for the Administrator of your Windows instance and keep the password handy. Ensure that the password meets the Windows password complexity requirements. Refer to the Windows server documentation. You'll need to set this password when you select a Windows image in Oracle Cloud Marketplace.
3. Click **Get App** and follow the process to create an instance using the web console. See [Creating an Instance from the Instances Page](#).

 **Note:**

The custom attributes required to specify RDP and the Administrator password that you provided earlier are pre-populated in the Create Instance wizard. You can specify a different password in the **Custom Attributes** field while creating the instance. If you want to add other users to your Windows instance and enable RDP access for them, then enter the list of users and passwords. See [User Data Attributes Used on Windows Instances](#).


After creating the instance, you can do the following:

- You should change the Administrator password when you log in to your instance the first time. You can also add additional administrators and users who are enabled for remote access, so that even if you lose or forget the Administrator password, you don't get locked out of your instance. If your instance uses a

persistent boot disk, any instance configuration, including tasks such as adding users or changing passwords, will be retained as long as the boot disk isn't deleted. However, if you're using a nonpersistent boot disk with your Windows instance, then if you terminate the orchestration and start it again later, the Administrator password will be reset to the password that you specified in the orchestration. This is true for any user password that you specify in an orchestration.

- Configure the security policy on your instance as required. For information about Windows security policies, see the Microsoft Windows Server documentation: <https://technet.microsoft.com/en-us/library/dn452420%28v=ws.11%29.aspx>
- Create and attach storage volumes. See [Creating a Storage Volume](#) and [Attaching a Storage Volume to an Instance](#).
- Add your instance to a security rule to control network access to the instance. See [Managing Security Rules for IP Networks](#).
- Create a security rule to enable RDP access and access your instance securely. See [Accessing a Windows Instance Using RDP](#).
- Create other Windows instances. After you've selected a Windows image from Oracle Cloud Marketplace and added it to your account, the Windows machine image is added to the list of images available while creating an instance or while creating a bootable storage volume. You can then directly select this image to create another Windows instance or a bootable storage volume. See [Creating an Instance from the Instances Page](#) and [Creating a Bootable Storage Volume](#).

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: Creating Instances Using Orchestration v1](#)

## Selecting a Method of Creating Instances

You can create Compute Classic instances in the following ways.

Task	Information
Quickly create an instance with a default configuration using the web console.	<a href="#">Creating an Instance Using QuickStarts</a>
Create an instance using the Create Instance wizard from the Instances page of the web console, using one of the following: <ul style="list-style-type: none"> <li>• An Oracle-provided machine image</li> <li>• A custom machine image</li> <li>• A machine image from Oracle Cloud Marketplace (Not available on Oracle Cloud at Customer)</li> </ul>	<a href="#">Creating an Instance from the Instances Page</a>

Task	Information
Create an instance using a non-persistent boot disk, configure the instance, and take a snapshot of the instance. Register this snapshot as a private machine image and use it to create instances.	<a href="#">Cloning an Instance by Using Instance Snapshots</a>
Select a custom machine image that you've already created, uploaded, and registered with Compute Classic from the Images page of the web console and use it to launch the Create Instance wizard.	<a href="#">Creating an Instance Using a Private Image</a>
Define one or more instances and other objects offline in a JSON-formatted file, upload the orchestration to Compute Classic, and then start the orchestration using the web console.	<a href="#">Creating Instances Using Orchestrations</a>
Instead of defining the orchestration in a JSON-formatted file and then uploading the orchestration to Compute Classic, you can create a blank orchestration, and then add objects to it by updating the orchestration. While updating the orchestration, you can define attributes for a single instance or create complex topologies that consist of multiple instances and multiple networks.	<a href="#">Creating an Instance Using a Blank Orchestration v2</a>
Create an instance using Visual Object Editor by using Oracle-provided images or private images.	<a href="#">Creating an Instance Using Visual Object Editor</a>
Specify one or more instances by starting an orchestration using the CLI.	See the workflow <a href="#">Preparing to Use the Compute Classic CLI</a> in <i>CLI Reference for Oracle Cloud Infrastructure Compute Classic</i>
Specify one or more instances by creating and managing orchestrations using the REST API.	<a href="#">Creating Instances Using an Orchestration in REST API for Oracle Cloud Infrastructure Compute Classic</a>
Create one or more instances by specifying instance creation parameters in a launch plan using the REST API.	<a href="#">Creating Instances Using Launch Plans</a>

## Creating an Instance Using QuickStarts

You can use QuickStarts to quickly create an instance using a default configuration.

### Prerequisites

- If you want to access this instance from a Windows host using PuTTY, you must generate an SSH key pair and upload the SSH public key to your Compute Classic account. To generate an SSH key pair using PuTTY, see the section [Generating an SSH Key Pair](#) in [Tutorial: Creating an Oracle Linux Instance Using the Oracle Cloud Infrastructure Compute Classic Web Console](#). To upload the SSH public key to your account, see [Adding an SSH Public Key](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to

ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

- Create an IP network to which you want to attach your instance or you can use the default IP network. See [Creating an IP Network](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, click **Create Instance**.  
The QuickStarts page appears.
3. Specify a name for the instance or use the default value. Names can contain only alphanumeric characters, hyphens, and underscores. The first and last characters cannot be hyphen or underscore.
4. Select the appropriate image. If you select an image from Oracle Cloud Marketplace, accept the terms and wait for the app to be installed before continuing.
5. Select the IP network that you want to add your instance to.
6. Select an existing SSH public key, or create an SSH key pair and download the SSH private key. Save the private key on your local system and make a note of the path where you've saved it. You'll need the private key later, when you log in to the instance.

#### Note:

If you download the SSH key, you can use it to access your instance from UNIX and UNIX-like systems. If you want to access your instance by using PuTTY on Windows, generate the SSH key using PuTTY and add the SSH public key to your Compute Classic account.

#### Note:

If you select a Windows image, you don't need an SSH key. Enter the administrator password instead. You'll use this password to access the Windows instance using RDP later.

7. Click **Create**.

The QuickStarts instance is created. While the instance is being created, you can monitor the corresponding orchestration on the Orchestration page. When the instance is created, it is listed on the Instances page.

A QuickStarts instance is created with the following general configuration:

- Uses a persistent boot disk. You won't be able to create an instance snapshot of this instance.
- In sites which support public IP address reservations, security rules, and access control lists in IP networks:

- Doesn't have an interface on the shared network.
- (Available only on Oracle Cloud at Customer) Has one interface on an IP network with the same name and is added to a vNICset of the same name.
- (Not available on Oracle Cloud at Customer) Has one interface on the default IP network and is added to a vNICset of the name which is same as the name of the instance.
- Has one IP address from the `/oracle/public/cloud-ippool` IP address pool and another IP address from the `/oracle/public/public-ippool` IP address pool.
- Has the required security rules and ACL set up to enable SSH or RDP access to the instance and all egress traffic.
- In sites which don't support public IP address reservations, security rules, and access control lists in IP networks:
  - Has an interface on the shared network.
  - Is added to the default security list.
  - Has a temporary public IP address.
  - Has the required security rule set up to enable SSH or RDP access.
- Is nonpersistent. This allows you to update the instance by suspending the corresponding orchestration v2. When the orchestration is suspended, the instance status changes to **Inactive** and you can update any attribute of the instance.
- Has persistence specified as true for all other objects.


After your instance is created, you can log in to your instance. See [Logging In to an Instance](#).

 **Tip:**

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

For security and patching-related guidelines, see the documentation for your operating system.

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Creating an Instance from the Instances Page

You can create a single instance using the Compute Classic web console. While creating an instance, you can specify persistent storage volumes to be associated with your instance. You can also enable access to your instance using SSH, or add your instance to a security list. If you add your instance to a security list, you can use that security list in security rules to control access to your instance.

When you create an instance using the Create Instance wizard, a single orchestration v2 is created automatically to manage the instance and its associated resources. Storage volumes and networking objects used by the instance are created in the same orchestration. Instances are nonpersistent by default. However, storage volumes and other objects are created with persistence set to true, so that if you suspend the orchestration, instances are shut down, but storage volumes aren't deleted. Terminating the orchestration, however, will cause all objects to be deleted and any data on storage volumes will be lost.

For more information about using orchestrations to manage your instances and other resources, see [Managing Resources Using Orchestrations v2](#).

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

3. On the Instances page, click **Create Instance**.

The QuickStarts page appears.

4. On the QuickStarts page:
  - To quickly create an instance using the default configuration, select an image on the QuickStarts page, select or create an SSH key (or, for a Windows instance, specify the administrator password), select the IP network that you want to add your instance to, and then click **Create**. The instance is created.
  - To select a different image, click **Show All Images**. The Images page of the Create Instance wizard is displayed.
  - To customize one of the QuickStart images with a specific configuration, click **Customize**. The image is selected and the Shapes page of the Create Instance wizard is displayed.

5. On the Image page, select the image you want to use. The image specifies the operating system and disk size of the instance. If there are multiple entries in an image list, the latest entry is selected by default. You can select an earlier entry from the drop-down list. For information about machine images, see [Managing Machine Images](#).
  - To use an Oracle-provided image, select it from the **Oracle Images** tab. Where there are multiple image entries in an image list, the most recent entry is selected by default. You can select an older machine image from the drop-down list, if required. Machine images added from the 16.3.6 release onwards include the release version in the image list entry. If you want to add your instance to an IP network, select a machine image with version 16.3.6 or later.
  - To use a custom image, click the **Private Images** tab in the left pane and select the required image. The private images listed here include your custom images as well as images that you've previously selected from Oracle Cloud Marketplace and associated with your account.
  - To use an image from Oracle Cloud Marketplace, click the **Marketplace** tab in the left pane and select the required image. (Not available on Oracle Cloud at Customer) You can filter the Oracle Cloud Marketplace images by name, price, or category using the appropriate menu. Select an image and read and accept the terms of use to continue. Also enter any additional information or custom attributes for the image, if required.

After selecting an image, you can click **Review and Create** to accept the default settings and create your instance.

 **Note:**

If you select an image and then accept the default settings, your instance is created with the following configuration:

- Uses the smallest applicable shape.
- Uses the default name and label.
- Uses a persistent boot disk. You won't be able to create an instance snapshot of this instance.
- In sites which support public IP address reservations, security rules, and access control lists in IP networks, the instance is added to the default IP network:
  - Doesn't have an interface on the shared network.
  - (Available only on Oracle Cloud at Customer) Has one interface on an IP network with the same name and is added to a vNICset of the same name.
  - (Not available on Oracle Cloud at Customer) Has one interface on the default IP network and is added to a vNICset of the name which is same as the name of the instance.
  - Has one IP address from the `/oracle/public/cloud-ippool` IP address pool and another IP address from the `/oracle/public/public-ippool` IP address pool.
  - Has the required security rules and ACL set up to enable SSH or RDP access to the instance and all egress traffic.
- In sites which don't support public IP address reservations, security rules, and access control lists in IP networks:
  - Has an interface on the shared network.
  - Is added to the default security list.
  - Has a temporary public IP address.
  - Has the required security rule set up to enable SSH or RDP access.
- Is nonpersistent. This allows you to update the instance by suspending the corresponding orchestration v2. When the orchestration is suspended, the instance status changes to **Inactive** and you can update any attribute of the instance.
- Has persistence specified as true for all other objects.
- Doesn't have a description, tags, a DNS host name prefix, or custom attributes (unless specified in the image).
- Doesn't have any SSH keys associated with it.



 **Caution:**

If you accept the default settings and click Review and Create, your instance won't have any SSH keys associated with it. This means that you won't be able to access the instance using SSH. An SSH key is **required** when you create an Oracle Linux or an Oracle Solaris instance, because you must use SSH to access these instances. Go to the Instance page of the Create Instance wizard to specify an SSH public key to be associated with your instance.

If you want to customize your instance configuration after selecting an image, click the button to go to the next page.

6. On the Shape page, select the shape that you want to use. The shape specifies the OCPU and memory resources to be allocated to the instance. If you select a high I/O shape, an NVMe SSD disk is automatically attached to your instance. This is a local, nonpersistent NVMe SSD disk, which provides high I/O access rates. This disk is attached to your instance with the device name `/dev/xvdz`. After your instance is created, you can mount this disk and format it as required. The size of this NVMe SSD disk is fixed depending on the selected shape.

 **Note:**

High I/O shapes aren't available in all regions.

For more information about shapes, see [About Shapes](#).

After selecting a shape, click the button to go to the next page.

7. On the Instance page, select or enter the following, and then click the button to go to the next page:
  - **Persistent:** (Available only on Oracle Cloud at Customer)

Because instances are nonpersistent by default, the instance is deleted when you suspend the relevant orchestration. This allows you to update the properties of the instance by suspending the orchestration. If a persistent boot disk is attached to the instance, no data or configuration is lost. After updating the properties of the instance, you can recreate it by starting the orchestration.

If you use this option to specify that the instance must be persistent, then, if you want to update the instance later on without stopping any of the other objects created by this orchestration, remember to update the instance to be nonpersistent at that time.
  - **Placement:** As there is a single domain, the instance is created in this domain when you select **Auto** or **Specific Domain**.
  - **Name:** Enter a name for your instance, or accept the default.

Note that the full name of an instance has the following format: `/Compute-identity_domain/user/name/id`. Here *id* is an autogenerated ID.

**Examples of Instance Names:**

- If you accept the default value suggested in the Create Instance wizard (for example, 20160422104055):

```
/Compute-myDomain/jack/20160422104055/300a7479-ec90-4826-98b9-a725662628f1
```

- If you specify a name in the Create Instance wizard (for example, vm1 ) :

```
/Compute-myDomain/jack/vm1/300a7479-ec90-4826-98b9-a725662628f1
```

- **Label:** Enter a label for the instance or accept the default. A label can contain only alphanumeric characters, hyphens, and underscores. It can't contain unicode characters.

Enter a label that's meaningful and that you can use to identify the instance easily later. Try to assign a unique label for each instance. This label is displayed on the instance details page.

- **Description:** (Optional) Enter a description.
- **Tags:** (Optional) Specify one or more tags to help you identify and categorize the instance.
- **SSH Keys:** (Optional) Specify the SSH keys that you want to associate with this instance. Click this field or start typing to see a list of available SSH public keys.

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

To add a new SSH public key:

- a. Click **Add SSH Public Key**.
- b. Enter a name for the SSH public key.
- c. In the **Value** field, click **Select File**. Navigate to the path where your SSH key is saved, and select the SSH public key file that you want to add. The value of the SSH key appears in the field.

Alternatively, you can paste the value of the SSH public key that you want to add.

 **Important:**

Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks.

- d. Click **Add**.

The SSH public key is added and appears in the list of SSH keys that you want to associate with the instance.

 **Tip:**

The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at `http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key`.

- Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the `authorized_keys` file of the `opc` user.
- In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the `authorized_keys` file of the appropriate users.

- **RDP:** This field is displayed when you select a Windows image. Retain the default, **Enabled**, if you want to use RDP to access your Windows instance. If you select **Disabled**, you won't be able to access your Windows instance using RDP.
- **Administrator Password:** This field is displayed when you select a Windows image. The password displayed here is the password that you specified while getting the Windows image from Oracle Cloud Marketplace. You can retain this password, or specify a different password. You'll use this password to log in to your Windows instance as the Administrator. Ensure that the password meets the Windows password complexity requirements. See the Windows Server documentation.
- **Custom Attributes:** Enter any additional attributes that you want to store on the instance. This field allows you to customize your instance by providing additional information specific to each instance. You can enter arbitrary key-value pairs in plain text. The text you enter here *must* be in JSON format. This information is stored as user data on your instance.

If you're creating a Windows instance, the following required attributes are pre-populated for you.

```
{
  "enable_rdp": true,
  "administrator_password": "Specify_password_here"
}
```

The Administrator password that appears here by default is the password that you specified when you selected the Windows image from Oracle Cloud Marketplace. You can specify a different Administrator password for your instance, if required. You can edit these attributes either directly in this field, or by modifying the **RDP** and **Administrator Password** fields above.


On Oracle Linux instances created using Oracle-provided images with the release version 16.4.6 or later, the OPC Agent is installed and enabled by default. This agent collects and reports memory utilization metrics on your instance. If you want to disable this agent, enter the `opc_guest_agent_enabled` attribute set to `false`.

For information about user-defined attributes that can be used to automate instance configuration, see [Automating Instance Initialization Using opc-init](#).

 **Note:**

Solaris machine images don't include the opc-init scripts. So you can't use opc-init to automate instance configuration of Solaris instances. (Not available on Oracle Cloud at Customer)

After the instance is created, the attributes that you specify here are available within the instance at `http://192.0.0.192/latest/user-data`. For information about retrieving user data, see [Retrieving User-Defined Instance Attributes](#).

8. On the Network page, select or enter the required network settings, and then click the button to go to the next page:
  - **DNS Hostname Prefix:** (Optional) Specify a DNS host name prefix. The host name is visible internally within your DNS space. It is referenced by other instances in the domain, as well as by the OS and applications running on your instance. The host name that you specify is suffixed by the domain name. If you don't specify a host name, then a host name is generated automatically.
  - **Network Options:** (Optional) From the 16.3.6 release onwards, Oracle-provided Oracle Linux and Windows machine images support up to eight interfaces (eth0 to eth7). If you use a private image, you can set up the image to support multiple interfaces. When you create an instance using these images, you can add each instance to up to eight different networks, including the shared network. To configure the shared network or IP networks, select the appropriate option.
    - (Available only on Oracle Cloud at Customer) If you select the **IP Network** option, you must click **Configure Interface** to configure IP network interfaces to enable access to your instance and specify one of the IP network interfaces as the default gateway for the instance. If you don't click **Configure Interface** to specify an IP network, then the instance is added to the default security list in the shared network.
    - (Not available on Oracle Cloud at Customer) If you select the **IP Network** option, then the eth0 interface of the instance is added to the default IP network. You can accept the default values or if you want to specify the IP network options, from the  menu, select **Update**.

Click **Configure Interface** to configure the other network interfaces of the instance.
    - If you don't want to access the instance on the shared network, ensure that the **Shared Network** option is deselected. If you don't select the **Shared Network** option, your instance isn't added to the shared network.
    - If you don't select the **IP Network** option, your instance isn't added to any IP network. If you don't select either option, your instance isn't added to any IP network but it is added to the shared network. It is added to the default security list. This enables you to access the instance on the shared network after the instance is created. However, no public IP address is associated with it, so if you want to access the instance from the public

Internet, you must first associate a public IP address with it. The shared network interface is automatically configured as the default gateway for the instance.

## 9. IP Network Options

Click **Configure Interface**. In the Configure IP Network Interface dialog box, enter the following information and then click **Save**.

- **Interface:** Select the interface that you want to add to the IP network. After you select all the interfaces that you want to add to IP networks, the first available interface is assigned to the shared network. You can't add, delete, or modify interface allocations after an instance is created.
- **vNIC Name:** Retain the default vNIC name or enter another name. The three-part vNIC name is generated using this name. It has the format */Compute-identity\_domain/username/instanceName\_vnicName*.

If you enter a vNIC name, ensure that the name is unique to the site.

- **IP Network:** Specify the IP network that you want to add this interface to. When you add an instance to an IP network, the specified interface of the instance is assigned an IP address on the specified IP network. After the instance is created, you can view information about each interface on the Instance Details page.

If you haven't created the IP network that you want to add your instance to, you can do so now. Click **Create IP Network**. Enter a name and the IP address prefix for the IP network, select an IP network exchange that you want to add the IP network to, if any. Then click **Create**. The IP network is created and selected in the list of IP networks that you want to add your instance to.

If an IP network belongs to an IP network exchange and if you have specified a host name, then that host name is resolvable by all IP networks connected to the IP network exchange.

The same DNS server will be used that is part of the IP network and will be setup by DHCP automatically.

- **Static IP Address:** Specify a private IP address for this interface. The private IP address must be unused and it must belong to the subnet of the selected IP network. Remember, too, that certain IP addresses in a subnet are reserved. For example, the first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.

If no static IP address is specified, an IP address from the specified IP network is allocated dynamically, when the instance is created. Dynamically allocated IP addresses might change if the instance is deleted and re-created.

Dynamic IP addresses are allocated from the lowest IP address in the range upwards. For example, if your IP network subnet is 192.168.1.0/25, dynamic allocation of IP addresses would start with 192.168.1.2 (as the first two IP addresses in the range, 192.168.1.0 and 192.168.1.1, are reserved).

To ensure that a static IP address that you've specified isn't already dynamically allocated, it is recommended that you specify static IP addresses from the end of your subnet range. For example, if your IP network subnet is 192.168.1.0/25, start allocating static IP addresses from 192.168.1.126 downwards (as the last IP address in the range, 192.168.1.127, is reserved).

- **Public IP Address:** Select an available IP reservation for IP networks. When the instance is created, you can configure security rules and access control

lists for your IP network to enable access to this IP address over the public Internet. If you don't select an IP reservation now, you can associate a public IP address with this interface later by creating or updating an IP reservation.

- **Cloud IP Address:** Select an available IP reservation from the cloud IP pool. When the instance is created, this IP address can be accessed by other Oracle Cloud services without being accessible over the public Internet. If you don't select a cloud IP address now, you can associate a cloud IP address with this interface later by creating or updating an IP reservation.
- **MAC Address:** Specify the MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter 01:02:03:04:ab:cd as the MAC address but not 01-02-03-04-ab-cd. Ensure that the MAC addresses that you specify are unique within each IP network exchange and each IP network. If you specify a duplicate MAC address, each vNIC with that MAC address is disabled.
- **Virtual NIC Sets:** Select the vNICsets that you want to add this interface to. Each interface is added to the default vNICset by default. If you select other vNICsets to add this interface to, you can remove it from the default vNICset. However, ensure that you add each interface to at least one vNICset, to enable communication to that interface. After the instance is created, communication with each vNIC depends on the vNICsets it belongs to and the access control lists that apply to each vNICset. While creating an instance, you can add a vNIC to up to 4 vNICsets. To add a vNIC to more than 4 vNICsets, update the required vNICsets after the instance is created. You can also remove vNICs from a vNICset after the instance is created.
- **DNS:** Enter the DNS A record names for the instance. You can specify up to eight DNS A record names for each interface on an IP network. These names can be queried by instances on any IP network in the same IP network exchange. If no static IP address is specified for the interface, an IP address on the specified IP network is assigned automatically. After the instance is launched, the defined names are associated with the IP address that was automatically allocated to the interface.
- **Name Servers:** Enter the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.
- **Search Domains:** Enter the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.
- **Default Gateway:** Select this option if you want to use this interface as the default gateway. All network traffic uses the specified default gateway, unless a different interface is explicitly configured for an application within the instance.

## 10. Shared Network Options

- **Public IP Address:**
  - If you want to connect to this instance over the Internet, then select either **Auto Generated** or **Persistent Public IP Reservation**.
  - If you select an autogenerated public IP address, the IP address persists while the instance is running, but will change if you delete the instance and create it again later.

- To associate a permanent public IP address with the instance, select **Persistent Public IP Reservation**.
- If you've already created an IP address reservation, select it from the list. Otherwise, to create one now, click **Create IP Reservation**. In the Create Public IP Reservation dialog box, enter a name for the IP reservation and then click **Create**. The IP reservation is created. Select this IP reservation from the list of persistent public IP reservations.
- If you don't want your instance to be accessed over the Internet, then you don't need to associate a public IP address with it. In the **Public IP Address** list, select **None**.

For more information about public IP addresses, see [About Public IP Addresses](#).

- **Security Lists:**
  - (Optional) Specify the security lists that you want to add the instance to. Click this field or start typing to see a list of available security lists. When you add an instance to a security list, you can control access to or from this instance by creating security rules that use the specified security list as a source or destination.
  - (Optional) If you haven't created the security lists that you want to use, you can do so now. Click **Create Security List**. Enter a name for the security list and optionally enter a description, and then click **Create**. The security list is created and appears in the list of security lists that you want to add your instance to.

If you don't specify any security list, the instance is added to the default security list, `default/default`. You can use this security list as a source or destination in security rules that you create. If any existing security rules already specify this security list as a source or destination, those rules will apply to this instance when the instance is created. If you don't want to use those security rules with this instance, or if you want to configure access to this instance separately later on, remember to remove the instance from the `default/default` security list after it is created. See [Removing an Instance from a Security List](#).

For more information about security lists, see [About Security Lists](#).

11. The Storage page shows the persistent boot disk that will be created and used to boot your instance. You can retain this setting and attach additional storage volumes later, when the instance is running. Alternatively, you can update or remove the persistent boot disk that is created by default or attach additional boot or data disks now.

You can attach existing storage volumes to your instance, if required, or create storage volumes and attach them to the instance.

- **To attach storage volumes that you've already created:**  
Click **Attach Existing Volume**.

 **Note:**

You can't detach storage volumes that are attached during instance creation.

In the Attach Existing Storage Volume dialog box, select or enter the following and then click **Add**:

- **Attach Storage Volume:** Select the storage volume that you want to attach.

 **Note:**

A storage volume must be in the online state before it can be attached to an instance. If a storage volume is already attached to another instance or if it is in an error state, it's not displayed in the **Attach Storage Volume** list.

- **Attach as Disk #:** Enter a disk index number. The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on. Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.
- **Boot Drive:** Select this option to use the specified storage volume as the boot disk. The storage volume you select here must have the same image as the image that you selected on the Image page of the Create Instance wizard.

- **To create a storage volume and attach it to the instance:**  
Click **Add New Volume**.

In the Add New Storage Volume dialog box, select or enter the following and then click **Add**:

- **Name:** Enter a name for this storage volume.
- **Size:** The size is set automatically to accommodate the disk size that's specified in the image that you selected earlier. If you want a larger boot disk than that specified in the image, then enter a larger size.

 **Note:**

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

- **Storage Property:** Select a storage property.

Based on your latency and IOPS requirements, select one of the following storage properties.

Storage Property	Latency	Throughput
storage/default	Standard	Standard
storage/latency	Low	High
storage/ssd/gpl	Lowest	Highest





 **Note:**

SSD storage volumes aren't available in all sites.

The web console might show other storage properties. Don't select any of them.

- **Description:** (Optional) Enter a description.
- **Attach as Disk #:** Accept the default disk number or enter a higher number. The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on. Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.
- **Boot Drive:** Select this option to use the specified storage volume as the boot disk. When you select this option, the disk number is automatically set to **1**.
- **To customize the persistent boot disk that is created by default:**

From the  menu, select **Update**. In the Update Storage Volume dialog box, you can modify the disk size, storage property, or the description. You can also uncheck the **Boot Drive** option. The persistent storage volume will still be created when the instance is created, but it won't be used to boot the instance. Instead, a nonpersistent boot disk will be created and used to boot the instance.
- **To create an instance that uses a nonpersistent boot disk:**

Remove the default boot disk. From the  menu, select **Remove**. When you do this and you don't specify another persistent storage volume as the boot drive, a nonpersistent boot disk is used to boot your instance.

 **Note:**

If you want to create an instance snapshot to use this instance as a template for creating other instances, remove the persistent boot disk while creating this instance and use a nonpersistent boot disk instead. You can't create an instance snapshot of an instance that uses a persistent boot disk.

However, if you want to use a persistent boot disk to boot your instance, you can still use this boot disk as a template to create other instances by creating a storage volume snapshot of the boot disk and using that snapshot to create a new bootable storage volume. See [Backing Up and Restoring Storage Volumes Using Snapshots](#).

When you're done, click the button to go to the next page.

12. On the Review page, verify the information that you've entered, and then click **Create**.
13. Monitor the status of the instance.

- When you create an instance, the initial status is **Preparing**. Compute Classic allocates resources and prepares to create the instance.
- While the specified image is being installed, the state changes to **Initializing**.
- After the image is installed and the instance is starting, the status changes to **Starting**.
- When the instance is ready, the status changes to **Running**. When an instance is in the **Running** state, you can connect to it. You can also attach or detach storage volumes and security lists.
- When an instance is running, you can shut down the instance. Its status changes to **Stopping**. When the operation is completed, its status changes to **Stopped**. When an instance is shut down or stopped, you can either start the instance, or delete it.
- When an instance is running or shut down, you can delete the instance. Its status changes to **Deleting**. When the operation is completed, the instance is deleted.
- At times, an instance can have the **Error** status.

For example, when you create or re-create an instance by starting its orchestration, if some of the resources required to create the instance aren't available, then the status of the instance changes to **Error**.

 **Note:**

If you get an error message `Unable to place instance...` it indicates that the site you've selected doesn't have sufficient resources to create this instance. If your domain spans multiple sites, then use the **Site** menu near the top of the page to select another site and run the Create Instance wizard again.

After your instance is created, you can log in to your instance. See [Logging In to an Instance](#).


 **Tip:**

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

For Oracle-provided images, apply the necessary security patches and review the security configuration right after you create the instances, *before* deploying any applications.

For security and patching-related guidelines, see the documentation for your operating system.

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Creating an Instance Using a Private Image

You can build a custom image and use it to create Compute Classic instances.

When you create an instance using the Create Instance wizard, a single orchestration v2 is created automatically to manage the instance and its associated resources. Storage volumes and networking objects used by the instance are created in the same orchestration. Instances are nonpersistent by default. However, storage volumes and other objects are created with persistence set to true, so that if you suspend the orchestration, instances are shut down, but storage volumes aren't deleted. Terminating the orchestration, however, will cause all objects to be deleted and any data on storage volumes will be lost.

For more information about using orchestrations to manage your instances and other resources, see [Managing Resources Using Orchestration v2](#).

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- The custom machine image that you want to use must already be available as a machine image in Compute Classic. See [Workflow for Creating Instances Using a Private Machine Image](#) for information about creating, uploading, and registering your custom machine images.


### Procedure

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

3. Click the **Images** tab.

The Private Images page is displayed.

4. The Private Images page lists images you've created as well as images you've got from Oracle Cloud Marketplace. Go to the image that you want to use, and from the  menu, select **Create Instance**.

The Create Instance wizard starts.

5. The Image page of the Create Instance wizard shows the custom image that you selected. After selecting this image, you can click **Review and Create** to accept the default settings and create your instance.

 **Note:**

If you select an image and then accept the default settings, your instance is created with the following configuration:

- Uses the smallest applicable shape.
- Has the high-availability (HA) policy set to Active.
- Uses the default name and label.
- Uses a persistent boot disk. You won't be able to create an instance snapshot of this instance.
- Has an autogenerated public IP address.
- Is added to the default security list.
- Isn't added to any IP networks.
- Doesn't have a description, tags, a DNS host name prefix, or custom attributes (unless specified in the image).
- Doesn't have any SSH keys associated with it.

 **Caution:**

If you accept the default settings and click Review and Create, your instance won't have any SSH keys associated with it. This means that you won't be able to access the instance using SSH. An SSH key is **required** when you create an Oracle Linux or an Oracle Solaris instance, because you must use SSH to access these instances. Go to the Instance page of the Create Instance wizard to specify an SSH public key to be associated with your instance.

6. On the Shape page, select the shape that you want to use. The shape specifies the OCPU and memory resources to be allocated to the instance. If you select a high I/O shape, an NVMe SSD disk is automatically attached to your instance. This is a local, nonpersistent NVMe SSD disk, which provides high I/O access rates. This disk is attached to your instance with the device name `/dev/xvdz`. After your instance is created, you can mount this disk and format it as required. The size of this NVMe SSD disk is fixed depending on the selected shape.

 **Note:**

High I/O shapes aren't available in all regions.

For more information about shapes, see [About Shapes](#).

After selecting a shape, click the button to go to the next page.

7. On the Instance page, select or enter the following, and then click the button to go to the next page:

- **Persistent:** (Available only on Oracle Cloud at Customer)

Because instances are nonpersistent by default, the instance is deleted when you suspend the relevant orchestration. This allows you to update the properties of the instance by suspending the orchestration. If a persistent boot disk is attached to the instance, no data or configuration is lost. After updating the properties of the instance, you can recreate it by starting the orchestration.

If you use this option to specify that the instance must be persistent, then, if you want to update the instance later on without stopping any of the other objects created by this orchestration, remember to update the instance to be nonpersistent at that time.

- **Placement:** As there is a single domain, the instance is created in this domain when you select **Auto** or **Specific Domain**.
- **Name:** Enter a name for your instance, or accept the default.

Note that the full name of an instance has the following format: `/Compute-identity_domain/user/name/id`. Here `id` is an autogenerated ID.

**Examples of Instance Names:**

- If you accept the default value suggested in the Create Instance wizard (for example, 20160422104055):

```
/Compute-myDomain/jack/20160422104055/300a7479-ec90-4826-98b9-a725662628f1
```

- If you specify a name in the Create Instance wizard (for example, `vm1`) :

```
/Compute-myDomain/jack/vm1/300a7479-ec90-4826-98b9-a725662628f1
```

- **Label:** Enter a label for the instance or accept the default. A label can contain only alphanumeric characters, hyphens, and underscores. It can't contain unicode characters.

Enter a label that's meaningful and that you can use to identify the instance easily later. Try to assign a unique label for each instance. This label is displayed on the instance details page.

- **Description:** (Optional) Enter a description.
- **Tags:** (Optional) Specify one or more tags to help you identify and categorize the instance.
- **SSH Keys:** (Optional) Specify the SSH keys that you want to associate with this instance. Click this field or start typing to see a list of available SSH public keys.

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

To add a new SSH public key:

- a. Click **Add SSH Public Key**.
- b. Enter a name for the SSH public key.
- c. In the **Value** field, click **Select File**. Navigate to the path where your SSH key is saved, and select the SSH public key file that you want to add. The value of the SSH key appears in the field.

Alternatively, you can paste the value of the SSH public key that you want to add.

 **Important:**

Paste the key value exactly as it was generated. Don't append or insert any spaces, characters, or line breaks.

- d. Click **Add**.

The SSH public key is added and appears in the list of SSH keys that you want to associate with the instance.

 **Tip:**

The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at `http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key`.

- Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the `authorized_keys` file of the `opc` user.
- In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the `authorized_keys` file of the appropriate users.

- **RDP:** This field is displayed when you select a Windows image. Retain the default, **Enabled**, if you want to use RDP to access your Windows instance. If you select **Disabled**, you won't be able to access your Windows instance using RDP.
- **Administrator Password:** This field is displayed when you select a Windows image. The password displayed here is the password that you specified while getting the Windows image from Oracle Cloud Marketplace. You can retain this password, or specify a different password. You'll use this password to log

in to your Windows instance as the Administrator. Ensure that the password meets the Windows password complexity requirements. See the Windows Server documentation.

- **Custom Attributes:** Enter any additional attributes that you want to store on the instance. This field allows you to customize your instance by providing additional information specific to each instance. You can enter arbitrary key-value pairs in plain text. The text you enter here *must* be in JSON format. This information is stored as user data on your instance.

If you're creating a Windows instance, the following required attributes are pre-populated for you.

```
{
    "enable_rdp": true,
    "administrator_password": "Specify_password_here"
}
```

The Administrator password that appears here by default is the password that you specified when you selected the Windows image from Oracle Cloud Marketplace. You can specify a different Administrator password for your instance, if required. You can edit these attributes either directly in this field, or by modifying the **RDP** and **Administrator Password** fields above.

On Oracle Linux instances created using Oracle-provided images with the release version 16.4.6 or later, the OPC Agent is installed and enabled by default. This agent collects and reports memory utilization metrics on your instance. If you want to disable this agent, enter the `opc_guest_agent_enabled` attribute set to `false`.

For information about user-defined attributes that can be used to automate instance configuration, see [Automating Instance Initialization Using opc-init](#).


 **Note:**

Solaris machine images don't include the `opc-init` scripts. So you can't use `opc-init` to automate instance configuration of Solaris instances. (Not available on Oracle Cloud at Customer)

After the instance is created, the attributes that you specify here are available within the instance at `http://192.0.0.192/latest/user-data`. For information about retrieving user data, see [Retrieving User-Defined Instance Attributes](#).

8. On the Network page, select or enter the required network settings, and then click the button to go to the next page:
  - **DNS Hostname Prefix:** (Optional) Specify a DNS host name prefix. The host name is visible internally within your DNS space. It is referenced by other instances in the domain, as well as by the OS and applications running on your instance. The host name that you specify is suffixed by the domain name. If you don't specify a host name, then a host name is generated automatically.
  - **Network Options:** (Optional) From the 16.3.6 release onwards, Oracle-provided Oracle Linux and Windows machine images support up to eight

interfaces (eth0 to eth7). If you use a private image, you can set up the image to support multiple interfaces. When you create an instance using these images, you can add each instance to up to eight different networks, including the shared network. To configure the shared network or IP networks, select the appropriate option.

- (Available only on Oracle Cloud at Customer) If you select the **IP Network** option, you must click **Configure Interface** to configure IP network interfaces to enable access to your instance and specify one of the IP network interfaces as the default gateway for the instance. If you don't click **Configure Interface** to specify an IP network, then the instance is added to the default security list in the shared network.
- (Not available on Oracle Cloud at Customer) If you select the **IP Network** option, then the eth0 interface of the instance is added to the default IP network. You can accept the default values or if you want to specify the IP network options, from the  menu, select **Update**.

Click **Configure Interface** to configure the other network interfaces of the instance.

- If you don't want to access the instance on the shared network, ensure that the **Shared Network** option is deselected. If you don't select the **Shared Network** option, your instance isn't added to the shared network.
- If you don't select the **IP Network** option, your instance isn't added to any IP network. If you don't select either option, your instance isn't added to any IP network but it is added to the shared network. It is added to the default security list. This enables you to access the instance on the shared network after the instance is created. However, no public IP address is associated with it, so if you want to access the instance from the public Internet, you must first associate a public IP address with it. The shared network interface is automatically configured as the default gateway for the instance.

## 9. IP Network Options

Click **Configure Interface**. In the Configure IP Network Interface dialog box, enter the following information and then click **Save**.

- **Interface:** Select the interface that you want to add to the IP network. After you select all the interfaces that you want to add to IP networks, the first available interface is assigned to the shared network. You can't add, delete, or modify interface allocations after an instance is created.
- **vNIC Name:** Retain the default vNIC name or enter another name. The three-part vNIC name is generated using this name. It has the format `/Compute-identity_domain/username/instanceName_vnicName`.

If you enter a vNIC name, ensure that the name is unique to the site.

- **IP Network:** Specify the IP network that you want to add this interface to. When you add an instance to an IP network, the specified interface of the instance is assigned an IP address on the specified IP network. After the instance is created, you can view information about each interface on the Instance Details page.

If you haven't created the IP network that you want to add your instance to, you can do so now. Click **Create IP Network**. Enter a name and the IP address prefix for the IP network, select an IP network exchange that you want



to add the IP network to, if any. Then click **Create**. The IP network is created and selected in the list of IP networks that you want to add your instance to.

If an IP network belongs to an IP network exchange and if you have specified a host name, then that host name is resolvable by all IP networks connected to the IP network exchange.

The same DNS server will be used that is part of the IP network and will be setup by DHCP automatically.

- **Static IP Address:** Specify a private IP address for this interface. The private IP address must be unused and it must belong to the subnet of the selected IP network. Remember, too, that certain IP addresses in a subnet are reserved. For example, the first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.

If no static IP address is specified, an IP address from the specified IP network is allocated dynamically, when the instance is created. Dynamically allocated IP addresses might change if the instance is deleted and re-created.

Dynamic IP addresses are allocated from the lowest IP address in the range upwards. For example, if your IP network subnet is 192.168.1.0/25, dynamic allocation of IP addresses would start with 192.168.1.2 (as the first two IP addresses in the range, 192.168.1.0 and 192.168.1.1, are reserved).

To ensure that a static IP address that you've specified isn't already dynamically allocated, it is recommended that you specify static IP addresses from the end of your subnet range. For example, if your IP network subnet is 192.168.1.0/25, start allocating static IP addresses from 192.168.1.126 downwards (as the last IP address in the range, 192.168.1.127, is reserved).

- **Public IP Address:** Select an available IP reservation for IP networks. When the instance is created, you can configure security rules and access control lists for your IP network to enable access to this IP address over the public Internet. If you don't select an IP reservation now, you can associate a public IP address with this interface later by creating or updating an IP reservation.
- **Cloud IP Address:** Select an available IP reservation from the cloud IP pool. When the instance is created, this IP address can be accessed by other Oracle Cloud services without being accessible over the public Internet. If you don't select a cloud IP address now, you can associate a cloud IP address with this interface later by creating or updating an IP reservation.
- **MAC Address:** Specify the MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter 01:02:03:04:ab:cd as the MAC address but not 01-02-03-04-ab-cd. Ensure that the MAC addresses that you specify are unique within each IP network exchange and each IP network. If you specify a duplicate MAC address, each vNIC with that MAC address is disabled.
- **Virtual NIC Sets:** Select the vNICsets that you want to add this interface to. Each interface is added to the default vNICset by default. If you select other vNICsets to add this interface to, you can remove it from the default vNICset. However, ensure that you add each interface to at least one vNICset, to enable communication to that interface. After the instance is created, communication with each vNIC depends on the vNICsets it belongs to and the access control lists that apply to each vNICset. While creating an instance, you can add a vNIC to up to 4 vNICsets. To add a vNIC to more than 4 vNICsets, update the required vNICsets after the instance is created. You can also remove vNICs from a vNICset after the instance is created.

- **DNS:** Enter the DNS A record names for the instance. You can specify up to eight DNS A record names for each interface on an IP network. These names can be queried by instances on any IP network in the same IP network exchange. If no static IP address is specified for the interface, an IP address on the specified IP network is assigned automatically. After the instance is launched, the defined names are associated with the IP address that was automatically allocated to the interface.
- **Name Servers:** Enter the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.
- **Search Domains:** Enter the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.
- **Default Gateway:** Select this option if you want to use this interface as the default gateway. All network traffic uses the specified default gateway, unless a different interface is explicitly configured for an application within the instance.

## 10. Shared Network Options

- **Public IP Address:**
  - If you want to connect to this instance over the Internet, then select either **Auto Generated** or **Persistent Public IP Reservation**.
  - If you select an autogenerated public IP address, the IP address persists while the instance is running, but will change if you delete the instance and create it again later.
  - To associate a permanent public IP address with the instance, select **Persistent Public IP Reservation**.
  - If you've already created an IP address reservation, select it from the list. Otherwise, to create one now, click **Create IP Reservation**. In the Create Public IP Reservation dialog box, enter a name for the IP reservation and then click **Create**. The IP reservation is created. Select this IP reservation from the list of persistent public IP reservations.
  - If you don't want your instance to be accessed over the Internet, then you don't need to associate a public IP address with it. In the **Public IP Address** list, select **None**.

For more information about public IP addresses, see [About Public IP Addresses](#).

- **Security Lists:**
  - (Optional) Specify the security lists that you want to add the instance to. Click this field or start typing to see a list of available security lists. When you add an instance to a security list, you can control access to or from this instance by creating security rules that use the specified security list as a source or destination.
  - (Optional) If you haven't created the security lists that you want to use, you can do so now. Click **Create Security List**. Enter a name for the security list and optionally enter a description, and then click **Create**. The security list is created and appears in the list of security lists that you want to add your instance to.

If you don't specify any security list, the instance is added to the default security list, `default/default`. You can use this security list as a source or destination in security rules that you create. If any existing security rules already specify this security list as a source or destination, those rules will apply to this instance when the instance is created. If you don't want to use those security rules with this instance, or if you want to configure access to this instance separately later on, remember to remove the instance from the `default/default` security list after it is created. See [Removing an Instance from a Security List](#).

For more information about security lists, see [About Security Lists](#).

11. The Storage page shows the persistent boot disk that will be created and used to boot your instance. You can retain this setting and attach additional storage volumes later, when the instance is running. Alternatively, you can update or remove the persistent boot disk that is created by default or attach additional boot or data disks now.

You can attach existing storage volumes to your instance, if required, or create storage volumes and attach them to the instance.

- **To attach storage volumes that you've already created:**  
Click **Attach Existing Volume**.

 **Note:**

You can't detach storage volumes that are attached during instance creation.

In the Attach Existing Storage Volume dialog box, select or enter the following and then click **Add**:

- **Attach Storage Volume:** Select the storage volume that you want to attach.

 **Note:**

A storage volume must be in the online state before it can be attached to an instance. If a storage volume is already attached to another instance or if it is in an error state, it's not displayed in the **Attach Storage Volume** list.

- **Attach as Disk #:** Enter a disk index number. The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on. Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.
  - **Boot Drive:** Select this option to use the specified storage volume as the boot disk. The storage volume you select here must have the same image as the image that you selected on the Image page of the Create Instance wizard.
- **To create a storage volume and attach it to the instance:**

Click **Add New Volume**.

In the Add New Storage Volume dialog box, select or enter the following and then click **Add**:

- **Name:** Enter a name for this storage volume.
- **Size:** The size is set automatically to accommodate the disk size that's specified in the image that you selected earlier. If you want a larger boot disk than that specified in the image, then enter a larger size.

 **Note:**

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

- **Storage Property:** Select a storage property.

Based on your latency and IOPS requirements, select one of the following storage properties.

Storage Property	Latency	Throughput
storage/default	Standard	Standard
storage/latency	Low	High
storage/ssd/gpl	Lowest	Highest


 **Note:**

SSD storage volumes aren't available in all sites.

The web console might show other storage properties. Don't select any of them.


- **Description:** (Optional) Enter a description.
- **Attach as Disk #:** Accept the default disk number or enter a higher number. The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on. Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.
- **Boot Drive:** Select this option to use the specified storage volume as the boot disk. When you select this option, the disk number is automatically set to **1**.

- **To customize the persistent boot disk that is created by default:**

From the  menu, select **Update**. In the Update Storage Volume dialog box, you can modify the disk size, storage property, or the description. You can also uncheck the **Boot Drive** option. The persistent storage volume will still be created when the instance is created, but it won't be used to boot the instance.

Instead, a nonpersistent boot disk will be created and used to boot the instance.

- **To create an instance that uses a nonpersistent boot disk:**

Remove the default boot disk. From the  menu, select **Remove**. When you do this and you don't specify another persistent storage volume as the boot drive, a nonpersistent boot disk is used to boot your instance.

 **Note:**

If you want to create an instance snapshot to use this instance as a template for creating other instances, remove the persistent boot disk while creating this instance and use a nonpersistent boot disk instead. You can't create an instance snapshot of an instance that uses a persistent boot disk.

However, if you want to use a persistent boot disk to boot your instance, you can still use this boot disk as a template to create other instances by creating a storage volume snapshot of the boot disk and using that snapshot to create a new bootable storage volume. See [Backing Up and Restoring Storage Volumes Using Snapshots](#).

When you're done, click the button to go to the next page.

12. On the Review page, verify the information that you've entered, and then click **Create**.
13. Monitor the status of the instance.
  - When you create an instance, the initial status is **Preparing**. Compute Classic allocates resources and prepares to create the instance.
  - While the specified image is being installed, the state changes to **Initializing**.
  - After the image is installed and the instance is starting, the status changes to **Starting**.
  - When the instance is ready, the status changes to **Running**. When an instance is in the **Running** state, you can connect to it. You can also attach or detach storage volumes and security lists.
  - When an instance is running, you can shut down the instance. Its status changes to **Stopping**. When the operation is completed, its status changes to **Stopped**. When an instance is shut down or stopped, you can either start the instance, or delete it.
  - When an instance is running or shut down, you can delete the instance. Its status changes to **Deleting**. When the operation is completed, the instance is deleted.
  - At times, an instance can have the **Error** status.

For example, when you create or re-create an instance by starting its orchestration, if some of the resources required to create the instance aren't available, then the status of the instance changes to **Error**.

 **Note:**

If you get an error message `Unable to place instance...` it indicates that the site you've selected doesn't have sufficient resources to create this instance. If your domain spans multiple sites, then use the **Site** menu near the top of the page to select another site and run the Create Instance wizard again.


After your instance is created, you can log in to your instance. See [Logging In to an Instance](#).

 **Tip:**

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

For security and patching-related guidelines, see the documentation for your operating system.

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Creating Instances Using Orchestration

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

To create instances using an orchestration, you define the orchestration offline in a JSON-formatted file, upload the orchestration to Compute Classic, and then start the orchestration. All the objects defined in the orchestration are created automatically.

At any time, you can delete and re-create all the instances in an orchestration just by stopping and restarting the orchestration. Storage attachments, security lists, and so on are re-created and associated automatically with the appropriate instances.

In orchestrations v2, you can achieve granular control over each object in an orchestration by defining the persistence of each object. If you want to delete instances in an orchestration but not the associated storage volumes, you can specify

persistence for storage volumes and not for instances. That way, when you suspend an orchestration, all nonpersistent objects are deleted, while persistent objects continue to run.

To learn more about orchestration features, terminology, and concepts, see [About Orchestration v2](#).

To get started with creating instances using orchestrations, see [Workflow for Creating Instances Using Orchestration v2](#).

After your instance is created, you can log in to your instance. See [Logging In to an Instance](#).

 **Tip:**

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

For security and patching-related guidelines, see the documentation for your operating system.

## Creating an Instance Using a Blank Orchestration v2

You can create a blank orchestration in the web console. Instead of defining the orchestration in a JSON-formatted file and then uploading the orchestration to Compute Classic, you can create a blank orchestration, and then add objects to it by updating the orchestration. While updating the orchestration, you can define attributes for a single instance or create complex topologies that consist of multiple instances and multiple networks.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console.
2. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
3. Click the **Orchestrations** tab.
4. Click **Create Orchestration**.  
The Create Orchestration dialog box appears.
5. Enter the following information, and then click **Create**.
  - **Name:** Enter a name for the orchestration.

- **Description:** Enter a description.
- **Tags:** Specify one or more tags to help you identify and categorize the orchestration.

A blank orchestration is created and listed in the Orchestration page.

6. Update the orchestration to add objects to it. Go to the blank orchestration that you want to update. From the  menu, select **Update**.

The orchestration details page appears.

7. To add any object type, go to the appropriate section and click **Add**. Let's consider that you want to create an Oracle Linux instance which you want to access over the public Internet by using SSH and which you want to associate with an IP network. To achieve this: add an SSH key, create a storage volume, vNICset, IP network, an access control list (ACL), and set up the required security rules. To create a new IP network, click **Add** in the IP Network section. The Create IP Network dialog box appears. Enter the required information, and then click **Create**. Similarly, you can add all the other objects that you'll require while creating the instance.

The object is added to the orchestration with the status **Inactive**. The objects are created when you start the orchestration.

 **Note:**

By default, objects that you define in the Orchestration details page are not associated with the instances you define in the orchestration. You'll have to associate the objects, such as storage volume and IP network with the instance while defining the instance.

8. By default, the objects that you define in the Orchestration details page are not persistent.

 **Note:**

Oracle recommends setting persistent to false for instances and setting persistent to true for the objects associated with an instance, such as security rules, storage volumes, and IP network. Objects that are not persistent will be deleted when you suspend the orchestration. This allows you to shutdown instances without losing data or breaking other instances. Generally, instances boot from a persistent boot disk, ensuring that any changes that you make at the operating system-level persist when the instance is re-created. If you create and customize an instance using a nonpersistent boot disk, you can use instance snapshots to use the instance as a template to create multiple identical instances but changes that you make at the operating system-level are lost.

To make these objects persistent, perform the following task for every object that you have defined.


- a. Go to the object, and then from the  menu, select **Properties**.



The **Object Properties** dialog box appears.

- b. Select the **Persistent** check box.
- c. Click **Update**.

The property of the object is updated.

9. To add an instance:
  - a. In the Instance section, click **Add**. An instance with default configuration (`/oracle/public/OL_7.2_UEKR4_x86_64` as the image and `oc3` as shape) is added to the orchestration with the status **Inactive**. As this instance doesn't have an interface on the shared network or on any IP network, when you start the orchestration this instance is automatically added to the default security list on the shared network.
  - b. Update attributes of the instance such as image, shape, storage volume attached to the instance, IP network interfaces, shared network interface, and SSH keys. Go to the instance. From the  menu, click **Update**.
  - c. In the **Information** section, provide the following information for the instance.
    - **Name:** Enter a name for your instance, or accept the default.
    - **Image:** Select the image you want to use. The image specifies the operating system and disk size of the instance.
    - **Shape:** The shape specifies the OCPU and memory resources to be allocated to the instance. If you select a high I/O shape, an NVMe SSD disk is automatically attached to your instance. This is a local, nonpersistent NVMe SSD disk, which provides high I/O access rates. This disk is attached to your instance with the device name `/dev/xvdz`. After your instance is created, you can mount this disk and format it as required. The size of this NVMe SSD disk is fixed depending on the selected shape.

**Note:**

High I/O shapes aren't available in all regions.

For more information about shapes, see [About Shapes](#).

- **Desired State:** When you don't set the desired state, the instance inherits this value from the orchestration. If you select **Running**, the instance is started. If you select **Stopped**, the instance is shut down. You can start the instance again later by updating the instance with the desired state specified as running.
- **DNS Hostname Prefix:** (Optional) Specify a DNS host name prefix. The host name is visible internally within your DNS space. It is referenced by other instances in the domain, as well as by the OS and applications running on your instance. The host name that you specify is suffixed by the domain name. If you don't specify a host name, then a host name is generated automatically.
- **Reverse DNS:** If set to `true` (default), then reverse DNS records are created. If set to `false`, no reverse DNS records are created.

- **Custom Attributes:** Enter any additional attributes that you want to store on the instance. This field allows you to customize your instance by providing additional information specific to each instance. You can enter arbitrary key-value pairs in plain text. The text you enter here *must* be in JSON format. This information is stored as user data on your instance.

For information about user-defined attributes that can be used to automate instance configuration, see [Automating Instance Initialization Using `opc-init`](#).

After the instance is created, the attributes that you specify here are available within the instance at `http://192.0.0.192/latest/user-data`. For information about retrieving user data, see [Retrieving User-Defined Instance Attributes](#).

- **Tags:** (Optional) Specify one or more tags to help you identify and categorize the instance.
- d. Click **Update** to update the attributes of the instance.
  - e. In the **Storage Volumes** section, click **Attach a Storage Volume** to attach an existing storage volume to the instance. The Attach a Storage Volume dialog box appears. Specify the following information, and then click **Attach**.
    - **Attach Storage Volume:** Select the storage volume that you want to attach. Ensure that the storage volume that you select is not attached to any other instance.
    - **Attach as Disk #:** Enter a disk index number. The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on. Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.

 **Note:**

It is recommended that you create a persistent boot disk from which instances can boot, ensuring that any changes that you make at the operating system-level persist when the instance is re-created. If you create and customize an instance using a nonpersistent boot disk, you can use instance snapshots to use the instance as a template to create multiple identical instances but changes that you make at the operating system-level are lost. If you don't select a storage volume, neither a data disk nor a nonpersistent boot disk is attached to the instance.

- f. In the **Shared Network Interface** section, configure the shared network if required. Don't select an interface on the Shared Network if you want to set up the instance for SSH access on IP networks. When you select shared network, the interface on the shared network is used as the default gateway even if you have created an interface on the IP network.

 **Note:**

If you don't add an interface to the Shared Network or to any IP network, then the instance is added to the default security list on the Shared Network. You can add it to other security lists and create security rules and assign a public IP address later; however you can't add it to any IP network later.

- g. In the **IP Network Interfaces** section, click **Add IP Network Interface** to add the instance to an IP network and then provide the following information.
- **Interface:** Select the interface that you want to add to the IP network. You can select any interface from eth0 to eth7. You can't add, delete, or modify interface allocations after an instance is created.
  - **vNIC Name:** Retain the default vNIC name or enter another name. The three-part vNIC name is generated using this name. It has the format / *Compute-identity\_domain/username/instanceName\_vnicName*.
  - **IP Network:** Specify the IP network that you want to add this interface to. When you add an instance to an IP network, the specified interface of the instance is assigned an IP address on the specified IP network. After the instance is created, you can view information about each interface on the Instance Details page.
  - **Static IP Address:** Specify a private IP address for this interface. The private IP address must be unused and it must belong to the subnet of the selected IP network. Remember, too, that certain IP addresses in a subnet are reserved. For example, the first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.  
  
If no static IP address is specified, an IP address from the specified IP network is allocated dynamically, when the instance is created. Dynamically allocated IP addresses might change if the instance is deleted and re-created.  
  
Dynamic IP addresses are allocated from the lowest IP address in the range upwards. For example, if your IP network subnet is 192.168.1.0/25, dynamic allocation of IP addresses would start with 192.168.1.2 (as the first two IP addresses in the range, 192.168.1.0 and 192.168.1.1, are reserved).  
  
To ensure that a static IP address that you've specified isn't already dynamically allocated, it is recommended that you specify static IP addresses from the end of your subnet range. For example, if your IP network subnet is 192.168.1.0/25, start allocating static IP addresses from 192.168.1.126 downwards (as the last IP address in the range, 192.168.1.127, is reserved).
  - **Public IP Address:** Select an available IP reservation for IP networks. When the instance is created, you can configure security rules and access control lists for your IP network to enable access to this IP address over the public Internet. If you don't select an IP reservation now, you can associate a public IP address with this interface later by creating or updating an IP reservation.

- **Cloud IP Address:** Select an available IP reservation from the cloud IP pool. When the instance is created, this IP address can be accessed by other Oracle Cloud services without being accessible over the public Internet. If you don't select a cloud IP address now, you can associate a cloud IP address with this interface later by creating or updating an IP reservation.
- **MAC Address:** Specify the MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter `01:02:03:04:ab:cd` as the MAC address but not `01-02-03-04-ab-cd`. Ensure that the MAC addresses that you specify are unique within each IP network exchange and each IP network. If you specify a duplicate MAC address, each vNIC with that MAC address is disabled.
- **Virtual NIC Sets:** Select the vNICsets that you want to add this interface to. Each interface is added to the default vNICset by default. If you select other vNICsets to add this interface to, you can remove it from the default vNICset. However, ensure that you add each interface to at least one vNICset, to enable communication to that interface. After the instance is created, communication with each vNIC depends on the vNICsets it belongs to and the access control lists that apply to each vNICset. While creating an instance, you can add a vNIC to up to 4 vNICsets. To add a vNIC to more than 4 vNICsets, update the required vNICsets after the instance is created. You can also remove vNICs from a vNICset after the instance is created.
- **DNS:** Enter the DNS A record names for the instance. You can specify up to eight DNS A record names for each interface on an IP network. These names can be queried by instances on any IP network in the same IP network exchange. If no static IP address is specified for the interface, an IP address on the specified IP network is assigned automatically. After the instance is launched, the defined names are associated with the IP address that was automatically allocated to the interface.
- **Name Servers:** Enter the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.
- **Search Domains:** Enter the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.
- **Default Gateway:** Select this option if you want to use this interface as the default gateway. All network traffic uses the specified default gateway, unless a different interface is explicitly configured for an application within the instance.

If the instance has an interface on the shared network, that interface is always used as the default gateway.

- h. In the **SSH Public Keys** section, click **Add SSH Public Key** and then select the SSH public key that you want to associate with the instance.

 **WARNING:**

Remember to associate an SSH public key and a public IP address with every Linux instance that you will access over SSH. If you don't associate an SSH key, you can't access your Linux instance and you can't associate SSH keys later on.

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

10. Click **Start** to start the orchestration.

When you start the orchestration, the status of the orchestration changes to **Starting** and then to **Ready** when all the objects defined in the orchestration are created successfully. The instance and other objects are created and their status changes from **Inactive** to **Active**.

## Creating Instances Using Launch Plans

A launch plan is a JSON-formatted file that defines the properties of one or more instances. You can use a launch plan to quickly create and start multiple instances in Compute Classic.

### Topics

- [About Launch Plans](#)
- [Sample Launch Plan](#)
- [Launch Plan Attributes](#)
- [Instance Attributes Specified in a Launch Plan](#)
- [Prerequisite for Creating Instances Using Launch Plans](#)
- [Procedure for Creating Instances Using Launch Plans](#)

### About Launch Plans

A **launch plan** specifies the provisioning sequence and attributes of the instances that you want to create. Note that while you can reuse your launch plan JSON file to create new instances based on the attributes and provisioning sequence specified in the JSON file, the launch plan itself doesn't persist in Compute Classic.

To understand how using launch plans differs from using orchestrations to create instances, see the FAQ [How are orchestrations different from launch plans?](#).

### Sample Launch Plan

The following is an example of a JSON-formatted file showing the attributes for two instances with different shapes and SSH keys but using the same image.

```

{
  "instances":
  [
    {
      "shape": "oc4",
      "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64",
      "sshkeys": ["/Compute-acme/admin/dev-ssh"],
      "name": "/Compute-acme/admin/dev-vm",
      "label": "dev-vm"
    },
    {
      "shape": "oc5",
      "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64",
      "sshkeys": ["/Compute-acme/admin/prod-ssh"],
      "name": "/Compute-acme/admin/prod-vm",
      "label": "prod-vm"
    }
  ]
}

```

### Launch Plan Attributes

Parameter	Required or Optional	Description
instances	required	A list of instances. Each instance is defined using the instance attributes.
relationships	optional	The relationships between various instances. Valid values: <ul style="list-style-type: none"> <li>same_node: The specified instances are created on the same physical server. This is useful if you want to ensure low latency across instances.</li> <li>different_node: The specified instances aren't created on the same physical server. This is useful if you want to isolate instances for security or redundancy.</li> </ul>

### Instance Attributes Specified in a Launch Plan

Parameter	Required or Optional	Description
shape	required	The name of the shape that defines the number of OCPUs and the RAM that you require for the instance. For general purpose and high-memory shapes, you can select the block storage disk size, but for high I/O shapes, the size of the SSD storage is determined by the shape.

Parameter	Required or Optional	Description
name	optional	<p>The three-part name of the instance (<i>/Compute-identity_domain/user/name</i>).</p> <p>If you specify this parameter, then the full name of the instance would be in the format, <i>/Compute-identity_domain/user/name_you_specify/id</i>.</p> <p>If you don't specify this parameter, then the full name would be in the format, <i>/Compute-identity_domain/user/id</i>.</p> <p>In either case, <i>id</i> is an autogenerated ID.</p> <p><b>Examples of Instance Names:</b></p> <ul style="list-style-type: none"><li>• When you specify <i>/Compute-acme/jack/vm1</i> as the value of the name parameter: <i>/Compute-acme/jack/vm1/300a7479-ec90-4826-98b9-a725662628f1</i></li><li>• When you don't specify the name parameter: <i>/Compute-acme/jack/38ef677e-9e13-41a7-a40c-2d99afce1714</i></li></ul> <p>Although this is an optional parameter, specifying a meaningful name makes it easier for you to identify your instances.</p>
label	optional	<p>A text string to identify the instance.</p> <p>This label is used when defining relationships in an orchestration.</p> <p>A label can contain only alphanumeric characters, hyphens, and underscores. It can't contain unicode characters and spaces.</p> <p>Maximum length is 256 characters.</p>
tags	optional	<p>A JSON array or list of strings used to tag the instance.</p> <p>By assigning a human-friendly tag to an instance, you can identify the instance easily when you perform an instance listing. These tags aren't available from within the instance.</p>

Parameter	Required or Optional	Description
attributes	optional	<p>A JSON object or dictionary of user-defined attributes to be made available to the instance.</p> <p>If you're creating a Windows instance, you must specify the following required attributes:</p> <pre> {     "enable_rdp": true,     "administrator_password": "Specify_password_here" } </pre> <p>For more information about specifying user-defined attributes that can be used to automate instance configuration, see <a href="#">Automating Instance Initialization Using opc-init</a>.</p> <p><b>Note:</b></p> <p>Solaris machine images don't include the opc-init scripts. So you can't use opc-init to automate instance configuration of Solaris instances.</p> <p>The attributes that you specify can be accessed from within the instance at <a href="http://192.0.0.192/latest/attributes">http://192.0.0.192/latest/attributes</a>. For more information about retrieving user-defined attributes, see <a href="#">Retrieving User-Defined Instance Attributes</a>.</p>
imagelist	optional	<p>The three-part name (<code>oracle/public/imagelist_name</code>) of the image list containing the image to be used (example: <code>/oracle/public/OL_6.7_UK4R4_x86_64</code>).</p> <p>You <i>must</i> use this attribute if you don't specify a bootable storage volume by using the <code>boot_order</code> attribute. If you specify the <code>imagelist</code> attribute as well as the <code>boot_order</code> attribute, then the <code>imagelist</code> attribute is ignored.</p>
storage_attachments	optional	<p>If you specify the <code>storage_attachments</code> parameter, then specify the following subparameters for each attachment:</p> <ul style="list-style-type: none"> <li><b>volume:</b> The three-part name (<code>/Compute-identity_domain/user/object_name</code>) of the storage volume that you want to attach to the instance. <p>Note that volumes attached to an instance at launch time can't be detached.</p> </li> <li><b>index:</b> The index number for the volume. <p>The allowed range is 1 to 10. If you want to use a storage volume as the boot disk for an instance, you must specify the index number for that volume as 1.</p> <p>The index determines the device name by which the volume is exposed to the instance. Index 0 is allocated to a nonpersistent boot disk, <code>/dev/xvda</code>. An attachment with index 1 is exposed to the instance as <code>/dev/xvdb</code>, an attachment with index 2 is exposed as <code>/dev/xvdc</code>, and so on.</p> </li> </ul>



Parameter	Required or Optional	Description
<code>boot_order</code>	optional	<p>The index number of the bootable storage volume that should be used to boot the instance. The only valid value is 1.</p> <p>If you set this attribute, you must also specify a bootable storage volume with index number 1 in the <code>volume</code> sub-parameter of <code>storage_attachments</code>.</p> <p>When you specify <code>boot_order</code>, you don't need to specify the <code>imagelist</code> attribute, because the instance is booted using the image on the specified bootable storage volume. If you specify both <code>boot_order</code> and <code>imagelist</code>, the <code>imagelist</code> attribute is ignored.</p>
<code>hostname</code>	optional	<p>The host name assigned to the instance. On an Oracle Linux instance, this host name is displayed in response to the <code>hostname</code> command.</p> <p>Only relative DNS is supported. The domain name is suffixed to the host name that you specify. The host name must not end with a period. If you don't specify a host name, then a name is generated automatically. The DNS name of an instance depends on its host name, as follows:</p> <ul style="list-style-type: none"> <li>• If no DNS name is specified in the <code>networking</code> attribute, then the DNS name is set to the host name, and a reverse DNS record (PTR) is created for the host name.</li> <li>• If the DNS name specified in the <code>networking</code> attribute matches the host name, then that record also creates a reverse DNS record for the host name.</li> <li>• If the <code>dns</code> attribute under <code>networking</code> is set to an empty list (<code>[]</code>), then no DNS records are created even if a host name is specified. The instance still receives its host name through DHCP, and can perform a reverse lookup of its host name. However, no other instance can perform this reverse lookup.</li> </ul>
<code>reverse_dns</code>	optional	<p>If set to <code>true</code> (default), then reverse DNS records are created.</p> <p>If set to <code>false</code>, no reverse DNS records are created.</p>

 **Note:**

If an instance has network interfaces defined only for IP networks and doesn't have any interface on the shared network, then when `hostname` is specified, no DNS entries are set. In this case, DNS entries are set by the `dns` subparameter of the `networking` attribute.

---

Parameter	Required or Optional	Description
networking (attributes for the shared network)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>Only one interface on an instance can be added to the shared network. To add an interface to the shared network, you can specify the following subparameters:</p> <ul style="list-style-type: none"><li>• <code>seclists</code>: (Optional) The security lists that you want to add the instance to.</li><li>• <code>nat</code>: (Optional) Indicates whether a temporary or permanent public IP address should be assigned to the instance.</li><li>• <code>dns</code>: (Optional) A list of the DNS A record names for the instance. This name is relative to the internal DNS domain.</li><li>• <code>model</code>: (Optional) The type of network interface card (NIC). The only allowed value is <code>e1000</code>.</li><li>• <code>name_servers</code>: (Optional) The name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li><li>• <code>search_domains</code>: (Optional) The search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li></ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on the Shared Network</a>.</p>

---

Parameter	Required or Optional	Description
networking (attributes for IP networks)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>To add this instance to an IP network, specify the following subparameters:</p> <ul style="list-style-type: none"> <li><code>ipnetwork</code>: The name of the IP network that you want to add the instance to.</li> <li><code>ip</code>: (Optional) If you want to associate a static private IP address with the instance, specify an available IP address from the IP address range of the specified <code>ipnetwork</code>.</li> <li><code>address</code>: (Optional) The MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter <code>01:02:03:04:ab:cd</code> as the MAC address but not <code>01-02-03-04-ab-cd</code>.</li> <li><code>nat</code>: (Optional) A list of IP reservation that you want to associate with this interface, in the format: <code>"nat": ["network/v1/ipreservation:IP_reservation_name"]</code>. Here <code>IP_reservation_name</code> is the three-part name of the IP reservation in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnic</code>: (Optional) The three-part name of the vNIC in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnicsets</code>: (Optional) A list of the three-part names of the vNICsets that you want to add this interface to.</li> <li><code>is_default_gateway</code>: (Optional) If you want to specify the interface to be used as the default gateway for all traffic, set this to <code>true</code>. The default is <code>false</code>. If the instance has an interface on the shared network, that interface is always used as the default gateway.</li> <li><code>dns</code>: (Optional) A list of the DNS A record names for the instance.</li> <li><code>name_servers</code>: (Optional) A list of the name servers that should be sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li> <li><code>search_domains</code>: (Optional) A list of the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li> </ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on an IP Network</a>.</p>

Parameter	Required or Optional	Description
sshkeys	optional	<p>A list of the SSH public keys that you want to associate with the instance.</p> <p><b>Note:</b></p> <p>You don't need to provide any SSH public keys if you're creating a Windows instance, because you can't access a Windows instance using SSH. To access a Windows instance, see <a href="#">Accessing a Windows Instance Using RDP</a>.</p> <p>For each key, specify the three-part name in the <code>/Compute-identity_domain/user/object_name</code> format.</p> <p>You can associate the same key with multiple instances.</p> <p>The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at <code>http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key</code>.</p> <ul style="list-style-type: none"> <li>Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the <code>authorized_keys</code> file of the <code>opc user</code>.</li> <li>In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the <code>authorized_keys</code> file of the appropriate users.</li> </ul>

### Prerequisite for Creating Instances Using Launch Plans

Ensure that you've created your launch plan JSON file.

### Procedure for Creating Instances Using Launch Plans

To create instances from a launch plan by using the CLI, use the `opc compute launch-plan add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create instances from a launch plan by using the API, use the `POST /launchplan/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Creating an Instance Using Visual Object Editor



This topic does not apply to Oracle Cloud at Customer.

When you create an instance using visual object editor, a single orchestration v2 is created automatically to manage the instance and its associated resources.

Storage volumes and networking objects used by the instance are created in the same orchestration. Instances are nonpersistent by default. However, storage volumes and other objects are created with persistence set to true, so that if you suspend the orchestration, instances are shut down, but storage volumes aren't deleted. Terminating the orchestration, however, will cause all objects to be deleted and any data on storage volumes will be lost.



### Note:

You can't create Windows instances using the Visual Object Editor. Create Windows instances using QuickStarts or by using the Create Instance wizard.

## Prerequisites

- You must generate an SSH key pair and upload the SSH public key to your Compute Classic account to access the instance from a Windows host using PuTTY. To generate an SSH key pair using PuTTY, see the section [Generating an SSH Key Pair](#) in [Tutorial: Creating an Oracle Linux Instance Using the Oracle Cloud Infrastructure Compute Classic Web Console](#). To upload the SSH public key to your account, see [Adding an SSH Public Key](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles](#) in *Managing and Monitoring Oracle Cloud*.
- Create an IP network to which you want to attach your instance or you can use the default IP network. See [Creating an IP Network](#).

## Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click **Visualization** in the top right corner.
3. In the palette on the left, expand **Storage**, and then drag and drop **Instance** on to the canvas in the center pane.
4. Select or enter the following information in the **Create Instance** dialog box. Select the appropriate image. If you select an image from Oracle Cloud Marketplace, accept the terms and wait for the app to be installed before continuing.
  - **Instance Name:** Enter a name for the instance or use the default value.
  - **Boot Image:** Select the image you want to use. The image specifies the operating system and disk size of the instance.

Only Oracle-provided images and private images are available. Oracle Cloud Marketplace images are not listed.
  - **Shape:** Select the shape that you want to use. The shape specifies the OCPU and memory resources to be allocated to the instance. If you select a high I/O shape, an NVMe SSD disk is automatically attached to your instance. This is a local, nonpersistent NVMe SSD disk, which provides high I/O access rates. This disk is attached to your instance with the device name `/dev/xvdz`. After your instance is created, you can mount this disk and format it as required. The size of this NVMe SSD disk is fixed depending on the selected shape.

 **Note:**

High I/O shapes aren't available in all regions.

For more information about shapes, see [About Shapes](#).

- **SSH Key:** Select an existing SSH public key.
- **Storage Size:** The size is set automatically to accommodate the disk size that's specified in the image that you selected earlier. If you want a larger boot disk than that specified in the image, then enter a larger size.

 **Note:**

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

- **IP Network:** Select the IP network that you want to attach your instance to.

5. Click **Create**.

An orchestration v2 is created. For example, if you created an instance and specified the name of the instance as **vm1**, an orchestration v2 is created with the same name. While the instance is being created, you can monitor the corresponding orchestration on the Orchestration page. When the instance is created, it is listed on the Instances page.

An instance is created with the following general configuration:

- Uses a persistent boot disk. You won't be able to create an instance snapshot of this instance.
- In sites which support public IP address reservations, security rules, and access control lists in IP networks:
  - Doesn't have an interface on the shared network.
  - Has one interface on the default IP network and is added to a vNICset of the same name.
  - Has one IP address from the `/oracle/public/cloud-ippool` IP address pool and another IP address from the `/oracle/public/public-ippool` IP address pool.
  - Has the required security rules and ACL set up to enable SSH access to the instance and all egress traffic.
- In sites which don't support public IP address reservations, security rules, and access control lists in IP networks:
  - Has an interface on the shared network.
  - Is added to the default security list.
  - Has a temporary public IP address.

- Has the required security rule set up to enable SSH access.
- Is nonpersistent. This allows you to update the instance by suspending the corresponding orchestration v2. When the orchestration is suspended, the instance status changes to **Inactive** and you can update any attribute of the instance.
- Has persistence specified as true for all other objects.


After your instance is created, you can log in to your instance. See [Logging In to an Instance](#).

 **Tip:**

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

For security and patching-related guidelines, see the documentation for your operating system.

 **See Also:**

- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Workflow for Creating Instances Using Orchestration v2](#)
- [Creating Instances Using Launch Plans](#)
-  [Tutorial: \*Creating Instances Using Orchestration v1\*](#)

## Logging In to an Instance

After you've associated a public IP address with your instance, you can log in to the instance.

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

To connect to your Oracle-provided Oracle Linux instance using `ssh`, see [Accessing an Oracle Linux Instance Using SSH](#).

 **Note:**

If you've created your instance using an Oracle-provided image, then you can log in to the instance as the `opc` user. You can't log in as `root`.

To connect to your Oracle Solaris instance using `ssh`, see [Accessing an Oracle Solaris Instance Using SSH](#). (Not available on Oracle Cloud at Customer)

To connect to your Windows instance using an RDP connection, see [Accessing a Windows Instance Using RDP](#).

## Cloning an Instance by Using Instance Snapshots

### Topics

- [About Instance Snapshots](#)
- [Creating an Instance Snapshot](#)
- [Registering the Image Generated by an Instance Snapshot](#)
- [Creating an Instance from an Instance Snapshot](#)
- [Deleting an Instance Snapshot](#)

## About Instance Snapshots

Instance snapshots provide an easy way to create a customized machine image using an existing instance as a template. You can then use this customized machine image to create multiple instances with identical configurations.

To clone an instance using an instance snapshot, first create an instance using an appropriate machine image.

 **Note:**

If you want to create an instance snapshot, your instance must use a nonpersistent boot volume. If your instance uses a bootable storage volume and you want to clone the storage volume, see [Backing Up and Restoring Storage Volumes Using Snapshots](#).

When your instance is running, customize your instance as required, by adding users, or installing and configuring applications. These changes are stored on your nonpersistent boot disk.

When you're done customizing your instance, to use the instance as a template to create other instances, create an instance snapshot. Instance snapshots capture the current state of your boot disk and create a corresponding machine image, which is uploaded to your Oracle Cloud Infrastructure Object Storage Classic account. You can then register this machine image with your Compute Classic account and use it to



create instances. These instances will contain all the configuration and customization that you'd done on the original instance when you took the snapshot.

You can generate an instance snapshot while your instance is still running. If you do this, you can continue to work on your instance, but any changes you make after the snapshot is generated won't be captured by the snapshot. If you want to generate the snapshot just before you delete your instance, you can use the option to create a deferred snapshot. This allows you to continue making changes to the instance even after you've created the snapshot request, and ensuring that all the changes you make will be captured by the snapshot just before the instance is deleted.

When you create an instance using a nonpersistent boot disk, if you want to delete the instance, then using instance snapshots also allows you to preserve the changes you've made to your instance before you delete the instance. Later on, you can use this machine image to create another instance identical to the one you deleted.

## Creating an Instance Snapshot

Creating a snapshot of an instance allows you to capture the current state of the nonpersistent boot disk used by an instance, including all customization that you may have made at the operating-system level after creating the instance.



### Note:


Instance snapshots capture the state of your nonpersistent boot disk. You can't create an instance snapshot if your instance uses a persistent bootable storage volume. To create a snapshot of a storage volume, see [Backing Up and Restoring Storage Volumes Using Snapshots](#).

### Prerequisites


- Instance snapshots are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. Ensure that you've set a replication policy for your Oracle Cloud Infrastructure Object Storage Classic instance. See [Selecting a Replication Policy for Oracle Cloud Infrastructure Object Storage Classic in Using Oracle Cloud Infrastructure Object Storage Classic](#).
- Ensure that the instance you want to snapshot isn't in an error state. You can't create a snapshot of an instance that is in an error state.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

### Procedure

To create an instance snapshot:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Go to the instance that you want to create a snapshot of. From the  menu, select **Create Snapshot**.

Alternatively, you can also create an instance snapshot from the instance details page.

- a. On the Instances page, go to the instance that you want to create a snapshot of, and from the  menu, select **View**.
  - b. On the instance details page, go to the Instance Snapshots section and click **Create Snapshot**.
3. In the Create Instance Snapshot dialog box, enter a name for the snapshot.
  4. If you haven't yet finished customizing your instance and you want to create the snapshot just before you delete the instance, you can select the **Deferred Snapshot** option. This option allows you to continue working on the instance. The snapshot is taken only when you delete the instance or stop the instance orchestration.
  5. Click **Create**. A request to create an instance snapshot is created. If the deferred snapshot option was selected, the snapshot will be generated when you delete the instance. If the deferred snapshot option wasn't selected, the process of creating the instance snapshot begins right away.

When an instance snapshot is generated, it creates a custom image. While the image is being created, or when you select the option to create a deferred snapshot, the instance details page shows the state of the instance snapshot as *Active*. When the image has been created and is available in your Oracle Cloud Infrastructure Object Storage Classic account, the state of the instance snapshot changes to *Complete*. Next, to register this image, see [Registering the Image Generated by an Instance Snapshot](#).

To create an instance snapshot using the CLI, use the `opc compute snapshot add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create an instance snapshot using the API, use the `POST /snapshot/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Registering the Image Generated by an Instance Snapshot

An instance snapshot captures the current state of the nonpersistent boot disk of an instance and uses it to create a corresponding machine image. You can then use this machine image to create other instances. These instances are clones of the instance that you created the snapshot of. Any customization done on that instance is automatically part of instances created using the snapshot.


The image created by an instance snapshot is stored in your Oracle Cloud Infrastructure Object Storage Classic account. Before you can use this image to create an instance, you must register this image in your Compute Classic account.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).



1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Instance Snapshots** tab in the left pane.

The Instance Snapshots page displays a list of snapshots. Instance snapshots are listed alphabetically by instance name. If an instance has multiple snapshots, the most recent snapshot is listed on top.

3. Go to the snapshot that you want to use. From the  menu, select **Associate Image**.

Alternatively, you can also register a snapshot from the instance details page.

- a. On the Instances page, go to the instance that you want to clone. From the  menu, select **View**.
- b. On the instance details page, in the **Instance Snapshots** section, go to the snapshot that you want to use. From the  menu, select **Associate Image**.

4. Enter a description for the image and click **Ok**.

The image is added as a private image on the Private Images page.

To do this using the CLI, use the `opc compute image-list add` and `opc compute image-list-entry add` commands, in that order. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To do this using the API, invoke the `POST /imagelist/` and the `POST /imagelistentry/` methods, in that order. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After registering the image generated by an instance snapshot, to create an instance using this machine image, see [Creating an Instance Using a Private Image](#).

## Creating an Instance from an Instance Snapshot

An instance snapshot captures the current state of an instance and uses it to generate an image is uploaded to your Oracle Cloud Infrastructure Object Storage Classic account. You can then register this image with your Compute Classic account and use it to create instances.

After you've registered the image generated by an instance snapshot, the machine image is added to the list of custom images on your Private Images page. To create an instance using this machine image, see [Creating an Instance Using a Private Image](#).

## Deleting an Instance Snapshot

An instance snapshot allows you to capture the current state of an instance and use it to launch other instances. When an instance snapshot is completed, it creates a machine image and stores it in your Oracle Cloud Infrastructure Object Storage Classic account.

After an instance snapshot has completed creating a machine image of an instance, the instance snapshot record on the web console only provides information about when a machine image was created from a given instance. You can also view the autogenerated name of an instance snapshot, which helps to identify the corresponding machine image file in your Oracle Cloud Infrastructure Object Storage Classic account. If you don't require this information for record-keeping purposes, you

can delete the instance snapshot. Deleting an instance snapshot has no impact on the machine image file stored in your Oracle Cloud Infrastructure Object Storage Classic account, or on the private image that you might have registered in your Compute Classic account.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- Ensure that the instance snapshot that you want to delete is in the **complete** state.

### Procedure

To delete an instance snapshot:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Go to the instance that you want to view. From the ☰ menu, select **View**.
3. On the instance details page, in the Instance Snapshots section, go to the instance snapshot that you want to delete. Make a note of the name of the image file associated with the snapshot. You might need this later, if you want to register the image created by this snapshot, or if you want to delete the image created by this snapshot from your associated Oracle Cloud Infrastructure Object Storage Classic account. Then, from the ☰ menu, click **Delete**.

Alternatively, you can also delete a snapshot from the Instance Snapshots page.

- a. On the Instances page, click the **Instance Snapshots** tab in the left pane.
- b. On the Instance Snapshots page, go to the snapshot that you want to delete. Make a note of the image file name associated with the snapshot, and then, from the ☰ menu, click **Delete**.

To delete an instance snapshot using the CLI, use the `opc compute snapshot delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an instance snapshot using the API, use the `DELETE /snapshot/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

# 4

## Managing Instances

### Topics

- [Viewing Information About an Instance](#)
- [Updating an Instance](#)
- [Managing Instance Lifecycle Operations](#)
- [Retrieving Instance Metadata](#)
- [Updating Packages on an Oracle Solaris Instance](#)

## Viewing Information About an Instance

You can view information about Compute Classic instances in several ways.

### Topics

- [Listing Instances](#)
- [Monitoring Instances](#)
- [Viewing Instance Metrics](#)
- [Enabling Instance Metrics Collection](#)
- [Viewing the Boot Log of an Instance](#)


## Listing Instances

After creating instances in Compute Classic, you can view a list of your instances using the web console.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

1. Sign in to your Cloud Account.
  - For Oracle Cloud, see *Signing in to Your Cloud Account* in *Getting Started with Oracle Cloud*.
  - For Oracle Cloud at Customer, click the Infrastructure Classic Console URL from the welcome email.

The Infrastructure Classic Console is displayed.

2. Click  in the top left corner of the Dashboard.
3. Under **Services**, click **Compute Classic**.

The **Compute Classic** console is displayed.

4. (Optional) This step is relevant only if your domain spans multiple sites. To change the site, click the **Site** menu near the top of the page.

See [About Compute Classic Sites](#).

Your instances are listed on the Instances page. For each instance, you can view details including the label, the current status, the attached storage volumes, and the public and private IP addresses associated with it. If your instance has an interface on the shared network, the public IP address on the shared network is displayed. Otherwise, if the instance has one or more interfaces on IP networks, the public IP address associated with the interface that is used as the default gateway is displayed.

You can start, stop, or reboot an instance by selecting the instance and clicking the appropriate icon at the top of the list.

To list your instances using the CLI, use the `opc compute instance list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To list your instances using the API, use the `GET /instance/container/` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Monitoring Instances


After creating instances in Compute Classic, you can view a list of your instances and get details of each instance.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. The Instances page shows a list of instances, along with information about each instance.

### Tip:

You can filter the list of instances according to their category or status. To list instances with a specific status (such as running, error, or stopped), click the **Show** menu and select the appropriate filter. To view instances of a specific category (such as PaaS, IaaS, or personal), click the **Category** menu and select the appropriate filter.

3. Go to the instance that you want to view. From the  menu, select **View**.

The instance details page shows all the details of the selected instance, such as the public and private IP addresses associated with it and details of interfaces added to IP networks. You can stop, start, or reboot the instance by clicking the appropriate icon at the top of the page. This page also displays the orchestration used to create the instance, and the storage volumes, security lists, and SSH keys

associated with it. You can add or remove storage volumes and security lists from this page. For more information, see [Updating an Instance](#).

To view details of an instance using the CLI, use the `opc compute instance get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view details of an instance using the API, use the `GET /instance/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Viewing Instance Metrics

You can view real-time metrics of your Compute Classic instances.


You can access the Monitoring page for your Compute Classic account from the Infrastructure Classic Console. On this page, you can select the type of metric you want to view and to specify up to five instances for each metric. After you configure the metrics that you want to view, the graphs show you the selected metrics for the past two hours for the specified instances. These metrics are automatically updated every minute.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

To view instance metrics:

1. Sign in to the Oracle Cloud Infrastructure Classic Console at <https://cloud.oracle.com/sign-in>.

The Oracle Cloud Infrastructure Classic Console page is displayed.

2. Click  in the top left corner of the Dashboard.
3. Click **Monitoring**.

The Monitoring Metrics page of the Infrastructure Classic Console opens.

4. From the list of metrics, select the metrics that you want to see. You can select up to four metrics. The following metrics are available:
  - **CPU (%):** Indicates CPU utilization in percentage.
  - **Iostat Read (sectors):** Indicates the average number of sectors read in I/O operations per second.
  - **Iostat Write (sectors):** Indicates the average number of sectors written in I/O operations per second.
  - **Memory Percent (%):** Indicates memory utilization in percentage.
  - **Memory Usage (KB):** Indicates memory utilization in kilobytes.
  - **Memory Utilization Percentage (agent):** Indicates memory utilization in percentage, as reported by the OPC agent, if the agent is enabled. The memory utilization metrics reported by this agent are more accurate than the memory utilization reported by the Memory Percent metric. When the OPC agent isn't installed and enabled, no value is displayed. To install and enable the agent, see [Enabling Instance Metrics Collection](#).

- **Memory Utilization (agent):** Indicates memory utilization in kilobytes, as reported by the OPC agent, if the agent is enabled. The memory utilization metrics reported by this agent are more accurate than the memory utilization reported by the Memory Usage metric. When the OPC agent isn't installed and enabled, no value is displayed. To install and enable the agent, see [Enabling Instance Metrics Collection](#).
- **Network Rcvd (B/s):** Indicates the average network traffic received by the instance in bytes per second.
- **Network Sent (B/s):** Indicates the average network traffic sent by the instance in bytes per second.

For more information about selecting the instances and metrics that you want to view, see Monitoring Service Usage in *Managing and Monitoring Oracle Cloud*.

You can also view metrics using the Oracle Monitoring Cloud Service REST API. See [Rest API for Oracle Monitoring Cloud Service](#).

## Enabling Instance Metrics Collection

You can view instance-level metrics from the Monitoring page in the web console. This page shows statistics about I/O operations, CPU utilization, network traffic, and memory utilization on your instance. The `opc-init` package also includes an agent to collect metrics. This agent runs on the instance and provides highly accurate memory utilization statistics, based on data that is collected every minute.

- On instances created using Oracle-provided Oracle Linux images with release version 16.4.6 or later, the agent is installed and enabled by default.
- On instances created using Oracle-provided Oracle Linux images with release version 16.4.4, the agent is installed and disabled by default. To enable the agent, see [Enabling and Disabling the OPC Agent](#).
- On Oracle Linux instances created using older images, or on Oracle Linux instances created using private images where `opc-init` is not installed, the agent isn't installed. To install the agent, download and install `opc-init`. See [Downloading and Installing the OPC Agent on an Oracle Linux Instance](#).



### Note:

You can use the OPC agent only on Oracle Linux 6.x and Oracle Linux 7.x instances. This agent isn't supported on Windows or Solaris instances.

### Downloading and Installing the OPC Agent on an Oracle Linux Instance

If your instance was created using an Oracle-provided image with release version 16.3.6 or older, or if your instance uses a private machine image on which `opc-init` isn't installed, then to use the agent you must download and install `opc-init`.

1. Determine the version of Python on your instance.

```
python --version
```

2. Download the `opc-init` package from <http://www.oracle.com/technetwork/topics/cloud/downloads/opc-init-3096035.html>.



3. Extract the files in the package.
4. Copy the appropriate file to your instance.
  - `opc-init-py2.6-version.noarch.rpm` — for Linux with Python 2.6
  - `opc-init-py2.7-version.noarch.rpm` — for Linux with Python 2.7

Here, *version* indicates the release version of the `opc-init` package.

 **Note:**

The `opc-init` package includes a `.exe` file for installing `opc-init` on Windows. While you can use this file to install `opc-init` on a Windows instance, the OPC agent isn't supported on Windows and it won't be installed on a Windows instance.

5. Log in to your instance and check if `opc-init` is installed.

```
rpm -qa | grep opc-init
```

If this command doesn't return any output, it indicates that `opc-init` isn't installed on your instance.

6. Install the appropriate file.
  - If `opc-init` isn't installed on your instance, then, to install `opc-init` on an instance with Python 2.6:

```
sudo rpm -i opc-init-py2.6-version.noarch.rpm
```

- If an earlier version of `opc-init` is already installed on your instance, then use the upgrade option. For example, on an instance with Python 2.6:

```
sudo rpm -Uvh opc-init-py2.6-version.noarch.rpm
```

7. Verify the installation:

```
rpm -qa | grep opc-init
```

The output of this command should show the `opc-init` file that you just installed.

### Enabling and Disabling the OPC Agent

If you've downloaded and installed `opc-init` on your Oracle Linux instance, or if you've created an instance using an Oracle-provided Oracle Linux image with release version 16.4.6 or later, the OPC agent is installed on your instance and enabled by default. If you've created an instance using an Oracle-provided Oracle Linux image with release version 16.4.4, the OPC agent is installed and disabled by default.

- **On Oracle Linux 7.x**
  - To enable the agent:

```
systemctl start opc-guest-agent.service
```

- To disable the agent:

```
systemctl stop opc-guest-agent.service
```

- To check the status of the agent:

```
systemctl status opc-guest-agent.service
```

- **On Oracle Linux 6.x**

- To enable the agent:

```
sudo start opc-guest-agent-service
```

To check the status of the agent:

```
sudo status opc-guest-agent-service
```

When the agent is running, this command gives the following sample output:

```
opc-guest-agent-service start/running, process 1578
```

- To disable the agent:

```
sudo stop opc-guest-agent-service
```

To check the status of the agent:

```
sudo status opc-guest-agent-service
```

When the agent is stopped, this command gives the following output:

```
opc-guest-agent-service stop/waiting
```

- If you stop and restart, or delete and re-create, an instance that uses a persistent boot disk, you don't need to install `opc-init` again.
- If you delete and re-create an instance that uses a nonpersistent boot disk, and if `opc-init` wasn't installed by default on that instance, you must download and install `opc-init` again.

### Viewing the Metrics Reported by the OPC Agent

The following memory utilization metrics are reported by the OPC agent:

- **Memory Utilization Percentage (agent):** Indicates the actual memory utilization on the instance, in percentage.
- **Memory Utilization (agent):** Indicates the actual memory utilization on the instance, in kilobytes.

 **Note:**

When the OPC agent is disabled, no value is displayed for these metrics.


To view instance metrics, see [Viewing Instance Metrics](#).

## Viewing the Boot Log of an Instance

After creating an instance in Compute Classic, you can view the boot log of the instance. If there are any issues after an instance has started or while restarting an instance, viewing the boot log helps you to detect those issues.

Boot logs are enabled by default in Oracle-provided images from mid-May 2016 onwards. For instances created using any other images, boot logs might not be visible by default. In this case, you can log in to the instance and configure it to show boot logs on the web console. See [How can I access the boot log for my instance?](#)

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. The Instances page shows a list of instances. Go to the instance for which you want to view the boot log. From the  menu, select **View**.
3. On the instance details page, click the **Logs** tab in the left pane.

The Logs page displays the boot log for the instance. The output from the latest start or restart is displayed.

 **Note:**

Boot logs are truncated at 81920 bytes.

 **Note:**

If your instance isn't configured to display boot logs, the Logs page displays a message stating that no log was found.

The following is a sample output of a boot log, which has been truncated for readability:

```
user pid=8847 uid=0 auid=0 ses=674 msg='op=PAM:setcred acct="root" exe="/usr/
sbin/crond" hostname=? addr=? terminal=cron res=success'
[341545.271630] type=1106 audit(1462252801.375:4058): user pid=8847 uid=0 auid=0
```

```

ses=674 msg='op=PAM:session_close acct="root" exe="/usr/sbin/crond" hostname=?
addr=? terminal=cron res=success'
[342145.291725] type=1101 audit(1462253401.394:4059): user pid=8855 uid=0
aid=4294967295 ses=4294967295 msg='op=PAM:accounting acct="root" exe="/usr/sbin/
crond" hostname=? addr=? terminal=cron res=success'
[342145.315757] type=1103 audit(1462253401.418:4060): user pid=8855 uid=0
aid=4294967295 ses=4294967295 msg='op=PAM:setcred acct="root" exe="/usr/sbin/
crond" hostname=? addr=? terminal=cron res=success'
...
...
...
[345145.356807] type=1103 audit(1462256401.460:4096): user pid=8918 uid=0
aid=4294967295 ses=4294967295 msg='op=PAM:setcred acct="root" exe="/usr/sbin/
crond" hostname=? addr=? terminal=cron res=success'
[345145.372663] type=1006 audit(1462256401.476:4097): pid=8918 uid=0 old
aid=4294967295 new aid=0 old ses=4294967295 new ses=681
[345145.383251] type=1105 audit(1462256401.486:4098): user pid=8918 uid=0 aid=0
ses=681 msg='op=PAM:session_open acct="root" exe="/usr/sbin/crond" hostname=?
addr=? terminal=cron res=success'
[345145.440485] type=1104 audit(1462256401.543:4099): user pid=8918 uid=0 aid=0
ses=681 msg='op=PAM:setcred acct="root" exe="/usr/sbin/crond" hostname=? addr=?
terminal=cron res=success'
[345145.458101] type=1106 audit(1462256401.560:4100): user pid=8918 uid=0 aid=0
ses=681 msg='op=PAM:session_close acct="root" exe="/usr/sbin/crond" hostname=?
addr=? terminal=cron res=success'

```

To view instance console output using the CLI, use the `opc compute instance-console get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view instance console output using the API, use the `GET /instanceconsole/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.


## Generating a Screen Capture of an Instance

After creating an instance, if your instance is in the running state, you should be able to log in to your instance. However, if you're unable to access an instance that is in the running state and you need information to help you understand why you're not able to access the instance, you can generate a screen capture of the instance. The screen capture images are stored in the associated Oracle Storage Cloud Service account.

### Prerequisites

- Screen capture image files are stored in the associated Oracle Storage Cloud Service instance. Ensure that you've set a replication policy for your Oracle Storage Cloud Service instance. See *Selecting a Replication Policy for Oracle Storage Cloud Service in Using Oracle Storage Cloud Service*.
- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. The Instances page shows a list of instances. Go to the instance for which you want to generate a screen capture. From the  menu, select **View**.
3. On the instance details page, click the **Screen Captures** tab in the left pane. The Screen Captures page is displayed.
4. To generate a screen capture, click **Create Screen Capture**. A screen capture is created and displayed. It is also added to the list of screen captures.
5. To view an existing screen capture of the instance, select it from the list. Screen captures are listed in reverse chronological order, with the most recent screen captures on top.
6. If you no longer need a screen capture, you can delete it. The corresponding screen capture image in the associated Oracle Storage Cloud Service account is deleted.

To generate a screen capture of an instance using the API, use the `POST /console/screencapture/` method. You must specify the instance you want a screen capture of as well as the Oracle Storage Cloud Service account associated with your Oracle Compute Cloud Service account. The screen capture image is generated and saved in the associated Oracle Storage Cloud Service account. The path to access this screen capture file is returned in the response to your API call. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After you've generated a screen capture using the API, to view the file, download it from your Oracle Storage Cloud Service account. See *Downloading an Object in Using Oracle Storage Cloud Service*.

## Updating Packages on an Oracle Solaris Instance



This topic does not apply to Oracle Cloud at Customer.

When you create instances by using an Oracle-provided Oracle Solaris image, you get a support entitlement for Oracle Solaris. You can update packages from the support repository, file service requests to get support, and so on. The default IPS publisher, named `solaris`, is preconfigured to use the Oracle Solaris support repository (<https://pkg.oracle.com/solaris/support/>).

### Checking Whether the SSL Key and Certificate of the IPS Publisher Are Associated

1. Log in to your Oracle Solaris instance as the `opc` user:  

```
ssh opc@ip_address -i /path/to/private_key
```
2. Run the following command:  

```
pkg publisher solaris
```

- If the `SSL Key` and `SSL Cert` fields have values and if the certificate hasn't expired yet (see the `Cert. Expiration Date` field, as shown in the following example), then proceed to [Updating Packages](#).

```

Publisher: solaris
Alias:
Origin URI: https://pkg.oracle.com/solaris/support/
SSL Key: /var/pkg/ssl/
0ea8b04aa00e4ea1621aa66cab649778b67ef486
SSL Cert: /var/pkg/ssl/
66aac7c266473f285641fef2b8e6817248cb7f4e
Cert. Effective Date: March 27, 2016 09:10:48 AM
Cert. Expiration Date: April 4, 2018 09:10:48 AM
Client UUID: 0717ae7e-bb12-11e5-9a62-9bd968ceffe9
Catalog Updated: March 24, 2016 03:53:33 PM
Enabled: Yes

```

- If the `SSL Key` and `SSL Cert` fields show `None`, or if they show a value but the certificate has expired, then complete the steps in [Associating the SSL Key and Certificate for the IPS Publisher](#).

### Associating the SSL Key and Certificate for the IPS Publisher

You must complete the steps in this section if the `pkg publisher solaris` command shows that the SSL key and certificate are not associated, or if the command shows that the certificate is associated but has expired.

1. Go to <https://pkg-register.oracle.com/>.
2. Click **Request Certificates**.
3. On the **Available Repositories** page, look for the **Oracle Solaris 11 Support** row, and click **Request Access**.
4. Read and accept the My Oracle Support terms.
5. Go to <https://pkg-register.oracle.com/register/certificate/> and download your key and certificate to your local host.
6. Copy the key and certificate from your local host to your Oracle Solaris instance:
 

```
scp pkg.oracle.com.*.pem opc@ip_address:~
```

Here, `ip_address` is the public IP address of your Oracle Solaris instance. This command copies

`pkg.oracle.com.key.pem` and `pkg.oracle.com.certificate.pem` from your local host to the `/export/home/opc` directory of your Oracle Solaris instance.

7. Log in to your Oracle Solaris instance as the `opc` user:
 

```
ssh opc@ip_address -i /path/to/private_key
```
8. Assume the root role:
 

```
su -
```
9. Set up the publisher configuration:

```

pkg set-publisher \
-k /export/home/opc/pkg.oracle.com.key.pem \
-c /export/home/opc/pkg.oracle.com.certificate.pem \
-G "*" -g https://pkg.oracle.com/solaris/support/ solaris

```

## Updating Packages

1. Verify that the SSL key and certificate are set for the IPS publisher.  
See [Checking Whether the SSL Key and Certificate of the IPS Publisher Are Associated](#).
2. Update the packages:
  - To list the packages available in the repository:  
`pkg list -a 'pkg://solaris/*'`
  - To do a dry run of an update:  
`pkg update -nv`
  - To update all packages:  
`pkg update`
3. Wait for the update operation to be completed.  
After all the packages are updated, messages such as the following are displayed:

```

          Packages to install:  1
          Packages to update: 154
          Create boot environment: Yes
          Create backup boot environment: No

DOWNLOAD          PKGS          FILES          XFER (MB)    SPEED
Completed          155/155        7607/7607      307.0/307.0  3.0M/s

                                                    ITEMS
Removing old actions                732/732
Installing new actions              1317/1317
Updating modified actions            7658/7658
Updating package state database      Done
Updating package cache              154/154
Updating image state                 Done
Creating fast lookup database        Done
Updating package cache              1/1

```

```

A clone of solaris exists and has been updated and activated.
On the next boot the Boot Environment solaris-1 will be
mounted on '/'. Reboot when ready to switch to this updated BE.

```

```

Updating package cache              1/1

```

```

-----
NOTE: Please review release notes posted at:

```

```

https://support.oracle.com/rs?type=doc&id=2045311.1
-----

```

4. In the output, note the name of the new boot environment (BE), `solaris-1` in this example.
5. To verify that the new BE exists, run the following command:  
`beadm list`

Here's an example of the output of this command:

```
BE          Flags Mountpoint Space  Policy Created
--          -
solaris     N      -           58.68M static 2016-02-23 23:59
solaris-1   R      /           6.49G  static 2016-04-04 11:34
```

In this example, two BEs exist on the instance:

- The currently active `solaris` BE, indicated by the `N` (=active now) flag
  - The new `solaris-1` BE, indicated by the `R` (=active on reboot) flag
6. For the new BE to take effect, restart the instance. See [Rebooting an Instance](#).

## Updating an Instance

### Topics

- [Attaching a Public IP Address to an Instance on the Shared Network](#)
- [Attaching a Storage Volume to an Instance](#)
- [Detaching a Storage Volume from an Instance](#)
- [Adding an Instance to a Security List](#)
- [Removing an Instance from a Security List](#)
- [Resizing an Instance](#)

#### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

## Attaching a Public IP Address to an Instance on the Shared Network

You can associate a public IP address with an instance either while creating the instance, or while creating or updating an IP reservation. If you've already created an




instance with an interface on the shared network, then you can also associate a public IP address with the interface on the shared network by updating the instance.

 **Note:**

When you attach an IP reservation to a running instance, then if you delete and re-create or shut down and restart the instance, the IP reservation reverts to whatever was specified while creating the instance and any updates made to the IP reservation are lost. You must update the IP reservation again.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to associate a public IP address with. From the  menu, select **View**.
3. On the instance details page, click **Assign Public IP**.
4. In the Assign Public IP dialog box, the **Create IP Reservation** option is selected by default.
  - To create a new IP reservation and assign it to the instance, enter a name for the IP reservation and click **Assign**.
  - To select an existing IP reservation, deselect the **Create IP Reservation** check box, select an IP reservation from the **Existing IP Reservations** drop-down list, and then click **Assign**.

You can also associate an IP reservation with an instance when you create or update the IP reservation. See [Reserving a Public IP Address](#) or [Updating an IP Reservation](#).

Internally, an IP reservation is associated with an instance through the instance's `vcable`. A `vcable` provides an attachment point to a specific network interface on an instance. The `vcable` of an instance is created automatically when the instance is launched and is deleted when the instance is deleted.

The process of adding a virtual link between an instance and an IP reservation is also referred to as IP association.

To find out the `vcable` ID of your instance using the CLI, use the `opc compute instance get` command. To associate an IP reservation with an instance using the CLI, use the `opc compute ip-association add` command and specify the `vcable` ID. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To find out the `vcable` ID of your instance using the API, use the `GET /instance/name` method. To associate an IP reservation with an instance using the API, use the `POST /ip/association/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.


## Attaching a Storage Volume to an Instance

A **storage volume** is a virtual disk that provides persistent block storage space for instances in Compute Classic. You can provide or increase the block storage capacity for an instance by attaching storage volumes.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have created the storage volume that you want to attach to your instance. See [Creating a Storage Volume](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance to which you want to attach a storage volume. From the  menu, select **View**.
3. On the instance details page, click **Attach Storage Volume**.
4. Select the volume that you want to attach.
5. The **Attach as Disk #** field is filled automatically with the next available index at which the volume can be attached. You can leave this field at the automatically selected disk number or enter a higher number up to 10.

The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on.

Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.

6. Click **Attach**.

You can also attach a storage volume to a running instance from the **Storage** page. See [Attaching a Storage Volume to an Instance](#).

To attach a storage volume to an instance using the CLI, you must add a storage attachment object by using the `opc compute storage-attachment add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To attach a storage volume to an instance using the API, you must add a storage attachment object by using the `POST /storage/attachment` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After attaching a storage volume to an instance, to access the block storage, you must mount the storage volume on your instance. See [Mounting and Unmounting a Storage Volume](#).

## Detaching a Storage Volume from an Instance

When you no longer require access to a storage volume, you can unmount it and detach it from your instance.

After you detach a storage volume from an instance, you can no longer read from or write data to the storage volume, unless you attach the volume to any instance.

 **Note:**



You can't detach or delete a storage volume that was attached while creating an instance.

If you're sure that a storage volume is no longer required, then back up the data elsewhere and delete the storage volume.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- Ensure that you've unmounted the storage volume that you want to detach. See [Unmounting a Storage Volume](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to update. From the  menu, select **View**.
3. On the instance details page, identify the storage volume that you want to detach. From the  menu, select **Detach Storage Volume**.

To detach a storage volume from an instance using the CLI, you must remove a storage attachment object by using the `opc compute storage-attachment delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To detach a storage volume from an instance using the API, you must remove a storage attachment object, by using the `DELETE /storage/attachment/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Adding an Instance to a Security List

When you add an instance to a security list, the instance can communicate freely with all the other instances in the same security list. Any security rules that are defined for the security list are applicable to all the instances in that security list.

Internally, an instance is associated with security lists by using the instance's **vcable**, which provides an attachment point to a specific network interface on the instance. You can dynamically add or remove an instance from a security list, without stopping the instance.

You can add an instance to up to five security lists.


### ▲ Caution:

When you add an instance to a security list, all the security rules that use that security list—as either the source or destination—are applicable to the instance. Consider a security list that is the destination in two security rules, one rule that allows SSH access from the public Internet and another rule permitting HTTPS traffic from the public Internet. When you add an instance to this security list, the instance is accessible from the public Internet over both SSH *and* HTTPS. Keep this in mind when you decide the security lists that you want to add an instance to.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- You must have created the security list that you want to add your instance to. See [Creating a Security List](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to update. From the  menu, select **View**.
3. On the instance details page, click **Add to Security List**.
4. Select the security list that you want to add your instance to, and click **Attach**.

To add an instance to a security list using the CLI, you must first find out the `vcable` ID of the instance. To find out the `vcable` ID of an instance, use the `opc compute instance get` command. Next, to create an association between the `vcable` ID and the security list, use the `opc compute sec-association add` command and specify the `vcable` ID. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To add an instance to a security list using the API, you must first find out the `vcable` ID of the instance. To find out the `vcable` ID of an instance using the API, use the `GET /instance/name` method. Next, to create an association between the `vcable` ID and the security list, use the `POST /secassociation/` method and specify the `vcable` ID. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

 **Note:**

When an instance is deleted and re-created or shut down and restarted, any security lists to which you had added the instance manually (that is, not during instance creation), must be associated again.

## Removing an Instance from a Security List

To prevent other hosts from accessing an instance, you can remove the instance from the security lists that it is attached to. This may be required when you want to perform maintenance activities, change or upgrade applications, and so on.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.



Internally, an instance is associated with security lists by using the instance's `vcable`. When you add an instance to a security list, a security association is created between the `vcable` and the specified security list. To remove an instance from a security list, you must delete the security association that binds the instance to the security list.

 **Note:**

When you remove an instance from a security list, the security rules that are defined for the security list are no longer applicable to the instance, and the instance can't communicate with other instances in the security list. An instance that isn't associated with any security list is completely inaccessible.

When an instance that you had previously removed from the `/default/default` security list is deleted and re-created or shut down and restarted, you must remove the instance from the security list again after the instance starts.

To remove an instance from a security list using the web console:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to update. From the  menu, select **View**.
3. On the instance details page, go to the security list that you want to remove your instance from. From the  menu, select **Remove from Security List**.

To remove an instance from a security list using the CLI, you must remove a security association by using the `opc compute sec-association delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To remove an instance from a security list using the API, you must remove a security association, by using the `DELETE /secassociation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Resizing an Instance

A **shape** is a resource profile that specifies the number of OCPUs and the amount of memory to be allocated to an instance in Compute Classic. The shape determines the type of disk drive that your instance uses. You specify the shape of an instance while creating the instance. However, if your instance is managed by an orchestration, then you can change the shape of an instance even after the instance has been created. This is useful if you find that your application workload has increased and you would like to add OCPUs and memory to your instance.

Here's an overview of the process for resizing an instance:

1. If your instance was created using orchestrations v2, suspend the orchestration and ensure that the instance you want to resize is nonpersistent. If your instance was created using orchestrations v1, terminate the instance orchestration v1 to delete the instance.

### **Caution:**

If you delete an instance that uses a nonpersistent boot disk, any changes that you may have made to the boot disk after the instance was created are lost.

### **Note:**

If you want to change the shape of an instance that you have created using orchestration v1, ensure that you terminate only the instance orchestration and not the master orchestration. That way, only your instance is deleted and re-created and storage volumes or other resources defined in other orchestrations are not deleted.


2. Update the orchestration with the required shape.
3. Start the orchestration. This re-creates the instance with the updated instance configuration.


To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify and note the name of the instance that you want to update.
3. Go to the orchestration that controls the instance that you want to delete.

 **Caution:**

If your instance was created using Orchestrations v1, ensure that you don't unintentionally go to the master orchestration or any other orchestration. Stopping the master orchestration or a storage volume orchestration will cause storage volumes or other resources defined in other orchestrations to be deleted.

4. If your instance was created using orchestrations v1, then:
  - a. From the  menu, select **Terminate**.


The status of the orchestration changes to **Stopping**. After all objects have been deleted, the status of the orchestration changes to **Stopped**.
  - b. After the orchestration status changes to **Stopped**, from the  menu, select **Resize Instance**.



 **Note:**


This option is shown only for orchestrations that define instances. Master orchestrations or orchestrations that contain only storage volumes or other objects don't display the **Resize Instance** option.

- c. In the Resize an Instance dialog box, if your orchestration contains multiple instances, select the appropriate instance. If your orchestration contains only one instance, it is selected by default. Select the shape you want to use and then click **Resize**.

The orchestration is updated with the specified shape.

5. If your instance was created using orchestrations v2, then:
  - a. From the  menu, select **Suspend**.

The status of the orchestration changes to **Suspending**. After all nonpersistent objects have been deleted, the status of the orchestration changes to **Suspended**.
  - b. After the orchestration status changes to **Suspended**, from the  menu, select **Update**.
  - c. On the orchestrations details page, in the Instance section, go to the instance that you want to modify. From the  menu, select **Properties**.

- d. In the Object Properties dialog box, ensure that the **Persistent** check box isn't selected. If it is selected, deselect it, then click **Update**. This ensures that the status of the instance changes to **Inactive**.
- e. On the orchestrations details page, in the Instance section, go to the instance that you want to modify. From the  menu, select **Update**.
- f. On the Instance Details page, select the **Shape** that you want to use, and then click **Update**.

The orchestration is updated with the specified shape.

6. Start the orchestration. This re-creates the instance with the updated instance configuration.

The orchestration is started and the instance is re-created using the specified shape.

To verify the shape your instance uses, you can view the appropriate orchestration, or after the instance is running, go to the Instances page and view the details of the instance.

To use the CLI to change the shape of an instance that you have created using orchestration v1, stop, update, and restart the instance orchestration v1 using the `opc compute orchestration update --action STOP`, `opc compute orchestration update`, and `opc compute orchestration update --action START` commands. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To use the API to change the shape of an instance that you have created using orchestration v1, use the `PUT /orchestration/name` method with the query argument `action=STOP` to stop the orchestration v1. To update the orchestration, use the `PUT /orchestration/name` method with the updated instance shape. Finally, to restart the orchestration, use the `PUT /orchestration/name` method with the query argument `action=START`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing Instance Lifecycle Operations

When your instance is running, you can use the web console to restart it, delete it, or delete and re-create it.

### Topics

- [Rebooting an Instance](#)
- [Shutting Down and Restarting an Instance](#)
- [Deleting an Instance](#)
- [Deleting and Re-creating an Instance](#)



## Rebooting an Instance


After your instance is running, if required, you can reboot your instance from the web console.

### **WARNING:**

When you shut down or reboot an instance, you might lose data on any nonpersistent boot disks, including NVMe SSD disks, that are attached automatically as part of the high I/O shapes.

When you reboot an instance, data on storage volumes (whether persistent or nonpersistent) isn't lost. Your instance also retains all its configuration information, such as its public IP address and storage volumes that were attached and mounted on the instance.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, go to the instance that you want to reboot. From the  menu, select **Reboot**.

The Reboot Instance dialog box appears.

3. (Optional.) If the instance hangs after it starts running, select the **Hard Reboot** check box to perform a hard reset of the instance.
4. Click **Yes** to reboot the instance.

To reboot an instance using the CLI, use the `opc compute reboot-instance-request` add command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To reboot an instance using the API, use the `POST /rebootinstancerequest/` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Shutting Down and Restarting an Instance

If you created an instance using a persistent bootable storage volume, then, if you don't need the instance, you can shut down the instance. However, the instance isn't deleted. After shutting down an instance, you can restart the instance later, without losing any of the instance data or configuration.

If you no longer need an instance, you can *delete* the instance by terminating or suspending the relevant orchestration. You can always re-create an instance by starting the appropriate orchestration later.

*Shutting down an instance* is useful when you've created multiple instances in a single orchestration. In this case, stopping or suspending the orchestration would cause all instances to be deleted. If you want to shut down one or more instances, while letting other instances in the same orchestration run, you can shut down the required instances individually.

 **WARNING:**

Shut down or restart the instance using the Compute Classic web console, REST API, or CLI. Do not log in to the instance, and then use the `halt`, `shutdown` or `shutdown -h` commands to shut down the instance. Doing so will stop the instance indefinitely and will require manual intervention by Oracle Cloud system administrators to restart the compute node.

Here's what happens when you shut down an instance:

 **WARNING:**

When you shut down or reboot an instance, you might lose data on any nonpersistent boot disks, including NVMe SSD disks, that are attached automatically as part of the high I/O shapes.

- The instance ID is retained and reused when you restart the instance. So the multipart instance name doesn't change. This is useful in case the instance name is referenced by other objects, such as storage attachments.
- For instances created using orchestrations v1, the instance orchestration shows an error. However, even if the HA policy specified is `active`, the instance isn't automatically re-created.
- For instances created using orchestrations v2, the orchestration doesn't show an error. The orchestration JSON shows the status of the instance as `shutdown`.
- The resources associated with that instance, such as OCPUs, and IP reservations, are freed up and can be used by other instances if required. However, if you attempt to restart an instance, ensure that the required resources are available, otherwise the instance can't restart and will go into an error state.
- The private IP address on the shared network is released. If you restart the instance later, it is allotted a private IP address afresh. So the private IP address of the instance on the shared network is likely to change.
- Dynamically allocated IP addresses on IP networks are also released. So if you start the instance later, dynamically allocated IP addresses on IP networks are also likely to change. Static private IP addresses that are allocated to the instance interfaces in the orchestration won't change.
- Any changes that you'd made to the instance in Compute Classic after the instance was created might be lost. For example, if you added the instance to security lists, attached storage volumes to the instance, or detached and attached an IP reservation, you might need to make those changes again.

 **Note:**


Changes made to the instance by logging in to the instance won't be lost, however, as these are preserved on the persistent storage volumes attached to the instance. Data on storage volumes isn't affected by stopping an instance.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- Ensure that the instance that you want to shut down uses a persistent boot disk.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. On the Instances page, go to the instance that you want to stop. From the  menu, select **Shut Down**.

While the instance is being shut down, its status changes to **Stopping**. When the instance has shut down, it continues to be listed on the Instances page with the status **Stopped**.

3. After the instance has shut down, to start the instance again, on the Instances page, go to the instance that you want to restart. From the  menu, select **Start**.

To shut down an instance using the CLI, use the `opc compute instance update [--desired-state shutdown]` command. To restart an instance, use the `opc compute instance update [--desired-state running]` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To shut down an instance using the API, use the `PUT /instance/name` method and specify the `desired_state` as `shutdown`. To restart an instance, use the `PUT /instance/name` method and specify the `desired_state` as `running`. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting an Instance

Instances created using the web console are managed by orchestrations. To delete such an instance, you must terminate the corresponding orchestration. If your instance was created by using the Create Instance wizard before mid-May 2016, or if it was created by defining a launch plan using the API, then it is not managed by an orchestration. In this case, you can delete the instance directly from the Instances page of the web console.

If you want to delete an instance created using orchestrations v2, and you want to retain the data on the attached storage volumes, then ensure that your instance has

persistence specified as false and that the relevant storage volumes and other objects in the orchestration have persistence specified as true. This allows you to suspend the orchestration, which deletes only nonpersistent objects while persistent objects are preserved.

If you want to delete an instance created using orchestrations v1, and you want to retain the data on the attached storage volumes, ensure that you terminate only the instance orchestration, not the master orchestration. If you terminate the master orchestration, both the instance and other resources such as storage volumes that were created while creating the instance will be deleted. If you start the master orchestration again later on to re-create the instance, the storage volumes will be re-created; however, the data that you had written to those storage volumes will be lost.

 **Caution:**

If you delete an instance that uses a nonpersistent boot disk, any changes you may have made to the boot disk after the instance was created are lost. If you want to preserve changes to the nonpersistent boot disk, you can create an instance snapshot. See [Cloning an Instance by Using Instance Snapshots](#).

Here's what happens when you delete an instance:

 **WARNING:**

When you shut down or reboot an instance, you might lose data on any nonpersistent boot disks, including NVMe SSD disks, that are attached automatically as part of the high I/O shapes.

- The resources associated with that instance, such as storage volumes and IP reservations, are freed up and can be used by other instances if required. However, if you attempt to re-create an instance, ensure that the required resources are available, otherwise the instance can't be created. The instance orchestration will go into an error state.
- The private IP address on the shared network is released and might be allotted to another instance. If you re-create the instance later, it is allotted a private IP address afresh. So the private IP address of the instance on the shared network is likely to change.
- Dynamically allocated IP addresses on IP networks are also released. So if you re-create the instance, dynamically allocated private IP addresses on IP networks are also likely to change. Static private IP addresses that are allocated to instance interfaces in the orchestration won't change.
- Any changes that you'd made to the instance in Compute Classic after the instance was created will be lost. For example, if you added the instance to security lists, attached storage volumes to the instance, or detached and attached an IP reservation, you'll need to make those changes again. The instance will be re-created with the resources that are associated with it in the orchestration.


 **Note:**

If the instance uses a persistent boot disk, changes made to the instance by logging in to the instance won't be lost when you delete an instance.



### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- Any storage volumes that were attached to an instance after the instance was created, are detached (but not deleted) when you delete the instance. You must unmount these storage volumes before deleting an instance. See [Mounting and Unmounting a Storage Volume](#).


### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to delete.
3. (Option 1) If your instance isn't managed by an orchestration, you can delete it now. From the  menu, select **Delete**.

The instance status changes to **Stopping**. After the instance shuts down, it is deleted.

4. (Option 2) If your instance is managed by an orchestration, you must terminate the orchestration to delete the instance.
  - a. (Optional) To find out the name of the orchestration that controls the instance, go to the instance details page. The orchestration name is displayed in the top pane. Make a note of the orchestration name.
  - b. Click the **Orchestrations** tab.
  - c. Go to the orchestration that controls the instance that you want to delete. You can view the orchestration to see the objects created by that orchestration. To view the orchestration, from the  menu, select **View**.
  - d. To suspend orchestration v2, on the Orchestrations page, from the  menu, select **Suspend**.

The status of the orchestration changes to **Suspending**. After all the non-persistent objects have been deleted, the status of the orchestration changes to **Suspended**.

- e. To terminate orchestration v1, on the Orchestrations page, from the  menu, select **Terminate**. Remember, when you terminate an orchestration v1, all the objects created by that orchestration are deleted.

 **Note:**

If you want to retain the data on any attached storage volumes, ensure that you terminate only the instance orchestration, not the master orchestration. If you terminate the master orchestration, both the instance and other resources such as storage volumes that were created while creating the instance will be deleted.

The status of the orchestration changes to **Stopping**. After all objects have been deleted, the status of the orchestration changes to **Stopped**.

If you created an instance using a launch plan, then to delete that instance using the CLI, use the `opc compute instance delete` command. Otherwise, to delete an instance by terminating an orchestration v1, use the `opc compute orchestration update --action STOP` command. To delete an instance by terminating an orchestration v2, use the `opc compute orchestration-v2 update [--desired-state inactive]` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

If you created an instance using a launch plan, then to delete that instance using the API, use the `DELETE /instance/name` method. To delete a nonpersistent instance by suspending an orchestration v2, use the `PUT /platform/v1/orchestration` method with the query argument `desired_state=suspend`. Otherwise, to delete an instance by terminating an orchestration v1, use the `PUT /orchestration/name` method with the query argument `action=STOP`. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting and Re-creating an Instance

After creating an instance, if you no longer need the instance, you can delete it. If you want to use the same instance again later on, you can re-create the instance.

 **WARNING:**

When you shut down or reboot an instance, you might lose data on any nonpersistent boot disks, including NVMe SSD disks, that are attached automatically as part of the high I/O shapes.

All instances that are created by using the Create Instance wizard, or by adding an orchestration, can be deleted and re-created by terminating and starting the orchestration that controls the instance.


When you create an instance using the Create Instance wizard, a single orchestration v2 is created automatically to manage the instance and its associated resources. Storage volumes and networking objects used by the instance are created in the same orchestration. Instances are nonpersistent by default. However, storage volumes and other objects are created with persistence set to true, so that if you suspend the orchestration, instances are shut down, but storage volumes aren't deleted. Terminating the orchestration, however, will cause all objects to be deleted and any data on storage volumes will be lost.

Earlier, when you created an instance using the Create Instance wizard, one or more orchestrations v1 were created automatically to manage the instance and its associated resources. For example, if you used the Create Instance wizard to create an instance and attach a new storage volume to it, then two separate orchestrations were created, one for the instance and the other for the storage volume. A master orchestration was also created, and the instance and storage volume orchestrations were referenced as objects in the master orchestration.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- Any storage volumes that were attached to an instance after the instance was created, are detached (but not deleted) when you delete the instance. You must unmount these storage volumes before deleting an instance. See [Unmounting a Storage Volume from a Linux Instance](#), [Unmounting a Storage Volume from an Oracle Solaris Instance](#), or [Unmounting a Storage Volume from a Windows Instance](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. On the Instances page, identify the instance that you want to delete.
3. Click the **Orchestrations** tab.
4. Go to the orchestration that controls the instance that you want to delete.
  - If your instance was created using orchestration v1, then from the  menu, select **Terminate**. The status of the orchestration changes to **Stopping**. After all objects have been deleted, the status of the orchestration changes to **Stopped**.

#### **Caution:**


If you use the Create Instance wizard to create an instance and the required storage volumes, then separate orchestrations are automatically created for the instance and the storage volumes. When you delete an instance, if you want to retain the data on the attached storage volumes, ensure that you terminate only the instance orchestration, not the master orchestration. If you terminate the master orchestration, both the instance and the storage volumes that were created while creating the instance will be deleted. When you start the master orchestration again later on to re-create the instance, the storage volumes will be re-created; however, the data you had written to those storage volumes will be lost.

- If your instance was created using orchestrations v2, then from the  menu, select **Suspend**. The status of the orchestration changes to **Suspending**.

After all nonpersistent objects have been deleted, the status of the orchestration changes to **Suspended**.

**⚠ Caution:**

If you terminate the orchestration instead of suspending it, all objects created by the orchestration are deleted including persistent objects such as storage volumes.

5. When you are ready to re-create the instance, on the Orchestration page, go to the orchestration that controls the instance that you want to re-create. From the  menu, select **Start**.

The status of the orchestration changes to **Starting**. After all objects have been created, the status of the orchestration changes to **Ready**.

**⚠ Caution:**

If your instance was created using orchestrations v1 and you stop the master orchestration, all data on the storage volumes created by the **vm1\_storage** orchestration is deleted. In this case, to re-create the instance along with the required storage volumes and any other resources, you must start the master orchestration. All the required resources will be created afresh.

To use the CLI to delete and re-create an instance that is managed by an orchestration, use the `opc compute orchestration-v2 update --action STOP` command followed by the `opc compute orchestration-v2 update --action START` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To use the API to delete and re-create an instance that is managed by an orchestration, use the `PUT /orchestration/name` method with the query argument `action=STOP` followed by the `PUT /orchestration/name` method with the query argument `action=START`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Retrieving Instance Metadata

### Topics

- [About Instance Metadata](#)
- [Retrieving Predefined Instance Metadata](#)
- [Retrieving User-Defined Instance Attributes](#)
- [Sample Scenario for Specifying and Using Instance Attributes](#)



## About Instance Metadata

Two types of metadata are stored within your instances: *user-defined instance attributes* that you can define explicitly while creating instances, and *predefined instance metadata* fields that are stored by default for all instances. Scripts and applications running on the instances can use the available metadata to perform certain tasks. For example, SSH public keys that are specified while creating an instance are stored as metadata on the instance. A script running on the instance can retrieve these keys and append them to the `authorized_keys` file of specified users to allow key-based login to the instance using `ssh`.

### Predefined Instance Metadata

The following predefined metadata fields are stored on every instance that you create:

Metadata	Description	Example
<code>local-ipv4</code>	Private IP address of the instance.	10.196.47.210
<code>local-hostname</code>	DNS name of the instance.	bd6032.acme.oraclecloud.example.com
<code>instance-id</code>	Name of the instance.	/Compute-acme/ joe.jonathan@example.com/debc974c-852e-4bd2-acd6-45a2de2109fd
<code>instance-type</code>	Memory and CPU resources available for the instance.	7680 ram, 2.0 cpus
<code>public-keys/{index}/openssh-key</code>	SSH public key specified while creating the instance, where {index} is a number starting with 0.	ssh-rsa AAAAB3NzaClyc2EAAAABI.. . == admin@acme

#### Note:

You may see certain additional metadata fields, such as `reservation-id`, `product-codes`, `kernel-id`, `public-hostname`, and `security-groups`, that aren't documented. Don't retrieve and use the values in the undocumented fields.

The predefined instance metadata fields are stored at `http://192.0.0.192/{version}/metadata`.

The following metadata versions are currently available:

```
latest
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2009-04-04
```

**Tip:**

New metadata versions may be released in the future. Metadata versions may not be backward compatible. So use metadata from a specific version (for example, from `http://192.0.0.192/2008-02-01/`) and not from `http://192.0.0.192/latest/`.

For the steps to retrieve the predefined instance metadata, see [Retrieving Predefined Instance Metadata](#).

### User-Defined Instance Attributes

User-defined attributes are key-value pairs that you can specify in the `attributes` parameter of machine images, image-list entries, and instance launch plans.

When you create instances, all the attributes that are specified in the `attributes` parameter in the orchestration or launch plan, machine image, and image list entry that are used to create your instances are stored on those instances. If an attribute in an image-list entry has the same name as an attribute in the machine image corresponding to that image-list entry, then the attribute in the image-list entry overrides the attribute in the machine image. Similarly, if an attribute in a launch plan has the same name as an attribute in an image-list entry or a machine image, then the attribute in the launch plan takes precedence.

User-defined instance attributes are stored within the instance at `http://192.0.0.192/latest/user-data`. For the steps to retrieve these attributes, see [Retrieving User-Defined Instance Attributes](#).

The following are a few sample use cases for user-defined instance attributes:

- If you want identical user data to be available on a set of instances, then specify the required user data in the machine image or image list entry that you'll use to create the instances. For example, you might require a certain pre-bootstrap script to be executed or specific applications to be installed on all instances that use a particular image. By specifying this script as user data in the machine image or the image list entry, you ensure that every instance that's created with that image has the specified user data.
- If each instance should have unique user data, use an orchestration to provide specific user data for each instance. This is useful if, for example, you want to specify a unique user name and password, or inject a unique SSH public key into each instance.
- You can automate instance configuration by providing scripts or other instructions to perform prebootstrapping tasks or install applications when you create an instance. These instance configuration instructions are provided as user-defined data using the `userdata` field under the `attributes` parameter. For example, you can use this field to specify the location of a database server and login details.

## Retrieving Predefined Instance Metadata

1. Log in to the instance.  
See [Logging In to an Instance](#).
2. Get a list of the available metadata versions by running the following command:

```
curl http://192.0.0.192
```

 **Note:**

The cURL commands provided in this document are for Linux and Oracle Solaris instances. On Windows instances, go to the PowerShell, and use the `Invoke-RestMethod` command instead of cURL.

3. From the list of versions displayed, select the version that you want to use.
4. Get a list of the top-level metadata fields:

```
curl http://192.0.0.192/{version}/meta-data
```

In this command, replace `{version}` with the version that you identified in the previous step.

**Example:**

```
curl http://192.0.0.192/2007-08-29/meta-data
```

5. Retrieve the specific metadata that you want, by running one of the following command examples:

 **Note:**

When you run these commands, replace `2007-08-29` with the metadata version that you want to use.

- To retrieve the private IP address of the instance:

```
curl http://192.0.0.192/2007-08-29/meta-data/local-ipv4
10.106.15.70
```

- To retrieve the host name of the instance:

```
curl http://192.0.0.192/2007-08-29/meta-data/local-hostname
bd6032.acme.oraclecloud.com
```

- To retrieve information about the memory and CPU resources of the instance:

```
curl http://192.0.0.192/2007-08-29/meta-data/instance-type
7680 ram, 2.0 cpus
```

- To retrieve the instance name:

```
curl http://192.0.0.192/2007-08-29/meta-data/instance-id
/Compute-acme/joe.jonathan@example.com/4c318760-444b-4b48-83e1-e1b112c201f2
```

- To find out how many SSH public keys are stored on the instance:

```
curl http://192.0.0.192/2007-08-29/meta-data/public-keys
0
1
2
```

In this example, three SSH public keys are stored as metadata, with index numbers 0, 1, and 2.

- To retrieve the value of a specific SSH public key:

```
curl http://192.0.0.192/2007-08-29/meta-data/public-keys/0/openssh-key  
ssh-rsa AAAAB3NzaC1yc2EAAAABI... == joe.jonathan@acme.com
```

## Retrieving User-Defined Instance Attributes

1. Log in to the instance using SSH.  
See [Logging In to an Instance](#).
2. Get a list of all the top-level attributes that are specified for the instance, by running the following command:

```
curl http://192.0.0.192/latest/user-data
```

### Note:

The cURL commands provided in this document are for Linux and Oracle Solaris instances. On Windows instances, go to the PowerShell, and use the `Invoke-RestMethod` command instead of cURL.

The following is an example of the output:

```
pre-bootstrap  
packages
```

In this example, the output shows that the instance has two top-level user-defined attributes: `pre-bootstrap` and `packages`.

3. To retrieve the attributes defined under the top-level `pre-bootstrap` attribute, run the following command:

```
curl http://192.0.0.192/latest/user-data/{topLevelAttribute}
```

### Example:

```
curl http://192.0.0.192/latest/user-data/pre-bootstrap
```

The following sample output indicates that two attributes are specified under the `pre-bootstrap` attribute:

```
failonerror  
scriptURL
```

4. Run the same command for successive levels of attributes until you get the required attribute value, as shown in the following example:

```
curl http://192.0.0.192/latest/user-data/pre-bootstrap/failonerror  
true
```

## Sample Scenario for Specifying and Using Instance Attributes

Consider a distributed system where a *manager* instance must handle requests from a set of *worker* instances. The instances in this sample scenario are identical in all other respects. So they're based on the same machine image.

Create an image list containing two entries, both for the same machine image, but one entry with the attribute `{"role": "manager"}` and the other with the attribute `{"role": "worker"}` in the `attributes` field. To create an image list entry using the API, use the `POST /imagelist/name/entry` method. See *REST API for Oracle Cloud Infrastructure Compute Classic* .

In the launch plan that you use to provision the instances in the distributed system, define a number of worker instances that use the image list entry with the `{"role": "worker"}` attribute, and define a manager instance that uses the image list entry with the `{"role": "manager"}` attribute.

After the instances are created, the software running on each instance can determine the role that the instance should play based on the value of the `role` attribute stored at `http://192.0.0.192/version/user-data`.

# 5

## Managing Resources Using the Visual Object Editor

In Compute Classic, you can create a large number of various types of resources. Apart from multiple instances and the associated storage volumes, you can also create numerous networking objects such as IP networks, vNICsets, security rules, access control lists, IP reservations and so on. Each instance can be associated with multiple objects. For example a single instance might be associated with multiple storage volumes, might have multiple IP addresses, and it might be added to multiple IP networks, as well as to the shared network. You can use the REST API, CLI, and the web console to view your instances and other resources and to get detailed information about each object. However, when you want to understand how certain objects relate to each other, or if you want a holistic picture of a large number of resources in your account, you might find it easier to grasp this information if it's provided in a visual or graphical format. This is where the visual object editor comes in.

### About the Visual Object Editor

The visual object editor provides you a graphical layout of the instances, storage volumes, and networking objects in your site. When you view the objects in the visual object editor, you can clearly identify relationships between instances and associated resources such as storage volumes and networking objects. You can also move objects around to customize the layout, apply a filter to view selected objects, and view details of objects or update them.

### Objects Displayed in the Visual Object Editor

Here's the list of objects that are displayed by the visual object editor:


- Instances
- Storage volumes
- IP networks
- IP network exchanges
- vNICsets
- Access control lists
- Security Rules for IP networks
- Security Protocols for IP networks
- IP address prefix sets
- Public IP addresses on the shared network that are associated with instances and IP reservations
- Public IP addresses and cloud IP addresses on IP networks
- VPN connections

The display also uses connecting lines to indicate relationships between objects. For example, connecting lines are used to indicate instance interfaces on the shared network or the public Internet, storage volumes attached to instances, IP networks added to IP network exchanges and so on.

### Accessing the Visual Object Editor

To get to the visual object editor:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click **Visualization** in the top right corner.

The Visualization page is displayed. You can view objects in your account or create new objects. Right-click objects to see the available options for each object. Click the  menu for options to refresh, save, or reset your view, or to show or hide filtered objects.

The palette on the left displays, in different sections, the objects that you can create. You can expand and close these sections based on your requirement.

#### Note:

If you create or modify objects outside the visual object editor, by using the web console, REST API, or the CLI, then you must refresh the visual object editor to display those changes.

### Viewing Objects Using the Visual Object Editor

Here's how you can use the visual object editor to find out more about individual objects or groups of objects:

- **View information about all your resources in a single pane**

The visual object editor makes it easy for you to see details of all your resources by moving your mouse over any object. Some object types have a status icon to indicate whether they are ready or in an error state.

Connections between objects are indicated by dotted or solid lines. Solid lines are used to indicate connections where data is transferred, for example from an instance to a storage volume or from an instance to an IP network.

When an object is connected to a specified target, but the target doesn't currently exist, a floating connection is shown. For example, if an IP network is added to an IP network exchange, and the specified IP network exchange is later deleted, then the IP network shows a floating connecting line.

#### Tip:

When you're done viewing information about an object, you can use the **Esc** key to close the tool tip.


- **View details of each object**


When you hover over an object, details of that object are displayed. For some object types, such as instances and storage volumes, you can view details of the object. To do this, right click an object and select the appropriate option. The page with detailed information about that object is displayed.

- **Zoom in for details, zoom out to get the big picture**

You can use the zoom buttons to change the level of magnification of your view. If there are a lot of objects in your account, you can zoom out to see many objects in a single page without scrolling. On the other hand, if you need to view details, you can zoom in to one section of the layout.

- **Maximize your view**

From the  menu, select **Maximise** or **Hide Palette** to increase the size of the visual object editor. When you use the **Maximise** option, it hides the header and footer of the Compute Classic web console and displays the editor on the entire available space. To restore the header and footer, select **Minimise**.

When you don't want to add an object, From the  menu, select **Hide Palette** to hide the palette so that the visual object editor is displayed in the entire available space. To display the palette again, use select **Show Palette**.

- **Filter your view**

You can use the **Category** and the **Show** menus to select one or more objects that you want to view, and either hide or gray out other items. You can also right click an object to add it to or remove it from the set of objects to be displayed. Objects directly associated with the selected objects aren't filtered out. This enables you to focus on the relationships of selected objects with associated objects. For example, if you set the filter to display a specific instance, you can view storage volumes and networking objects associated with that instance, without being distracted by other resources that aren't associated with the selected instance.

 **Tip:**

To view two connected objects and filter out other objects, click and hold the mouse button on the connecting line between the objects. Other objects are grayed out and the two objects connected by the line are shown. To hide all other connecting lines, click and hold the **Alt** key and then click and hold the mouse button on the connecting line.

- **Customize your layout**

In the default layout, objects that are associated with other objects are shown in the top pane, while unassociated objects are shown in the bottom pane.

Two panels run across the layout from end to end. The top panel represents the shared network. Instances that have an interface on the shared network are connected to this panel. The second horizontal panel represents the Internet. Instances that have access to and from the Internet using a public IP address are connected to this panel.

You can change the layout to organize and group objects as you require.

- Select objects and move them around to create a better layout. Note that you can't move the panels representing the shared network and the Internet to the



left or right as these objects span the width of the layout. You can move these objects only upward or downward.



- Press **Shift** and click to select multiple objects. You can also click and drag the mouse to select a group of objects.

 **Note:**

If your selection includes either of the panels representing the shared network or the Internet, you won't be able to move the selection to the left or right. You can only move the selected objects upward or downward. To move objects to the left or right, ensure that you don't select the panels representing the shared network or the Internet.

- Make your layout more compact by stacking objects. When objects are stacked, you can move your mouse over any of the stacked objects to see its properties. Objects automatically move to the front of the stack when you mouse over them, and revert to their original position when you move the mouse away.
- Move selected objects to the back or bring them to the front. This is useful if you want to view a specific object when you've stacked objects. To move an object to the back or front of a stack, right click the object and select the appropriate option.
- Save your changes, so that you see the customized layout next time you load the Visualization page. You can also save the layout. All others domain users can view the saved layout.

To make your layout easily available for your reference at any time:

- \* From the  menu, select **Save As Image** to save your layout as an image in the JPEG or PNG file format. This makes your layout easily available for your reference at any time.
- \* From the  menu, select **Save as** to save your layout. You can save this layout as the default layout, overwrite an existing layout, or save the layout with a new name. And when you're done, if you don't like the revised layout, you can opt to revert to the default layout and delete the saved layout.

Select the **Save the preferences** check box to view and save multiple views of the same data. This allows you to zoom in to specific sections of the layout and save just that part of the layout along with your viewing preferences. You can save the viewing preferences that you have set while customizing your layout, such as zoom ratio, the values you have selected in the **Category** and the **Show** menus, and values entered in the search box. If you don't select the **Save the preferences** check box, your viewing preferences are not saved and the preferences are not changed when you reload a layout.

After saving a layout, you can load it and switch between the different layouts that you have saved.

## Creating and Updating Objects Using the Visual Object Editor

Here's how you can use the visual object editor to create and manage objects.

- **Create objects**

Select an object from the palette on the left and drag and drop it on to the canvas in the center pane. A dialog box opens. Enter the required information and click **Create** to create the object.

 **Tip:**

If you don't want to create the object, you can press the **Esc** key or click **Cancel** to exit the dialog box.

- **Create connections between objects**

For some objects, such as IP networks, IP exchanges, storage volumes or instances, there's a yellow highlight on the object's icon and a plus sign appears when you hover over the object's icon. This indicates that you can create a connection from this object to a corresponding object by dragging your mouse to the target object. For example, you can add an IP network to an IP exchange by hovering over the required IP network till the plus sign appears, and then dragging the mouse to the required IP exchange. The connection is created and a connecting line joins the two objects.

- **Reassign existing connections**

You can also change connections between objects by clicking a connecting point and dragging it to a new object. For example, if you want to change the IP exchange that an IP network is added to, you can drag the connecting point of that IP network from the current IP exchange to another IP exchange. Not all types of connections can be reassigned in the visual object editor. For objects where a connection can be reassigned, the cursor changes to a plus sign when you hover over the connecting point.

 **Note:**

Connections that can be created or reassigned by using the drag-and-drop operation can't be specified or modified in the Create or Update dialog box of the corresponding object. For example, to add an IP network to an IP network exchange, use the drag-and-drop operation. In the visual object editor, you can't create or modify connections, such as from an IP network exchange to an IP network, by using the Create IP Network or Update IP Network dialog box.

- **Update objects**

You can update some object types, such as IP networks, IP network exchanges, access control list, vNICset, security rules for IP network, security protocol for IP network, IP address prefix sets, and IP reservations. Right click the object and select **Update**.

 **Tip:**

When you right click an object, if the menu doesn't display the **Update** option, it means that you can't update that type of object in the visual object editor.

When a connection can be reassigned by using the drag-and-drop operation, it can't be specified in the Create or Update dialog box of the corresponding object.

- **Create storage snapshots and storage snapshot schedules**

You can right click a storage volume and select the option to create a storage snapshot or a storage snapshot schedule.

- **Delete objects and connections**

Right click an object or a connecting line to delete the object or the connection between objects.

 **Note:**

The delete operation isn't allowed for all object types or all connections. When you right click an object or a connecting line, if the menu doesn't display the **Delete** option, it means that you can't delete that type of object in the visual object editor.

# 6

## Managing Resources Using Orchestrations v1

### Topics

- [About Orchestrations v1](#)
- [Orchestrations v1 Templates](#)
- [Workflow for Creating Instances Using Orchestrations v1](#)
- [Building Your First Orchestration v1](#)
- [Attributes in Orchestrations v1](#)
- [Uploading an Orchestration v1](#)
- [Orchestrations v1 Life Cycle](#)
- [Starting an Orchestration v1](#)
- [Monitoring Orchestrations v1](#)
- [Return Parameters Displayed in Orchestrations v1](#)
- [Terminating an Orchestration v1](#)
- [Downloading an Orchestration v1](#)
- [Updating an Orchestration v1](#)
- [Deleting an Orchestration v1](#)

## About Orchestrations v1

### Topics

- [What Is an Orchestration?](#)
- [Orchestration Terminology](#)
- [Object Types in an Orchestration](#)
- [Relationships Between Object Plans](#)
- [Relationships Between Objects Within a Launch Plan Object](#)
- [About Nested Orchestrations](#)
- [About High-Availability Policies in an Orchestration](#)

### What Is an Orchestration?

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

For example, you can use orchestrations to create and manage a collection of instances hosting a multitiered application stack with all the necessary networking, storage, and security settings.

A newer version of orchestrations, Orchestrations v2, offers a similar method of creating and managing resources. To understand the differences between the two versions and for an overview of the benefits of Orchestrations v2, see [Comparing Orchestrations v1 and Orchestrations v2](#).

At any time, you can delete and re-create all the instances in an orchestration just by terminating and restarting the orchestration. Storage attachments, security lists, and so on are re-associated automatically. When the HA policy in an orchestration is set to `active`, if an instance in such an orchestration goes down, the instance is re-created automatically.

Note that networking and storage objects needn't be defined in the same orchestrations that you use to create instances. You can define the networking and storage objects in separate orchestrations, and then refer to them in the orchestrations that define the instances. With this approach, you can remove and re-create instances independent of the associated resources.

To create instances using orchestrations, you build an orchestration in a JSON-formatted file, upload it to Compute Classic, and then start the orchestration. For a simple example of an orchestration file that you can use to learn how to build your first orchestration, see [Building Your First Orchestration v1](#). But before that, do read the remainder of this topic and become familiar with the features, terminology, and concepts of orchestrations.

### Orchestration Terminology

Term	Description
object plan ( <code>oplan</code> )	<p>An object plan, or <code>oplan</code>, is the primary building block of an orchestration.</p> <p>Each <code>oplan</code> contains all the attributes for the object type defined in that <code>oplan</code>.</p> <p>An orchestration can contain up to 10 object plans, and each <code>oplan</code> can include up to 10 objects.</p>
object type ( <code>obj_type</code> )	<p>An object type refers to the Compute Classic resource that you want to create.</p> <p>For example, if you want to create a storage volume, the <code>obj_type</code> would be <code>storage/volume</code>. If you want to create an instance, the <code>obj_type</code> would be <code>launchplan</code>.</p> <p>See <a href="#">Object Types in an Orchestration</a>.</p>
object ( <code>objects</code> )	<p>The <code>objects</code> attribute defines the properties or characteristics of the the Compute Classic resource that you want to create, as specified by the <code>obj_type</code> attribute.</p> <p>The fields in the <code>objects</code> section vary depending on the specified <code>obj_type</code>.</p> <p>For example, if you want to create a storage volume, the <code>obj_type</code> would be <code>storage/volume</code>, and the <code>objects</code> would include <code>size</code> and <code>bootable</code>. If you want to create an instance, the <code>obj_type</code> would be <code>launchplan</code>, and the <code>objects</code> would include <code>instances</code>, along with instance-specific attributes, such as <code>imagelist</code> and <code>shape</code>.</p>

For information about the attributes of each object type, see [Attributes in Orchestrations v1](#).

### Object Types in an Orchestration

In an orchestration, you can define any of the following object types:

Object Type	Description
integrations/osscontainer	Creates a container in the specified Oracle Cloud Infrastructure Object Storage Classic account.
ip/reservation	Reserves an IP address. To associate an IP reservation with an instance that's defined in the same orchestration, you must specify a relationship between the <code>ip/reservation</code> and the <code>launchplan</code> object plans.
launchplan	Creates an instance. To add an instance to a security list that's defined in the same orchestration, you must specify a relationship between the <code>launchplan</code> and the <code>seclist</code> object plans.
network/v1/acl	Creates an access control list (ACL) that can be applied to interfaces that are part of your IP networks.
network/v1/ipaddressprefixset	Creates an IP address prefix set. This can be used as a source or destination in security rules that determine access to or from the virtual interfaces of instances that are attached to IP networks.
network/v1/ipassociation	Associates a public IP address reservation with an interface on an instance that is attached to an IP network.
network/v1/ipnetwork	Creates an IP network. You can specify an IP network in the networking attributes while creating an instance.
network/v1/ipnetworkexchange	Creates an IP network exchange. You can add IP networks to an IP network exchange either while creating the IP network, or later, by updating the IP network.
network/v1/ipreservation	Reserves a public IP address from a specified IP pool. This IP address can be associated with the virtual interface of an instance that is attached to an IP network.
network/v1/route	Creates a route to a specified destination using the specified vNICset.
network/v1/secprotocol	Specifies a permitted transport protocol and ports. Security protocols are used in security rules which determine the permitted flow of traffic across your IP networks.
network/v1/secrule	Creates a security rule which can be added to an ACL. Security rules are applied through an ACL to control the permitted flow of traffic across your IP networks.
network/v1/vnicset	Creates a vNICset of one or more virtual network interfaces (vNICs). A vNICset is used to specify the next hop in a route. While creating an instance, you can specify if you want a vNIC to be added to either the shared network or an IP network.
orchestration	Starts a set of orchestrations. See <a href="#">About Nested Orchestrations</a> .
secapplication	Creates a security application. To use this security application in a security rule that's defined in the same orchestration, you must specify a relationship between these objects.

Object Type	Description
seclist	Creates a security IP list. To use this security IP list in a security rule that's defined in the same orchestration, you must specify a relationship between these objects.
seclist	Creates a security list. To use this security list in a security rule that's defined in the same orchestration, you must specify a relationship between these objects.
secrule	Creates a security rule. If this security rule uses security applications, security lists, or security IP lists that are defined in the same orchestration, then you must specify a relationship between these objects.
storage/volume	Creates a storage volume. To attach this storage volume to an instance that's defined in the same orchestration, you must specify a relationship between the storage/volume and the launchplan object plans.

An orchestration can contain up to 10 object plans, and each `oplan` can contain up to 10 objects.

An orchestration can also contain up to three levels of nested orchestrations. So you can use a single orchestration to manage many individual components. See [About Nested Orchestrations](#).

### Relationships Between Object Plans

You can use the `relationships` attribute in an orchestration to specify the sequence in which the objects in the orchestration must be created.

The `relationships` attribute specifies the two objects that have a relationship, identified by their `oplan` labels. It also specifies the relationship `type`, which is set to `depends`.

For example, if you define a storage volume in an orchestration and you also define an instance that the storage volume is attached to, then in the `relationships` section of the orchestration, you can specify that the `launchplan` object plan depends on the `storage/volume` object plan. This ensures that the storage volume is created before the instance is created.

So if you define a storage volume in an orchestration with the `oplan` label `storagevolume1`, and a launch plan with the `oplan` label `boot-from-storagevolume1`, then define the relationship between these objects as follows:

```
"relationships": [
  {
    "oplan": "boot-from-storagevolume1",
    "to_oplan": "storagevolume1",
    "type": "depends"
  }
]
```

For more complex scenarios, you can define multiple relationships.

For example, to create a security list (`seclist1`), a security application (`secapplication1`), and a security rule (`secrule1`) in a single orchestration, define the

following relationships to ensure that both the security application and the security list are created before the security rule:

```
"relationships": [
  {
    "oplan": "secrule1",
    "to_oplan": "seclist1",
    "type": "depends"
  },
  {
    "oplan": "secrule1",
    "to_oplan": "secapplication1",
    "type": "depends"
  }
]
```

### Relationships Between Objects Within a Launch Plan Object

You can also specify relationships within `launchplan` objects (that is, instances).

For example, if you define two instances with the labels `instanceA` and `instanceB` in an orchestration and you want those instances to be created on separate nodes, then in the `launchplan` object plan, define the relationship between the instances as follows:

```
"relationships": [
  {
    "instances": [
      "instanceA",
      "instanceB"
    ],
    "type": "different_node"
  }
]
```

The `type` attribute under `relationships` in a launch plan can have one of the following values:

- `same_node`: The specified instances are created on the same physical server. This is useful if you want to ensure low latency across instances.
- `different_node`: The specified instances aren't created on the same physical server. This is useful if you want to isolate instances for security or redundancy.

### About Nested Orchestrations

You can specify `orchestration` as an object type within an orchestration. You can use such an orchestration to start and terminate multiple other orchestrations.

For example, if you've defined the following orchestrations:

- `/Compute-acme/joe.jonathan@example.com/instances_orch`: An orchestration that defines multiple instances.
- `/Compute-acme/joe.jonathan@example.com/networking_orch`: An orchestration that defines networking objects such as security lists and security rules.
- `/Compute-acme/joe.jonathan@example.com/storage_orch`: An orchestration that defines storage volumes.

You can synchronize the management of all the resources defined in these orchestrations, through the following master orchestration:



```

{
  "name": "/Compute-acme/joe.jonathan@example.com/master_orch",
  "opplans": [
    {
      "label": "master-orchestration",
      "obj_type": "orchestration",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/instances_orch"
        },
        {
          "name": "/Compute-acme/joe.jonathan@example.com/networking_orch"
        },
        {
          "name": "/Compute-acme/joe.jonathan@example.com/storage_orch"
        }
      ]
    }
  ]
}

```

When you start the master orchestration, all of the nested orchestrations are started. Note that when you add a master orchestration to Compute Classic, the nested orchestrations are *not* added automatically. You must add each of the nested and master orchestrations separately.

Depending on the nature of the orchestrations, you might also need to define relationships between the different orchestration object plans in the master orchestration, to ensure that the objects defined in the various orchestrations are created in the appropriate sequence.

For example, to ensure that your network and storage resources are created before the orchestration that defines the instances is started, you can create a master orchestration with relationships defined as follows:

```

{
  "name": "/Compute-acme/joe.jonathan@example.com/master_orch",
  "opplans": [
    {
      "label": "instances-orchestration",
      "obj_type": "orchestration",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/instances_orch"
        }
      ]
    },
    {
      "label": "network-orchestration",
      "obj_type": "orchestration",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/networking_orch"
        }
      ]
    },
    {
      "label": "storage-orchestration",
      "obj_type": "orchestration",
      "objects": [

```

```

        {
          "name": "/Compute-acme/joe.jonathan@example.com/storage_orch"
        }
      ]
    },
    "relationships": [
      {
        "oplan": "instances-orchestration",
        "to_oplan": "network-orchestration",
        "type": "depends"
      },
      {
        "oplan": "instances-orchestration",
        "to_oplan": "storage-orchestration",
        "type": "depends"
      }
    ]
  }
}

```

You can terminate and restart each nested orchestration individually as required. When you terminate the master orchestration, all the nested orchestrations are stopped, and the objects created by those orchestrations are deleted.

If you delete the master orchestration, the nested orchestrations aren't automatically deleted. However, if you use the web console to delete a master orchestration, you can select the option to delete all nested orchestrations as well. See [Deleting an Orchestration v1](#).

An orchestration can contain up to three levels of nested objects.

### About High-Availability Policies in an Orchestration

You can specify a high availability (HA) policy in the `ha_policy` attribute of an orchestration, to specify the behavior when an object stops unexpectedly.

You can specify one of following HA policies:

- `active`

You can specify this policy only for instances, that is, only for objects of type `launchplan`.

When the HA policy for an instance is set to `active`, if the instance stops unexpectedly, it is re-created automatically. Note, however, that the instance is re-created automatically only if the orchestration was in the `Ready` state and the instance was running without an error. If an instance is in an error state, it isn't re-created automatically.

- `monitor`

You can specify this policy only for instances, storage volumes, and orchestrations, that is, for objects of type `launchplan`, `storage/volume`, and `orchestration`.

When the HA policy for an object is set to `monitor`, if the object goes to an error state or stops unexpectedly, the orchestration changes to the `Error` state. However, the object isn't re-created automatically.

You can't specify an HA policy for any objects other than instances, storage volumes, and orchestrations. Attempting to do so results in an error. Also, if you don't specify an HA policy for instances, storage volumes, or orchestrations explicitly, then no HA policy is applied. That is, the policy is set to `none` by default.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

## Orchestrations v1 Templates

The following sample JSON file illustrates the high-level structure of an orchestration. For templates for individual object types, see [Orchestration Templates for Each Object Type](#).

The orchestration templates provided here might not illustrate the use of all the attributes of each object. For a complete list of attributes and their description, see [Attributes in Orchestrations v1](#). To get started with building an orchestration, see [Building Your First Orchestration v1](#).

```
{
  "description": "someDescriptionHere",
  "name": "/Compute-identity_domain/user/name",
  "relationships": [see Relationships Between Object Plans],
  "oplans": [
    {
      "label": "someText",
      "obj_type": "objectType", (see Object Types in an Orchestration)
      "ha_policy": "policy", (see About High-Availability Policies in an Orchestration)
      "objects": [
        {
          attributes (see Attributes in Orchestrations v1)
        }
      ]
    },
    {
      "label": "someText",
      "obj_type": "objectType", (see Object Types in an Orchestration)
      "objects": [
        {
          attributes (see Attributes in Orchestrations v1)
        }
      ]
    },
    .
    . up to 10 oplans
    .
  ]
}
```

## Template for Top-Level Attributes of an Orchestration

**Top-level attributes** contain the name and description of an orchestration, along with other information such as the relationship between objects defined in the orchestration, start and stop times for the orchestration, and the list of objects in the orchestration.

```

{
  "name": "/Compute-acme/joe.jonathan@example.com/myOrchestration",
  "description": "sample orchestration",
  "relationships": [],
  "schedule": {"start_time": "2015-06-21T12:00:00Z"},
  "oplans": [
    {
      <Define your oplans here. See Orchestration Template for oplans.>
    }
  ]
}

```

## Orchestration Template for oplans

An **object plan**, or `oplan`, is a top-level orchestration attribute. Within an object plan, you can specify various object types and define one or more object for each object type.

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>

  "oplans": [

    {
      "label": "My orchestration",
      "obj_type": "orchestration",
      "objects": [
        <Define your objects here. See Orchestration Templates for Each Object Type.>
      ]
    }
  ]
}

```

## Orchestration Templates for Each Object Type

- [Orchestration Template for integrations/osscontainer](#)
- [Orchestration Template for ip/reservation](#)
- [Orchestration Template for launchplan](#)
  - [Orchestration Template for Instances](#)
- [Orchestration Template for network/v1/acl](#)
- [Orchestration Template for network/v1/ipaddressprefixset](#)
- [Orchestration Template for network/v1/ipassociation](#)
- [Orchestration Template for network/v1/ipnetwork](#)
- [Orchestration Template for network/v1/ipnetworkexchange](#)
- [Orchestration Template for network/v1/ipreservation](#)

- [Orchestration Template for network/v1/route](#)
- [Orchestration Template for network/v1/secprotocol](#)
- [Orchestration Template for network/v1/secrule](#)
- [Orchestration Template for network/v1/vnicset](#)
- [Orchestration Template for orchestration](#)
- [Orchestration Template for secapplication](#)
- [Orchestration Template for seclist](#)
- [Orchestration Template for seclist](#)
- [Orchestration Template for secrule](#)
- [Orchestration Template for storage/volume](#)

### Orchestration Template for integrations/osscontainer

Use this object type if you want to create a container in your associated Oracle Cloud Infrastructure Object Storage Classic account.

```
{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level
  Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "My-OSS-Container",
      "obj_type": "integrations/osscontainer",
      "objects": [
        {
          "account": "/Compute-acme/cloud_storage",
          "container": "Container_1",
          "delete_remote": false
        }
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
  <Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for ip/reservation

Use this object type if you want to reserve permanent IP addresses in the shared network to attach with your instances. For more information, see [About Public IP Addresses](#).

To associate an IP reservation with an instance that's defined in the same orchestration, you must specify a relationship between the `ip/reservation` and the `launchplan` object plans.

```
{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level
  Attributes of an Orchestration.>
  "oplans": [
    {
```

```

"label": "My IP reservations",
"obj_type": "ip/reservation",

"objects": [
  {
    "name": "/Compute-acme/joe.jonathan@example.com/ipres1",
    "parentpool": "/oracle/public/ippool",
    "permanent": true
  },
  {
    "name": "/Compute-acme/joe.jonathan@example.com/ipres2",
    "parentpool": "/oracle/public/ippool",
    "permanent": true
  }
  <Define other IP reservations here.>
]
}
<Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for launchplan

Use this object type if you want to define one or more instances. In an orchestration, instance is an attribute of the launchplan object type.

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "My instances",
      "obj_type": "launchplan",

      "objects": [
        {
          "instances": [
            {
              <Define your instance here. See Orchestration Template for Instances.>
            }
            <Define other instances here.>
          ]
        }
      ]
    }
  ]
}
<Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for Instances

In an orchestration, instance is an attribute of the launchplan object type. If any of the objects referred to in instance attributes are defined in the same orchestration as the instance, you must specify a relationship between each such object and the instance launch plan. For more information, see [Relationships Between Object Plans](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level

```

[Attributes of an Orchestration.](#)>

```

"opplans": [
  {
    "label": "My instances",
    "obj_type": "launchplan",
    "objects": [
      {
        "instances": [
          {
            "shape": "oc3",
            "boot_order": [1],
            "label": "vm-1",
            "networking": {
              "eth0": {
                "seclists": ["/Compute-acme/joe.jonathan@example.com/
wlsadmin_seclist"],
                "nat": "ipreservation:/Compute-acme/joe.jonathan@example.com/
ipres1"
              },
              "eth1": {
                "ipnetwork": "/Compute-acme/joe.jonathan@example.com/ipnet-1",
                "ip": "192.168.4.2",
                "vnic": "/Compute-acme/joe.jonathan@example.com/eth1-ipnet1"
              }
            },
            "sshkeys": ["/Compute-acme/joe.jonathan@example.com/key1"],
            "storage_attachments": [
              {
                "index": 1,
                "volume": "/Compute-acme/joe.jonathan@example.com/boot"
              }
            ]
          }
        ]
      }
    ]
  }
]
<Define other objects here. See Orchestration Templates for Each Object Type.>
<Define other opplans here. See Orchestration Template for opplans.>

```

**Orchestration Template for network/v1/acl**

Use this object type if you want to create an access control list (ACL). For more information about using ACLs in IP networks, see [Configuring IP Networks](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "opplans": [
    {
      "label": "ACL1-for-vnicset1",
      "obj_type": "network/v1/acl",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/acl_1",

```

```

        "enabledFlag": true
      }
      <Define other ACLs here.>
    ]
  }
  <Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for network/v1/ipaddressprefixset

Use this object type if you want to create IP address prefix set to use as a source or destination in a security rule. For more information about using IP address prefix sets in IP networks, see [Configuring IP Networks](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "IPAddress-prefix-set-1",
      "obj_type": "network/v1/ipaddressprefixset",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/ext_ip_addresses",
          "ipAddressPrefixes": ["203.0.113.0/30", "192.51.100.1/24"]
        }
        <Define other IP address prefix sets here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for network/v1/ipassociation

Use this object type if you want to create an IP association between an IP reservation and a vNIC in an IP network. For more information about IP associations in IP networks, see [Configuring IP Networks](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "IP-Association-for-vnic1-on-instancetype",
      "obj_type": "network/v1/ipassociation",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/IP-association-
vnic1",
          "ipAddressReservation": "/Compute-acme/joe.jonathan@example.com/
IPres-for-instancetype-vnic1",
          "vnic": "/Compute-acme/joe.jonathan@example.com/instancetype-
vnic1"
        }
        <Define other IP associations here.>
      ]
    }
  ]
}

```



```

    ]
  }
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for network/v1/ipnetwork

Use this object type if you want to create IP networks. For more information about setting up and using IP networks, see [Configuring IP Networks](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "ipnet1",
      "obj_type": "network/v1/ipnetwork",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/ipnet1",
          "ipAddressPrefix": "192.168.3.0/24",
          "ipNetworkExchange": "/Compute-acme/joe.jonathan@example.com/ipnetworkexchange1",
          "description": "IP network with IP network exchange"
        }
        <Define other IP networks here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for network/v1/ipnetworkexchange

Use this object type if you want to create IP network exchanges. For more information about setting up and using IP networks and IP network exchanges, see [Configuring IP Networks](#).

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "ipnetworkexchange",
      "obj_type": "network/v1/ipnetworkexchange",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/ipnetworkexchange1"
        }
        <Define other IP network exchanges here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}

```

```
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for network/v1/ipreservation

Use this object type if you want to create an IP reservation for IP networks. For more information about IP networks, see [Configuring IP Networks](#).

```
{
<Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "IP-Reservation-for-IP-network",
      "obj_type": "network/v1/ipreservation",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/IPres-for-
instancel-vnic1",
          "ipAddressPool": "/oracle/public/public-ippool"
        }
        <Define other IP reservations for IP networks here.>
      ]
    }
  ]
<Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for network/v1/route

Use this object type if you want to create routes to direct traffic across your IP networks. See [Configuring IP Networks](#).

```
{
<Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "route",
      "obj_type": "network/v1/route",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/routel",
          "nextHopVnicSet": "/Compute-acme/joe.jonathan@example.com/
vnicset1",
          "ipAddressPrefix": "203.0.113.0/24",
          "adminDistance": "0"
        }
        <Define other routes here.>
      ]
    }
  ]
<Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}
```

**Orchestration Template for network/v1/secprotocol**

Use this object type if you want to create a security protocol for IP networks. For more information about IP networks, see [Configuring IP Networks](#).

```
{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level
  Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "Security-protocol-for-IP-networks",
      "obj_type": "network/v1/secprotocol",
      "objects": [
        {
          "description": "Security Protocol 1",
          "dstPortSet": ["20", "155-1100"],
          "ipProtocol": "tcp",
          "name": "/Compute-acme/joe.jonathan@example.com/secprotocol_1",
          "srcPortSet": ["10", "55-100"]
        }
        <Define other security protocols for IP networks here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}
```

**Orchestration Template for network/v1/secrule**

Use this object type if you want to create a security rule for IP networks. For more information about IP networks, see [Configuring IP Networks](#).

```
{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level
  Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "IP-network-secrule-1",
      "obj_type": "network/v1/secrule",
      "objects": [
        {
          "acl": "/Compute-acme/joe.jonathan@example.com/acl_1",
          "description": "Sec Rule 1",
          "flowDirection": "egress",
          "name": "/Compute-acme/joe.jonathan@example.com/
ipnetSecrule1",
          "secProtocols": ["/Compute-acme/joe.jonathan@example.com/
secprotocol_1"],
          "srcIpAddressPrefixSets": ["/Compute-acme/
joe.jonathan@example.com/ext_ip_address_list_1"]
        }
        <Define other security rules for IP networks here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
```

```
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for network/v1/vnicset

Use this object type if you want to create vNICsets in IP networks. For information about using vNICsets in IP networks, see [Configuring IP Networks](#).

```
{
<Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "vnicset",
      "obj_type": "network/v1/vnicset",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/vnicset1",
          "vnics": [ "/Compute-acme/joe.jonathan@example.com/vnic1",
                    "/Compute-acme/joe.jonathan@example.com/vnic2" ]
        }
        <Define other vnicsets here.>
      ]
    }
  ]
}
<Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for orchestration

Use this object type if you want to use an orchestration to start or stop multiple nested orchestrations. For more information, see [About Nested Orchestrations](#).

```
{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "My orchestration",
      "obj_type": "orchestration",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/myInstances"
        },
        {
          "name": "/Compute-acme/joe.jonathan@example.com/myStorageVolumes"
        }
        <Add names of other nested orchestrations here.>
      ]
    }
  ]
}
<Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for `secapplication`

Use this object type to define security applications to use in security rules in the shared network. For more information, see [About Security Applications](#).

To associate an IP reservation with an instance that's defined in the same orchestration, you must specify a relationship between the `ip/reservation` and the `launchplan` object plans.

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "My security applications",
      "obj_type": "secapplication",

      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/wlsadmin_ssl",
          "dport": 7002,
          "protocol": "tcp"
        }
        <Define other security applications here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for `seciplist`

Use this object type to define a set of IP addresses that you want to use as a source in a security rule in the shared network. For more information, see [About Security IP Lists](#).

To use this security IP list in a security rule that's defined in the same orchestration, you must specify a relationship between these objects.

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "admin-ip-list",
      "obj_type": "seciplist",

      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/admin_ips",
          "secipentries": ["203.0.113.0/30"]
        }
        <Define other security IP lists here.>
      ]
    }
  ]
  <Define other objects here. See Orchestration Templates for Each Object Type.>
}

```

```
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for `seclist`

Use this object type to define security lists to group your instances in the shared network. For more information, see [About Security Lists](#).

To use this security list in a security rule that's defined in the same orchestration, you must specify a relationship between these objects.

```
{
<Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [

    {
      "label": "admin-seclists",
      "obj_type": "seclist",

      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/sysadmin_seclist"
        },
        {
          "name": "/Compute-acme/joe.jonathan@example.com/wlsadmin_seclist"
        }
      ]
      <Define other security lists here.>
    }
  ]
}
<Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}
```

### Orchestration Template for `secrule`

Use this object type to define security rules that control access to your instances in the shared network. For more information, see [About Security Rules](#).

If this security rule uses security applications, security lists, or security IP lists that are defined in the same orchestration, then you must specify a relationship between these objects.

```
{
<Define the top-level attributes of your orchestration here. See Template for Top-Level Attributes of an Orchestration.>
  "oplans": [

    {
      "label": "My security rules",
      "obj_type": "secrule",

      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/
admin_ssh_to_sysadmin_rule",
          "application": "/oracle/public/ssh",
          "src_list": "seclist:/Compute-acme/joe.jonathan@example.com/admin_ips",
          "dst_list": "seclist:/Compute-acme/joe.jonathan@example.com/
sysadmin_seclist",
```

```

        "action": "PERMIT"
      },
      {
        "name": "/Compute-acme/joe.jonathan@example.com/dbadmin_ssh_to_db_rule",
        "application": "/oracle/public/ssh",
        "src_list": "seclist:/Compute-acme/joe.jonathan@example.com/
dbadmin_seclist",
        "dst_list": "seclist:/Compute-acme/joe.jonathan@example.com/db_seclist",
        "action": "PERMIT"
      }
    ]
  }
  <Define other security rules here.>
}
]
<Define other objects here. See Orchestration Templates for Each Object Type.>
]
<Define other oplans here. See Orchestration Template for oplans.>
}

```

### Orchestration Template for storage/volume

Use this object type to create storage volumes that you want to attach to your instances. For more information, see [About Storage Volumes](#).

#### Note:

Don't define storage volumes and instances in the same orchestration. By keeping storage volumes and instances in separate orchestrations, you can shut down and start the instances when required and yet preserve the attached storage volumes. Note that the recommendation here is to define the storage *volumes* outside the instance orchestration. To ensure that the storage volumes remain attached after an instance is re-created, you must define the storage *attachments* within the instance orchestration.

```

{
  <Define the top-level attributes of your orchestration here. See Template for Top-Level
Attributes of an Orchestration.>
  "oplans": [
    {
      "label": "My storage volumes",
      "obj_type": "storage/volume",
      "objects": [
        {
          "name": "/Compute-acme/joe.jonathan@example.com/boot",
          "bootable": true,
          "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64",
          "properties": ["/oracle/public/storage/default"],
          "size": "22548578304"
        },
        {
          "name": "/Compute-acme/joe.jonathan@example.com/data",
          "properties": ["/oracle/public/storage/latency"],
          "size": "32212254720"
        }
      ]
    }
  ]
  <Define other storage volumes here.>
}

```

```

    }
    <Define other objects here. See Orchestration Templates for Each Object Type.>
  }
  <Define other oplans here. See Orchestration Template for oplans.>
}

```

## Workflow for Creating Instances Using Orchestration v1

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

To use an orchestration to create and manage compute, networking, or storage resources:

1. Build your orchestration.

An orchestration is defined in a JavaScript Object Notation (JSON) file that contains the attributes of the Compute Classic objects that you want to create. See [Building Your First Orchestration v1](#).

2. Upload the orchestration to Compute Classic. See [Uploading an Orchestration v1](#).
3. To create the objects defined in the orchestration, start the orchestration. See [Starting an Orchestration v1](#).
4. To delete the objects defined in the orchestration, stop the orchestration. See [Terminating an Orchestration v1](#).

## Building Your First Orchestration v1

### Topics

- [Before You Begin](#)
- [Sample Orchestration for Creating a Single Instance](#)
- [Steps for Building Your First Orchestration](#)

### Before You Begin

Before building your orchestration JSON file, do the following:

- Read [Best Practices for Using Compute Classic](#).
- Create the security, storage, and networking resources that you plan to reference in your orchestration.

These tasks require the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

- If you want to create a Linux instance with SSH access enabled, upload your SSH public keys to Compute Classic. See [Adding an SSH Public Key](#).



 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH.

- If you want your instances to boot from a persistent storage disk, create bootable storage volumes. See [Creating a Bootable Storage Volume](#).
- Create storage volumes for the data and applications that you plan to deploy on your instances. See [Creating a Storage Volume](#). When you create the storage volumes, don't attach them to any existing instance. You'll specify the storage volumes later in the orchestration.
- If you want your instances to have fixed public IP addresses, then create the required IP reservation. See [Reserving a Public IP Address](#).
- Create the required security lists. See [Creating a Security List](#).

### Sample Orchestration for Creating a Single Instance

You can use the following sample orchestration as a starting point for building your first orchestration.

```
{
  "description": "Simple oplan with an ssh key and a security list",
  "name": "/Compute-acme/joe.jonathan@example.com/simple_orchestration",
  "opplans": [
    {
      "label": "simple_oplan",
      "obj_type": "launchplan",
      "objects": [
        {
          "instances": [
            {
              "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64",
              "label": "OL_6.7",
              "networking": {
                "eth0": {
                  "seclists": [
                    "/Compute-acme/joe.jonathan@example.com/my_instances"
                  ],
                  "nat": "ipreservation:/Compute-acme/joe.jonathan@example.com/ip1"
                }
              },
              "shape": "oc3",
              "storage_attachments": [
                {
                  "index": 1,
                  "volume": "/Compute-acme/joe.jonathan@example.com/OL66_boot"
                },
                {
                  "index": 2,
                  "volume": "/Compute-acme/joe.jonathan@example.com/data1"
                }
              ],
              "boot_order": [1],
              "sshkeys": [
                "/Compute-acme/joe.jonathan@example.com/ssh-key1"
              ]
            }
          ]
        }
      ]
    }
  ]
}
```

```
    ]  
  }  
] }  
]  
]  
]  
]  
]  
]  
]  
]
```

This sample orchestration does the following:

- Defines an instance with the label `OL_6.7`, the `oc3` shape, and using the `/oracle/public/OL_6.7_UEKR4_x86_64` image.
- Adds the instance to the security list `/Compute-acme/joe.jonathan@example.com/my_instances`.
- Associates the IP reservation `/Compute-acme/joe.jonathan@example.com/ip1` with the instance.
- Attaches the bootable storage volume `/Compute-acme/joe.jonathan@example.com/OL66_boot` to the instance.
- Attaches the data storage volume `/Compute-acme/joe.jonathan@example.com/data1` to the instance.
- Associates the SSH public key `/Compute-acme/joe.jonathan@example.com/ssh-key1` with the instance.

 **Note:**

To learn about the structure of an orchestration, see [Orchestrations v1 Templates](#). For information about all the attributes that you can define in an orchestration, see [Attributes in Orchestrations v1](#).

### Steps for Building Your First Orchestration

1. Copy the sample orchestration to a plain text file, and open the file in any text editor.
2. Replace the name of the orchestration with an appropriate three-part name (`/Compute-identity_domain/user/object`).

 **Note:**

While editing this sample file, remember to replace all placeholder values with values specific to your environment. For example, replace the identity domain name `acme` with your identity domain name and the user name `joe.jonathan@example.com` with your user name.

3. Change the value of the `imagelist` attribute to any image that you want to use.
4. Under `instances`, change the value of the `label` attribute to any label that you want.

5. Replace the security list `/Compute-acme/joe.jonathan@example.com/my_instances` with a security list that you've already created.

If you want to attach the instance to more security lists, remember to enclose each security-list name in double quotation marks and separate the security-list names by using commas. See the following example:

```
"seclists": [
  "/Compute-acme/joe.jonathan@example.com/my_instances",
  "/Compute-acme/joe.jonathan@example.com/dev_instances",
  "/Compute-acme/joe.jonathan@example.com/prod_instances"
]
```

6. Replace the IP reservation `/Compute-acme/joe.jonathan@example.com/ip1` with an IP reservation that you've already created.
7. Replace the `oc3` shape with the shape that you want to use.
8. Replace the storage volume `/Compute-acme/joe.jonathan@example.com/OL66_boot` with the bootable storage volume that your instance should boot from.
9. Replace the storage volume `/Compute-acme/joe.jonathan@example.com/data1` with a storage volume that you want to attach to the instance.

If you don't want to attach any storage volume, then remove the following section (and the comma preceding it) from the orchestration.

```
{
  "index": 2,
  "volume": "/Compute-acme/joe.jonathan@example.com/data1"
}
```

If you want to attach more storage volumes, then specify the index for the storage attachment and the name of the storage volume as follows. Separate the storage volume definitions using commas. See the following example:

```
{
  "index": 2,
  "volume": "/Compute-acme/joe.jonathan@example.com/admin/data1"
},
{
  "index": 3,
  "volume": "/Compute-acme/joe.jonathan@example.com/data2"
}
```

10. If you're creating a Linux instance enabled for SSH access, replace the SSH key `/Compute-acme/joe.jonathan@example.com/ssh-key1` with a key that you've created and added to Compute Classic.

If you want to add more SSH keys, then enclose each key in double quotation marks and separate the keys by using commas. See the following example:

```
"sshkeys": [
  "/Compute-acme/joe.jonathan@example.com/ssh-key1",
  "/Compute-acme/joe.jonathan@example.com/ssh-key2",
  "/Compute-acme/joe.jonathan@example.com/ssh-key3"
]
```

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

**11. Save the orchestration file.**

You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. If your JSON format isn't valid, then an error message is displayed when you upload the orchestration.

 **Note:**

Oracle doesn't support or endorse any third-party JSON-validation tool.

Your orchestration file is now ready.

To create instances by using this orchestration, you must upload it to Compute Classic. See [Uploading an Orchestration v1](#).

## Attributes in Orchestrations v1

You specify attributes in orchestrations at several levels. At the highest level, you specify certain attributes for the orchestration as a whole. Then, you specify attributes for each object plan defined in the orchestration. Finally, there are attributes that are specific to each object type.

- **Top-level attributes**

**Top-level attributes** contain the name and description of an orchestration, along with other information such as the relationship between objects defined in the orchestration, start and stop times for the orchestration, and the list of objects in the orchestration. See [Top-Level Attributes in Orchestrations v1](#). For a template of top-level orchestration attributes, see [Template for Top-Level Attributes of an Orchestration](#).

- **Object plan attributes**

An **object plan**, or `oplan`, is a top-level orchestration attribute. Within an object plan, you can specify various object types and define one or more object for each object type. Object plan attributes define the characteristics of each `oplan`, including its label, object type and list of objects, and the HA policy, if applicable. See [Object Plan Attributes](#). For an `oplan` template, see [Orchestration Template for `oplans`](#).

- **Attributes specific to each object type**

These are the characteristics specific to each object type. See [Orchestration v1 Attributes Specific to Each Object Type](#). The attributes that you can specify for each object type in an orchestration are the same as the parameters that you can specify with the `POST` method for that resource using the API.

For orchestration templates that you can use to create individual objects, see [Orchestration Templates for Each Object Type](#).

## Top-Level Attributes in Orchestrations v1

**Top-level attributes** contain the name and description of an orchestration, along with other information such as the relationship between objects defined in the orchestration, start and stop times for the orchestration, and the list of objects in the orchestration.

Attributes for objects defined in an orchestration vary according to the object type. For a list of objects and object-specific attributes in orchestrations v1, see [Orchestration v1 Attributes Specific to Each Object Type](#).

The following sample JSON shows the required top-level orchestration attributes, `name` and `oplans`. A description of each of the required and optional top-level attributes is provided in the table below.

```
{
  "name": "/Compute-acme/joe/myOrchestration",
  "oplans": [
    {
      ...
    }
  ]
}
```

Parameter	Required or Optional	Description
<code>name</code>	required	The three-part name of the orchestration ( <code>/Compute-identity_domain/user/object_name</code> ).
<code>oplans</code>	required	The list of object plans ( <code>oplans</code> ) in the orchestration. An <code>oplan</code> is the primary building block of an orchestration. Each <code>oplan</code> contains all the attributes for the object type defined in it. An orchestration can contain up to 10 object plans.
<code>description</code>	optional	Text string describing the orchestration.
<code>relationships</code>	optional	The relationship between the objects that are created by this orchestration. The only supported relationship type for orchestrations is <code>depends</code> . The <code>depends</code> relationship type specifies that one object must be instantiated first. For example, you could define a storage volume in one <code>oplan</code> and attach that storage volume to an instance in another <code>oplan</code> . The second <code>oplan</code> would then depend on the first.

Parameter	Required or Optional	Description
schedule	optional	<p>The start and stop dates and times, in ISO 8601 format. You must specify the time zone as UTC.</p> <ul style="list-style-type: none"> <li> <b>start_time</b>            (Optional) Date and time when you want the orchestration to start. For example, to start an orchestration at noon on 6/21/2015, UTC, enter the start time as 2015-06-21T12:00:00Z. Here Z denotes UTC.             If you enter a start time that is earlier than the time you upload the orchestration, then the orchestration starts immediately.         </li> <li> <b>stop_time</b>            (Optional) Date and time when you want the orchestration to stop. For example, to stop an orchestration at 11:59 p.m. on 12/31/2015, enter the stop time as 2015-12-31T23:59:59Z. Here Z denotes UTC.             The stop time must be at least 120 seconds after the start time.         </li> </ul>

## Object Plan Attributes

An **object plan**, or `oplan`, is a top-level orchestration attribute. Within an object plan, you can specify various object types and define one or more object for each object type. You must provide a label for each `oplan`. You can also specify a High Availability policy, if applicable.

The following sample JSON shows the required attributes of an object plan. A description of each of the required and optional attributes is provided in the table below.

```
{
  "name": "/Compute-amce/joe/myOrchestration",
  "oplans": [
    {
      "label": "My orchestration",
      "obj_type": "orchestration",
      "objects": [
        ...
      ]
    }
  ]
}
```

Parameter	Required or Optional	Description
label	required	Text string describing the object plan. Maximum length: 256 characters.

Parameter	Required or Optional	Description
obj_type	required	<p>The type of object that you want to create. Specify one of the following object types.</p> <ul style="list-style-type: none"> <li>• integrations/osscontainer</li> <li>• ip/reservation</li> <li>• launchplan</li> <li>• network/v1/acl</li> <li>• network/v1/ipaddressprefixset</li> <li>• network/v1/ipassociation</li> <li>• network/v1/ipnetwork</li> <li>• network/v1/ipnetworkexchange</li> <li>• network/v1/iptables</li> <li>• network/v1/iptablesrule</li> <li>• network/v1/iptablesruleset</li> <li>• network/v1/ipreservation</li> <li>• network/v1/route</li> <li>• network/v1/secprotocol</li> <li>• network/v1/secrule</li> <li>• network/v1/vnicset</li> <li>• orchestration</li> <li>• secapplication</li> <li>• seclist</li> <li>• seclist</li> <li>• secrule</li> <li>• storage/volume</li> </ul> <p>For a brief description of each object type, see <a href="#">Object Types in an Orchestration</a>.</p> <p>Each object type has a specific set of attributes. See <a href="#">Orchestration v1 Attributes Specific to Each Object Type</a>.</p>
objects	required	<p>The list of objects, depending on the type of object that you're creating, as defined in the obj_type attribute. See <a href="#">Orchestration v1 Attributes Specific to Each Object Type</a></p>
ha_policy	optional	<p>The high availability policy: active or monitor. You can specify either active or monitor for instances, and monitor for storage volumes or orchestrations. You can't specify a high availability policy for other objects. Attempting to do so results in an error. If you don't specify a high availability policy for instances, storage volumes, or orchestrations, no high availability policy is applied. That is, by default, ha_policy is set to none. See <a href="#">High Availability Policies</a>.</p>

## Orchestration v1 Attributes Specific to Each Object Type

You can specify various object types in an orchestration, including launch plans, networking objects such as security lists and security rules, storage volumes, and

even other orchestrations. The attributes for each object vary depending on the object type.

The following sections describe the attributes for each object type that you can create using an orchestration. Each set of attributes corresponds to an object under the specified `obj_type` in the orchestration file.

- [Orchestration v1 Attributes for `integrations/osscontainer`](#)
- [Orchestration v1 Attributes for `ip/reservation`](#)
- [Orchestration v1 Attributes for `launchplan`](#)
- [Orchestration v1 Attributes for `instances`](#)
- [Orchestration v1 Attributes for `network/v1/acl`](#)
- [Orchestration v1 Attributes for `network/v1/ipaddressprefixset`](#)
- [Orchestration v1 Attributes for `network/v1/ipassociation`](#)
- [Orchestration v1 Attributes for `network/v1/ipnetwork`](#)
- [Orchestration v1 Attributes for `network/v1/ipnetworkexchange`](#)
- [Orchestration v1 Attributes for `network/v1/ipreservation`](#)
- [Orchestration v1 Attributes for `network/v1/route`](#)
- [Orchestration v1 Attributes for `network/v1/secprotocol`](#)
- [Orchestration v1 Attributes for `network/v1/secrule`](#)
- [Orchestration v1 Attributes for `network/v1/vnicset`](#)
- [Orchestration v1 Attributes for `orchestration`](#)
- [Orchestration v1 Attributes for `secapplication`](#)
- [Orchestration v1 Attributes for `seciplist`](#)
- [Orchestration v1 Attributes for `seclist`](#)
- [Orchestration v1 Attributes for `secrule`](#)
- [Orchestration v1 Attributes for `storage/volume`](#)

All objects in an orchestration are contained within an object plan. For information about object plan attributes, see [Object Plan Attributes](#). For information about defining the name and other characteristics of an orchestration, see [Top-Level Attributes in Orchestrations v1](#).

## Orchestration v1 Attributes for `integrations/osscontainer`

The following sample JSON shows the key attributes of the `integrations/osscontainer` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
    "account": "/Compute-acme/cloud_storage",
    "container": "Container_1",
    "delete_remote": false
}
```



Parameter	Required or Optional	Description
account	required	The two-part name of the account ( <code>/Compute-identity_domain/cloud_storage</code> ) that contains the credentials and access details of the associated Oracle Cloud Infrastructure Object Storage Classic instance.
container	required	The name of the container that you want to create. Container names must: <ul style="list-style-type: none"> <li>Contain only UTF-8 characters</li> <li>Be a maximum of 256 bytes</li> <li>Avoid using a slash (<code>/</code>) character because this character acts as a delimiter between the container name and the object name</li> </ul> Ensure that a container of the same name doesn't already exist.
delete_remove	required	When set to <code>true</code> , deletes the Oracle Cloud Infrastructure Object Storage Classic container along with all the objects in the container when you delete the <code>integration/osscontainer</code> object created by this orchestration.  When set to <code>false</code> , only the <code>integrations/osscontainer</code> object created by this orchestration is deleted. The container in Oracle Cloud Infrastructure Object Storage Classic remains intact, along with all objects in the container.
name	optional	The three-part name of the <code>integrations/osscontainer</code> object created by this orchestration. This name is in the format <code>/Compute-identity_domain/user/object</code> .  If you don't specify a name for this object, then the name is generated automatically.  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.

## Orchestration v1 Attributes for `ip/reservation`

The following sample JSON shows the key attributes of the `ip/reservation` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipres1",
  "parentpool": "/oracle/public/ippool",
  "permanent": true
}
```

Parameter	Required or Optional	Description
parentpool	required	Specify <code>/oracle/public/ippool</code>

Parameter	Required or Optional	Description
permanent	required	Set to True
account	optional	Specify <code>/Compute-identity_domain/default</code>
name	optional	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>

## Orchestration v1 Attributes for `launchplan`

Launch plan objects are used to define instances. The following sample JSON shows the required attributes of the `launchplan` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "instances":
  [
    {
      ...
    }
  ]
}
```

Parameter	Required or Optional	Description
instances	required	<p>A list of instances.</p> <p>For instance attributes, see <a href="#">Orchestration v1 Attributes for instances</a>.</p>
relationships	optional	<p>The relationships between instances.</p> <p>Valid values:</p> <ul style="list-style-type: none"> <li><code>same_node</code>: The specified instances are created on the same physical server. This is useful if you want to ensure low latency across instances.</li> <li><code>different_node</code>: The specified instances aren't created on the same physical server. This is useful if you want to isolate instances for security or redundancy.</li> </ul>

After defining a `launchplan` object, to define an instance, see [Orchestration v1 Attributes for instances](#).

## Orchestration v1 Attributes for `instances`



### Note:

An instance is not in itself an object type. It is an attribute of the `launchplan` object type.

### Topics:

- [Instance Attributes](#)
- [Networking Attributes for Instances](#)
  - [Subparameters for a Network Interface on the Shared Network](#)
  - [Subparameters for a Network Interface on an IP Network](#)

### Instance Attributes

Instances are an attribute of the `launchplan` object type. Instances have a number of required and optional attributes. The following sample JSON shows some of the key instance attributes. A description of each of the required and optional instance attributes is provided in the table below.

```
{
  "instances": [
    {
      "shape": "oc3",
      "boot_order": [
        1
      ],
      "label": "vm-1",
      "networking": {
        "eth0": {
          "seclists": [
            "/Compute-acme/joe/wlsadmin_seclist"
          ],
          "nat": "ipreservation:/Compute-acme/joe/ipres1"
        },
        "eth1": {
          "ipnetwork": "/Compute-acme/joe/ipnet1",
          "ip": "192.168.4.2",
          "vnic": "/Compute-acme/joe/eth1-ipnet1"
        }
      },
      "sshkeys": [
        "/Compute-acme/joe/key1"
      ],
      "storage_attachments": [
        {
          "index": 1,
          "volume": "/Compute-acme/joe/boot"
        }
      ]
    }
  ]
}
```

```
]
}
```

Parameter	Required or Optional	Description
shape	required	The name of the shape that defines the number of OCPUs and the RAM that you require for the instance. For general purpose and high-memory shapes, you can select the block storage disk size, but for high I/O shapes, the size of the SSD storage is determined by the shape.
name	optional	<p>The three-part name of the instance (<code>/Compute-identity_domain/user/name</code>).</p> <p>If you specify this parameter, then the full name of the instance would be in the format, <code>/Compute-identity_domain/user/name_you_specify/id</code>.</p> <p>If you don't specify this parameter, then the full name would be in the format, <code>/Compute-identity_domain/user/id</code>.</p> <p>In either case, <i>id</i> is an autogenerated ID.</p> <p><b>Examples of Instance Names:</b></p> <ul style="list-style-type: none"> <li>When you specify <code>/Compute-acme/jack/vm1</code> as the value of the name parameter: <code>/Compute-acme/jack/vm1/300a7479-ec90-4826-98b9-a725662628f1</code></li> <li>When you don't specify the name parameter: <code>/Compute-acme/jack/38ef677e-9e13-41a7-a40c-2d99afce1714</code></li> </ul> <p>Although this is an optional parameter, specifying a meaningful name makes it easier for you to identify your instances.</p>
label	optional	<p>A text string to identify the instance.</p> <p>This label is used when defining relationships in an orchestration.</p> <p>A label can contain only alphanumeric characters, hyphens, and underscores. It can't contain unicode characters and spaces.</p> <p>Maximum length is 256 characters.</p>
tags	optional	<p>A JSON array or list of strings used to tag the instance.</p> <p>By assigning a human-friendly tag to an instance, you can identify the instance easily when you perform an instance listing. These tags aren't available from within the instance.</p>

Parameter	Required or Optional	Description
attributes	optional	<p>A JSON object or dictionary of user-defined attributes to be made available to the instance.</p> <p>If you're creating a Windows instance, you must specify the following required attributes:</p> <pre> {     "enable_rdp": true,     "administrator_password": "Specify_password_here" } </pre> <p>For more information about specifying user-defined attributes that can be used to automate instance configuration, see <a href="#">Automating Instance Initialization Using opc-init</a>.</p> <p><b>Note:</b> Solaris machine images don't include the opc-init scripts. So you can't use opc-init to automate instance configuration of Solaris instances.</p> <p>The attributes that you specify can be accessed from within the instance at <a href="http://192.0.0.192/latest/attributes">http://192.0.0.192/latest/attributes</a>. For more information about retrieving user-defined attributes, see <a href="#">Retrieving User-Defined Instance Attributes</a>.</p>
imagelist	optional	<p>The three-part name (<code>oracle/public/imagelist_name</code>) of the image list containing the image to be used (example: <code>/oracle/public/OL_6.7_UKR4_x86_64</code>).</p> <p>You <i>must</i> use this attribute if you don't specify a bootable storage volume by using the <code>boot_order</code> attribute. If you specify the <code>imagelist</code> attribute as well as the <code>boot_order</code> attribute, then the <code>imagelist</code> attribute is ignored.</p>
storage_attachments	optional	<p>If you specify the <code>storage_attachments</code> parameter, then specify the following subparameters for each attachment:</p> <ul style="list-style-type: none"> <li><b>volume:</b> The three-part name (<code>/Compute-identity_domain/user/object_name</code>) of the storage volume that you want to attach to the instance. Note that volumes attached to an instance at launch time can't be detached.</li> <li><b>index:</b> The index number for the volume. The allowed range is 1 to 10. If you want to use a storage volume as the boot disk for an instance, you must specify the index number for that volume as 1. The index determines the device name by which the volume is exposed to the instance. Index 0 is allocated to a nonpersistent boot disk, <code>/dev/xvda</code>. An attachment with index 1 is exposed to the instance as <code>/dev/xvdb</code>, an attachment with index 2 is exposed as <code>/dev/xvdc</code>, and so on.</li> </ul>

Parameter	Required or Optional	Description
<code>boot_order</code>	optional	<p>The index number of the bootable storage volume that should be used to boot the instance. The only valid value is 1.</p> <p>If you set this attribute, you must also specify a bootable storage volume with index number 1 in the <code>volume</code> sub-parameter of <code>storage_attachments</code>.</p> <p>When you specify <code>boot_order</code>, you don't need to specify the <code>imagelist</code> attribute, because the instance is booted using the image on the specified bootable storage volume. If you specify both <code>boot_order</code> and <code>imagelist</code>, the <code>imagelist</code> attribute is ignored.</p>
<code>hostname</code>	optional	<p>The host name assigned to the instance. On an Oracle Linux instance, this host name is displayed in response to the <code>hostname</code> command.</p> <p>Only relative DNS is supported. The domain name is suffixed to the host name that you specify. The host name must not end with a period. If you don't specify a host name, then a name is generated automatically. The DNS name of an instance depends on its host name, as follows:</p> <ul style="list-style-type: none"> <li>• If no DNS name is specified in the <code>networking</code> attribute, then the DNS name is set to the host name, and a reverse DNS record (PTR) is created for the host name.</li> <li>• If the DNS name specified in the <code>networking</code> attribute matches the host name, then that record also creates a reverse DNS record for the host name.</li> <li>• If the <code>dns</code> attribute under <code>networking</code> is set to an empty list (<code>[]</code>), then no DNS records are created even if a host name is specified. The instance still receives its host name through DHCP, and can perform a reverse lookup of its host name. However, no other instance can perform this reverse lookup.</li> </ul>
<code>reverse_dns</code>	optional	<p>If set to <code>true</code> (default), then reverse DNS records are created.</p> <p>If set to <code>false</code>, no reverse DNS records are created.</p>

 **Note:**

If an instance has network interfaces defined only for IP networks and doesn't have any interface on the shared network, then when `hostname` is specified, no DNS entries are set. In this case, DNS entries are set by the `dns` subparameter of the `networking` attribute.

---

Parameter	Required or Optional	Description
networking (attributes for the shared network)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>Only one interface on an instance can be added to the shared network. To add an interface to the shared network, you can specify the following subparameters:</p> <ul style="list-style-type: none"><li>• <code>seclists</code>: (Optional) The security lists that you want to add the instance to.</li><li>• <code>nat</code>: (Optional) Indicates whether a temporary or permanent public IP address should be assigned to the instance.</li><li>• <code>dns</code>: (Optional) A list of the DNS A record names for the instance. This name is relative to the internal DNS domain.</li><li>• <code>model</code>: (Optional) The type of network interface card (NIC). The only allowed value is <code>e1000</code>.</li><li>• <code>name_servers</code>: (Optional) The name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li><li>• <code>search_domains</code>: (Optional) The search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li></ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on the Shared Network</a>.</p>

---

Parameter	Required or Optional	Description
networking (attributes for IP networks)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>To add this instance to an IP network, specify the following subparameters:</p> <ul style="list-style-type: none"> <li><code>ipnetwork</code>: The name of the IP network that you want to add the instance to.</li> <li><code>ip</code>: (Optional) If you want to associate a static private IP address with the instance, specify an available IP address from the IP address range of the specified <code>ipnetwork</code>.</li> <li><code>address</code>: (Optional) The MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter <code>01:02:03:04:ab:cd</code> as the MAC address but not <code>01-02-03-04-ab-cd</code>.</li> <li><code>nat</code>: (Optional) A list of IP reservation that you want to associate with this interface, in the format: <code>"nat": ["network/v1/ipreservation:IP_reservation_name"]</code>. Here <code>IP_reservation_name</code> is the three-part name of the IP reservation in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnic</code>: (Optional) The three-part name of the vNIC in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnicsets</code>: (Optional) A list of the three-part names of the vNICsets that you want to add this interface to.</li> <li><code>is_default_gateway</code>: (Optional) If you want to specify the interface to be used as the default gateway for all traffic, set this to <code>true</code>. The default is <code>false</code>. If the instance has an interface on the shared network, that interface is always used as the default gateway.</li> <li><code>dns</code>: (Optional) A list of the DNS A record names for the instance.</li> <li><code>name_servers</code>: (Optional) A list of the name servers that should be sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li> <li><code>search_domains</code>: (Optional) A list of the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li> </ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on an IP Network</a>.</p>



Parameter	Required or Optional	Description
sshkeys	optional	<p>A list of the SSH public keys that you want to associate with the instance.</p> <p><b>Note:</b></p> <p>You don't need to provide any SSH public keys if you're creating a Windows instance, because you can't access a Windows instance using SSH. To access a Windows instance, see <a href="#">Accessing a Windows Instance Using RDP</a>.</p> <p>For each key, specify the three-part name in the <code>/Compute-identity_domain/user/object_name</code> format.</p> <p>You can associate the same key with multiple instances.</p> <p>The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at <code>http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key</code>.</p> <ul style="list-style-type: none"> <li>• Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the <code>authorized_keys</code> file of the <code>opc</code> user.</li> <li>• In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the <code>authorized_keys</code> file of the appropriate users.</li> </ul>

### Networking Attributes for Instances

There are several subparameters that you can specify under the `ethn` parameter in the networking section of instance attributes. The list of subparameters varies depending on whether you're defining a network interface on a shared network or an IP network.

Only one interface can be added to the shared network. If no subparameters are specified for the `ethn` parameter, the interface is implicitly added to the default security list in the shared network. You can't explicitly or implicitly define two interfaces to be added to the shared network.

### Subparameters for a Network Interface on the Shared Network

- `seclists`: (Optional) The security lists that you want to add the instance to.
 

For each security list, specify the three-part name in the `/Compute-identity_domain/user/object_name` format. You can attach an instance to a maximum of five security lists. If you launch an instance without specifying any security list, the instance is assigned to the `/Compute-identity_domain/default/default` security list.
- `nat`: (Optional) Indicates whether a temporary or permanent public IP address should be assigned to the instance.
  - To associate a temporary IP address with the instance for use during the lifetime of the instance, specify `ippool:/oracle/public/ippool`.
  - To associate a persistent IP address, specify `ipreservation:ipreservation_name`, where `ipreservation_name` is the three-part name of an existing IP reservation in the `/Compute-identity_domain/user/object_name` format.

If `nat` is not specified, then no public IP address is associated with your instance when it is created. If required, you can associate an IP address with the instance after the instance has been created.

- `dns`: (Optional) A list of the DNS A record names for the instance. The name is relative to the internal DNS domain.
- `model`: (Optional) The type of network interface card (NIC). The only allowed value is `e1000`.
- `name_servers`: (Optional) Enter the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface. If `name_servers` are set in both the IP network settings as well as the shared network settings, the name servers in the shared network will be used. To ensure that the name servers specified in the IP network are used, specify the same values for name servers on each interface.
- `search_domains`: (Optional) Enter the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface. If `search_domains` are set in both the IP network settings as well as the shared network settings, the search domains in the shared network will be used. To ensure that the search domains specified in the IP network are used, specify the same values for search domains on each interface.

### Subparameters for a Network Interface on an IP Network

- `ipnetwork`: The name of the IP network that you want to add the instance to.

If no name is specified, the interface isn't added to any IP network. Instead, it is implicitly added to the shared network. However, only one instance interface can be added to the shared network. If another interface is either implicitly or explicitly added to the shared network, the instance won't be created and will display an error.

Specify the three-part name of the IP network, in the `/Compute-identity_domain/user/object_name` format.

If an IP network belongs to an IP network exchange and if you have specified a host name, then that host name is resolvable by all IP networks connected to the IP network exchange.

- `ip`: (Optional) The static private IP address of the instance. This is a persistent private IP address, which is reserved for use with this instance. The private IP address must be unused and it must belong to the subnet of the selected `IP network`. Remember, too, that certain IP addresses in a subnet are reserved. For example, the first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.

If you don't specify an IP address, an IP address is assigned dynamically from the available IP addresses of the specified `ipnetwork`. However in this case, if you delete and re-create the instance, its IP address might change.

#### Note:

Dynamically allocated IP addresses are assigned from the top of the subnet range. It is recommended that you specify static IP addresses starting from the end of the subnet range to avoid conflicts.

- **address:** (Optional) The MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter 01:02:03:04:ab:cd as the MAC address but not 01-02-03-04-ab-cd. Ensure that the MAC addresses that you specify are unique within each IP network exchange and each IP network. If you specify a duplicate MAC address, each vNIC with that MAC address is disabled.
- **nat:** (Optional) A list of IP reservations that you want to associate with this interface. Specify `network/v1/ipmapreservation:ipmapreservation_name`, where `ipmapreservation_name` is the three-part name of an existing IP reservation in the `/Compute-identity_domain/user/object_name` format.

When you create an IP reservation, you specify the IP pool from which you want to reserve the IP address. You can associate a maximum of two IP reservations with each vNIC, one from each IP pool.

### Example:

```
"networking":
{
    "eth0": {
        "ipnetwork": "/test-customer/ipnet-1",
        "ip": "192.168.2.14",
        "nat": ["network/v1/ipmapreservation:/Compute-acme/joe/public-
ipmapreservation-1"]
    }
}
```

- **vnic:** (Optional) The three-part name of the vNIC in the `/Compute-identity_domain/user/object_name` format.

If you don't specify a name for this object, then the name is generated automatically.

When the vNIC name is generated automatically, the autogenerated instance id is included as part of the `object_name`. So if you delete and re-create an instance, the vNIC name will change. However, if you specify a vNIC name, the name won't change if you delete and re-create the instance.

Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.

- **vnicsets:** (Optional) A list of the three-part names of the vNICsets that you want to add this vnic to. Specifying vNICsets ensures that this vNIC is added to the required vNICsets whenever the instance is created and removed from the vNICset whenever the instance is deleted.

While creating an instance, you can add a vNIC to up to 4 vNICsets. To add a vNIC to more than 4 vNICsets, update the required vNICsets after the instance is created.

The vNICsets that you specify here must already exist when you create or re-create an instance.

If no vNICset is specified, then the vNIC is added to the default vNICset, `/Compute-identity_domain/default`.

If an empty list (`"vnicsets": []`) is specified, this vNIC isn't added to any vNICset, including the default vNICset.

- **is\_default\_gateway:** (Optional) If you want to specify the interface to be used as the default gateway for all traffic, set this to `true`. The default is `false`. Only one

interface on an instance can be specified as the default gateway. If the instance has an interface on the shared network, that interface is always used as the default gateway. You can specify an interface on an IP network as the default gateway only when the instance doesn't have an interface on the shared network.

- `dns`: (Optional) A list of the DNS A record names for the instance.

Each IP network has its own DNS server listening on the first IP address of the subnet. You can specify up to eight DNS A record names for each instance on an IP network. These names can be queried by instances on any IP network in the same IP network exchange.

If no static IP address is specified for the instance on the IP network, an IP address on the specified IP network is assigned automatically. After the instance is launched, the defined names are associated with the IP address that was automatically allocated to the instance.

The same DNS A record name can be specified for multiple instances.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
        "dns": [ "dns1.example.com", "dns2.bar.com" ]
    }
}
```

- `name_servers`: (Optional) A list of the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface. If `name_servers` are set in both the IP network as well as the shared network, the name servers in the shared network will be used. To ensure that the name servers specified in the IP network are used, specify the same values for name servers on each interface.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
        "dns": ["dns1.example.com", "dns2.bar.com"],
        "name_servers": ["192.168.12.1", "192.168.12.2"]
    }
}
```

In this example, the name servers 192.168.12.1 and 192.168.12.2 will be pushed to the instance through DHCP.

- `search_domains`: (Optional) A list of the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface. If `search_domains` are set in both the IP network as well as the shared network, the search domains in the shared network will be used. To ensure that the search domains specified in the IP network are used, specify the same values for search domains on each interface.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
```

```

        "dns": ["dns1.example.com", "dns2.bar.com"],
        "name_servers": ["192.168.12.1", "192.168.12.2"],
        "search_domains": ["example.com", "us.example1.com"]
    }
}

```

In this example, the search domain zones `example.com` and `us.example1.com` will be pushed to the instance through DHCP.

## Orchestration v1 Attributes for `network/v1/acl`

The following sample JSON shows the key attributes of the `network/v1/acl` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```

{
    "name": "/Compute-acme/joe/acl_1",
    "enabledFlag": true
}

```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
enabledFlag	optional	Allows the ACL to be enabled or disabled. This parameter is set to true by default. Specify false to disable the ACL.
description	optional	Description of the ACL.
tags	optional	Strings that you can use to tag the ACL.

## Orchestration v1 Attributes for `network/v1/ipaddressprefixset`

The following sample JSON shows the key attributes of the `network/v1/ipaddressprefixset` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```

{
    "name": "/Compute-acme/joe/ext_ip_address_list_1",
    "ipAddressPrefixes": ["203.0.113.0/30", "192.51.100.1/24"]
}

```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
ipAddressPrefixes	optional	Set of IPv4 addresses in CIDR address prefix format.
description	optional	Description of the IP address prefix set.
tags	optional	Strings that you can use to tag the IP address prefix set.

## Orchestration v1 Attributes for `network/v1/ipassociation`

The following sample JSON shows the key attributes of the `network/v1/ipassociation` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/IP-association-vnic1",
  "ipAddressReservation": "/Compute-acme/joe/IPres-for-instancetype-
vnic1",
  "vnic": "/Compute-acme/joe/instancetype-vnic1"
}
```


Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
ipAddressReservation	optional	The name of the IP reservation that you want to associate with an instance.
vnic	optional	The name of the vNIC that you want to associate the IP reservation with.
description	optional	Description of the IP association.
tags	optional	Strings that you can use to tag the IP association.

## Orchestration v1 Attributes for `network/v1/ipnetwork`

The following sample JSON shows the attributes of the `network/v1/ipnetwork` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipnet1",
  "ipAddressPrefix": "192.168.3.0/24",
  "ipNetworkExchange": "/Compute-acme/joe/ipnetworkexchange1"
}
```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
ipAddressPrefix	required	<p>The set of IP addresses allocated to your IP network, specified in the CIDR format. When you create instances, you can associate a vNIC on the instance with an IP network. That vNIC on the instance is then allocated an IP address from the specified IP network.</p> <p>Select the IP address prefix for your IP networks carefully. Consider the number of instances that you might want to add to the network. This will help determine the size of the subnet required.</p> <p>If you create multiple IP networks and you might want to add these IP networks to the same IP network exchange, then ensure that you don't allocate overlapping address ranges to these IP networks.</p> <p>Similarly, if you plan to connect to your IP networks using VPN, then ensure that the addresses you specify for your IP networks don't overlap with each other, or with the IP addresses used in your on-premises network.</p>

Parameter	Required or Optional	Description
ipNetworkExchange	optional	The IP network exchange that you want to add this IP network to. An IP network can belong to only one IP network exchange. Before you specify an IP network exchange for an IP network, ensure that the IP addresses in this IP network don't overlap the IP addresses in any other network in the same IP network exchange.
<div style="border-left: 2px solid #0070C0; padding-left: 10px; margin-left: 200px;"> <p> <b>Note:</b></p> <p>You should ensure that the IP network exchange you reference currently exists. If the IP network exchange hasn't been created or has been deleted, then when you add an instance interface to this IP network while creating the instance, the instance will go into an error state and won't be created.</p> </div>		
<p>If you want to connect IP networks by using an IP network exchange, it is recommended that you do this before creating instances with an interface on those IP networks. This ensures that routes are appropriately configured on instances by the DHCP client during instance initialization.</p>		
description	optional	Description of the IP network.
tags	optional	Strings that you can use to tag the IP network.

## Orchestration v1 Attributes for `network/v1/ipnetworkexchange`

The following sample JSON shows the required attribute of the `network/v1/ipnetworkexchange` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipnetworkexchange1"
}
```



Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
description	optional	Description of the IP network exchange.
tags	optional	Strings that you can use to tag the IP network exchange.

## Orchestration v1 Attributes for `network/v1/ipreservation`

The following sample JSON shows the key attributes of the `network/v1/ipreservation` object type for IP networks. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/IPres-for-instance1-vnic1",
  "ipAddressPool": "/oracle/public/public-ippool"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
ipAddressPool	required	The IP address pool from which you want to reserve an IP address. Enter one of the following: <ul style="list-style-type: none"> <li><code>/oracle/public/public-ippool</code>: When you attach an IP address from this pool to an instance, you enable access between the public Internet and the instance.</li> <li><code>/oracle/public/cloud-ippool</code>: When you attach an IP address from this pool to an instance, the instance can communicate privately (that is, without traffic going over the public Internet) with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region.</li> </ul>

Parameter	Required or Optional	Description
description	optional	Description of the IP reservation.
tags	optional	Strings that you can use to tag the IP reservation.

## Orchestration v1 Attributes for `network/v1/route`

The following sample JSON shows the required attributes of the `network/v1/route` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/route1",
  "nextHopVnicSet": "/Compute-acme/joe/vnicset1",
  "ipAddressPrefix": "192.168.0.0/16"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
ipAddressPrefix	required	The IP address prefix, in CIDR format, of the destination network that you want to specify the route to.
nextHopVnicSet	required	The vNICset that you want to use to route packets to the destination network. When a vNICset containing multiple vNICs is used in a route, Equal Cost Multipath (ECMP) anycast routing is implemented. Traffic routed by that route is load balanced across all the vNICs in the vNICset. Using vNICsets with multiple vNICs also ensures high availability for traffic across the specified vNICs.
administrativeDistance	optional	The route's administrative distance. Specify 0 (the default), 1, or 2.  The administrative distance indicates the priority of a route. The highest priority is 0. The route with the highest priority is used. If multiple routes have the highest priority, all those routes are used.
description	optional	Description of the route.
tags	optional	Strings that you can use to tag the route.

## Orchestration v1 Attributes for `network/v1/secprotocol`

The following sample JSON shows the key attributes of the `network/v1/secprotocol` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "description": "Sec Protocol 1",
  "dstPortSet": ["20", "155-1100"],
  "ipProtocol": "tcp",
  "name": "/Compute-acme/joe/secprotocol_1",
  "srcPortSet": ["10", "55-100"]
}
```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
ipProtocol	optional	<p>The protocol used in the data portion of the IP datagram.</p> <p>The value that you specify can be either a text representation of a protocol or any unsigned 8-bit assigned protocol number in the range 0–254. See <i>Assigned Internet Protocol Numbers</i> (<a href="http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml">http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml</a>).</p> <p>The following text representations are allowed:</p> <ul style="list-style-type: none"> <li>• tcp</li> <li>• udp</li> <li>• icmp</li> <li>• igmp</li> <li>• ipip</li> <li>• rdp</li> <li>• esp</li> <li>• ah</li> <li>• gre</li> <li>• icmpv6</li> <li>• ospf</li> <li>• pim</li> <li>• sctp</li> <li>• mplsip</li> <li>• all</li> <li>• Any number from 0 to 254</li> </ul> <p>If no protocol is specified, all protocols are allowed.</p>

Parameter	Required or Optional	Description
srcPortSet	optional	<p>List of port numbers or port range strings to match the packet's source port.</p> <ul style="list-style-type: none"> <li>For <code>tcp</code>, <code>sctp</code>, and <code>udp</code>, each port is a source transport port, between 0 and 65535, inclusive.</li> <li>For <code>icmp</code>, each port is an ICMP type, between 0 and 255, inclusive.</li> </ul> <p>If no source ports are specified, all source ports or ICMP types are allowed.</p>
dstPortSet	optional	<p>List of port numbers or port range strings to match the packet's destination port.</p> <p>For <code>tcp</code>, <code>sctp</code>, and <code>udp</code>, each port is a destination transport port, between 0 and 65535, inclusive. For <code>icmp</code>, each port is an ICMP code, between 0 and 255, inclusive.</p> <p>If no destination ports are specified, all destination ports or ICMP codes are allowed.</p>
description	optional	Description of the security protocol.
tags	optional	Strings that you can use to tag the security protocol.

## Orchestration v1 Attributes for `network/v1/secrule`

The following sample JSON shows the key attributes of the `network/v1/secrule` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "acl": "/Compute-acme/joe/acl_1",
  "description": "Sec Rule 1",
  "flowDirection": "egress",
  "name": "/Compute-acme/joe/ipnetSecrule1",
  "secProtocols": ["/Compute-acme/joe/secprotocol_1"],
  "srcIpAddressPrefixSets": ["/Compute-acme/joe/
ext_ip_address_list_1"]
}
```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>

Parameter	Required or Optional	Description
flowDirection	required	The direction of flow of traffic that this rule applies to. Allowed values are <code>ingress</code> or <code>egress</code> .
srcVnicSet	optional	The vNICset from which you want to permit traffic. Only packets from vNICs in the specified vNICset are permitted. When no source vNICset is specified, traffic from any vNIC is permitted.
dstVnicSet	optional	The vNICset to which you want to permit traffic. Only packets to vNICs in the specified vNICset are permitted. When no destination vNICset is specified, traffic to any vNIC is permitted.
srcIpAddressPrefixSets	optional	A list of IP address prefix sets from which you want to permit traffic. Only packets from IP addresses in the specified IP address prefix sets are permitted. When no source IP address prefix sets are specified, traffic from any IP address is permitted.
dstIpAddressPrefixSets	optional	A list of IP address prefix sets to which you want to permit traffic. Only packets to IP addresses in the specified IP address prefix sets are permitted. When no destination IP address prefix sets are specified, traffic to any IP address is permitted.
securityProtocols	optional	A list of security protocols for which you want to permit traffic. Only packets that match the specified protocols and ports are permitted. When no security protocols are specified, traffic using any protocol over any port is permitted.
enabledFlag	optional	Allows the security rule to be enabled or disabled. This parameter is set to <code>true</code> by default. Specify <code>false</code> to disable the security rule.
acl	optional	The name of the access control list (ACL) that contains this security rule.
description	optional	Description of the security rule.
tags	optional	Strings that you can use to tag the security rule.

## Orchestration v1 Attributes for `network/v1/vnicset`

The following sample JSON shows the key attributes of the `network/v1/vnicsets` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/vnicset1",
  "appliedAcls": ["/Compute-acme/joe/acl_1", "/Compute-acme/joe/acl_2"]
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
vnic	optional	The list of vNICs associated with this vNICset.
appliedAcls	optional	The names of the ACLs applied to the vNICs in the vNICset. A vNICset can have multiple ACLs applied to it and an ACL can be applied to multiple vNIC sets.
description	optional	Description of the route.
tags	optional	Strings that you can use to tag the IP network exchange.

## Orchestration v1 Attributes for `orchestration`

The `orchestration` object type is used in nested orchestrations, when you want to launch one or more orchestrations from within an orchestration. See [About Nested Orchestrations](#). The `orchestration` object type has only a single attribute, `name`. The following sample JSON shows this attribute and the table below provides a description of this attribute.

```
{
  "name": "/Compute-acme/joe/myInstances"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.

## Orchestration v1 Attributes for `secapplication`

The following sample JSON shows the key attributes of the `secapplication` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/wlsadmin_ssl",
  "dport": 7002,
  "protocol": "tcp"
}
```

Parameter	Required or Optional	Description
<code>name</code>	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
<code>protocol</code>	required	<p>The protocol to use.</p> <p>The value that you specify can be either a text representation of a protocol or any unsigned 8-bit assigned protocol number in the range 0–254. See <i>Assigned Internet Protocol Numbers</i> (<a href="http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml">http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml</a>).</p> <p>For example, you can specify either <code>tcp</code> or the number 6.</p> <p>The following text representations are allowed: <code>tcp</code>, <code>udp</code>, <code>icmp</code>, <code>igmp</code>, <code>ipip</code>, <code>rdp</code>, <code>esp</code>, <code>ah</code>, <code>gre</code>, <code>icmpv6</code>, <code>ospf</code>, <code>pim</code>, <code>sctp</code>, <code>mplsip</code>, <code>all</code>.</p> <p>To specify all protocols, set this to <code>all</code>.</p>
<code>dport</code>	optional	<p>The TCP or UDP destination port number.</p> <p>You can also specify a port range, such as 5900-5999 for TCP.</p> <p>If you specify <code>tcp</code> or <code>udp</code> as the protocol, then the <code>dport</code> parameter is required; otherwise, it is optional.</p> <p>This parameter isn't used by the ICMP protocol or the GRE protocol.</p> <p><b>Note:</b> This request fails if the range-end is lower than the range-start. For example, if you specify the port range as 5000–4000.</p>

Parameter	Required or Optional	Description
<code>icmp</code> <code>type</code>	optional	<p>The ICMP type.</p> <p>This parameter is relevant only if you specify <code>icmp</code> as the protocol. You can specify one of the following values:</p> <ul style="list-style-type: none"> <li><code>echo</code></li> <li><code>reply</code></li> <li><code>ttl</code></li> <li><code>traceroute</code></li> <li><code>unreachable</code></li> </ul> <p>If you specify <code>icmp</code> as the protocol and don't specify <code>icmp</code><code>type</code> or <code>icmp</code><code>code</code>, then all ICMP packets are matched.</p>
<code>icmp</code> <code>code</code>	optional	<p>The ICMP code.</p> <p>This parameter is relevant only if you specify <code>icmp</code> as the protocol. You can specify one of the following values:</p> <ul style="list-style-type: none"> <li><code>network</code></li> <li><code>host</code></li> <li><code>protocol</code></li> <li><code>port</code></li> <li><code>df</code></li> <li><code>admin</code></li> </ul> <p>If you specify <code>icmp</code> as the protocol and don't specify <code>icmp</code><code>type</code> or <code>icmp</code><code>code</code>, then all ICMP packets are matched.</p>
<code>description</code>	optional	A description of the security application.

## Orchestration v1 Attributes for `seclist`

The following sample JSON shows the required attributes of the `seclist` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/admin_ips",
  "seclistentries": ["203.0.113.0/30"]
}
```

Parameter	Required or Optional	Description
<code>name</code>	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>



Parameter	Required or Optional	Description
secipentri es	required	A comma-separated list of the subnets (in CIDR format) or IPv4 addresses for which you want to create this security IP list.  For example, to create a security IP list containing the IP addresses 203.0.113.1 and 203.0.113.2, enter one of the following:  "203.0.113.0/30"  "203.0.113.1", "203.0.113.2"
descriptio n	optional	A description of the security IP list.

## Orchestration v1 Attributes for `seclist`

The following sample JSON shows the required attribute of the `seclist` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/sysadmin_seclist"
}
```

Parameters	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
policy	optional	The policy for inbound traffic to the security list. You can specify one of the following values:  deny (default): Packets are dropped. No response is sent. reject: Packets are dropped, but a response is sent. permit: Packets are allowed. This policy effectively turns off the firewall for all instances in this security list.
outbound_c idr_policy	optional	The policy for outbound traffic from the security list. You can specify one of the following values:  deny: Packets are dropped. No response is sent. reject: Packets are dropped, but a response is sent. permit (default): Packets are allowed.
descriptio n	optional	A description of the security list.

## Orchestration v1 Attributes for `secrule`

The following sample JSON shows the required attributes of the `secrule` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/admin_ssh_to_sysadmin_rule",
  "application": "/oracle/public/ssh",
  "src_list": "seciplist:/Compute-acme/joe/admin_ips",
  "dst_list": "seclist:/Compute-acme/joe/sysadmin_seclist",
  "action": "PERMIT"
}
```

Parameter	Required or Optional	Description
<code>name</code>	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
<code>src_list</code>	required	The three-part name ( <code>/Compute-identity_domain/user/object_name</code> ) of the source security list or security IP list. You must use the prefix <code>seclist:</code> or <code>seciplist:</code> to identify the list type.
<code>dst_list</code>	required	The three-part name ( <code>/Compute-identity_domain/user/object_name</code> ) of the destination security list or security IP list. You must use the prefix <code>seclist:</code> or <code>seciplist:</code> to identify the list type. <b>Note:</b> You can specify a security IP list as the destination in a <code>secrule</code> , provided <code>src_list</code> is a security list that has DENY as its outbound policy.
<code>application</code>	required	The three-part name of the security application: ( <code>/Compute-identity_domain/user/object_name</code> ) for user-defined security applications and <code>/oracle/public/object_name</code> for predefined security applications.
<code>action</code>	required	Set this parameter to PERMIT.
<code>description</code>	optional	A description of the security rule.
<code>disabled</code>	optional	Indicates whether the security rule is enabled (set to True) or disabled (False). The default setting is False.

## Orchestration v1 Attributes for `storage/volume`

The following sample JSON shows the key attributes of the `storage/volume` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/boot",
  "bootable": true,
  "imagelist": "/oracle/public/oel_6.6_20GB_x11_RD",
  "properties": ["/oracle/public/storage/default"],
  "size": "22548578304"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.
size	required	The size of this storage volume. Use one of the following abbreviations for the unit of measurement: <ul style="list-style-type: none"> <li>• B or b for bytes</li> <li>• K or k for kilobytes</li> <li>• M or m for megabytes</li> <li>• G or g for gigabytes</li> <li>• T or t for terabytes</li> </ul> For example, to create a volume of size 10 gigabytes, you can specify 10G, or 10240M, or 10485760K, and so on. The allowed range is from 1 GB to 2 TB, in increments of 1 GB.
properties	required	Based on your latency and IOPS requirements, select one of the following storage properties: <ul style="list-style-type: none"> <li>• For standard latency and throughput, specify <code>/oracle/public/storage/default</code>.</li> <li>• For high latency and throughput, specify <code>/oracle/public/storage/latency</code>.</li> <li>• For the highest latency and throughput, specify <code>/oracle/public/storage/ssd/gpl</code>.</li> </ul>
description	optional	The description of the storage volume.

Parameter	Required or Optional	Description
bootable	optional	<p>Indicates whether the storage volume can be used as the boot disk for an instance.</p> <p>The default value is <code>False</code> (not a bootable volume).</p> <p>If you set the value to <code>True</code>, then you must specify values for the following parameters:</p> <ul style="list-style-type: none"> <li><code>imagelist</code> The machine image that you want to extract on to the storage volume that you're creating.</li> <li><code>imagelist_entry</code> (Optional) The version of the image list entry that you want to extract. The default value is 1.</li> </ul>
tags	optional	Strings that you can use to tag the storage volume.

## Uploading an Orchestration v1

To use an orchestration to control the provisioning and life cycle of resources in Compute Classic, you must define the orchestration in a JSON-format file and then upload the orchestration to Compute Classic.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have already created the orchestration file that you want to upload. See [Building Your First Orchestration v1](#).

You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. If your JSON isn't valid, then an error occurs when you upload the orchestration. Oracle doesn't support or endorse any third-party JSON-validation tool.

### Procedure

- Sign in to the Compute Classic console.
- (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

- Click the **Orchestrations** tab.
- Click **Upload Orchestration** and select the orchestration file that you want to upload.

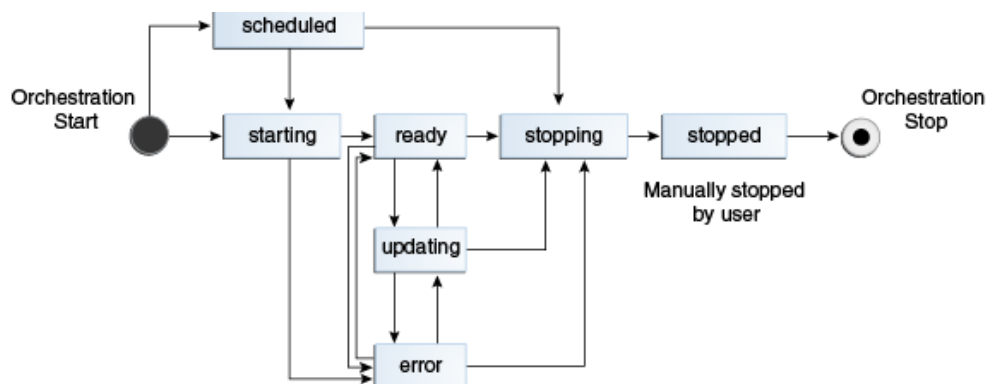
To upload an orchestration using the CLI, use the `opc compute orchestration add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To upload an orchestration using the API, use the `POST /orchestration/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Orchestrations v1 Life Cycle

When you start an orchestration, the objects defined in it are created and the orchestration moves to the `ready` state. When you terminate an orchestration, the objects defined in it are deleted and the orchestration moves to the `stopped` state.

The following figure shows the states that an orchestration can be in.



### starting

The orchestration is starting.

### scheduled

A future `start_time` has been specified for the orchestration.

- When the current time is equal to or past the `start_time` value, then the state of the orchestration changes to `starting`.
- To cancel a current schedule, terminate the orchestration. The state of the orchestration then changes to `stopping`.

### ready

The orchestration is running.

- Note that, for any object where the HA policy isn't specified or is set to `none`, you can still update or delete the object using the web console or the API. In this case, the orchestration continues to be in the `ready` state, even though some or all of the objects created using that orchestration may have been deleted.
- For instances where the HA policy is set to `active`, if the orchestration is in the `ready` state, you can update the instance using the web console or the API, but you can't delete the instance, because it is re-created automatically. To delete such instances, you must terminate the orchestration.

### updating

The orchestration is being updated.

- When an orchestration is in the `ready` or `error` state, you can update it by using the `PUT /orchestration/name` API call. This causes the state of the orchestration to change to `updating`.
- When an orchestration is in the `updating` state, no further updates can be made. Attempts to update such an orchestration are rejected with a validation error.
- If an orchestration in the `updating` state encounters an error, its state changes `error`. If no errors are encountered, then the orchestration completes the updates and returns to the `ready` state.
- When you terminate an orchestration that's in the `updating` state, it transitions to the `stopping` state.

#### **error**

One or more instances in the orchestration have encountered an error.

- The orchestration remains in the `error` state until all the instances defined in it are running.
- Wait to see if all the instances start running and the state of the orchestration changes automatically to `ready`. If that doesn't happen, then terminate the orchestration, identify and fix the error, and start the orchestration again.

#### **stopping**

The orchestration is stopping.

If any of the objects defined in an orchestration are used or referenced by another object, the orchestration won't be able to delete the referenced objects, and it can get stuck in the **Stopping** state. See [My orchestration is stuck in the stopping state](#).

#### **stopped**

The orchestration has stopped. All the objects defined in the orchestration have been deleted.

## Starting an Orchestration v1

When you start an orchestration, the objects defined in it are created, and when you stop an orchestration, those objects are deleted.

Plan your orchestrations carefully, so that you can control the creation and deletion of objects that consume resource quotas. For example, if you're about to start an orchestration that creates a large number of storage volumes, consider whether you really need all those resources. If not, redefine your orchestration to create only the resources that you need.

#### **Prerequisites**

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have uploaded the orchestration to Compute Classic. See [Uploading an Orchestration v1](#).
- You must have already created all the objects or nested orchestrations that this orchestration depends on.

- If you created an instance using the Create Instance wizard, the appropriate master, instance, and storage orchestrations would have been created automatically. Next, if you stopped the master orchestration, the instance and storage orchestrations would have stopped. If you want to start those orchestrations again, ensure that all of the objects referenced by those orchestrations exist and are available before starting the master orchestration.


 **Note:**

If any of the objects defined in an orchestration already exist, another object of the same type with the same name won't be created and the existing object won't be modified. The orchestration will continue starting, without reporting an error. To ensure that an orchestration creates each object that is defined in it, ensure that each object defined in an orchestration has a unique name, so that objects of the same type with the same name don't already exist.

### Procedure

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

3. Click the **Orchestrations** tab.
4. Go to the orchestration that you want to start. From the  menu, select **Start**.

When you start an orchestration, its status changes to **Starting** and the objects defined in the orchestration are provisioned. When all the objects have been created, the status of the orchestration changes to **Ready**.

If the orchestration can't create an object, its status changes to **Error**. An orchestration might transition from the **Error** to the **Ready** state when it completes creating all the specified objects.

If the status of your orchestration continues to show **Error**, then stop the orchestration, identify and fix the issue in the orchestration JSON file, and then start the orchestration again.

To start an orchestration using the CLI, use the `opc compute orchestration update --action START` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To start an orchestration using the API, use the `PUT /orchestration/name` method with the query argument `action=START`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After starting an orchestration, you can view its status on the Orchestrations page. If you no longer require any of the objects created by an orchestration, then to delete the objects, stop the orchestration. See [Terminating an Orchestration v1](#).

## Monitoring Orchestrations v1

The Orchestrations page shows you a list of your orchestrations and the status of each orchestration.


To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.

All orchestrations are displayed, with information about their description and status.

### Tip:

You can filter the list of orchestrations according to their category or status. To view orchestrations with a specific status (such as ready, error, or stopped), click the **Show** menu and select the appropriate filter. To view orchestrations of a specific category (such as all or personal), click the **Category** menu and select the appropriate filter.

3. Go to the orchestration that you want to view and, from the  menu, select **View**.

The orchestration details page shows you the details of the current state of the orchestration, including return parameters, in JSON format. For information about the return parameters of an instance, see [Return Parameters Displayed in Orchestrations v1](#).

To get a list of your orchestrations using the CLI, use the `opc compute orchestration list` command and to view the details of an orchestration, use the `opc compute orchestration get` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To get a list of your orchestrations using the API, use the `GET /orchestration/container` method and to view the details of an orchestration, use the `GET /orchestration/name` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

For information about the status of an orchestration, see [Orchestrations v1 Life Cycle](#).

To start an orchestration, see [Starting an Orchestration v1](#) and to stop an orchestration, see [Terminating an Orchestration v1](#).



## Return Parameters Displayed in Orchestrations v1

When you view an orchestration using the web console, you'll see that the orchestration contains additional information about your instances, such as its status and the most recent start or stop time. This information contains return parameters that tell you about the current state of your instance. These return parameters are not part of the input that you provided in the JSON file that you uploaded.

You'll also see that the orchestration includes certain optional parameters that you might not have specified in the JSON that you uploaded. These optional parameters are displayed either empty, or with a default value. For a description of optional input parameters, see [Attributes in Orchestrations v1](#).

You might also notice that the sequence of objects is different from the sequence of objects in the JSON file that you uploaded. This happens because Compute Classic rearranges the objects according to a certain internal sequence. However, this has no impact on the values you provided or the way your orchestration works.

The following table shows the return parameters displayed for your instance when you view an orchestration using the web console.

Return Parameters	Description
<b>Top-level Parameters</b>	
status	Shows the current status of the orchestration.
account	Shows the default account for your identity domain.
uri	Shows the complete URI of the orchestration.
info	The nested parameter <code>errors</code> shows which object in the orchestration has encountered an error. Empty if there are no errors.
status_timestamp	This information is generally displayed at the end of the orchestration JSON. It indicates the time that the current view of the orchestration was generated. This information shows only when the orchestration is running.
<b>Oplan Parameters</b>	
status	Shows the current status of the <code>oplan</code> .
info	If the orchestration has encountered an error, the nested parameter <code>errors</code> shows the errors. Empty if there are no errors.
status_timestamp	This information is generally displayed towards the end of the orchestration JSON. It indicates the time that the current view of the orchestration was generated. This information shows only when the orchestration is running.
<b>Instance Parameters</b>	
placement_requirements	Empty. This parameter is not used.
ip	If the instance is running, this parameter shows its private IP address. This information doesn't show when an instance is not running.
state	If the orchestration is running, this parameter shows the current state of the instance. This information doesn't show when an orchestration is stopped or if the instance couldn't be created due to an error.
start_time	If the orchestration is running, this parameter shows the time the instance was created. This information doesn't show when an orchestration is stopped or if the instance couldn't be created due to an error.

Return Parameters	Description
<code>error_reason</code>	If the instance goes into an error state, this parameter shows the reason for the error. This information doesn't show when an instance is not in an error state.

## Terminating an Orchestration v1


When you terminate or stop an orchestration, all the instances and other resources that were provisioned by that orchestration are deleted.

### Note:

When you terminate an orchestration, only the resources that are created by the orchestration are deleted. For example, if you use an orchestration to create storage volumes and attach them to your instances, then such storage volumes are deleted when you terminate the orchestration, and you lose the data stored on those storage volumes. However, if an orchestration specifies only *attachments* to storage volumes that are created outside the orchestration, then when you terminate the orchestration, the storage volumes aren't deleted.

If you use the Create Instance wizard to create an instance and the required storage volumes, then separate orchestrations are automatically created for the instance and the storage volumes. When you delete an instance, if you want to retain the data on the attached storage volumes, ensure that you terminate only the instance orchestration, not the master orchestration. If you terminate the master orchestration, both the instance and the storage volumes that were created while creating the instance will be deleted. When you start the master orchestration again later on to re-create the instance, the storage volumes will be re-created; however, the data you had written to those storage volumes will be lost.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to terminate. From the  menu, select **Terminate**.

The status of the orchestration changes to **Stopping**.

 **Note:**

If any of the objects defined in an orchestration are used or referenced by another object, the orchestration won't be able to delete the referenced objects, and it can get stuck in the **Stopping** state. See [My orchestration is stuck in the stopping state](#).

After all objects have been deleted, the status of the orchestration changes to **Stopped**. You can view the orchestration, download it, or start it again.

To terminate an orchestration using the CLI, use the `opc compute orchestration update --action STOP` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).


To terminate an orchestration using the API, use the `PUT /orchestration/name` method with the query argument `action=STOP`. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

When you no longer need an orchestration, you can delete it. See [Deleting an Orchestration v1](#).

## Downloading an Orchestration v1

You can download the orchestration file to your local host, edit it, and upload a modified orchestration file as a new orchestration.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to download. From the  menu, select **Download**, and save the orchestration file on your local host.

You can edit the downloaded orchestration file on your local host, as required, by using any text editor, and then upload the edited orchestration file as a new orchestration. Remember to change the `name` attribute in the JSON file.

For the procedure to upload an orchestration to Compute Classic, see [Uploading an Orchestration v1](#).

To download an orchestration using the CLI, use the `opc compute orchestration get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To download an orchestration using the API, use the `GET /orchestration/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating an Orchestration v1

You can update an orchestration in either of the following ways:


- Directly in the web console, by selecting the **Update** option.  
This option is enabled only when the orchestration is in the **Stopped** state.
- By downloading the orchestration file to your local host and updating it using a text editor.

You'll have to stop and delete the existing orchestration before you can upload the modified orchestration and start it.

### **Caution:**


You must stop an orchestration to update it. When you stop an orchestration, all the resources that were provisioned by that orchestration are deleted.



To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. You can update an orchestration either directly in the web console, or by downloading it to your local host.
  - To update an orchestration directly in the web console:
    - a. Identify the orchestration that you want to update. If the orchestration is in the **Ready** state, to stop the orchestration, from the  menu, select **Stop**.

### **Caution:**



When you stop an orchestration, all the resources that were provisioned by that orchestration are deleted.

- b. When the orchestration is in the **Stopped** state, from the  menu, select **Update**.
- c. Modify the orchestration as required. Ensure that your modifications contain valid JSON and that you don't update any return parameters or read-only information. When you're done, click **Update**.

- To download an orchestration and edit it on your local host:
  - a. Identify the orchestration that you want to update. From the  menu, select **Download**, and save the orchestration file on your local host.
  - b. Modify the downloaded orchestration file on your local host, as required, by using any text editor. You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. If your JSON isn't valid, then an error occurs when you upload the orchestration. Oracle doesn't support or endorse any third-party JSON-validation tool.
  - c. Stop the existing orchestration. Go to the orchestration that you want to update and, from the  menu, select **Stop**.

 **Caution:**

When you stop an orchestration, all the resources that were provisioned by that orchestration are deleted.

- d. Delete the existing orchestration. Go to the orchestration that you want to update and, from the  menu, select **Delete**.
  - e. Upload the edited orchestration file. Click **Upload Orchestration** and select the updated orchestration file, then click **Upload**.
4. To start the updated orchestration, from the  menu, select **Start**.

To download an orchestration using the CLI, use the `opc compute orchestration get` command. After editing an orchestration, to upload it using the CLI, use the `opc compute orchestration update` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To download an orchestration using the API, use the `GET /orchestration/name` method. After editing an orchestration, to upload it using the API, use the `PUT /orchestration/name` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Deleting an Orchestration v1

After you've stopped an orchestration, if you don't need it any more, you can delete the orchestration. When you delete an orchestration, it's no longer listed on the Orchestrations page, and you can't perform any action on it.


 **Note:**

Deleting an orchestration doesn't delete the objects created by the orchestration. To delete the objects created by an orchestration, you must stop the orchestration.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- You must have stopped the orchestration that you want to delete. See [Terminating an Orchestration v1](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to delete. From the  menu, select **Delete**.
4. You're prompted to confirm the deletion.

If the orchestration that you're deleting is a master orchestration that references other orchestrations, and if all the nested orchestrations are in the `stopped` state, then to delete all the nested orchestrations, select **Delete all nested orchestrations**.

#### Note:

If any of the nested orchestrations aren't in the `stopped` state, then you can stop and delete each nested orchestration individually, if required.

Click **Yes** to confirm the deletion.

To delete an orchestration using the CLI, use the `opc compute orchestration delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an orchestration using the API, use the `DELETE /orchestration/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

# 7

## Managing Resources Using Orchestrations v2

### Topics

- [About Orchestrations v2](#)
- [Comparing Orchestrations v1 and Orchestrations v2](#)
- [Object References and Relationships](#)
- [Object Persistence in Orchestrations v2](#)
- [Object Types in Orchestrations v2](#)
- [Orchestration v2 Templates and Samples](#)
- [Workflow for Creating Instances Using Orchestrations v2](#)
- [Building Your First Orchestration v2](#)
- [Attributes in Orchestrations v2](#)
- [Orchestration v2 Life Cycle](#)
- [Managing Orchestrations v2](#)

## About Orchestrations v2

### Topics

- [What Is an Orchestration?](#)
- [Orchestrations v2 Terminology](#)
- [Object Types in Orchestrations v2](#)
- [Recovering Failed Objects in Orchestrations v2](#)

### What Is an Orchestration?

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

For example, you can use an orchestration to create and manage a collection of instances hosting a multitiered application stack with all the necessary networking, storage, and security resources.

At any time, you can delete and re-create all the objects in an orchestration by terminating and activating the orchestration. Storage attachments, security lists, and so on are re-created and re-associated automatically. If you want to delete and re-create specific objects, you can suspend and activate the orchestration. When orchestrations v2 are suspended, only non-persistent objects are deleted. See [Object Persistence in Orchestrations v2](#).

In orchestrations v2, you can add, remove, or update objects without terminating the entire orchestration.

For an overview of the benefits of using orchestrations v2 and to understand how orchestrations v2 differ from orchestrations v1, see [Comparing Orchestrations v1 and Orchestrations v2](#).

To create instances using orchestrations v2, you build an orchestration in a JSON-formatted file and upload it to Compute Classic. If the orchestration has the desired state specified as `active`, it starts automatically. For an example of a simple orchestration file that you can use to learn how to build your first orchestration, see [Building Your First Orchestration v2](#). But before that, do read the remainder of this topic and become familiar with the features, terminology, and concepts of orchestrations v2.

### Orchestrations v2 Terminology

Term	Description
objects	An object is the primary building block of an orchestration. Each object contains all the attributes for the compute, networking, or storage resource that you want to create. An orchestration can contain up to 100 objects.
type	The <code>type</code> attribute defines the type of the object that you want to create. For example, if you want to create a storage volume, the <code>type</code> would be <code>StorageVolume</code> . If you want to create an instance, the <code>type</code> would be <code>Instance</code> . See <a href="#">Attributes in Orchestrations v2</a> .
template	The <code>template</code> attribute defines the properties or characteristics of the Compute Classic resource that you want to create, as specified by the <code>type</code> attribute. The fields in the <code>template</code> section vary depending on the specified <code>type</code> . For example, if you want to create a storage volume, the <code>type</code> would be <code>StorageVolume</code> , and the <code>template</code> would include <code>size</code> and <code>bootable</code> . If you want to create an instance, the <code>type</code> would be <code>Instance</code> , and the <code>template</code> would include instance-specific attributes, such as <code>imagelist</code> and <code>shape</code> .

### Object Types in Orchestrations v2

An orchestration can contain up to 100 objects. You can define the type of an object by setting the `type` parameter. You can define any of the following object types:

Type	Description
<code>Acl</code>	Creates an access control list (ACL) that can be applied to interfaces that are part of your IP networks.
<code>Backup</code>	Creates a back up of a storage volume using a specified backup configuration.
<code>BackupConfiguration</code>	Specifies the storage volume to back up, along with the backup schedule, retention count, and the name of the snapshot to be created.
<code>Instance</code>	Creates an instance.



Type	Description
IpAddressAssociation	Associates a public IP address reservation with an interface on an instance that is attached to an IP network.
IpAddressPrefixSet	Creates an IP address prefix set. This can be used as a source or destination in security rules that determine access to or from the virtual interfaces of instances that are attached to IP networks.
IpAddressReservation	Reserves a public IP address from a specified IP pool. This IP address can be associated with the virtual interface of an instance that is attached to an IP network.
IpNetwork	Creates an IP network. You can specify an IP network in the networking attributes while creating an instance.
IpNetworkExchange	Creates an IP network exchange. You can add IP networks to an IP network exchange either while creating the IP network, or later, by updating the IP network.
IPReservation	Reserves a public IP address that can be used in the shared network.
OSSContainer	Creates a container in the specified Oracle Cloud Infrastructure Object Storage Classic account.
Restore	Restores a storage volume from the specified backup.
Route	Creates a route to a specified destination using the specified vNICset.
SecApplication	Creates a security application that can be used in a security rule created for the shared network.
SecIPList	Creates a security IP list.
SecList	Creates a security list.
SecRule	Creates a security rule.
SecurityProtocol	Creates a security protocol that can be used in a security rule created for IP networks.
SecurityRule	Creates a security rule which can be added to an access control list (ACL). ACLs are used to control the flow of traffic across your IP networks.
SSHKey	Adds an SSH key.
StorageAttachment	Attaches a storage volume to an instance.
StorageSnapshot	Creates a snapshot of a storage volume.
StorageVolume	Creates a storage volume.
VirtualNicSet	Creates a vNICset, which contains one or more virtual network interfaces (vNICs). While creating an instance, you can specify the vNICsets that you want to add each vNIC to.

## Recovering Failed Objects in Orchestrations v2

All the objects defined in orchestrations v2 recover automatically from failure. Orchestrations v2 use *object references* to recover interdependent objects to a healthy state. See [Object References and Relationships](#).

For example, consider an orchestration for an instance. The instance object may reference several other objects, such as storage volumes and IP reservations. For each storage volume to be attached to the instance, a storage attachment object in the orchestration references the instance and the appropriate storage volume. If the instance fails, these storage attachments are re-created automatically.

During the recovery of an object, all the other objects that reference the failed object are considered.

Note that when recovering from a failure, orchestrations don't consider object relationships, which define the sequence in which the objects in an orchestration must be created.

For information on the life cycle of orchestrations v2, see [Orchestration v2 Life Cycle](#).

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update, delete, or change the attributes of an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

## Comparing Orchestrations v1 and Orchestrations v2

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

In earlier releases of Compute Classic, you could use orchestrations v1 to create and manage resources. From release 17.1.6 onwards, you can also create and manage resources using orchestrations v2. With orchestrations v2 you can take advantage of several key enhancements that allow you greater flexibility in referencing and managing resources.

 **Note:**

You shouldn't try to use or manage resources created using orchestrations v1 by referencing them in orchestrations v2, or vice versa.

There are some similarities and some key differences between orchestrations v1 and orchestrations v2.

Task	Orchestrations v1	Orchestrations v2
Creating an orchestration	Create your orchestration in a JSON file. Your orchestration can contain all the objects you want to create, or can reference nested orchestrations or objects created by other means. See <a href="#">Building Your First Orchestration v1</a> .	Create your orchestration in a JSON file. It is recommended that you create orchestrations that are entirely self-contained. Each orchestration should contain all the objects that you want to create, along with any objects referenced by those objects. The only external objects that an orchestration should reference are shared objects such as security lists that have been created earlier, or Oracle-provided resources such as images or shapes. See <a href="#">Building Your First Orchestration v2</a> .
Creating objects in an orchestration	See <a href="#">Object Types in an Orchestration</a> for a list of objects that you can create using an orchestration.	In addition to the objects that you can create using orchestrations v1, using orchestrations v2 you can also create storage snapshots and scheduled storage volume backups, restore storage volumes from scheduled backups, and add SSH keys. See <a href="#">Object Types in Orchestrations v2</a> for a list of objects that you can create using orchestrations v2.
Updating objects in a running orchestration	You can add or remove oplans when an orchestration is running. However, you must stop an orchestration if you want to update objects. See <a href="#">Updating an Orchestration v1</a> .	You can add or delete objects in an orchestration when the orchestration is running. You can also update objects to modify certain attributes. See <a href="#">Updating an Orchestration v2</a> .
Managing orchestrated objects individually	You can use master orchestrations to reference multiple individual orchestrations within a single orchestration. This enables you to synchronize starting and stopping multiple orchestrations. However, when required, you can manage each of the nested orchestrations separately. This way you can, for example, delete instances defined in one orchestration, while retaining storage volumes defined in another orchestration. See <a href="#">About Nested Orchestrations</a> .	You can use object persistence to specify objects that should not be deleted when the orchestration is suspended. For example, you can specify persistence for some instances and all storage volumes in an orchestration. Suspending the orchestration deletes nonpersistent objects, while persistent objects are preserved. To delete all objects, terminate the orchestration. See <a href="#">Object Persistence in Orchestrations v2</a> .

Task	Orchestrations v1	Orchestrations v2
Defining dependencies between objects	Relationships determine the sequence in which objects are created. You can define a one-on-one relationships either between different object plans, or between different instances. You can use relationships in a master orchestration to control the sequence in which a series of nested orchestrations is started. See <a href="#">Relationships Between Object Plans</a> .	You define associations between objects using object referencing. Unlike in orchestrations v1, in orchestrations v2 all the objects associated with a given object must be created in the same orchestration. This allows the orchestration to track the status of all referenced objects. You can use relationships to determine the sequence in which objects are created. However, relationships shouldn't be used to create dependencies, they should be used only to establish the sequence in which resources must be created. For dependencies, use references. See <a href="#">Object References and Relationships</a> .
Re-creating an object when it stops unexpectedly	When you specify the high availability policy for an instance as <code>active</code> , if the instance stops unexpectedly, it is re-created automatically. See <a href="#">About High-Availability Policies in an Orchestration</a> .	Failure recovery is implemented automatically for all objects. If any object fails unexpectedly, it is re-created automatically along with any objects that reference the failed object. See <a href="#">Recovering Failed Objects in Orchestrations v2</a> .
Uploading and starting an orchestration	You must upload your orchestration to Compute Classic and then start your orchestration to create the objects defined in the orchestration.	When an orchestration has the desired state specified as <code>active</code> in its top-level attributes, then the orchestration starts automatically when it is successfully uploaded to Compute Classic.
Stopping an orchestration	When you stop an orchestration, all objects created by that orchestration are deleted.	You can either suspend or terminate an orchestration. When you suspend an orchestration, persistent objects aren't deleted. When you terminate an orchestration, all objects are deleted.
Deleting an orchestration	An orchestration must be stopped before it can be deleted. Stopping an orchestration causes objects to be deleted, but deleting an orchestration has no impact on objects that are defined in the orchestration.	You can delete an orchestration when it is in the Ready, Suspended, Stopped or Error state. Any objects created by the orchestration are deleted when the orchestration is deleted.

## Object References and Relationships

### Object References

When you define an object in an orchestration, you can create dependencies with other objects by using *references*. With references, you can link an object to another using just the label of the target object. For example, you can reference the `name` of a storage volume from a storage attachment object using the format `{{volume_label:name}}`.

When recovering an object from a failure, Compute Classic recovers all the referenced objects automatically.

In the following example, the `StorageAttachment` object references the `name` attribute of an instance and the `name` attribute of a storage volume that's to be attached to the instance.

```
{
  "description": "a storage attachment object with references",
  "label": "attachment_object",
  "type": "StorageAttachment",
  "template": {
    "index": 1,
    "instance_name": "{{myInstance1:name}}",
    "storage_volume_name": "{{myVolume1:name}}"
  }
}
```

- `myInstance1` is the label of the instance object.
- `myVolume1` is the label of the storage volume object.

### Object Relationships

You can use the `relationships` attribute of an object to specify other related objects that must be created first.

Ensure that you don't create a relationship between a persistent and a nonpersistent object. A persistent object can be in a relationship only with another persistent object.

For example, if you define two instances – `instance1` and `instance2` – in an orchestration and you want `instance1` to be created first, then in the `relationships` attribute of `instance2`, specify that it depends on `instance1`.

```
"relationships": [
  {
    "type": "depends",
    "targets": ["instance1"]
  }
]
```

#### Note:

When recovering from a failure, orchestrations don't consider object relationships. So in the preceding example, if `instance2` fails, then the orchestration re-creates it, but it doesn't ensure first that `instance1` is available. To ensure that dependent objects are re-created, use object referencing.

For more complex scenarios, you can define multiple relationships.

For example, to ensure that `instance4` starts after `instance1`, `instance2`, and `instance3` are started, specify the following in the `relationships` attribute of `instance4`.

```
"relationships": [  
  {  
    "type": "depends",  
    "targets": ["instance1", "instance2", "instance3"]  
  }  
]
```

If all the related instances fail, then the orchestration will re-create them. But when re-creating `instance4`, the orchestration does *not* check whether the other instances exist.

## Object Persistence in Orchestrations v2

Orchestrations v2 enable you to provision an entire stack of cloud resources and manage them individually. Unlike orchestrations v1, you don't need to have separate orchestrations for different sets of objects such as storage, networking, or instances to ensure that they persist.

In some situations, you might want to stop certain objects while retaining others defined in the same orchestration. Using object persistence, you can ensure that when an orchestration is suspended, certain objects are not deleted.

To make an object persistent, set the `persistent` attribute to `true`. When an object is set to persist, it is not deleted when the orchestration is suspended. If the orchestration is terminated, then all the objects are deleted. For information about suspending and terminating orchestrations v2, see [Managing Orchestrations v2 Using the REST API](#).

If you set the `persistent` attribute of an object to `true`, then you must set the `persistent` attribute of all the dependent objects as well to `true`. For example, if a persistent instance references a bootable storage volume, the storage volume must also be persistent.

The following sample JSON illustrates a persistent storage volume.

```
{  
  "objects":  
  {  
    {  
      "type": "StorageVolume",  
      "description": "a persistent storage volume",  
      "label": "myVolume1",  
      "persistent": true,  
      "template": {  
        "name": "/Compute-acme/jack.jones@example.com/volume1",  
        "properties": [  
          "oracle/public/storage/default"  
        ],  
        "size": "2G"  
      }  
    }  
  }  
}
```

## Object Types in Orchestrations v2

You can define any of the following types of objects using orchestrations v2. The attributes for each object vary depending on the object type.

For the attributes used to define each of these object types, see [Orchestration v2 Attributes Specific to Each Object Type](#).

Type	Description
Acl	Creates an access control list (ACL) that can be applied to interfaces that are part of your IP networks.
Backup	Creates a back up of a storage volume using a specified backup configuration.
BackupConfiguration	Specifies the storage volume to back up, along with the backup schedule, retention count, and the name of the snapshot to be created.
Instance	Creates an instance.
IpAddressAssociation	Associates a public IP address reservation with an interface on an instance that is attached to an IP network.
IpAddressPrefixSet	Creates an IP address prefix set. This can be used as a source or destination in security rules that determine access to or from the virtual interfaces of instances that are attached to IP networks.
IpAddressReservation	Reserves a public IP address from a specified IP pool. This IP address can be associated with the virtual interface of an instance that is attached to an IP network.
IpNetwork	Creates an IP network. You can specify an IP network in the networking attributes while creating an instance.
IpNetworkExchange	Creates an IP network exchange. You can add IP networks to an IP network exchange either while creating the IP network, or later, by updating the IP network.
IPReservation	Reserves a public IP address that can be used in the shared network.
OSSContainer	Creates a container in the specified Oracle Cloud Infrastructure Object Storage Classic account.
Restore	Restores a storage volume from the specified backup.
Route	Creates a route to a specified destination using the specified vNICset.
SecApplication	Creates a security application that can be used in a security rule created for the shared network.
SecIPList	Creates a security IP list.
SecList	Creates a security list.
SecRule	Creates a security rule.
SecurityProtocol	Creates a security protocol that can be used in a security rule created for IP networks.
SecurityRule	Creates a security rule which can be added to an access control list (ACL). ACLs are used to control the flow of traffic across your IP networks.
SSHKey	Adds an SSH key.
StorageAttachment	Attaches a storage volume to an instance.

Type	Description
StorageSnapshot	Creates a snapshot of a storage volume.
StorageVolume	Creates a storage volume.
VirtualNicSet	Creates a vNICset, which contains one or more virtual network interfaces (vNICs). While creating an instance, you can specify the vNICsets that you want to add each vNIC to.

## Orchestration v2 Templates and Samples

The templates and samples provided here might not illustrate the use of all the attributes of each object. For a complete list of attributes and their description, see [Attributes in Orchestration v2](#). To get started with building an orchestration, see [Building Your First Orchestration v2](#).

### Note:

The orchestration templates provided here use placeholders for object names, labels, and other user-specific values. Replace these placeholders with values specific to your environment. For example, replace the identity domain `acme` with your identity domain and the user name `joe` or `joe.jonathan@example.com` with your user name.

### Orchestration v2 Template

The following is a JSON template for the structure of an orchestration, with the top-level attributes highlighted.

```
{
  "name": "/Compute-identity_domain/user/orchestration_name",
  "description": "OrchestrationDescription",
  "desired_state": "state",
  "tags": ["tag-1", "tag-2"],
  "objects": [
    {
      "type": "objectType",
      "description": "ObjectDescription",
      "label": "ObjectLabel",
      "persistent": true,
      "template": {
        attributes
      }
    },
    {
      "type": "objectType",
      "description": "ObjectDescription",
      "label": "ObjectLabel",
      "persistent": true,
      "template": {
        attributes
      }
    }
  ]
}
```



```
    }  
  },  
  .  
  . up to 100 objects  
  .  
] }  
}
```

### Orchestration v2 Template for Objects

The `objects` attribute is a top-level attribute in an orchestration. Within this attribute, you can specify one or more objects.

```
"objects": [  
  {  
    "type": "ObjectType",  
    "description": "ObjectDescriptionHere",  
    "label": "ObjectLabel",  
    "persistent": true,  
    "template": {  
      }  
  }  
]
```

### Orchestration v2 Samples for Each Object Type

- [Orchestration v2 Sample for Acl](#)
- [Orchestration v2 Sample for Backup](#)
- [Orchestration v2 Sample for BackupConfiguration](#)
- [Orchestration v2 Sample for Instance](#)
- [Orchestration v2 Sample for IpAddressAssociation](#)
- [Orchestration v2 Sample for IpAddressPrefixSet](#)
- [Orchestration v2 Sample for IpAddressReservation](#)
- [Orchestration v2 Sample for IpNetwork](#)
- [Orchestration v2 Sample for IpNetworkExchange](#)
- [Orchestration v2 Sample for IPReservation](#)
- [Orchestration v2 Sample for OSSContainer](#)
- [Orchestration v2 Sample for Restore](#)
- [Orchestration v2 Sample for Route](#)
- [Orchestration v2 Sample for SecApplication](#)
- [Orchestration v2 Sample for SecIPList](#)
- [Orchestration v2 Sample for SecList](#)
- [Orchestration v2 Sample for SecRule](#)
- [Orchestration v2 Sample for SecurityProtocol](#)
- [Orchestration v2 Sample for SecurityRule](#)

- [Orchestration v2 Sample for SSHKey](#)
- [Orchestration v2 Sample for StorageAttachment](#)
- [Orchestration v2 Sample for StorageSnapshot](#)
- [Orchestration v2 Sample for StorageVolume](#)
- [Orchestration v2 Sample for VirtualNicSet](#)

### Orchestration v2 Sample for Acl

Use this `type` of object to create an access control list that can be applied to vNICsets in IP networks. See [Configuring IP Networks](#).

```
"objects":
[
  {
    "label": "My-access-control-list",
    "type": "Acl",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/Acl-for-vnicset1"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for Backup

Use this `type` of object to create a backup of a storage volume using a specified backup configuration. See [Scheduling Backups of Storage Volumes and Restoring from Backups](#).

```
"objects":
[
  {
    "label": "Backup-from-backup-config-1",
    "type": "Backup",
    "template":
    {
      "backupConfigurationName": "/Compute-acme/joe.jonathan@example.com/
backupConfig-for-voll",
      "name": "/Compute-acme/joe.jonathan@example.com/backup-1"
    }
  },
  <Define other objects here.>
]
```

**Orchestration v2 Sample for BackupConfiguration**

Use this `type` of object to create a backup configuration. This configuration is used to schedule backups for a specified storage volume. See [Scheduling Backups of Storage Volumes and Restoring from Backups](#).

```
"objects":
[
  {
    "label": "Backup-schedule-for-voll",
    "type": "BackupConfiguration",
    "template":
    {
      "volumeUri": "http://api-z999.compute.us0.oraclecloud.com/storage/
volume/Compute-acme/joe.jonathan@example.com/voll",
      "name": "/Compute-acme/joe.jonathan@example.com/backupConfig-for-voll",
      "enabled": false,
      "backupRetentionCount": 2,
      "interval": {
        "Hourly": {"hourlyInterval": 1}
      }
    }
  },
  <Define other objects here.>
]
```

**Orchestration v2 Sample for Instance**

Use this `type` of object to create instances.

```
"objects":
[
  {
    "label": "MyInstance",
    "type": "Instance",
    "description": "My instance",
    "template":
    {
      "shape": "oc3",
      "boot_order": [1],
      "label": "vm-1",
      "networking": {
        "eth0": {
          "seclists": ["/Compute-acme/joe.jonathan@example.com/
wlsadmin_seclist"],
          "nat": "ipreservation:/Compute-acme/joe.jonathan@example.com/
ipres1"
        },
        "eth1": {
          "ipnetwork" : "/Compute-acme/joe.jonathan@example.com/ipnet-1",
          "ip": "192.168.4.2",
          "vnic": "/Compute-acme/joe.jonathan@example.com/eth1-ipnet1"
        }
      }
    }
  },
]
```

```

    "sshkeys": ["/Compute-acme/joe.jonathan@example.com/key1"],
    "storage_attachments": [
      {
        "index": 1,
        "volume": "/Compute-acme/joe.jonathan@example.com/boot"
      }
    ]
  }
},
<Define other objects here.>
]

```

### Orchestration v2 Sample for `IpAddressAssociation`

Use this `type` of object to create an IP association between an IP reservation and a vNIC in an IP network. See [Configuring IP Networks](#).

```

"objects":
[
  {
    "label": "IP-Association-for-vnic1-on-instancel",
    "type": "IpAddressAssociation",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/IP-association-vnic1",
      "ipAddressReservation": "/Compute-acme/joe.jonathan@example.com/IPres-
for-instancel-vnic1",
      "vnic": "/Compute-acme/joe.jonathan@example.com/instancel-vnic1"
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for `IpAddressPrefixSet`

Use this `type` of object to create an IP address prefix set to use in IP networks. See [Configuring IP Networks](#).

```

"objects":
[
  {
    "label": "IpAddress-prefix-set-1",
    "type": "IpAddressPrefixSet",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/ext_ip_addresses",
      "ipAddressPrefixes": ["203.0.113.0/30", "192.51.100.1/24"]
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for IpAddressReservation

Use this type of object to reserve an IP address to use in IP networks. See [Configuring IP Networks](#).

```
"objects":
[
  {
    "label": "IP-Reservation-for-instance-1-on-IP-network-1",
    "type": "IpAddressReservation",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/IPres-for-instancel-
vnic1",
      "ipAddressPool": "/oracle/public/public-ippool"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for IpNetwork

Use this type of object to create IP networks. See [About IP Networks](#).

```
"objects":
[
  {
    "label": "ipnet1",
    "type": "IpNetwork",
    "description": "My IP network with IP network exchange",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/ipnet1",
      "ipAddressPrefix": "192.168.3.0/24",
      "ipNetworkExchange": "/Compute-acme/joe.jonathan@example.com/
ipnetworkexchange1",
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for IpNetworkExchange

Use this type of object to create an IP network exchanges. IP network exchanges are used to connect IP networks. See [Configuring IP Networks](#).

```
"objects":
[
  {
    "label": "ipnetworkexchange",
    "type": "IpNetworkExchange",
    "description": "My IP network exchange",
    "template":
```

```
    {
      "name": "/Compute-acme/joe.jonathan@example.com/ipnetworkexchange1"
    }
  ],
  <Define other objects here.>
]
```

### Orchestration v2 Sample for IPReservation

Use this `type` of object to reserve permanent public IP addresses to use on the shared network. See [About Public IP Addresses](#).

```
"objects":
[
  {
    "label": "IP-Reservation",
    "type": "IPReservation",
    "description": "My IP address reservation",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/ipres1",
      "parentpool": "/oracle/public/ippool",
      "permanent": true
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for OSSContainer

Use this `type` of object to create a container in your associated Oracle Cloud Infrastructure Object Storage Classic account.

```
"objects":
[
  {
    "label": "My-OSS-Container",
    "type": "OSSContainer",
    "template":
    {
      "account": "/Compute-acme/cloud_storage",
      "container": "Container_1",
      "delete_remote": false
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for Restore

Use this `type` of object to restore a storage volume from a backup created using a backup configuration. See [Scheduling Backups of Storage Volumes and Restoring from Backups](#).

```
"objects":
[
  {
    "label": "Restored-vol-1",
    "type": "Restore",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/restored-vol-1",
      "backupName": "/Compute-acme/joe.jonathan@example.com/backup-1",
      "volumeUri": "http://api-z999.compute.us0.oraclecloud.com/storage/
volume/Compute-acme/joe.jonathan@example.com/restored-vol-1"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for Route

Use this `type` of object to create routes to direct traffic across your IP networks. See [Configuring IP Networks](#).

```
"objects":
[
  {
    "label": "MyRoute",
    "type": "Route",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/route1",
      "nextHopVnicSet": "/Compute-acme/joe.jonathan@example.com/vnicset1",
      "ipAddressPrefix": "203.0.113.0/24",
      "adminDistance": "0"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for SecApplication

Use this `type` of object to define security applications for use in security rules in the shared network. See [About Security Applications](#).

```
"objects":
[
  {
    "label": "MySecApplication",
    "type": "SecApplication",
```

```

"template":
{
  "name": "/Compute-acme/joe.jonathan@example.com/wlsadmin_ssl",
  "dport": 7002,
  "protocol": "tcp"
}
},
<Define other objects here.>
]

```

### Orchestration v2 Sample for `SecIPList`

Use this `type` of object to define a set of IP addresses that you want to use as a source in a security rule in the shared network. See [About Security IP Lists](#).

```

"objects":
[
{
  "label": "MySecurityIPList",
  "type": "SecIPList",
  "template":
{
  "name": "/Compute-acme/joe.jonathan@example.com/admin_ips",
  "secipentries": ["203.0.113.0/30"]
}
},
<Define other objects here.>
]

```

### Orchestration v2 Sample for `SecList`

Use this `type` of object to define security lists. See [About Security Lists](#).

```

"objects":
[
{
  "label": "MySecurityList",
  "type": "SecList",
  "template":
{
  "name": "/Compute-acme/joe.jonathan@example.com/sysadmin_seclist"
}
},
<Define other objects here.>
]

```

### Orchestration v2 Sample for `SecRule`

Use this `type` of object to define security rules that control access to your instances in the shared network. See [About Security Rules](#).

```

"objects":
[
{

```



```

    "label": "MySecurityRules",
    "type": "SecRule",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/
admin_ssh_to_sysadmin_rule",
      "application": "/oracle/public/ssh",
      "src_list": "seclist:/Compute-acme/joe.jonathan@example.com/
admin_ips",
      "dst_list": "seclist:/Compute-acme/joe.jonathan@example.com/
sysadmin_seclist",
      "action": "PERMIT"
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for SecurityProtocol

Use this `type` of object to create a security protocol that can be used in a security rule in IP networks. See [Configuring IP Networks](#).

```

"objects":
[
  {
    "label": "Security-protocol-for-IP-networks",
    "type": "SecurityProtocol",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/secprotocol_1",
      "description": "Security Protocol 1",
      "ipProtocol": "tcp",
      "srcPortSet": ["10", "55-100"],
      "dstPortSet": ["20", "155-1100"]
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for SecurityRule

Use this `type` of object to create a security rule that can be used to control traffic to vNICs in IP networks. See [Configuring IP Networks](#).

```

"objects":
[
  {
    "label": "IP-network-secrule-1",
    "type": "SecurityRule",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/ipnetSecrule1",
      "acl": "/Compute-acme/joe.jonathan@example.com/acl_1",
      "description": "Security Rule for ACL-1",

```

```

        "flowDirection": "egress",
        "secProtocols": ["/Compute-acme/joe.jonathan@example.com/
secprotocol_1"],
        "srcIpAddressPrefixSets": ["/Compute-acme/joe.jonathan@example.com/
ext_ip_address_list_1"]
    }
},
<Define other objects here.>
]

```

### Orchestration v2 Sample for SSHKey

Use this `type` of object to add an SSH public key to your account. See [Enabling Secure Access to Instances Using SSH](#).

```

"objects":
[
  {
    "label": "My-SSH-key",
    "type": "SSHKey",
    "template":
    {
      "key": "ssh-rsa
AAAAB3NzaClyc2EAAAADAQABAAQGDzU21CEj6JsqIMQAYwNbmZ5P2BVxA...",
      "name": "/Compute-acme/joe.jonathan@example.com/key1"
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for StorageAttachment

Use this `type` of object to attach a storage volume to an instance after the instance has been created.

```

"objects":
[
  {
    "label": "Attach-vol1-to-instance1",
    "type": "StorageAttachment",
    "template":
    {
      "index": 1,
      "storage_volume_name": "/Compute-acme/joe.jonathan@example.com/vol1",
      "instance_name": "/Compute-acme/joe.jonathan@example.com/instance1/
a6462ba5-5933-41a1-b853-fcfc421cb07/5fd18f4a-2ac2-4548-a0cf-57774c024742"
    }
  },
  <Define other objects here.>
]

```

### Orchestration v2 Sample for StorageSnapshot

Use this type of object to create a snapshot of a storage volume. See [Backing Up and Restoring Storage Volumes Using Snapshots](#).

```
"objects":
[
  {
    "label": "My-Storage-Snapshot",
    "type": "StorageSnapshot",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/voll-snapshot",
      "volume": "/Compute-acme/joe.jonathan@example.com/voll",
      "property": "/oracle/public/storage/snapshot/default"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for StorageVolume

Use this type of object to create storage volumes that you want to attach to your instances. See [About Storage Volumes](#).

```
"objects":
[
  {
    "label": "MyStorageVolume",
    "type": "StorageVolume",
    "template":
    {
      "name": "/Compute-acme/joe.jonathan@example.com/boot",
      "bootable": true,
      "imagelist": "/oracle/public/oel_6.6_20GB_x11_RD",
      "properties": ["/oracle/public/storage/default"],
      "size": "22548578304"
    }
  },
  <Define other objects here.>
]
```

### Orchestration v2 Sample for VirtualNicSet

Use this type of object to create vNICsets to use in IP networks. See [Configuring IP Networks](#).

```
"objects":
[
  {
    "label": "vNICset-1",
    "type": "VirtualNicSet",
    "template":
```

```

    {
      "name": "/Compute-acme/joe.jonathan@example.com/vnicset1",
      "vnics": ["/Compute-acme/joe.jonathan@example.com/vnic1",
                "/Compute-acme/joe.jonathan@example.com/vnic2"]
    }
  },
  <Define other objects here.>
]

```

## Workflow for Creating Instances Using Orchestrations v2

An **orchestration** defines the attributes and interdependencies of a collection of compute, networking, and storage resources in Compute Classic. You can use orchestrations to automate the provisioning and lifecycle operations of an entire virtual compute topology.

To use an orchestration to create and manage compute, networking, or storage resources:

1. Build your orchestration.  
An orchestration is defined in a JavaScript Object Notation (JSON) file that contains the attributes of the Compute Classic objects that you want to create. See [Building Your First Orchestration v2](#).
2. Upload the orchestration to Compute Classic. See [Uploading an Orchestration v2](#).
3. Start an orchestration. See [Starting an Orchestration v2](#).

### Note:

If the `desired_state` parameter is set to `active` in the orchestration JSON, the orchestration is activated automatically when you upload it.

4. While the orchestration is running, you can add, update, or delete an instance. See [Updating an Orchestration v2](#).
5. To delete objects that are not set to be persistent, suspend the orchestration.
6. To delete all objects defined in the orchestration, stop the orchestration. See [Terminating an Orchestration v2](#).

## Building Your First Orchestration v2

### Sample Orchestration v2 for Creating a Single Instance

You can define all your cloud resources in a single orchestration and manage the resources individually.

Use the following sample JSON as a starting point for building your first orchestration.

```

{
  "description": "Simple orchestration with an instance, storage volumes,
  ssh key, and a security list",
  "name": "/Compute-acme/joe.jonathan@example.com/simple_orchestration",

```

```
"desired_state": "active",
"objects": [
  {
    "type": "SSHKey",
    "label": "my_key",
    "template": {
      "enabled": false,
      "key": "ssh-rsa
AAAAB3NzaClyc2EAAAADAQABAAQgQDzU2lCEj6JsqIMQAYwNbmZ5P2BVxA...",
      "name": "/Compute-acme/jack.jones@example.com/key1"
    }
  },
  {
    "type": "SecList",
    "label": "my_seclist",
    "template": {
      "name": "/Compute-acme/joe.jonathan@example.com/my_instances"
    }
  },
  {
    "type": "StorageVolume",
    "label": "boot_volume",
    "description": "Boot disk for your instance",
    "persistent": true,
    "template": {
      "name": "/Compute-acme/jack.jones@example.com/BootVolume",
      "bootable": true,
      "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64",
      "properties": [
        "/oracle/public/storage/default"
      ],
      "size": "23G"
    }
  },
  {
    "type": "StorageVolume",
    "label": "data_volume",
    "description": "Data disk for your instance",
    "persistent": true,
    "template": {
      "name": "/Compute-acme/jack.jones@example.com/DataVolume",
      "properties": [
        "/oracle/public/storage/default"
      ],
      "size": "2G"
    }
  },
  {
    "type": "IPReservation",
    "label": "ip_reservation",
    "description": "IP reservation for your instance",
    "persistent": true,
    "template": {
      "parentpool": "/oracle/public/ippool",
      "permanent": true
    }
  }
]
```

```

    }
  },
  {
    "type": "Instance",
    "description": "demo instance",
    "label": "demo_instance",
    "template": {
      "label": "demo_instance",
      "shape": "oc3",
      "networking": {
        "eth0": {
          "seclists": [ "{{my_seclist:name}}" ],
          "nat": "ipreservation:{{ip_reservation:name}}"
        }
      },
      "storage_attachments": [
        {
          "index": 1,
          "volume": "{{boot_volume:name}}"
        },
        {
          "index": 2,
          "volume": "{{data_volume:name}}"
        }
      ],
      "boot_order": [1],
      "sshkeys": [
        "{{my_key:name}}"
      ]
    }
  }
],
"tags": ["sample"]
}

```

This sample orchestration does the following:

- Defines an instance with the label `demo_instance`, the `oc3` shape, and using the `/oracle/public/OL_6.7_UEKR4_x86_64` image.
- Defines and associates an SSH public key with the label `my_key` with the instance.
- Defines a security list with the label `my_seclist` and adds the instance to it.
- Defines and attaches the bootable storage volume with the label `boot_volume` to the instance.
- Defines and attaches the data storage volume with the label `data_volume` to the instance.
- Defines an IP reservation with the label `ip_reservation` and associates it with the instance.

 **Note:**

To learn about the structure of an orchestration, see [Orchestration v2 Templates and Samples](#). For information about all the attributes that you can define in an orchestration, see [Attributes in Orchestration v2](#).

**Steps for Building Your First Orchestration v2**

1. Copy the sample orchestration JSON to a plain text file, and open the file in any text editor.
2. Replace the name of the orchestration with an appropriate three-part name (`/Compute-identity_domain/user/object`).

 **Note:**

While editing the sample, remember to replace all the placeholder values with values specific to your environment. For example, replace the identity domain `acme` with your identity domain and the user name `joe.jonathan@example.com` with your user name.

3. Change the value of the `imagelist` attribute to any image that you want to use.
4. Under `template`, change the value of the `label` attribute to any label that you want.
5. If you want to attach the instance to more security lists, define each security list in the orchestration, and reference each security list in the instance. In the instance object, remember to enclose each security-list name in double quotation marks and separate the security-list names by using commas. See the following example:

```
"seclists": [
  "{{my_seclist:name}}",
  "{{my_devlist:name}}",
  "{{my_prodlis:name}}"
]
```

6. If you're creating a Linux instance enabled for SSH access, replace the `key` in the `my_key` object with your public key.

If you want to add more SSH keys, define each key as an object in the orchestration, and reference each key in double quotation marks and separate the keys by using commas. See the following example:

```
"sshkeys": [
  "{{my_key1:name}}",
  "{{my_key2:name}}",
  "{{my_key3:name}}"
]
```

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

7. Save the JSON file.

You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. If your JSON format isn't valid, then an error message is displayed when you upload the orchestration.

 **Note:**

Oracle doesn't support or endorse any third-party JSON-validation tool.

Your orchestration JSON file is ready now.

To create instances by using this orchestration, you must upload it to Compute Classic. See [Uploading an Orchestration v2](#).

## Attributes in Orchestrations v2

You specify attributes in orchestrations at several levels. At the top level, you specify certain attributes for the orchestration as a whole. Then, you specify attributes for each object defined in the orchestration. Finally, there are attributes that are specific to each type of object.

- **Top-level orchestration attributes**

The top-level orchestration attributes define the name and description of an orchestration, along with other information such as the desired state, the current status of the orchestration, and the tags associated with the orchestration. See [Top-Level Orchestration v2 Attributes](#).

- **General attributes for all object types**

These attributes specify the object type; a label, name, and description for the object, and so on. See [General Attributes for Objects in Orchestrations v2](#).

- **Attributes specific to each object type**

These are the attributes defined in the template for each object type. See [Orchestration v2 Attributes Specific to Each Object Type](#).



## Top-Level Orchestration v2 Attributes

The top-level orchestration attributes define the name and description of an orchestration, along with other information such as the desired state, the current status of the orchestration, and the tags associated with the orchestration.

The following JSON template shows the top-level orchestration attributes. A description of each attribute is provided in the table that follows the JSON template.

### Note:

Attributes for each object that you define in an orchestration vary depending on the object type. For information about object-specific attributes, see [Object Types in Orchestrations v2](#).

```
{
  "name": "/Compute-identity_domain/user/orchestration_name",
  "description": "OrchestrationDescriptionHere",
  "desired_state": "state",
  "tags": ["sometag", "sometag2"],
  "objects": [
    {
      ...
    },
    {
      ...
    },
    .
    . up to 100 objects
    .
  ]
}
```

Attribute	Required or Optional	Description
name	required	The three-part name of the orchestration ( <code>/Compute-identity_domain/user/orchestration_name</code> ).
desired_state	required	The desired state for the orchestration. You can specify one of the following desired states: <ul style="list-style-type: none"> <li>active: The orchestration starts immediately and all objects defined by the orchestration are created.</li> <li>inactive: The orchestration stops immediately and all objects defined by the orchestration are deleted.</li> <li>suspend: Nonpersistent objects are deleted and persistent objects are retained.</li> <li>delete: All objects are deleted and the orchestration itself is also deleted.</li> </ul>
description	optional	Text string describing the orchestration.

Attribute	Required or Optional	Description
objects	required	The list of objects in the orchestration. An object is the primary building block of an orchestration. An orchestration can contain up to 100 objects.
tags	optional	A list of the tags that you want to associate with your orchestration.

## General Attributes for Objects in Orchestrations v2

Any object that you define in an orchestration, regardless of the object type, has certain general attributes.

The following is a JSON template for an orchestration, with the general attributes for objects highlighted. The table that follows the template contains the descriptions for these attributes.

```
{
  "name": "/Compute-identity_domain/user/orchestration_name",
  "description": "OrchestrationDescriptionHere",
  "desired_state": "state",
  "tags": ["sometag", "sometag2"],
  "objects": [
    {
      "label": "someText",
      "type": "objectType",
      "desired_state": "inherit",
      "template": {
        attributes
      }
      "name": "objectName",
      "description": "ObjectDescriptionHere",
      "persistent": true,
      "relationships": [
        "type": "rel_type",
        "targets": ["object1", "object2", ...]
      ]
    },
    .
    . up to 100 objects
    .
  ]
}
```

Parameter	Required or Optional	Description
label	required	<p>A text string describing the object. A label can contain only alphanumeric characters, hyphens, and underscores. It can't contain unicode characters and spaces.</p> <p>In an orchestration, the label for each object must be unique.</p> <p>Maximum length: 256 characters.</p>
type	required	<p>The type of object that you want to create. Specify one of the following object types.</p> <ul style="list-style-type: none"> <li>• Acl</li> <li>• Backup</li> <li>• BackupConfiguration</li> <li>• Instance</li> <li>• IpAddressAssociation</li> <li>• IpAddressPrefixSet</li> <li>• IpAddressReservation</li> <li>• IpNetwork</li> <li>• IpNetworkExchange</li> <li>• IPReservation</li> <li>• OSSContainer</li> <li>• Restore</li> <li>• Route</li> <li>• SecApplication</li> <li>• SecIPList</li> <li>• SecList</li> <li>• SecRule</li> <li>• SecurityProtocol</li> <li>• SecurityRule</li> <li>• SSHKey</li> <li>• StorageAttachment</li> <li>• StorageSnapshot</li> <li>• StorageVolume</li> <li>• VirtualNicSet</li> </ul> <p>For a brief description of each object type, see <a href="#">Object Types in Orchestrations v2</a>.</p>
template	required	<p>The parameters specific to each object type.</p> <p>See <a href="#">Orchestration v2 Attributes Specific to Each Object Type</a>.</p>

---

Parameter	Required or Optional	Description
name	optional	<p>The four-part name of the object (<i>/Compute-identity_domain/user/orchestration/object</i>)</p> <p>If you don't specify a name for this object, the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, then another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
orchestration	optional	<p>The three-part name of the orchestration (<i>/Compute-identity_domain/user/orchestration_name</i>) to which the object belongs.</p>
desired_state	optional	<p>Specifies the desired state of an object. This allows you to manage the state of an object independently from the state of the orchestration. Specify one of the following:</p> <ul style="list-style-type: none"><li>• <code>inherit</code>: The default. The desired state of the object is the same as the desired state of the orchestration. Note that you can't specify states such as <code>active</code>, <code>inactive</code>, or <code>suspend</code> at the object level. These states must be inherited from the orchestration's desired state.</li><li>• <code>delete</code>: The object definition is removed from the orchestration JSON and the underlying object that was created by this orchestration is also deleted.</li></ul>
description	optional	<p>A text string describing the object.</p>
persistent	optional	<p>Specifies whether the object should persist when the orchestration is suspended. Specify one of the following:</p> <ul style="list-style-type: none"><li>• <code>true</code>: The object persists when the orchestration is suspended.</li><li>• <code>false</code>: The object is deleted when the orchestration is suspended.</li></ul> <p>By default, <code>persistent</code> is set to <code>false</code>. It is recommended that you specify <code>true</code> for storage volumes and other critical objects.</p> <p>Persistence applies only when you're suspending an orchestration. When you terminate an orchestration, all the objects defined in it are deleted.</p>

---

Parameter	Required or Optional	Description
relationships	optional	<p>The relationship between the objects that are created by this orchestration.</p> <p>The depends relationship indicates that the specified target objects must be created first. For example, if you define two instances – <code>instance1</code> and <code>instance2</code> – in an orchestration and you want <code>instance1</code> to be created first, then in the <code>relationships</code> attribute of <code>instance2</code>, specify that it depends on <code>instance1</code>.</p> <pre>"relationships": [   {     "type": "depends",     "targets": ["instance1"]   } ]</pre> <p>Note that when recovering from a failure, the orchestration doesn't consider object relationships. Orchestrations v2 use <i>object references</i> to recover interdependent objects to a healthy state. See <a href="#">Object References and Relationships</a>.</p>

## Orchestration v2 Attributes Specific to Each Object Type

You can specify various object types in an orchestration, including launch plans, networking objects such as security lists and security rules, storage volumes, and even other orchestrations. The attributes for each object vary depending on the object type.

The following sections describe the attributes for each object type that you can create using an orchestration.

- [Orchestration v2 Attributes for Acl](#)
- [Orchestration v2 Attributes for Backup](#)
- [Orchestration v2 Attributes for BackupConfiguration](#)
- [Orchestration v2 Attributes for Instance](#)
- [Orchestration v2 Attributes for IpAddressAssociation](#)
- [Orchestration v2 Attributes for IpAddressPrefixSet](#)
- [Orchestration v2 Attributes for IpAddressReservation](#)
- [Orchestration v2 Attributes for IpNetwork](#)
- [Orchestration v2 Attributes for IpNetworkExchange](#)
- [Orchestration v2 Attributes for IPReservation](#)
- [Orchestration Attributes for OSSContainer](#)
- [Orchestration v2 Attributes for Restore](#)
- [Orchestration v2 Attributes for Route](#)

- [Orchestration v2 Attributes for SecApplication](#)
- [Orchestration v2 Attributes for SecIPList](#)
- [Orchestration v2 Attributes for SecList](#)
- [Orchestration v2 Attributes for SecRule](#)
- [Orchestration v2 Attributes for SecurityProtocol](#)
- [Orchestration v2 Attributes for SecurityRule](#)
- [Orchestration v2 Attributes for SSHKey](#)
- [Orchestration v2 Attributes for StorageAttachment](#)
- [Orchestration v2 Attributes for StorageSnapshot](#)
- [Orchestration v2 Attributes for StorageVolume](#)
- [Orchestration v2 Attributes for VirtualNicSet](#)

## Orchestration v2 Attributes for `Acl`

The following sample JSON shows the key attributes of the `Acl` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/acl_1",
  "enabledFlag": true
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
enabledFlag	optional	Allows the ACL to be enabled or disabled. This parameter is set to <code>true</code> by default. Specify <code>false</code> to disable the ACL.
description	optional	Description of the ACL.
tags	optional	Strings that you can use to tag the ACL.

## Orchestration v2 Attributes for `Backup`

The following sample JSON shows the required attributes of the `Backup` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "backupConfigurationName": "{{My_Backup_Configuration:name}}",
  "name": "/Compute-acme/jack.jones@example.com/BACKUP-A"
}
```

Parameter	Required or Optional	Description
backupConfigurationName	required	A reference to the name of the BackupConfiguration object or the multi-part name of the backup configuration.
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
description	optional	Description of the backup storage volume.

## Orchestration v2 Attributes for BackupConfiguration

The following sample JSON shows the required attributes of the BackupConfiguration object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "volumeUri": "{{My_Storage_Volume:uri}}",
  "name": "/Compute-acme/jack.jones@example.com/backupConfigVoll1",
  "enabled": false,
  "backupRetentionCount": 2,
  "interval": {
    "Hourly": {
      "hourlyInterval": 1
    }
  }
}
```

Parameter	Required or Optional	Description
interval	required	The interval between back ups.  There are two kinds of intervals. Each Interval has its own JSON format. Your Interval field should look like one of the following: <ul style="list-style-type: none"> <li>"interval": {"Hourly": {"hourlyInterval": 2}}</li> <li>{"DailyWeekly": {"daysOfWeek": ["MONDAY"], "timeOfDay": "03:15", "userTimeZone": "America/Los_Angeles"}}</li> </ul>

Parameter	Required or Optional	Description
volumeUri	required	The URI of the storage volume that you want to back up, or a reference to the uri parameter of the StorageVolume object.
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
backupRetentionCount	required	The number of backups that should be retained. Minimum is 1.
enabled	optional	Set to true to enable backups. If not specified, the default is true.
description	optional	Description of the backup configuration.

## Orchestration v2 Attributes for Instance

### Topics

- [Instance Attributes](#)
- [Networking Attributes for Instances](#)
  - [Subparameters for a Network Interface on the Shared Network](#)
  - [Subparameters for a Network Interface on an IP Network](#)

### Instance Attributes

Instances have a number of required and optional attributes. The following sample JSON shows some of the key instance attributes. A description of each of the required and optional instance attributes is provided in the table below.

```
{
  "instances":
  [
    {
      "shape": "oc3",
      "boot_order": [1],
      "label": "vm-1",
      "networking": {
        "eth0": {
          "seclists": ["/Compute-acme/joe/wlsadmin_seclists"],
          "nat": "ipreservation:/Compute-acme/joe/ipres1"
        },
        "eth1": {
          "ipnetwork": "/Compute-acme/joe/ipnet1",
          "ip": "192.168.4.2",
          "vnic": "/Compute-acme/joe/eth1-ipnet1"
        }
      }
    }
  ]
}
```



```

    }
  },
  "sshkeys": ["/Compute-acme/joe/key1"],
  "relationships": [
    {
      "type": "different_node",
      "instances": ["instance:/Compute-acme/jack.jones@example.com/
instance1"]
    }
  ]
  "storage_attachments":
  [
    {
      "index": 1,
      "volume": "/Compute-acme/joe/boot"
    }
  ]
}
]
}
}

```


Parameter	Required or Optional	Description
shape	required	The name of the shape that defines the number of OCPUs and the RAM that you require for the instance. For general purpose and high-memory shapes, you can select the block storage disk size, but for high I/O shapes, the size of the SSD storage is determined by the shape.
name	optional	<p>The three-part name of the instance (<i>/Compute-identity_domain/user/name</i>).</p> <p>If you specify this parameter, then the full name of the instance would be in the format, <i>/Compute-identity_domain/user/name_you_specify/id</i>.</p> <p>If you don't specify this parameter, then the full name would be in the format, <i>/Compute-identity_domain/user/id</i>.</p> <p>In either case, <i>id</i> is an autogenerated ID.</p> <p><b>Examples of Instance Names:</b></p> <ul style="list-style-type: none"> <li>When you specify <i>/Compute-acme/jack/vm1</i> as the value of the name parameter:           <pre><i>/Compute-acme/jack/vm1/300a7479-ec90-4826-98b9-a725662628f1</i></pre> </li> <li>When you don't specify the name parameter:           <pre><i>/Compute-acme/jack/38ef677e-9e13-41a7-a40c-2d99afce1714</i></pre> </li> </ul> <p>Although this is an optional parameter, specifying a meaningful name makes it easier for you to identify your instances.</p>
label	optional	<p>A text string to identify the instance.</p> <p>This label is used when defining relationships in an orchestration.</p>
tags	optional	<p>A JSON array or list of strings used to tag the instance.</p> <p>By assigning a human-friendly tag to an instance, you can identify the instance easily when you perform an instance listing. These tags aren't available from within the instance.</p>

Parameter	Required or Optional	Description
desired_state	optional	The only allowed values are <code>running</code> or <code>shutdown</code> . If you specify <code>running</code> , the instance is started. If you specify <code>shutdown</code> , the instance is stopped. You can start the instance again later by updating the instance with the <code>desired_state</code> specified as <code>running</code> .
attributes	optional	<p>A JSON object or dictionary of user-defined attributes to be made available to the instance.</p> <p>If you're creating a Windows instance, you must specify the following required attributes:</p> <pre>{   "enable_rdp": true,   "administrator_password":     "Specify_password_here" }</pre> <p>For more information about specifying user-defined attributes that can be used to automate instance configuration, see <a href="#">Automating Instance Initialization Using <code>opc-init</code></a>.</p> <p><b>Note:</b></p> <p>Solaris machine images don't include the <code>opc-init</code> scripts. So you can't use <code>opc-init</code> to automate instance configuration of Solaris instances.</p> <p>The attributes that you specify can be accessed from within the instance at <code>http://192.0.0.192/latest/attributes</code>. For more information about retrieving user-defined attributes, see <a href="#">Retrieving User-Defined Instance Attributes</a>.</p>
imagelist	optional	<p>The three-part name (<code>oracle/public/imagelist_name</code>) of the image list containing the image to be used (example: <code>/oracle/public/OL_6.7_UK4_x86_64</code>).</p> <p>You <i>must</i> use this attribute if you don't specify a bootable storage volume by using the <code>boot_order</code> attribute. If you specify the <code>imagelist</code> attribute as well as the <code>boot_order</code> attribute, then the <code>imagelist</code> attribute is ignored.</p>

---

Parameter	Required or Optional	Description
storage_attachments	optional	<p>If you specify the <code>storage_attachments</code> parameter, then specify the following subparameters for each attachment:</p> <ul style="list-style-type: none"><li><code>volume</code>: The three-part name (<code>/Compute-identity_domain/user/object_name</code>) of the storage volume that you want to attach to the instance. Note that volumes attached to an instance at launch time can't be detached.</li><li><code>index</code>: The index number for the volume. The allowed range is 1 to 10. If you want to use a storage volume as the boot disk for an instance, you must specify the index number for that volume as 1. The index determines the device name by which the volume is exposed to the instance. Index 0 is allocated to a nonpersistent boot disk, <code>/dev/xvda</code>. An attachment with index 1 is exposed to the instance as <code>/dev/xvdb</code>, an attachment with index 2 is exposed as <code>/dev/xvdc</code>, and so on.</li></ul>
boot_order	optional	<p>The index number of the bootable storage volume that should be used to boot the instance. The only valid value is 1.</p> <p>If you set this attribute, you must also specify a bootable storage volume with index number 1 in the <code>volume</code> sub-parameter of <code>storage_attachments</code>.</p> <p>When you specify <code>boot_order</code>, you don't need to specify the <code>imagelist</code> attribute, because the instance is booted using the image on the specified bootable storage volume. If you specify both <code>boot_order</code> and <code>imagelist</code>, the <code>imagelist</code> attribute is ignored.</p>

---

Parameter	Required or Optional	Description
hostname	optional	<p>The host name assigned to the instance. On an Oracle Linux instance, this host name is displayed in response to the <code>hostname</code> command.</p> <p>Only relative DNS is supported. The domain name is suffixed to the host name that you specify. The host name must not end with a period. If you don't specify a host name, then a name is generated automatically. The DNS name of an instance depends on its host name, as follows:</p> <ul style="list-style-type: none"> <li>• If no DNS name is specified in the <code>networking</code> attribute, then the DNS name is set to the host name, and a reverse DNS record (PTR) is created for the host name.</li> <li>• If the DNS name specified in the <code>networking</code> attribute matches the host name, then that record also creates a reverse DNS record for the host name.</li> <li>• If the <code>dns</code> attribute under <code>networking</code> is set to an empty list (<code>[]</code>), then no DNS records are created even if a host name is specified. The instance still receives its host name through DHCP, and can perform a reverse lookup of its host name. However, no other instance can perform this reverse lookup.</li> </ul>
		<div style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> <b>Note:</b></p> <p>If an instance has network interfaces defined only for IP networks and doesn't have any interface on the shared network, then when <code>hostname</code> is specified, no DNS entries are set. In this case, DNS entries are set by the <code>dns</code> subparameter of the <code>networking</code> attribute.</p> </div>
reverse_dns	optional	<p>If set to <code>true</code> (default), then reverse DNS records are created.</p> <p>If set to <code>false</code>, no reverse DNS records are created.</p>

---

Parameter	Required or Optional	Description
networking (attributes for the shared network)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>Only one interface on an instance can be added to the shared network. To add an interface to the shared network, you can specify the following subparameters:</p> <ul style="list-style-type: none"><li>• <code>seclists</code>: (Optional) The security lists that you want to add the instance to.</li><li>• <code>nat</code>: (Optional) Indicates whether a temporary or permanent public IP address should be assigned to the instance.</li><li>• <code>dns</code>: (Optional) A list of the DNS A record names for the instance. This name is relative to the internal DNS domain.</li><li>• <code>model</code>: (Optional) The type of network interface card (NIC). The only allowed value is <code>e1000</code>.</li><li>• <code>name_servers</code>: (Optional) The name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li><li>• <code>search_domains</code>: (Optional) The search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li></ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on the Shared Network</a>.</p>

---

Parameter	Required or Optional	Description
networking (attributes for IP networks)	optional	<p><code>ethn</code>: The interface that you're defining. Oracle-provided images with release version 16.3.6 and later support eight vNICs. You can also create private images that support multiple vNICs. If the image you've specified supports eight vNICs, then you can specify up to eight network interfaces, from <code>eth0</code> to <code>eth7</code>.</p> <p><b>Note:</b></p> <p>For each interface, you can specify parameters for either the shared network, or for an IP network. You can't specify parameters for both networks for the same <code>ethn</code> interface.</p> <p>To add an interface to an IP network, specify the following subparameters:</p> <ul style="list-style-type: none"> <li><code>ipnetwork</code>: The name of the IP network that you want to add the instance to.</li> <li><code>ip</code>: (Optional) If you want to associate a static private IP address with the instance, specify an available IP address from the IP address range of the specified <code>ipnetwork</code>.</li> <li><code>address</code>: (Optional) The MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter <code>01:02:03:04:ab:cd</code> as the MAC address but not <code>01-02-03-04-ab-cd</code>.</li> <li><code>nat</code>: (Optional) A list of IP reservation that you want to associate with this interface, in the format: <code>"nat": ["network/v1/ipreservation:IP_reservation_name"]</code>. Here <code>IP_reservation_name</code> is the three-part name of the IP reservation in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnic</code>: (Optional) The three-part name of the vNIC in the <code>/Compute-identity_domain/user/object_name</code> format.</li> <li><code>vnicsets</code>: (Optional) A list of the three-part names of the vNICsets that you want to add this interface to.</li> <li><code>is_default_gateway</code>: (Optional) If you want to specify the interface to be used as the default gateway for all traffic, set this to <code>true</code>. The default is <code>false</code>. If the instance has an interface on the shared network, that interface is always used as the default gateway.</li> <li><code>dns</code>: (Optional) A list of the DNS A record names for the instance.</li> <li><code>name_servers</code>: (Optional) A list of the name servers that should be sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface.</li> <li><code>search_domains</code>: (Optional) A list of the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface.</li> </ul> <p>For more information about each of these subparameters, see <a href="#">Subparameters for a Network Interface on an IP Network</a>.</p>

---

Parameter	Required or Optional	Description
relationships	optional	<p>You can also define relationships to indicate that you want the specified instances to be created on the same or different physical server.</p> <ul style="list-style-type: none"><li>• <b>Relationship: same_node</b> The <code>same_node</code> relationship indicates that you want the specified instances to be created on the same physical server. This is useful if you want to ensure low latency across instances. Example: to ensure that <code>instance1</code> is created on the same physical server. <pre>"relationships": [   {     "type": "same_node",     "instances": [ "instance:/Compute-acme/ jack.jones@example.com/instance1" ]   } ]</pre></li><li>• <b>Relationship: different_node</b> The <code>different_node</code> relationship indicates that you do not want the specified instances to be created on the same physical server. This is useful if you want to isolate instances for security or redundancy. Example: to ensure that <code>instance1</code> is not created on the same physical server. <pre>"relationships": [   {     "type": "different_node",     "instances": [ "instance:/Compute-acme/ jack.jones@example.com/instance1" ]   } ]</pre></li></ul>

---

Parameter	Required or Optional	Description
sshkeys	optional	<p>A list of the SSH public keys that you want to associate with the instance.</p> <p><b>Note:</b></p> <p>You don't need to provide any SSH public keys if you're creating a Windows instance, because you can't access a Windows instance using SSH. To access a Windows instance, see <a href="#">Accessing a Windows Instance Using RDP</a>.</p> <p>For each key, specify the three-part name in the <code>/Compute-identity_domain/user/object_name</code> format.</p> <p>You can associate the same key with multiple instances.</p> <p>The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at <code>http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key</code>.</p> <ul style="list-style-type: none"> <li>• Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the <code>authorized_keys</code> file of the <code>opc</code> user.</li> <li>• In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the <code>authorized_keys</code> file of the appropriate users.</li> </ul>

### Networking Attributes for Instances

There are several subparameters that you can specify under the `ethn` parameter in the networking section of instance attributes. The list of subparameters varies depending on whether you're defining a network interface on a shared network or an IP network.

Only one interface can be added to the shared network. If no subparameters are specified for the `ethn` parameter, the interface is implicitly added to the default security list in the shared network. You can't explicitly or implicitly define two interfaces to be added to the shared network.

### Subparameters for a Network Interface on the Shared Network

- `seclists`: (Optional) The security lists that you want to add the instance to.
 

For each security list, specify the three-part name in the `/Compute-identity_domain/user/object_name` format. You can attach an instance to a maximum of five security lists. If you launch an instance without specifying any security list, the instance is assigned to the `/Compute-identity_domain/default/default` security list.
- `nat`: (Optional) Indicates whether a temporary or permanent public IP address should be assigned to the instance.
  - To associate a temporary IP address with the instance for use during the lifetime of the instance, specify `ippool:/oracle/public/ippool`.
  - To associate a persistent IP address, specify `ipreservation:ipreservation_name`, where `ipreservation_name` is the three-part name of an existing IP reservation in the `/Compute-identity_domain/user/object_name` format.



If `nat` is not specified, then no public IP address is associated with your instance when it is created. If required, you can associate an IP address with the instance after the instance has been created.

- `dns`: (Optional) A list of the DNS A record names for the instance. The name is relative to the internal DNS domain.
- `model`: (Optional) The type of network interface card (NIC). The only allowed value is `e1000`.
- `name_servers`: (Optional) Enter the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface. If `name_servers` are set in both the IP network settings as well as the shared network settings, the name servers in the shared network will be used. To ensure that the name servers specified in the IP network are used, specify the same values for name servers on each interface.
- `search_domains`: (Optional) Enter the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface. If `search_domains` are set in both the IP network settings as well as the shared network settings, the search domains in the shared network will be used. To ensure that the search domains specified in the IP network are used, specify the same values for search domains on each interface.

### Subparameters for a Network Interface on an IP Network

- `ipnetwork`: The name of the IP network that you want to add the instance to.

If no name is specified, the interface isn't added to any IP network. Instead, it is implicitly added to the shared network. However, only one instance interface can be added to the shared network. If another interface is either implicitly or explicitly added to the shared network, the instance won't be created and will display an error.

Specify the three-part name of the IP network, in the `/Compute-identity_domain/user/object_name` format.

If an IP network belongs to an IP network exchange and if you have specified a host name, then that host name is resolvable by all IP networks connected to the IP network exchange.

- `ip`: (Optional) The static private IP address of the instance. This is a persistent private IP address, which is reserved for use with this instance. The private IP address must be unused and it must belong to the subnet of the selected `IP network`. Remember, too, that certain IP addresses in a subnet are reserved. For example, the first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.

If you don't specify an IP address, an IP address is assigned dynamically from the available IP addresses of the specified `ipnetwork`. However in this case, if you delete and re-create the instance, its IP address might change.

#### Note:

Dynamically allocated IP addresses are assigned from the top of the subnet range. It is recommended that you specify static IP addresses starting from the end of the subnet range to avoid conflicts.

- **address:** (Optional) The MAC address of the interface, in hexadecimal format, where each digit is separated by colon. For example, you can enter 01:02:03:04:ab:cd as the MAC address but not 01-02-03-04-ab-cd. Ensure that the MAC addresses that you specify are unique within each IP network exchange and each IP network. If you specify a duplicate MAC address, each vNIC with that MAC address is disabled.
- **nat:** (Optional) A list of IP reservations that you want to associate with this interface. Specify `network/v1/ipreservation:ipreservation_name`, where `ipreservation_name` is the three-part name of an existing IP reservation in the `/Compute-identity_domain/user/object_name` format.

When you create an IP reservation, you specify the IP pool from which you want to reserve the IP address. You can associate a maximum of two IP reservations with each vNIC, one from each IP pool.

### Example:

```
"networking":
{
    "eth0": {
        "ipnetwork": "/test-customer/ipnet-1",
        "ip": "192.168.2.14",
        "nat": ["network/v1/ipreservation:/Compute-acme/joe/public-
ipres-1"]
    }
}
```

- **vnic:** (Optional) The three-part name of the vNIC in the `/Compute-identity_domain/user/object_name` format.

If you don't specify a name for this object, then the name is generated automatically.

When the vNIC name is generated automatically, the autogenerated instance id is included as part of the `object_name`. So if you delete and re-create an instance, the vNIC name will change. However, if you specify a vNIC name, the name won't change if you delete and re-create the instance.

Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.

- **vnicsets:** (Optional) A list of the three-part names of the vNICsets that you want to add this vnic to. Specifying vNICsets ensures that this vNIC is added to the required vNICsets whenever the instance is created and removed from the vNICset whenever the instance is deleted.

While creating an instance, you can add a vNIC to up to 4 vNICsets. To add a vNIC to more than 4 vNICsets, update the required vNICsets after the instance is created.

The vNICsets that you specify here must already exist when you create or re-create an instance.

If no vNICset is specified, then the vNIC is added to the default vNICset, `/Compute-identity_domain/default`.

If an empty list (`"vnicsets": []`) is specified, this vNIC isn't added to any vNICset, including the default vNICset.

- **is\_default\_gateway:** (Optional) If you want to specify the interface to be used as the default gateway for all traffic, set this to `true`. The default is `false`. Only one

interface on an instance can be specified as the default gateway. If the instance has an interface on the shared network, that interface is always used as the default gateway. You can specify an interface on an IP network as the default gateway only when the instance doesn't have an interface on the shared network.

- `dns`: (Optional) A list of the DNS A record names for the instance.

Each IP network has its own DNS server listening on the first IP address of the subnet. You can specify up to eight DNS A record names for each instance on an IP network. These names can be queried by instances on any IP network in the same IP network exchange.

If no static IP address is specified for the instance on the IP network, an IP address on the specified IP network is assigned automatically. After the instance is launched, the defined names are associated with the IP address that was automatically allocated to the instance.

The same DNS A record name can be specified for multiple instances.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
        "dns": [ "dns1.example.com", "dns2.bar.com" ]
    }
}
```

- `name_servers`: (Optional) A list of the name servers that are sent through DHCP as option 6. You can specify a maximum of eight name server IP addresses per interface. If `name_servers` are set in both the IP network as well as the shared network, the name servers in the shared network will be used. To ensure that the name servers specified in the IP network are used, specify the same values for name servers on each interface.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
        "dns": ["dns1.example.com", "dns2.bar.com"],
        "name_servers": ["192.168.12.1", "192.168.12.2"]
    }
}
```

In this example, the name servers 192.168.12.1 and 192.168.12.2 will be pushed to the instance through DHCP.

- `search_domains`: (Optional) A list of the search domains that should be sent through DHCP as option 119. You can enter a maximum of eight search domain zones per interface. If `search_domains` are set in both the IP network as well as the shared network, the search domains in the shared network will be used. To ensure that the search domains specified in the IP network are used, specify the same values for search domains on each interface.

**Example:**

```
"networking":
{
    "eth1": {
        "ipnetwork": "/Compute-acme/joe/ipnet1",
```

```

        "dns": ["dns1.example.com", "dns2.bar.com"],
        "name_servers": ["192.168.12.1", "192.168.12.2"],
        "search_domains": ["example.com", "us.example1.com"]
    }
}

```

In this example, the search domain zones `example.com` and `us.example1.com` will be pushed to the instance through DHCP.

## Orchestration v2 Attributes for `IpAddressAssociation`

The following sample JSON shows the key attributes of the `IpAddressAssociation` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```

{
    "name": "/Compute-acme/joe/IP-association-vnic1",
    "ipAddressReservation": "/Compute-acme/joe/IPres-for-instance1-
vnic1",
    "vnic": "/Compute-acme/joe/instance1-vnic1"
}

```

Parameter	Required or Optional	Description
<code>name</code>	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
<code>ipAddressReservation</code>	optional	The name of the IP reservation that you want to associate with an instance.
<code>vnic</code>	optional	The name of the vNIC that you want to associate the IP reservation with.
<code>description</code>	optional	Description of the IP association.
<code>tags</code>	optional	Strings that you can use to tag the IP association.

## Orchestration v2 Attributes for `IpAddressPrefixSet`

The following sample JSON shows the key attributes of the `IpAddressPrefixSet` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```

{
    "name": "/Compute-acme/joe/ext_ip_address_list_1",
    "ipAddressPrefixes": ["203.0.113.0/30", "192.51.100.1/24"]
}

```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
ipAddressP refixes	optional	Set of IPv4 addresses in CIDR address prefix format.
descriptio n	optional	Description of the IP address prefix set.
tags	optional	Strings that you can use to tag the IP address prefix set.

## Orchestration v2 Attributes for `IpAddressReservation`

The following sample JSON shows the key attributes of the `IpAddressReservation` object type for IP networks. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/IPres-for-instancel-vnic1",
  "ipAddressPool": "/oracle/public/public-ippool"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.


Parameter	Required or Optional	Description
ipAddressPool	required	The IP address pool from which you want to reserve an IP address. Enter one of the following: <ul style="list-style-type: none"> <li><code>/oracle/public/public-ippool</code>: When you attach an IP address from this pool to an instance, you enable access between the public Internet and the instance.</li> <li><code>/oracle/public/cloud-ippool</code>: When you attach an IP address from this pool to an instance, the instance can communicate privately (that is, without traffic going over the public Internet) with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region.</li> </ul>
description	optional	Description of the IP reservation.
tags	optional	Strings that you can use to tag the IP reservation.

## Orchestration v2 Attributes for `IpNetwork`

The following sample JSON shows the attributes of the `IpNetwork` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipnet1",
  "ipAddressPrefix": "192.168.3.0/24",
  "ipNetworkExchange": "/Compute-acme/joe/ipnetworkexchange1"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.

Parameter	Required or Optional	Description
ipAddressPrefix	required	<p>The set of IP addresses allocated to your IP network, specified in the CIDR format. When you create instances, you can associate a vNIC on the instance with an IP network. That vNIC on the instance is then allocated an IP address from the specified IP network.</p> <p>Select the IP address prefix for your IP networks carefully. Consider the number of instances that you might want to add to the network. This will help determine the size of the subnet required.</p> <p>If you create multiple IP networks and you might want to add these IP networks to the same IP network exchange, then ensure that you don't allocate overlapping address ranges to these IP networks.</p> <p>Similarly, if you plan to connect to your IP networks using VPN, then ensure that the addresses you specify for your IP networks don't overlap with each other, or with the IP addresses used in your on-premises network.</p>
ipNetworkExchange	optional	<p>The IP network exchange that you want to add this IP network to. An IP network can belong to only one IP network exchange. Before you specify an IP network exchange for an IP network, ensure that the IP addresses in this IP network don't overlap the IP addresses in any other network in the same IP network exchange.</p>
<div style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> <b>Note:</b></p> <p>You should ensure that the IP network exchange you reference currently exists. If the IP network exchange hasn't been created or has been deleted, then when you add an instance interface to this IP network while creating the instance, the instance will go into an error state and won't be created.</p> </div>		
<p>If you want to connect IP networks by using an IP network exchange, it is recommended that you do this before creating instances with an interface on those IP networks. This ensures that routes are appropriately configured on instances by the DHCP client during instance initialization.</p>		
description	optional	Description of the IP network.
tags	optional	Strings that you can use to tag the IP network.

## Orchestration v2 Attributes for `IpNetworkExchange`

The following sample JSON shows the required attribute of the `IpNetworkExchange` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipnetworkexchange1"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
description	optional	Description of the IP network exchange.
tags	optional	Strings that you can use to tag the IP network exchange.

## Orchestration v2 Attributes for `IPReservation`

The following sample JSON shows the key attributes of the `IPReservation` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/ipres1",
  "parentpool": "/oracle/public/ippool",
  "permanent": true
}
```

Parameter	Required or Optional	Description
parentpool	required	Specify <code>/oracle/public/ippool</code>
permanent	required	Set to True
account	optional	Specify <code>/Compute-identity_domain/default</code>



Parameter	Required or Optional	Description
name	optional	<p>The three-part name of the object (<i>/Compute-identity_domain/user/object</i>).</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>

## Orchestration Attributes for `OSSContainer`

The following sample JSON shows the key attributes of the `integrations/osscontainer` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "account": "/Compute-acme/cloud_storage",
  "container": "Container_1",
  "delete_remote": false
}
```

Parameter	Required or Optional	Description
account	required	The two-part name of the account ( <i>/Compute-identity_domain/cloud_storage</i> ) that contains the credentials and access details of the associated Oracle Cloud Infrastructure Object Storage Classic instance.
container	required	<p>The name of the container that you want to create. Container names must:</p> <ul style="list-style-type: none"> <li>Contain only UTF-8 characters</li> <li>Be a maximum of 256 bytes</li> <li>Avoid using a slash (/) character because this character acts as a delimiter between the container name and the object name</li> </ul> <p>Ensure that a container of the same name doesn't already exist.</p>
delete_remote	required	<p>When set to <code>true</code>, deletes the Oracle Cloud Infrastructure Object Storage Classic container along with all the objects in the container when you delete the <code>integration/osscontainer</code> object created by this orchestration.</p> <p>When set to <code>false</code>, only the <code>integrations/osscontainer</code> object created by this orchestration is deleted. The container in Oracle Cloud Infrastructure Object Storage Classic remains intact, along with all objects in the container.</p>

Parameter	Required or Optional	Description
name	optional	<p>The three-part name of the <code>integrations/osscontainer</code> object created by this orchestration. This name is in the format <code>/Compute-identity_domain/user/object</code>.</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>

## Orchestration v2 Attributes for `Restore`

The following sample JSON shows the required attributes of the `Restore` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/jack.jones@example.com/RESTORE-A",
  "backupName": "{My Backup:name}",
  "volumeUri": "http://api-z999.compute.us0.oraclecloud.com/storage/volume/
Compute-acme/jack.jones@example.com/restored-example-volume",
  "description": null
}
```

Parameter	Required or Optional	Description
backupName	required	The multi-part name of the backup that you want to restore. The backup must be in the completed state.
volumeUri	required	The URI of the storage volume that should be created when the backup is restored. Ensure that another volume with the same URI does not exist.
name	optional	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
description	optional	Description of the restored storage volume.

## Orchestration v2 Attributes for `Route`

The following sample JSON shows the required attributes of the `Route` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/route1",
  "nextHopVnicSet": "/Compute-acme/joe/vnicset1",
  "ipAddressPrefix": "192.168.0.0/16"
}
```

Parameter	Required or Optional	Description
<code>name</code>	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
<code>ipAddressPrefix</code>	required	The IP address prefix, in CIDR format, of the destination network that you want to specify the route to.
<code>nextHopVnicSet</code>	required	The vNICset that you want to use to route packets to the destination network. When a vNICset containing multiple vNICs is used in a route, Equal Cost Multipath (ECMP) anycast routing is implemented. Traffic routed by that route is load balanced across all the vNICs in the vNICset. Using vNICsets with multiple vNICs also ensures high availability for traffic across the specified vNICs.
<code>administrativeDistance</code>	optional	The route's administrative distance. Specify 0 (the default), 1, or 2.  The administrative distance indicates the priority of a route. The highest priority is 0. The route with the highest priority is used. If multiple routes have the highest priority, all those routes are used.
<code>description</code>	optional	Description of the route.
<code>tags</code>	optional	Strings that you can use to tag the route.

## Orchestration v2 Attributes for `SecApplication`

The following sample JSON shows the key attributes of the `SecApplication` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/wlsadmin_ssl",
  "dport": 7002,
}
```

```

    "protocol": "tcp"
  }

```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<i>/Compute-identity_domain/user/object</i>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
protocol	required	<p>The protocol to use.</p> <p>The value that you specify can be either a text representation of a protocol or any unsigned 8-bit assigned protocol number in the range 0–254. See <i>Assigned Internet Protocol Numbers</i> (<a href="http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml">http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml</a>).</p> <p>For example, you can specify either <code>tcp</code> or the number 6.</p> <p>The following text representations are allowed: <code>tcp</code>, <code>udp</code>, <code>icmp</code>, <code>igmp</code>, <code>ipip</code>, <code>rdp</code>, <code>esp</code>, <code>ah</code>, <code>gre</code>, <code>icmpv6</code>, <code>ospf</code>, <code>pim</code>, <code>sctp</code>, <code>mplsip</code>, <code>all</code>.</p> <p>To specify all protocols, set this to <code>all</code>.</p>
dport	optional	<p>The TCP or UDP destination port number.</p> <p>You can also specify a port range, such as 5900-5999 for TCP.</p> <p>If you specify <code>tcp</code> or <code>udp</code> as the protocol, then the <code>dport</code> parameter is required; otherwise, it is optional.</p> <p>This parameter isn't used by the ICMP protocol or the GRE protocol.</p> <p><b>Note:</b> This request fails if the range-end is lower than the range-start. For example, if you specify the port range as 5000–4000.</p>
icmptype	optional	<p>The ICMP type.</p> <p>This parameter is relevant only if you specify <code>icmp</code> as the protocol. You can specify one of the following values:</p> <ul style="list-style-type: none"> <li>echo</li> <li>reply</li> <li>ttl</li> <li>traceroute</li> <li>unreachable</li> </ul> <p>If you specify <code>icmp</code> as the protocol and don't specify <code>icmptype</code> or <code>icmpcode</code>, then all ICMP packets are matched.</p>

Parameter	Required or Optional	Description
icmpcode	optional	<p>The ICMP code.</p> <p>This parameter is relevant only if you specify <code>icmp</code> as the protocol. You can specify one of the following values:</p> <ul style="list-style-type: none"> <li>network</li> <li>host</li> <li>protocol</li> <li>port</li> <li>df</li> <li>admin</li> </ul> <p>If you specify <code>icmp</code> as the protocol and don't specify <code>icmptype</code> or <code>icmpcode</code>, then all ICMP packets are matched.</p>
description	optional	A description of the security application.

## Orchestration v2 Attributes for `SecIPList`

The following sample JSON shows the required attributes of the `SecIPList` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/admin_ips",
  "secipentries": ["203.0.113.0/30"]
}
```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
secipentries	required	<p>A comma-separated list of the subnets (in CIDR format) or IPv4 addresses for which you want to create this security IP list.</p> <p>For example, to create a security IP list containing the IP addresses 203.0.113.1 and 203.0.113.2, enter one of the following:</p> <ul style="list-style-type: none"> <li>"203.0.113.0/30"</li> <li>"203.0.113.1", "203.0.113.2"</li> </ul>
description	optional	A description of the security IP list.

## Orchestration v2 Attributes for `SecList`

The following sample JSON shows the required attribute of the `SecList` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/sysadmin_seclist"
}
```

Parameters	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
policy	optional	The policy for inbound traffic to the security list. You can specify one of the following values:  deny (default): Packets are dropped. No response is sent. reject: Packets are dropped, but a response is sent. permit: Packets are allowed. This policy effectively turns off the firewall for all instances in this security list.
outbound_cidr_policy	optional	The policy for outbound traffic from the security list. You can specify one of the following values:  deny: Packets are dropped. No response is sent. reject: Packets are dropped, but a response is sent. permit (default): Packets are allowed.
description	optional	A description of the security list.

## Orchestration v2 Attributes for `SecRule`

The following sample JSON shows the required attributes of the `SecRule` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/admin_ssh_to_sysadmin_rule",
  "application": "/oracle/public/ssh",
  "src_list": "seclist:/Compute-acme/joe/admin_ips",
  "dst_list": "seclist:/Compute-acme/joe/sysadmin_seclist",
  "action": "PERMIT"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <i>/Compute-identity_domain/user/object</i> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
src_list	required	The three-part name ( <i>/Compute-identity_domain/user/object_name</i> ) of the source security list or security IP list. You must use the prefix <code>seclist:</code> or <code>seciplist:</code> to identify the list type.
dst_list	required	The three-part name ( <i>/Compute-identity_domain/user/object_name</i> ) of the destination security list or security IP list. You must use the prefix <code>seclist:</code> or <code>seciplist:</code> to identify the list type. <b>Note:</b> You can specify a security IP list as the destination in a <code>secrule</code> , provided <code>src_list</code> is a security list that has DENY as its outbound policy.
application	required	The three-part name of the security application: ( <i>/Compute-identity_domain/user/object_name</i> ) for user-defined security applications and <i>/oracle/public/object_name</i> for predefined security applications.
action	required	Set this parameter to PERMIT.
description	optional	A description of the security rule.
disabled	optional	Indicates whether the security rule is enabled (set to True) or disabled (False). The default setting is False.

## Orchestration v2 Attributes for `SecurityProtocol`

The following sample JSON shows the key attributes of the `SecurityProtocol` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "description": "Sec Protocol 1",
  "dstPortSet": ["20", "155-1100"],
  "ipProtocol": "tcp",
  "name": "/Compute-acme/joe/secprotocol_1",
  "srcPortSet": ["10", "55-100"]
}
```

Parameter	Required or Optional	Description
name	required	<p>The three-part name of the object (<i>/Compute-identity_domain/user/object</i>).</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
ipProtocol	optional	<p>The protocol used in the data portion of the IP datagram.</p> <p>The value that you specify can be either a text representation of a protocol or any unsigned 8-bit assigned protocol number in the range 0–254. See <i>Assigned Internet Protocol Numbers</i> (<a href="http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml">http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml</a>).</p> <p>The following text representations are allowed:</p> <ul style="list-style-type: none"> <li>• tcp</li> <li>• udp</li> <li>• icmp</li> <li>• igmp</li> <li>• ipip</li> <li>• rdp</li> <li>• esp</li> <li>• ah</li> <li>• gre</li> <li>• icmpv6</li> <li>• ospf</li> <li>• pim</li> <li>• sctp</li> <li>• mplsip</li> <li>• all</li> <li>• Any number from 0 to 254</li> </ul> <p>If no protocol is specified, all protocols are allowed.</p>
srcPortSet	optional	<p>List of port numbers or port range strings to match the packet's source port.</p> <ul style="list-style-type: none"> <li>• For <i>tcp</i>, <i>sctp</i>, and <i>udp</i>, each port is a source transport port, between 0 and 65535, inclusive.</li> <li>• For <i>icmp</i>, each port is an ICMP type, between 0 and 255, inclusive.</li> </ul> <p>If no source ports are specified, all source ports or ICMP types are allowed.</p>
dstPortSet	optional	<p>List of port numbers or port range strings to match the packet's destination port.</p> <p>For <i>tcp</i>, <i>sctp</i>, and <i>udp</i>, each port is a destination transport port, between 0 and 65535, inclusive. For <i>icmp</i>, each port is an ICMP code, between 0 and 255, inclusive.</p> <p>If no destination ports are specified, all destination ports or ICMP codes are allowed.</p>



Parameter	Required or Optional	Description
description	optional	Description of the security protocol.
tags	optional	Strings that you can use to tag the security protocol.

## Orchestration v2 Attributes for `SecurityRule`

The following sample JSON shows the key attributes of the `SecurityRule` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "acl": "/Compute-acme/joe/acl_1",
  "description": "Sec Rule 1",
  "flowDirection": "egress",
  "name": "/Compute-acme/joe/ipnetSecrule1",
  "secProtocols": ["/Compute-acme/joe/secprotocol_1"],
  "srcIpAddressPrefixSets": ["/Compute-acme/joe/
ext_ip_address_list_1"]
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
flowDirection	required	The direction of flow of traffic that this rule applies to. Allowed values are <code>ingress</code> or <code>egress</code> .
srcVnicSet	optional	The vNICset from which you want to permit traffic. Only packets from vNICs in the specified vNICset are permitted. When no source vNICset is specified, traffic from any vNIC is permitted.
dstVnicSet	optional	The vNICset to which you want to permit traffic. Only packets to vNICs in the specified vNICset are permitted. When no destination vNICset is specified, traffic to any vNIC is permitted.
srcIpAddressPrefixSets	optional	A list of IP address prefix sets from which you want to permit traffic. Only packets from IP addresses in the specified IP address prefix sets are permitted. When no source IP address prefix sets are specified, traffic from any IP address is permitted.
dstIpAddressPrefixSets	optional	A list of IP address prefix sets to which you want to permit traffic. Only packets to IP addresses in the specified IP address prefix sets are permitted. When no destination IP address prefix sets are specified, traffic to any IP address is permitted.

Parameter	Required or Optional	Description
secProtocols	optional	A list of security protocols for which you want to permit traffic. Only packets that match the specified protocols and ports are permitted. When no security protocols are specified, traffic using any protocol over any port is permitted.
enabledFlag	optional	Allows the security rule to be enabled or disabled. This parameter is set to <code>true</code> by default. Specify <code>false</code> to disable the security rule.
acl	optional	The name of the access control list (ACL) that contains this security rule.
description	optional	Description of the security rule.
tags	optional	Strings that you can use to tag the security rule.

## Orchestration v2 Attributes for `SSHKey`

The following sample JSON shows the required attributes of the `SSHKey` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/key1"
  "enabled": false,
  "key": "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGDzU2lCEj6JsqIMQAYwNbmZ5P2BVxA...",
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.  When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
key	required	The SSH public key value.
enabled	optional	Indicates whether the key must be enabled or disabled. SSH keys are enabled by default. To explicitly enable the key, specify <code>true</code> . To disable a key, specify <code>false</code> . Disabled keys can't be associated with instances.

## Orchestration v2 Attributes for `StorageAttachment`

The following sample JSON shows the key attributes of the `StorageAttachment` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "storage_volume_name": "{{My_Storage_Volume:name}}",
  "instance_name": "{{My_Instance:name}}",
  "index": 1
}
```

Parameter	Required or Optional	Description
<code>name</code>	optional	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>
<code>instance_name</code>	required	<p>The name of the instance to which you want to attach the storage volume.</p> <p>You can specify the object name as a reference to an object. For example, for an instance with the label <code>my_instance</code>, you can specify its name as follows:</p> <pre>"instance_name": "{{my_instance:name}}",</pre>
<code>storage_volume_name</code>	required	<p>The name of the storage volume that you want to attach to the instance.</p> <p>You can specify the object name as a reference to an object. For example, for a storage volume with the label <code>my_volume</code>, you can specify its name as follows:</p> <pre>"storage_volume_name":   "{{my_volume:name}}",</pre>

Parameter	Required or Optional	Description
index	required	<p>The index number for the volume.</p> <p>The allowed range is 1 to 10. If you want to use a storage volume as the boot disk for an instance, you must specify the index number for that volume as 1.</p> <p>The index determines the device name by which the volume is exposed to the instance. Index 0 is allocated to a nonpersistent boot disk, <code>/dev/xvda</code>. An attachment with index 1 is exposed to the instance as <code>/dev/xvdb</code>, an attachment with index 2 is exposed as <code>/dev/xvdc</code>, and so on.</p>

## Orchestration v2 Attributes for `StorageSnapshot`

The following sample JSON shows the key attributes of the `StorageSnapshot` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "volume": "/Compute-acme/joe/voll",
  "name": "/Compute-acme/joe/voll-snapshot"
  "description": "Remote snapshot of voll"
}
```

Parameter	Required or Optional	Description
volume	required	The three-part name <code>/Compute-identity_domain/user/object_name</code> of the storage volume that you want to create a snapshot of.
name	optional	<p>The three-part name of the object (<code>/Compute-identity_domain/user/object</code>).</p> <p>If you don't specify a name for this object, then the name is generated automatically.</p> <p>Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.</p> <p>When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.</p>

Parameter	Required or Optional	Description
property	optional	Specify <code>/oracle/private/storage/snapshot/collocated</code> to create a collocated snapshot. Collocated snapshots are stored in the same physical location as the original storage volume. Collocated snapshots and volumes from collocated snapshots can be created very quickly.  If you don't specify a value, a remote snapshot is created. Remote snapshots aren't stored in the same location as the original storage volume. Instead, they are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. Creating a remote snapshot and restoring a storage volume from a remote snapshot can take a longer time than for collocated snapshots, as data is written to and from the Oracle Cloud Infrastructure Object Storage Classic instance.
platform	optional	Specify the operating system platform for a bootable storage volume, such as Linux or Windows.
tags	optional	Strings that you can use to tag the storage snapshot.
description	optional	Description of the storage snapshot.

## Orchestration v2 Attributes for `StorageVolume`

The following sample JSON shows the key attributes of the `StorageVolume` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```
{
  "name": "/Compute-acme/joe/boot",
  "bootable": true,
  "imagelist": "/oracle/public/oel_6.6_20GB_x11_RD",
  "properties": ["/oracle/public/storage/default"],
  "size": "22548578304"
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ).  Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive.

Parameter	Required or Optional	Description
size	required	<p>The size of this storage volume.</p> <p>Use one of the following abbreviations for the unit of measurement:</p> <ul style="list-style-type: none"> <li>• B or b for bytes</li> <li>• K or k for kilobytes</li> <li>• M or m for megabytes</li> <li>• G or g for gigabytes</li> <li>• T or t for terabytes</li> </ul> <p>For example, to create a volume of size 10 gigabytes, you can specify 10G, or 10240M, or 10485760K, and so on.</p> <p>The allowed range is from 1 GB to 2 TB, in increments of 1 GB.</p>
properties	required	<p>Based on your latency and IOPS requirements, select one of the following storage properties:</p> <ul style="list-style-type: none"> <li>• For standard latency and throughput, specify <code>/oracle/public/storage/default</code>.</li> <li>• For high latency and throughput, specify <code>/oracle/public/storage/latency</code>.</li> <li>• For the highest latency and throughput, specify <code>/oracle/public/storage/ssd/gpl</code>.</li> </ul>
description	optional	The description of the storage volume.
bootable	optional	<p>Indicates whether the storage volume can be used as the boot disk for an instance.</p> <p>The default value is <code>False</code> (not a bootable volume).</p> <p>If you set the value to <code>True</code>, then you must specify values for the following parameters:</p> <ul style="list-style-type: none"> <li>• <code>imagelist</code> The machine image that you want to extract on to the storage volume that you're creating.</li> <li>• <code>imagelist_entry</code> (Optional) The version of the image list entry that you want to extract. The default value is 1.</li> </ul>
tags	optional	Strings that you can use to tag the storage volume.

## Orchestration v2 Attributes for `VirtualNicSet`

The following sample JSON shows the key attributes of the `VirtualNicSet` object type. A description of each of the required and optional attributes of this object type is provided in the table that follows the JSON sample.

```

{
  "name": "/Compute-acme/joe/vnicset1",
  "appliedAcls": ["/Compute-acme/joe/acl_1", "/Compute-acme/joe/
acl_2"]
}
```

Parameter	Required or Optional	Description
name	required	The three-part name of the object ( <code>/Compute-identity_domain/user/object</code> ). Object names can contain only alphanumeric characters, hyphens, underscores, and periods. Object names are case-sensitive. When you specify the object name, ensure that an object of the same type and with the same name doesn't already exist. If such an object already exists, another object of the same type and with the same name won't be created and the existing object won't be updated.
vnic	optional	The list of vNICs associated with this vNICset.
appliedAcls	optional	The names of the ACLs applied to the vNICs in the vNICset. A vNICset can have multiple ACLs applied to it and an ACL can be applied to multiple vNIC sets.
description	optional	Description of the route.
tags	optional	Strings that you can use to tag the IP network exchange.

## Orchestration v2 Life Cycle

When you activate an orchestration, all the objects defined in it are created and the orchestration moves to the `active` state. When you suspend an orchestration, the nonpersistent objects defined in it are deleted and the orchestration moves to the `suspended` state. When you deactivate an orchestration, all the objects defined in it are deleted and the orchestration moves to the `inactive` state.



### activating

The orchestration is starting.

Compute Classic is provisioning the objects defined in the orchestration. The time to complete this action varies depending on the number and type of objects that are being provisioned.

### active

The orchestration is running.

Compute Classic successfully provisioned all the objects in the orchestration. For example, an orchestration displays that it is in the `active` state when all its instances have been created, storage volumes are online, and so on.

### suspending

The orchestration is being suspended.

- All the nonpersistent (`persistent: false`) objects are being deleted.

- All persistent (`persistent: true`) objects are being created if they were not already created.

#### **suspended**

- All the nonpersistent objects are deleted.
- All persistent objects are created if they were not already created.

#### **terminating**

The orchestration is being terminated.

Compute Classic is deleting all the objects defined in the orchestration.

#### **inactive**

The orchestration is inactive.

Compute Classic successfully deleted all the objects defined in the orchestration.

#### **terminal\_error**

The orchestration reached an error state from which it can't recover. You must identify and address the issue.

The following are a few examples of the possible issues:

- Errors in the template of your orchestration are preventing the orchestration from being created.
- You don't have the necessary permissions to provision a specified object.

#### **transient\_error**

The orchestration is automatically recovering from a failure. You don't need to intervene.

For example, if an instance crashes and Compute Classic is automatically re-creating the instance, a `transient_error` is displayed.

## Managing Orchestrations v2

### Topics

- [Uploading an Orchestration v2](#)
- [Starting an Orchestration v2](#)
- [Monitoring Orchestrations v2](#)
- [Suspending an Orchestration v2](#)
- [Terminating an Orchestration v2](#)
- [Downloading an Orchestration v2](#)
- [Workflows for Updating Orchestrations v2](#)
- [Updating an Orchestration v2](#)
- [Deleting an Orchestration v2](#)
- [Managing Orchestrations v2 Using the REST API](#)



## Uploading an Orchestration v2

To use an orchestration to control the provisioning and life cycle of objects in Compute Classic, you must define the orchestration in a JSON-format file and then upload the orchestration to Compute Classic.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have already created the orchestration file that you want to upload. See [Building Your First Orchestration v2](#)
- You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. If your JSON isn't valid, then an error occurs when you upload the orchestration. Oracle doesn't support or endorse any third-party JSON-validation tool.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Click **Upload Orchestration** and select the orchestration file that you want to upload.

The orchestration is uploaded. If you upload an orchestrations v2 file with the `desired_state` specified as `active`, the orchestration is started automatically and the objects defined in it are created.

To upload an orchestration using the CLI, use the `opc compute orchestration-v2 add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To upload orchestrations v2 using the API, use the `POST /platform/v1/orchestration/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Starting an Orchestration v2

When you start an orchestration, the objects defined in it are created, and when you stop an orchestration, those objects are deleted.

If your orchestration has the `desired_state` defined as `active`, it starts immediately when you upload the orchestration. If the desired state specified isn't `active`, then you must explicitly start the orchestration. You can also start an orchestration if you had previously suspended or stopped it.

 **Note:**

If you're about to start an orchestration that creates a large number of storage volumes or instances, consider whether you really need all those resources. If not, redefine your orchestration to create only the resources that you need.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have uploaded the orchestration to Compute Classic. See [Uploading an Orchestration v2](#).


 **Note:**

Ensure that each object defined in an orchestration has a unique name, and that objects of the same type with the same name don't already exist. If any of the objects defined in an orchestration already exists, when the orchestration attempts to start, it reports an error.

### Procedure

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

3. Click the **Orchestrations** tab.
4. Go to the orchestration that you want to start. From the  menu, select **Start**.

When you start an orchestration, its status changes to **Starting** and the objects defined in the orchestration are provisioned. When all the objects have been created, the status of the orchestration changes to **Ready**.

If the orchestration can't create an object, its status changes to **Transient Error** or **Terminal Error**. An orchestration might transition from the **Transient Error** to the **Ready** state when it completes creating all the specified objects.

If the status of your orchestration shows **Terminal Error**, then you must stop the orchestration, identify and fix the issues in the orchestration JSON file, and then start the orchestration again.

To start an orchestration using the CLI, use the `opc compute orchestration-v2 update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To start orchestrations v2 using the API, use the `PUT /platform/v1/orchestration/orchestrationName` method with the query argument `desired_state=active`. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After starting an orchestration, you can view its status on the Orchestrations page. If you no longer require any of the objects created by an orchestration, then to delete all the objects, stop the orchestration. Alternatively, to delete only nonpersistent objects, suspend the orchestration. See [Terminating an Orchestration v2](#) or [Suspending an Orchestration v2](#).

## Monitoring Orchestrations v2

The Orchestrations page shows you a list of your orchestrations and the status of each orchestration.


To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.

All orchestrations are displayed, with information about their description and status.

### Tip:

You can filter the list of orchestrations according to their category or status. To view orchestrations with a specific status (such as ready, error, or stopped), click the **Show** menu and select the appropriate filter. To view orchestrations of a specific category (such as all or personal), click the **Category** menu and select the appropriate filter.

3. Go to the orchestration that you want to view and, from the  menu, select **View**.  
The orchestration details page displays the orchestration JSON along with other information about objects created by the orchestration, such as instances, IP networks, interfaces on IP networks, storage volumes, and so on.

To get a list of your orchestrations using the CLI, use the `opc compute orchestration-v2 list` command and to view the details of an orchestration, use the `opc compute rchestration-v2 get` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To get a list of your orchestrations v2 using the API, use the `GET /platform/v1/orchestration/container/` method and to view the details of an orchestration, use the `GET /platform/v1/orchestration/name/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

For information about the status of an orchestration, see [Orchestration v2 Life Cycle](#).

To start an orchestration, see [Starting an Orchestration v2](#) and to stop an orchestration, see [Terminating an Orchestration v2](#).


## Suspending an Orchestration v2

When you suspend an orchestration, all nonpersistent resources that were provisioned by that orchestration are deleted. Persistent objects aren't deleted.

### Prerequisites

- The orchestration that you want to suspend must be in the **Ready** state or **Error** state. You can't suspend orchestration in a transient state such as **Starting** or **Stopping**.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to suspend. From the  menu, select **Suspend**.

The status of the orchestration changes to **Suspending**.

After all nonpersistent objects have been deleted, the status of the orchestration changes to **Suspended**. You can view the orchestration, stop it, delete it, or start it again.

To suspend an orchestration using the CLI, use the `opc compute orchestration-v2 update name [--desired-state suspend]` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete only the nonpersistent objects in orchestrations v2 using the API, use the `PUT /platform/v1/orchestration/orchestrationName` method with the query argument `desired_state=suspend`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.


## Terminating an Orchestration v2

When you terminate or stop an orchestration, all the instances and other resources that were provisioned by that orchestration are deleted.

### ▲ Caution:

When you terminate an orchestration, *all* the resources that are created by the orchestration are deleted. For example, if you use an orchestration to create storage volumes and attach them to your instances, then such storage volumes are deleted when you terminate the orchestration, and you lose the data stored on those storage volumes.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to terminate. From the  menu, select **Terminate**.

The status of the orchestration changes to **Stopping**.

After all objects have been deleted, the status of the orchestration changes to **Stopped**. You can start the orchestration again, or if you don't require the orchestration any more, you can delete it.

To terminate an orchestration using the CLI, use the `opc compute orchestration-v2 delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete all the objects in orchestrations v2 using the API, use the `PUT /platform/v1/orchestration/orchestrationName` method with the query argument `desired_state=inactive`. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).


If you no longer need an orchestration, you can delete it. See [Deleting an Orchestration v2](#).

## Downloading an Orchestration v2

You can download the orchestration file to your local host, edit it, and upload a modified orchestration file as a new orchestration.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that

the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to download. From the  menu, select **Download**, and save the orchestration file on your local host.

You can edit the downloaded orchestration file on your local host, as required, by using any text editor, and then upload the edited orchestration file as a new orchestration. Remember to change the `name` attribute in the JSON file.

For the procedure to upload an orchestration to Compute Classic, see [Uploading an Orchestration v2](#).

To download an orchestration using the CLI, use the `opc compute orchestration-v2 get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To download orchestrations v2 using the API, use the `GET /platform/v1/orchestration/name/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Workflows for Updating Orchestrations v2

You can update an orchestration that is in the **Active**, **Suspended**, **Stopped**, or **Error** state.

When an orchestration is stopped, all objects created by that orchestration are deleted, so all the attributes of an object can be updated. When you start the orchestration, the objects are created with the updated attributes.

When an orchestration is suspended, nonpersistent objects have the status **Inactive** and all attributes of those objects can be updated. Those objects are created with the updated attributes when you start the orchestration or when you change the object's properties from nonpersistent to persistent.

When an orchestration is suspended, persistent objects aren't deleted. Those objects have the status **Active** and some attributes of those objects can't be updated. For example, you can't update the name of a storage volume while the storage volume is online. Updates to attributes of persistent objects take effect immediately. Attempting to modify attributes that can't be modified while an object has the status **Active** might cause the orchestration to go into a terminal error state, indicating that the update failed.

The following table displays the situations in which some or all of an object's attributes can be modified and the actions required for the updates to be effective.

Orchestration Status	Object Persistence	Object Status	Modifiable Attributes of an Object	Updates Effective
Ready	Persistent	Active	Some	Immediately.

Orchestration Status	Object Persistence	Object Status	Modifiable Attributes of an Object	Updates Effective
Ready	Nonpersistent	Active	Some	Immediately.
Suspended	Persistent	Active	Some	Immediately.
Suspended	Nonpersistent	Inactive	All	When orchestration is started or object is updated to be persistent.
Stopped	Persistent	Inactive	All	When orchestration is started.
Stopped	Nonpersistent	Inactive	All	When orchestration is started.

## Topics

- [Workflow for Changing the Persistence of an Object](#)
- [Workflow for Adding an Instance](#)
- [Workflow for Adding an Object to an Orchestration](#)
- [Workflow for Updating a Nonpersistent Object](#)
- [Workflow for Updating a Persistent Object](#)
- [Workflow for Resizing an Instance](#)
- [Workflow for Updating an Instance](#)


## Workflow for Changing the Persistence of an Object

You can specify persistence for each object in an orchestration. When you suspend an orchestration, nonpersistent objects are deleted, but persistent objects are retained. For most object types, certain attributes of an object can't be modified while the object exists. This means that, if you want to update those attributes of an object, after suspending the orchestration you must also ensure that the object is nonpersistent and its current status is **Inactive**.

Similarly, if you've suspended an orchestration and added or updated an object in that orchestration, then to start that object without changing the status of the orchestration, you can specify the object to be persistent. The object is created right away while the orchestration remains in the **Suspended** state.

### Caution:

When you specify an object as nonpersistent, if the orchestration is in the Suspended state, the object is deleted immediately.


1. Select the name of the required orchestration to open the orchestrations details page.
2. On the orchestrations details page, go to the object for which you want to specify persistence. From the  menu, select **Properties**.
3. In the Object Properties dialog box, to set the object as persistent, select the **Persistent** check box. To set the object as nonpersistent, deselect the **Persistent** check box. Then click **Update**.

If the orchestration is in the **Suspended** state, the change in object persistence is effective immediately. If you updated the object to be nonpersistent, it is deleted and the orchestration details page shows the object's status as **Inactive**. If you updated the object to be persistent, it is created and the orchestration details page shows its status as **Active**.

If the orchestration is in the **Stopped** state, the change in object persistence is updated in the orchestration and it is effective when the orchestration is started.

### Workflow for Adding an Instance

To add an instance to an existing orchestration:

1. Select the name of the required orchestration to open the orchestrations details page.
2. On the orchestrations details page, go to the Instance section and click **Add**. An instance with default configuration is added to the orchestration with the status **Inactive**.
3. (Optional) To view or modify the instance configuration, from the  menu, select **Update**. The instance details page displays the instance configuration. You can modify the configuration as required.

Alternatively, you can view and modify the instance configuration JSON by selecting **Edit JSON**.

4. To create the instance, start the orchestration, or if the orchestration is in the **Suspended** state, set the instance to be persistent.


### Workflow for Adding an Object to an Orchestration

To add an object to an existing orchestration:

1. Select the name of the required orchestration to open the orchestrations details page.
2. On the orchestrations details page, go to the object type that you want to add and click **Add**.
3. A dialog box is displayed. Enter the required information to create the object. The object is added to the orchestration.
4. To create the object, start the orchestration, or if the orchestration is in the **Suspended** state, set the object to be persistent.

### Workflow for Updating a Nonpersistent Object

You can specify persistence for each object in an orchestration. All nonpersistent objects are deleted when you either suspend or stop an orchestration. To update a nonpersistent object:

1. Select the name of the required orchestration to open the orchestrations details page.
2. On the orchestration details page, go to the object that you want to update and from the  menu, select **Update**.

Alternatively, you can view and modify the JSON by selecting **Edit JSON**.

3. A dialog box is displayed. Modify the object attributes as required. The orchestration is updated.




4. To create the object with the updated attributes, start the orchestration, or if the orchestration is in the **Suspended** state, set the object to be persistent.

### Workflow for Updating a Persistent Object


You can specify persistence for each object in an orchestration. While persistent objects are deleted when you stop an orchestration, when you suspend an orchestration, persistent objects aren't deleted. The orchestration details page shows the status for these object as **Active**. Although you can update almost all the attributes of an object while it has the status **Active**, the name of the object can't be updated.

To update a persistent object:

1. Select the name of the required orchestration to open the orchestrations details page.
2. If you want to change the name of the object, then modify the object to be nonpersistent:
  - a. On the orchestration details page, go to the object that you want to update and from the  menu, select **Properties**.
  - b. In the Object Properties dialog box, to set the object as nonpersistent, deselect the **Persistent** check box. Then click **Update**.

#### **Caution:**

When you specify an object as nonpersistent, if the orchestration is in the **Suspended** state, the object is deleted immediately.



3. On the orchestration details page, go to the object that you want to update and from the  menu, select **Update**.

Alternatively, you can view and modify the JSON by selecting **Edit JSON**.

4. A dialog box is displayed. Modify the object attributes as required.
5. To create the object with the updated attributes, start the orchestration, or if the orchestration is in the **Suspended** state, set the object to be persistent.

### Workflow for Resizing an Instance



You can't resize a running instance. Before you resize an instance, you must ensure that the instance is deleted. You can re-create the instance later, after it has been resized. To resize an instance:

1. Select the name of the required orchestration to open the orchestrations details page.
2. If the instance status is displayed as **Active**, then:
  - a. On the orchestration details page, go to the instance that you want to resize and from the  menu, select **Properties**.
  - b. In the Object Properties dialog box, deselect the **Persistent** check box. Then click **Update**. The status of the instance changes from **Active** to **Inactive**.
3. On the orchestration details page, go to the instance that you want to update and from the  menu, select **Update**.

4. In the Resize an Instance dialog box, select the required shape. Ensure that the shape you select is bigger than the current shape and click **Resize**. The orchestration is updated with the selected shape.
5. To create the instance with the updated shape, start the orchestration, or if the orchestration is in the **Suspended** state, update the instance to be persistent.

### Workflow for Updating an Instance

When an instance is running, you can modify some attributes of the instance. For example, you can add storage volumes to the instance and you can add the instance to security lists in the shared network. However, many other instance attributes can be modified only when the instance is deleted. To update an instance:

1. Select the name of the required orchestration to open the orchestrations details page.
2. If the orchestration is suspended (not stopped), and if the updates that you want to make require the instance to be deleted, then ensure that the instance is nonpersistent:
  - a. On the orchestration details page, go to the instance that you want to update and from the  menu, select **Properties**.
  - b. In the Object Properties dialog box, ensure that the **Persistent** check box isn't selected. If it is, then deselect it, and then click **Update**. The status of the instance changes from **Active** to **Inactive**.
3. On the orchestration details page, go to the instance that you want to update and from the  menu, select **Update**.

Alternatively, you can view and modify the instance configuration JSON by selecting **Edit JSON**.
4. Make the required changes to the instance configuration. The orchestration is updated with your changes.
5. To create the instance with the updated attributes, start the orchestration, or if the orchestration is in the **Suspended** state, set the instance to be persistent.

## Updating an Orchestration v2

The web console allows you to update each object of an orchestration separately. You can add or remove objects, modify general attributes of objects such as persistence and dependencies, or update the object-specific attributes.

You can update an orchestration in either of the following ways:

- By downloading the orchestration file to your local host and updating it using a text editor.

You'll have to delete the existing orchestration before you can upload the modified orchestration. Alternatively, you can upload the modified orchestration with a new name. See [Downloading an Orchestration v2](#), [Deleting an Orchestration v2](#), and [Uploading an Orchestration v2](#)
- Directly in the web console, by selecting the **Update** option, as described in the following procedure.

## Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Note:

When an orchestration is stopped, all objects created by that orchestration are deleted, so all the attributes of an object can be updated. When you start the orchestration, the objects are created with the updated attributes.


When an orchestration is suspended, nonpersistent objects have the status **Inactive** and all attributes of those objects can be updated. Those objects are created with the updated attributes when you start the orchestration or when you change the object's properties from nonpersistent to persistent.





When an orchestration is suspended, persistent objects aren't deleted. Those objects have the status **Active** and some attributes of those objects can't be updated. For example, you can't update the name of a storage volume while the storage volume is online. Updates to attributes of persistent objects take effect immediately. Attempting to modify attributes that can't be modified while an object has the status **Active** might cause the orchestration to go into a terminal error state, indicating that the update failed.

The following table displays the situations in which some or all of an object's attributes can be modified and the actions required for the updates to be effective.

Orchestration Status	Object Persistence	Object Status	Modifiable Attributes of an Object	Updates Effective
Ready	Persistent	Active	Some	Immediately.
Ready	Nonpersistent	Active	Some	Immediately.
Suspended	Persistent	Active	Some	Immediately.
Suspended	Nonpersistent	Inactive	All	When orchestration is started or object is updated to be persistent.
Stopped	Persistent	Inactive	All	When orchestration is started.
Stopped	Nonpersistent	Inactive	All	When orchestration is started.



## Procedure

- Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
- Click the **Orchestrations** tab.
- Go to the orchestration that you want to update. From the  menu, select **Update**.
- On the orchestration details page, go to the object that you want to update.

- To add an instance, in the Instance section, click **Add**. An instance with default configuration is added to the orchestration with the status **Inactive**. When you start the orchestration, the instance is created.
- To add any other object type, go to the appropriate section and click **Add**. For example, to create an access control list (ACL), click **Add** in the ACL section. The Create dialog box appears. Enter the required information and click **Create**. The object is added to the orchestration with the status **Inactive**. When you start the orchestration, the object is created.
- To update the general properties of an object, from the  menu, select **Properties**. The Object Properties dialog box appears. You can update the description, persistence, and dependencies of the object.
- To update the general properties or attributes of an object by editing the JSON, from the  menu, select **Edit JSON**. The Edit Orchestration Object JSON dialog box appears, with the JSON for the general properties as well as the object-specific attributes. You can update the JSON for the specified object as required and then click **Update**. The changes are saved in the orchestration.
- To change the shape of an instance, ensure that the status of the instance is **Inactive**. From the  menu, select **Resize Instance**. In the Resize an Instance dialog box, select the shape you want to use and click **Resize**. The instance configuration is updated in the orchestration.
- To update attributes of an instance, go to the appropriate instance and from the  menu, select **Update**. The instance details page is displayed. You can update the instance attributes as required. When you're done, click **Back to Orchestration Details**. The instance attributes are updated in the orchestration.

 **Note:**

It is recommended that you ensure that an instance is nonpersistent before updating its attributes. Nonpersistent instances are deleted when the orchestration is suspended and all attributes can be modified. Persistent instances are still running when the orchestration is suspended and many attributes can't be modified while the instance is running. Attempting to modify attributes that can't be modified might cause the orchestration to go into an error state.

- To update attributes of any other object, from the  menu, select **Update**. The Update dialog box appears. Update the information as required and click **Update**. The object attributes are updated in the orchestration.
  - To remove an object from the orchestration, from the  menu, select **Delete**. The Delete Orchestration Object dialog box appears. When you confirm the deletion, the object is removed from the orchestration.
5. When you're done making changes to the objects in the orchestration, use the buttons at the top of the page to do the following:
- If the orchestration is in the **Suspended** state, you can start the orchestration to create all objects, or stop the orchestration to delete all persistent objects.

- If the orchestration is in the **Stopped** state, you can start the orchestration to create all objects.

 **Note:**

If the changes you made cause the orchestration to go into an **Error** state, you can update the orchestration to undo your changes.

To update an orchestration using the API, download the orchestration, modify it, and then upload the modified JSON.

To download an orchestration using the CLI, use the `opc compute orchestration-v2 get` command. After editing an orchestration, to upload it using the CLI, use the `opc compute orchestration-v2 update` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To download orchestrations v2 using the API, use the `GET /platform/v1/orchestration/orchestrationName/` method. After modifying an orchestration, to upload it using the API, use the `PUT /platform/v1/orchestration/orchestrationName` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

 **Note:**

When you use the API to update an orchestration, you don't need to stop the orchestration to update it. However, you can update an orchestration only if the orchestration is not in a transient state (activating, suspending, or deactivating).

When you update an orchestration without stopping it, the orchestration attempts to update the objects that you've modified. Remember, however, that various attributes of any object are immutable and modifying those attributes isn't permitted. For example, the name of an object, the `bootable` attribute of a storage volume, or the `shape` of a running instance are immutable. If you attempt to modify immutable attributes of any object, the update will fail and the orchestration will go into the `terminal_error` state. If this happens, you must identify and fix the issues and then update or activate the orchestration again.

When you stop an orchestration, all the objects created by the orchestration are deleted. You can then update any attribute of any object. The objects are created afresh when the updated orchestration starts.

## Deleting an Orchestration v2


If you don't need an orchestration any more, you can delete the orchestration. When you delete an orchestration, it's no longer listed on the Orchestrations page, and you can't perform any action on it. Orchestrations v2 can be deleted even when they are in a ready or error state. In such cases, any resources that have been created by the

orchestration and haven't been stopped or deleted yet, are deleted when you delete the orchestration.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- The orchestration that you want to delete must be in the **Ready**, **Suspended**, **Stopped**, or **Error** state. You can't delete an instance in a transient state such as **Starting** or **Stopping**.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Orchestrations** tab.
3. Identify the orchestration that you want to delete. From the  menu, select **Delete**.

 **Note:**

When you delete an orchestration that isn't in the **Stopped** state, all existing resources created by the orchestration are deleted.

When all objects created by the orchestration have been deleted, the orchestration itself is deleted and it is no longer listed on the Orchestrations page.

To delete an orchestration using the CLI, use the `opc compute orchestration-v2 delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete orchestrations v2 using the API, use the `DELETE /platform/v1/orchestration/orchestrationName` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing Orchestrations v2 Using the REST API

You can create and manage your orchestrations as well as objects within an orchestration using the REST API.

Use the following APIs to upload and manage orchestrations. See *Orchestration v2 in REST API for Oracle Cloud Infrastructure Compute Classic*.

- To upload an orchestration, use the `POST /platform/v1/orchestration/` method.

Note that if you upload an orchestration with the `desired_state` attribute set to `active`, the orchestration is activated automatically and all objects are created.

- To view the details of all of the orchestrations in a container, use the `GET /platform/v1/orchestration/container/` method.
- To view the details of an orchestration, use the `GET /platform/v1/orchestration/orchestrationName` method.
- To start an orchestration, use the `PUT /platform/v1/orchestration/orchestrationName?desired_state=active` method.
- To update an orchestration, use the `PUT /platform/v1/orchestration/orchestrationName` method.

You can update an orchestration only if the orchestration is not in a transient state (activating, suspending, or deactivating).

 **Note:**

When you update an orchestration, the orchestration attempts to update the objects that you've modified without deleting the object or stopping the orchestration. Remember, however, that various attributes of any object are immutable and modifying those attributes isn't permitted. For example, the `name` of an object, the `bootable` attribute of a storage volume, or the `shape` of a running instance are immutable. If you attempt to modify immutable attributes of any object, the update will fail and the orchestration will go into the `terminal_error` state. If this happens, you must identify and fix the issues and then update or activate the orchestration again.

- To delete all of the nonpersistent objects defined in the orchestration, use the `PUT /platform/v1/orchestration/orchestrationName?desired_state=suspend` method.

When you suspend an `active` orchestration, only the nonpersistent objects are deleted; the persistent objects are not deleted.

- To delete all the objects in an orchestration, use the `PUT /platform/v1/orchestration/orchestrationName?desired_state=inactive` method.

### Managing Objects in an Orchestration

Use the following APIs to manage the objects in orchestrations. See *Orchestration Objects in REST API for Oracle Cloud Infrastructure Compute Classic*.

- To add an object to an orchestration, use the `POST /platform/v1/object/` method.

In the JSON file, specify the orchestration to which you want to add the object.

- To update an object in an orchestration, use the `PUT /platform/v1/object/orchestrationName/objectName` method.

 **Note:**

If you want to modify the attributes of an instance, remember that some attributes can't be updated while the instance is running. To modify these attributes, update the instance with the `desired_state` specified as `shutdown` to stop the instance. Then change the attributes as required and update the instance with the `desired_state` specified as `running`.

Before you update an object, note the following:

- In the JSON file, specify the `label` and the `version` of the object.
- You can't update objects when the orchestration is in a transient state, such as activating, suspending, or deactivating.
- You can't update the `type` of an object.
- To update multiple objects in a single operation, update the orchestration.
- To view the details of an object, use the `GET /platform/v1/object/orchestrationName/objectName` method.
- To delete an object from an orchestration, use the `DELETE /platform/v1/object/orchestrationName/objectName` method.

If the object currently exists, specify the query argument `terminate=True` to delete the object and then remove it from the orchestration.

See the following table for the appropriate method of deleting an object from an orchestration.

Desired State of the Orchestration	Method
active	<code>DELETE /platform/v1/object/orchestrationName/objectName?terminate=True</code>
suspended	<ul style="list-style-type: none"> <li>– If object persistence is set to true: <code>DELETE /platform/v1/object/orchestrationName/objectName?terminate=True</code></li> <li>– If object persistence set to false or not specified: <code>DELETE /platform/v1/object/orchestrationName/objectName</code></li> </ul>
inactive	<code>DELETE /platform/v1/object/orchestrationName/objectName</code>

## Managing Orchestrations v2 Using CLI

You can create and manage your orchestrations as well as objects within an orchestration using the CLI commands.

Use the following CLI commands to upload and manage orchestrations. See *Orchestration v2* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

- To upload an orchestration, use the `opc compute orchestration-v2 add` command.



- To view the details of all of the orchestrations in a container, use the `opc compute orchestration-v2 list` command.
- To view the details of all of the orchestrations and subcontainers in a container, use the `opc compute orchestration-v2 discover` command.
- To view the details of an orchestration, use the `opc compute orchestration-v2 get` command.
- To update an orchestration, use the `opc compute orchestration-v2 update` command.

You can update an orchestration only if the orchestration is not in a transient state (activating, suspending, or deactivating).

You can specify the `desired_state` as `inactive` to add the orchestration without starting or activating. You can activate the orchestration later by changing the `desired_state` to `active`.

 **Note:**

When you update an orchestration, the orchestration attempts to update the objects that you've modified without deleting the object or stopping the orchestration. Remember, however, that various attributes of any object are immutable and modifying those attributes isn't permitted. For example, the `name` of an object, the `bootable` attribute of a storage volume, or the `shape` of a running instance are immutable. If you attempt to modify immutable attributes of any object, the update will fail and the orchestration will go into the `terminal_error` state. If this happens, you must identify and fix the issues and then update or activate the orchestration again.

- To delete all of the nonpersistent objects defined in the orchestration, use the `opc compute orchestration-v2 update name [--desired-state suspend]` command.  
When you suspend an `active` orchestration, only the nonpersistent objects are deleted; the persistent objects are not deleted.
- To delete all the objects in an orchestration, use the `opc compute orchestration-v2 update name [--desired-state inactive]` command.

### Managing Objects in an Orchestration

Use the following CLI commands to manage the objects in orchestrations. See *Orchestration Object in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

- To add an object to an orchestration, use the `opc compute orchestration-object add [--request-body=FILE.json]` command.

In the JSON file, specify the orchestration to which you want to add the object.

- To update an object in an orchestration, use the `opc compute orchestration-object update name [--request-body=FILE.json]` command.

 **Note:**

If you want to modify the attributes of an instance, remember that some attributes can't be updated while the instance is running. To modify these attributes, update the instance with the `desired_state` specified as `shutdown` to stop the instance. Then change the attributes as required and update the instance with the `desired_state` specified as `running`.

Before you update an object, note the following:

- In the JSON file, specify the `label` and the `version` of the object.
- You can't update objects when the orchestration is in a transient state, such as activating, suspending, or deactivating.
- You can't update the `type` of an object.
- To update multiple objects in a single operation, update the orchestration.
- To view the details of all objects in a container, use the `opc compute orchestration-object list container [--orchestration orchestration-name]` command.
- To view the details of all objects and subcontainers in a container, use the `opc compute orchestration-object discover container` command.
- To view the details of an object, use the `opc compute orchestration-object get name` command.
- To delete an object from an orchestration, use the `opc compute orchestration-object delete name [--terminate=true]` command.

If the object currently exists, specify the query argument `[--terminate=true]` to delete the object and then remove it from the orchestration.

# 8

## Managing Machine Images

A **machine image** is a template of a virtual hard disk of a specific size with an installed operating system. You use machine images to create virtual machine instances in Compute Classic.

You can create instances by using either your own machine images or images provided by Oracle.

### Topics

- [About Oracle-Provided Linux Images](#)
- [About Oracle-Provided Solaris Images](#)
- [About Oracle-Provided Windows Images](#)
- [Workflow for Creating Instances Using a Private Machine Image](#)
- [Building Your Own Machine Images](#)
- [Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic](#)
- [Registering a Machine Image in Compute Classic](#)
- [Listing Machine Images](#)
- [Updating a Private Machine Image](#)
- [Deleting a Private Machine Image](#)
- [Maintaining Versions of Private Machine Images](#)

## About Oracle-Provided Linux Images

### Releases

Oracle provides images for the following x86, 64-bit releases of Oracle Linux:

- 5.3
- 5.11 UEK R2
- 6.4 UEK R3 and UEK R4
- 6.6 UEK R3 and UEK R4
- 6.7 UEK R3 and UEK R4
- 6.8 UEK R3 and UEK R4
- 7.1 UEK R3 and UEK R4
- 7.2 UEK R3 and UEK R4

Each image list might contain multiple versions of an image. From release 16.3.6, the release number is included in the name of the image. Whenever possible, use the latest version of an image. When you create an instance, the web console selects the

most recent version of an image by default. If required, you can select an older version.

The Oracle-provided images include the essential packages that are necessary to get started using the instance that you create in Compute Classic.

For more information about available images, see the Compute Classic [FAQ](#).

### Users

In instances created by using any of the Oracle-provided Oracle Linux images, a user named `opc` is preconfigured. The `opc` user has `sudo` privileges and is configured for remote access over the SSH v2 protocol using RSA keys. The SSH public keys that you specify while creating instances are added to the `/home/opc/.ssh/authorized_keys` file.

Note that `root` login is disabled.

### Remote Access

Access to the instance is permitted only over the SSH v2 protocol. All other remote access services are disabled.

### OS Updates

The Oracle-provided images are preconfigured to enable you to install and update packages from the repositories on the Oracle public Yum server. The repository configuration file is in the `/etc/yum.repos.d` directory on your instance. You can install, update, and remove packages by using the `yum` utility.



#### Note:

Oracle-provided images are updated regularly with the necessary patches. But after creating instances using the Oracle-provided images, you're responsible for applying the required security updates published through the Oracle public Yum server.

### Language Support

Arabic, Chinese - Simplified, Chinese - Traditional, Czech, Danish, Dutch, English, Finnish, French, German, Greek, Hebrew, Hungarian, Italian, Japanese, Korean, Norwegian, Polish, Portuguese - Brazilian, Romanian, Russian, Slovak, Spanish, Swedish, Thai, Turkish

## About Oracle-Provided Solaris Images



This topic does not apply to Oracle Cloud at Customer.

### Releases

Oracle provides machine images for Oracle Solaris 11.3 (x86, 64-bit).

The Oracle-provided images include the essential packages for getting started using the instance that you create in Compute Classic. These images are updated according to Oracle's quarterly critical patch update schedule.

 **Note:**

Oracle Solaris Kernel Zones are not supported. The only virtualization that's supported within Oracle Solaris instances in Compute Classic is native non-global zones and Oracle Solaris 10 Zones.

See [Workflow for Creating Your First Oracle Solaris Instance](#).

### Users

In instances created by using any of the Oracle-provided Oracle Solaris images, a user named `opc` is preconfigured. The `opc` user is assigned the System Administrator profile and can perform basic administration tasks without entering a password by using `pfexec`. The `opc` user is configured for remote access over the SSH v2 protocol using RSA keys. The SSH public keys that you specify while creating instances are added to the `/export/home/opc/.ssh/authorized_keys` file.

 **Note:**

Direct login as `root` is disabled. You can assume the `root` role by running `su -`. The password is `solaris_opc` and is marked as expired. You must change the password the first time that you assume the `root` role.

### Disk Layout

The images include a single disk that's mapped to the root ZFS storage pool (`rpool`).

### Support and Package Updates

When you create instances by using an Oracle-provided Oracle Solaris image, you get a support entitlement for Oracle Solaris. You can update packages from the support repository, file service requests to get support, and so on. The default IPS publisher, named `solaris`, is preconfigured to use the Oracle Solaris support repository (<https://pkg.oracle.com/solaris/support/>).

### Language Support

See [Managing Available Locales](#) in *International Language Environments Guide for Oracle Solaris 11.3*.

## About Oracle-Provided Windows Images



This topic does not apply to Oracle Cloud at Customer.

### Releases

Oracle provides machine images in Oracle Cloud Marketplace for Microsoft Windows Server (x86, 64-bit) releases 2012 R2 Standard Edition and 2008 R2 Standard Edition. See [Workflow for Creating Your First Windows Instance](#).

### Licensing Requirements

When you obtain a Windows image from Oracle Cloud Marketplace, the terms and conditions for using the image are displayed. You must read and accept those terms.

### Users

On instances created by using any of the Oracle-provided Windows images, a user named `Administrator` is created automatically. This user is configured for accessing the instance through a remote desktop protocol (RDP) connection. You must set the password for this user while creating the instance.

### Remote Access

Access to the instance is permitted only over RDP. All other remote access services are disabled.

### Disk Layout

The images contain a single disk that's mapped to the `C` drive.

### Language Support

English only.

## Workflow for Creating Instances Using a Private Machine Image

You can create instances in Compute Classic by using either Oracle-provided machine images or your own custom machine images. In either case, you can set up the instances to boot from a persistent disk. This workflow summarizes the high-level steps for building a custom machine image, adding it to Compute Classic, and using that machine image to create instances.

1. Build your machine image. See [Building Your Own Machine Images](#).
2. Upload the `tar.gz` machine image file to Oracle Cloud Infrastructure Object Storage Classic. See [Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic](#).
3. Create a machine image in Compute Classic corresponding to the machine image file stored in Oracle Cloud Infrastructure Object Storage Classic. See [Registering a Machine Image in Compute Classic](#).

4. (Optional) Create a bootable storage volume using the machine image. See [Creating a Bootable Storage Volume](#).
5. Create instances. See [Creating Instances](#).

## Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic

After building your private images, to use the images to launch instances in Compute Classic, you must first upload the image files to Oracle Cloud Infrastructure Object Storage Classic.

Oracle Cloud Infrastructure Object Storage Classic provides an enterprise-grade, large-scale, object storage solution for files and unstructured data.

(Not available on Oracle Cloud at Customer) When your Compute Classic account was activated, an Oracle Cloud Infrastructure Object Storage Classic instance would have been provisioned automatically.

To access Oracle Cloud Infrastructure Object Storage Classic in Oracle Cloud at Customer, you'll need a separate subscription to the service.

### Note:

For information about the operating systems that you can use to build machine images, see [Guidelines for Building Private Images](#).

### Tip:

You can also upload image files to Oracle Cloud Storage Service by using the `uploadcli` tool. With this tool, you can upload multiple files by using a single command. See the [Uploading a Machine Image to Oracle Cloud Infrastructure Object Storage Classic](#) tutorial.

### Prerequisites

- Make sure that the `.tar.gz` file that you want to upload is available on the host from which you're accessing the Compute Classic web console.
- Make sure that you have the required role to upload images to Oracle Cloud Infrastructure Object Storage Classic.
  - If this is the first image being uploaded to Oracle Cloud Infrastructure Object Storage Classic, then you must have the `Storage Administrator` role.
  - If one or more images have previously been uploaded to Oracle Cloud Infrastructure Object Storage Classic, then any user with the `Storage_ReadWriteGroup` role can upload images.

If you don't have the required role or aren't sure, then ask your service administrator to ensure that you have the required role in Oracle Cloud

Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

- Make sure that a replication policy has been set for your Oracle Cloud Infrastructure Object Storage Classic instance. See *Selecting a Replication Policy for Oracle Cloud Infrastructure Object Storage Classic in Using Oracle Cloud Infrastructure Object Storage Classic*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Images** tab.

The Private Images page is displayed.

3. Click **Upload Image**.

4. Enter your password, and then click **Continue**.

The **Upload Image** page is displayed in a new tab.

5. In the **Image File** field, browse to select the `.tar.gz` image file that you want to upload.

The path where the image file will be uploaded and the size of the image are displayed.

6. In the **Target Object** field, enter the name of the object that the image file should be stored as in Oracle Cloud Infrastructure Object Storage Classic.

By default, this field is filled automatically with the name of the selected image file. You can use that name or enter a new name. The name must be unique and it must end with `.tar.gz` (example: `myImage.tar.gz`).

Note this name. You'll need it later when you want to add a machine image to Compute Classic using the `POST /machineimage/` HTTP request or delete the image file from Oracle Cloud Infrastructure Object Storage Classic.

7. Click **Upload**.

#### **Note:**

If an error message is displayed, check whether a replication policy has been selected for your Oracle Cloud Infrastructure Object Storage Classic instance. See *Selecting a Replication Policy for Oracle Cloud Infrastructure Object Storage Classic in Using Oracle Cloud Infrastructure Object Storage Classic*.

If an image file already exists with the name specified in the **Target Object** field, you're prompted to enter another name. If you proceed with the upload without changing the name, the existing image file is overwritten.

The progress indicator shows the percentage of the upload operation that is complete. The time taken to upload the file varies depending the size of the image file. Do not close this browser window while the upload is still in progress.

If you want to cancel the upload, click **Cancel**.



After the file is uploaded to the `compute_images` container in Oracle Cloud Storage Service, a message is displayed to indicate that the image file was successfully uploaded. If you want to upload another image file, click **Upload More**.

To launch instances using the image files that you uploaded to Oracle Cloud Infrastructure Object Storage Classic, you must register the machine images in Compute Classic. See [Registering a Machine Image in Compute Classic](#).

 **Tip:**

By default, any user in your Oracle Cloud Infrastructure Object Storage Classic account who has the `Storage_ReadWriteGroup` role has full read and write access to the `compute_images` container in which you store image files. To restrict access to the `compute_images` container, create a custom role in Oracle Cloud Infrastructure Classic Console, assign that role to only the users who must be allowed to access the `compute_images` container, and then assign the role to the `X-Container-Write` ACL of the container. See the [Restrict Read and Write Access to Containers by Using the REST API](#) tutorial.

## Registering a Machine Image in Compute Classic

You can create your own machine images, register them in Compute Classic, and then use the images to launch instances.

 **Note:**

For information about the operating systems that you can use to build machine images, see [Guidelines for Building Private Images](#).

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You must have uploaded the machine image file to Oracle Cloud Infrastructure Object Storage Classic. See [Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Images** tab.  
The Private Images page is displayed.
3. Click **Associate Image**.

4. Enter a name and description for the new image, select the image file, and click **Add**.

You can now use your machine image to launch instances.

To do this using the CLI, use the `opc compute machine-image add`, `opc compute image-list add`, and `opc compute image-list-entry add` commands, in that order. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To do this using the API, invoke the `POST /machineimage/`, `POST /imagelist/`, and `POST /imagelistentry/` methods, in that order. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Listing Machine Images

A machine image can be either an Oracle-provided image or a private image that you added.

### Listing Oracle-Provided Images

When you create instances by using the web console, the **Oracle Images** tab on the **Image** page lists all the Oracle-provided images.

To get a list of Oracle-provided machine images using the CLI, use the `opc compute machine-image list` command with `/oracle/public` as the container. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To get a list of Oracle-provided machine images using the API, use the `GET /machineimage/oracle/public` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

### Listing Private Images

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Images** tab.

The Private Images page is displayed.

The Private Images page displays all the images that you've added.


To get a list of private machine images using the CLI, use the `opc compute machine-image list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To get a list of private machine images using the API, use the `GET /machineimage/Compute-account/user` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating a Private Machine Image

After you've registered a private machine image in Compute Classic, if required, you can update the description of the machine image.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Images** tab.  
The Private Images page is displayed.
3. Go to the image that you want to update, and from the  menu, select **Update**.
4. In the Update Image dialog box, edit the description of the image. Enter a meaningful description that will help you to identify the image and its attributes easily. Click **Update**.

To update an image using the CLI, use the `opc compute image-list update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an image description using the API, use the `PUT /imagelist/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a Private Machine Image

When you no longer need a private machine image that you've registered in Compute Classic, you can delete the image. Deleting an image removes the image from your Compute Classic account. However, the image file in your Oracle Cloud Infrastructure Object Storage Classic account isn't deleted. You can register this image in your Compute Classic account again later, if required.


### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- Make sure that the machine image isn't used in any orchestration.

#### **Caution:**

If you delete a machine image that's used in an orchestration, then when that orchestration is stopped and re-started, the instances won't be created.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Images** tab.  
The Private Images page is displayed.
3. Go to the image that you want to delete. Make a note of the image name. You might need this later, if you want to register the image again, or if you want to delete this image from your associated Oracle Cloud Infrastructure Object Storage Classic account.
4. From the  menu, select **Delete**.

To delete an image using the CLI, use the `opc compute machine-image delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an image using the API, use the `DELETE /machineimage/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

#### Note:

When you delete a machine image from Compute Classic, the image file that's stored in Oracle Cloud Infrastructure Object Storage Classic is **not** removed.

- At any time, you can register the machine image again in Compute Classic and then use the image to launch instances.
- For instructions to permanently remove a machine image file from Oracle Cloud Infrastructure Object Storage Classic, see the [Deleting Machine Image Files from Oracle Cloud Infrastructure Object Storage Classic](#) tutorial.

## Maintaining Versions of Private Machine Images

You can group multiple versions or flavors of machine images that you build, using **image lists**. An **image list** is a collection of Compute Classic machine images. Each machine image in an image list is identified by a unique entry number. Image lists enable you to administer and use related machine images easily.

#### Note:

For information about the operating systems that you can use to build machine images, see [Guidelines for Building Private Images](#).

For example, you can group multiple versions of an Oracle Linux 6.6 machine image, each containing a different set of packages, in an image list. To view the details of all your Oracle Linux 6.6 image versions, all you need to do is view the details of the image list that contains those images. In an orchestration, you can quickly change the machine image that must be used, say from one Oracle Linux 6.6 image version to another, by simply changing the `imagelist_entry` number.

When you add a machine image using the web console, an image list is created automatically by using the name that you specified for the image. The new machine image becomes the default (and only) entry in the image list.

## Building Your Own Machine Images

### Topics

- [Guidelines for Building Private Images](#)
- [Building an Oracle Linux Machine Image](#)

## Guidelines for Building Private Images

### Note:

Oracle provides support for instances created using Oracle-provided images.

When you build images, consider the following guidelines:

- **Supported operating systems**

Oracle has certified the use of private images that are built using x86, 64-bit versions of the following operating systems:

- Oracle Linux
  - \* 5.3
  - \* 5.11 UEK R2
  - \* 6.4 UEK R3 and UEK R4
  - \* 6.6 UEK R3 and UEK R4
  - \* 6.7 UEK R3 and UEK R4
  - \* 6.8 UEK R3 and UEK R4
  - \* 7.1 UEK R3 and UEK R4
  - \* 7.2 UEK R3 and UEK R4

Oracle Linux images must be set up to boot using kernel version 2.6.36 or later. Kernels starting from v2.6.36 contain PVHVM drivers, which are required for instances to work in Compute Classic.

- Oracle Solaris 11.3

 **Note:**

You can launch instances from images built using other operating systems as well. Do follow the guidelines provided in this document when building such images.

- **Network configuration**

If you expect the instances that're created from your image to be attached to multiple networks, then configure your image to support multiple virtual NICs:

- For Oracle Linux and other Linux distributions that're based on Red Hat Enterprise Linux (RHEL), create a separate interface file under `/etc/sysconfig/network-scripts` for each network.

The interface files should be named `ifcfg-interface`, where `interface` is the interface name:

- \* For Oracle Linux 6.x and RHEL-based images, the interface name should be in the `ethN` format—that is, `eth0`, `eth1`, and so on.
- \* For Oracle Linux 7+ images, the interface name should be in the `emN` format: `em1`, `em2`, and so on.

You can attach your instance to up to eight networks.

Each interface file should be contain the following attributes:

```
DEVICE=interface
ONBOOT=yes
TYPE=Ethernet
BOOTPROTO=dhcp
PERSISTENT_DHCLIENT=1
```

Replace `interface` with the appropriate interface name as described earlier.

- For Debian-based images, edit the `/etc/network/interfaces` file to include the following commands:

```
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet dhcp
auto eth2
iface eth2 inet dhcp
auto eth3
iface eth3 inet dhcp
auto eth4
iface eth4 inet dhcp
auto eth5
iface eth5 inet dhcp
auto eth6
iface eth6 inet dhcp
auto eth7
iface eth7 inet dhcp
```

- **Image disk count and size**

The image must contain only one disk.

Keep your image disk size just as small as is essential. A large image requires more time to be uploaded to Oracle Cloud Infrastructure Object Storage Classic,

and costs more to store. In addition, creating instances and bootable storage volumes from a large image requires more time. Before uploading image files to Oracle Cloud Infrastructure Object Storage Classic, make them *sparse* files. On Linux, you can convert a file to the sparse format by running the command, `cp --sparse=always original_file sparse_file`. And when creating the `tar` archive, to ensure that the `tar` utility stores the sparse file appropriately, specify the `-S` option.

- **User access**

Before creating the image file, plan ahead and provision any users that you'd like to be available when instances are created using the image.

 **Note:**

While creating instances, you can specify one or more SSH public keys.

The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at `http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key`.

- Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the `authorized_keys` file of the `opc` user.
- In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the `authorized_keys` file of the appropriate users.

Alternatively, if you're building an Oracle Linux 6.7 image, you can install and use `opc-init` to perform instance initialization and configuration tasks, including copying the SSH public key to the `authorized_keys` file of the `opc` user. See [Using opc-init in a Private Machine Image](#).

- **Format**

The image must be a full disk image, including a partition table and boot loader. The virtual disk image must be converted to the `raw` format, packaged in a `tar` archive that contains only the image, and compressed using `gzip`. The final image must be a `tar.gz` file.

Choose a `tar.gz` file name that you can use later to easily identify the key characteristics of the image, such as the OS name, OS version, and the disk size. For example, for a root-disabled, Oracle Linux 6.6 image with a 20-GB disk, consider using a file name such as `OL66_20GB_RD.tar.gz`.

- **Security patches**

Apply the necessary security patches and review the security configuration *before* creating the image file.

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

## Building an Oracle Linux Machine Image

You can build Oracle Linux machine images in one of the following ways:

- By using one of the several ready-to-use Oracle Linux machine images provided by Oracle.
- By building your own machine images (without using the Oracle-provided images). For detailed instructions about installing Oracle Linux on Oracle VM VirtualBox; customizing the operating system for enabling key-based SSH access; changing the default kernel; installing Apache HTTP Server, MySQL, and PHP; and then creating a raw image that you can use to launch instances in Compute Classic, see the [Building a Custom Oracle Linux Machine Image with the LAMP Stack](#) tutorial.

After building a machine image, to use it to launch instances, you must upload the `tar.gz` image file to Oracle Cloud Infrastructure Object Storage Classic. See [Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic](#).



# 9

## Managing Storage Volumes

### Topics

- [About Storage Volumes](#)
- [Creating a Storage Volume](#)
- [Creating a Bootable Storage Volume](#)
- [Backing Up and Restoring Storage Volumes Using Snapshots](#)
- [Scheduling Backups of Storage Volumes and Restoring from Backups](#)
- [Attaching a Storage Volume to an Instance](#)
- [Viewing Details of a Storage Volume](#)
- [Mounting and Unmounting a Storage Volume](#)
- [Increasing the Size of a Storage Volume](#)
- [Detaching a Storage Volume from an Instance](#)
- [Deleting a Storage Volume](#)

## About Storage Volumes

A **storage volume** is a virtual disk that provides persistent block storage space for instances in Compute Classic.

You can use storage volumes to store data and applications.

You can also associate a storage volume with a machine image and then, while creating an instance, you can specify that volume as a persistent boot disk for the instance.

- When you create a storage volume, you can specify the capacity that you need. The allowed range is from 1 GB to 2 TB, in increments of 1 GB.
- You can attach one or more storage volumes to an instance either while creating the instance or later, while the instance is running.
- After creating an instance, you can easily scale up or scale down the block storage capacity for the instance by attaching or detaching storage volumes. However, you can't detach a storage volume that was attached during instance creation. Note that, when a storage volume is detached from an instance, data stored on the storage volume isn't lost.

There are certain limitations in using storage volumes with your instances:

- Each storage volume can be up to 2 TB in capacity and you can attach up to 10 storage volumes to each instance. So there is an upper limit on the capacity of block storage that you can add to an instance.
- A storage volume can be attached in read-only mode to only one instance. So multiple instances can't write to a volume.

To provide highly scalable and shared storage in the cloud over NFSv4 for your instances, consider using Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution. This appliance is provisioned on a Compute Classic instance and plays the role of a file server in the cloud. It provides shared, highly scalable, low-cost, and reliable storage capacity in Oracle Cloud Infrastructure Object Storage Classic for your Compute Classic instances running Oracle Linux. For information about the use cases that the appliance is best suited for, see *About Oracle Cloud Infrastructure Storage Software Appliance– Cloud Distribution* in *Using Oracle Cloud Infrastructure Storage Software Appliance*.

## Creating a Storage Volume

A **storage volume** is a virtual disk that provides persistent block storage space for instances in Compute Classic. You can create storage volumes and attach them to instances to provide block storage capacity for storing data and applications. You can also associate a storage volume with a machine image, and then use the storage volume as the boot disk for an instance.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Note:

When an instance is deleted and re-created or shut down and restarted, storage volumes that were attached manually (that is, not attached automatically through the orchestration that was used to create the instance) must be attached again.

### Tip:

Before you begin, read the storage-related recommendations in [Best Practices for Using Compute Classic](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Click **Create Storage Volume**.
4. Select or enter the required information:
  - Enter a name for the storage volume. Note this name. You'll need it later to search for the storage volume on the Storage page.  
Pick a name that you can use later to quickly identify the key characteristics of the storage volume.

- To make this storage volume a boot disk, select a machine image in the **Boot Image** field. Later, while creating an instance, you can specify this volume as the boot disk for the instance.

If you select a machine image with a large disk size, it may take a while for the storage volume to be created.

- Enter the size, in GB, of the storage volume. The allowed range is 1 GB to 2 TB.

Consider the storage capacity needs of the applications that you plan to deploy on the instance, and leave some room for attaching more storage volumes in the future. This approach helps you use the available block storage capacity efficiently in the long run.

If you intend to use this storage volume as a boot disk, then the size must be at least 5% higher than the boot image disk size.

 **Note:**

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

- Select a storage property.

Based on your latency and IOPS requirements, select one of the following storage properties.

Storage Property	Latency	Throughput
<code>storage/default</code>	Standard	Standard
<code>storage/latency</code>	Low	High
<code>storage/ssd/gpl</code>	Lowest	Highest

If you select the storage property `storage/latency`, an SSD write cache is provided. When you hit this cache, you experience very high performance. When you miss the cache, you experience the same performance as you would if you had selected the `storage/default` property.

If you select the storage property `storage/ssd/gpl`, you experience consistent high performance at all times, because the SSD cache is always used. This means that under high load conditions, a storage volume created with the property `storage/ssd/gpl` performs better than a storage volume created with the property `storage/latency`.

 **Note:**

SSD storage volumes aren't available in all sites.

The web console might show other storage properties. Don't select any of them.


- Enter a description for the storage volume.

5. Click **Create**.

The Storage page is displayed.

While the new storage volume is being created, the **Status** field for the storage volume shows **Initializing**.

When the storage volume is ready, the **Status** field changes to **Online**.

To view details of the new storage volume, search for it by using the name that you noted earlier. From the  menu, select **View**.

To create a storage volume using the CLI, use the `opc compute storage-volume add` command. To attach a storage volume to an instance, you must add a storage attachment object, by using the `opc compute storage-attachment add` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a storage volume using the API, use the `POST /storage/volume/` method. To attach a storage volume to an instance, you must add a storage attachment object, by using the `POST /storage/attachment/` method. For more information about these API methods, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After creating a storage volume, you must attach the storage volume to an instance and then mount the storage volume on the instance. See [Attaching a Storage Volume to an Instance](#) and [Mounting and Unmounting a Storage Volume](#).

## Creating a Bootable Storage Volume

A **storage volume** is a virtual disk that provides persistent block storage space for instances in Compute Classic. While creating a storage volume, you can associate it with a machine image and later use this storage volume as the boot disk for an instance. When you boot an instance from such a storage volume, any changes you make to the boot disk aren't lost when the instance is deleted and re-created..

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Click **Create Storage Volume**.
4. Select or enter the required information:
  - Enter a name for the storage volume. Note this name. You'll need it later to search for the storage volume on the Storage page.

Pick a name that you can use later to quickly identify the key characteristics of the storage volume. For example, consider a name such as `boot-OL66-20G` for a bootable storage volume with an Oracle Linux 6.6 machine image on a 20-GB disk).

- Select a machine image in the **Boot Image** field.

If you select a machine image with a large disk size, it may take a while for the storage volume to be created.

- Enter the size, in GB, of the storage volume. The allowed range is 1 GB to 2 TB.

The size you enter must be at least 5% higher than the boot image disk size.

 **Note:**

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

- Select a storage property.

Based on your latency and IOPS requirements, select one of the following storage properties.

Storage Property	Latency	Throughput
storage/default	Standard	Standard
storage/latency	Low	High
storage/ssd/gpl	Lowest	Highest

 **Note:**

SSD storage volumes aren't available in all sites.

The web console might show other storage properties. Don't select any of them.


- Enter a description for the storage volume.

5. Click **Create**.

The Storage page is displayed.

While the new storage volume is being created, the **Status** field for the storage volume shows **Initializing**.

When the storage volume is ready, the **Status** field changes to **Online**. You can then specify this storage volume as the boot disk while creating an instance.

To view details of the new storage volume, search for it using the name you noted earlier. From the  menu, select **View**.

To create a storage volume using the CLI, use the `opc compute storage-volume add` command. To attach a storage volume to an instance, you must add a storage attachment object, by using the `opc compute storage-attachment add` command. For help with these commands, run each command with the `-h` option. For the instructions


to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create a storage volume using the API, use the `POST /storage/volume/` method. To attach a storage volume to an instance, you must add a storage attachment object, by using the `POST /storage/attachment/` method. For more information about these API methods, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Attaching a Storage Volume to an Instance

You can provide or increase block storage capacity for an instance by attaching storage volumes.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Identify the storage volume that you want to attach. From the  menu, select **Attach Instance**.
4. Select the instance to which you want to attach the volume.
5. The **Attach as Disk #** field is filled automatically with the next available index at which the volume can be attached. You can leave this field at the automatically selected disk number or enter a higher number up to 10.

The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on.

Make a note of the disk number. You'll need it later when you mount the storage volume on the instance.

6. Click **Attach**.

You can also attach a storage volume to a running instance from the **Instances** page. See [Attaching a Storage Volume to an Instance](#).

To attach a storage volume to a running instance using the CLI, use the `opc compute storage-attachment add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To attach a storage volume to a running instance using the API, use the `POST /storage/attachment/` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After attaching a storage volume to an instance, to access the block storage, you must mount the storage volume on your instance. See [Mounting and Unmounting a Storage Volume](#).

## Viewing Details of a Storage Volume

You can use the web console to view details of a storage volume, such as the status, size, and the instance to which it is attached.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.

All storage volumes are displayed, along with information about each storage volume.

### Tip:

You can filter the list of storage volumes according to their category or status. To view storage volumes with a specific status (such as online, offline, or attached), click the **Show** menu and select the appropriate filter. To view storage volumes of a specific category (such as IaaS, PaaS, or Personal), click the **Category** menu and select the appropriate filter.

3. Go to the storage volume that you want to view. From the  menu, select **View**.

To view the details of a storage volume using the CLI, use the `opc compute storage-volume get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view the details of a storage volume using the API, use the `GET /storage/volume/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Mounting and Unmounting a Storage Volume

### Topics

- [Mounting a Storage Volume on a Linux Instance](#)
- [Unmounting a Storage Volume from a Linux Instance](#)
- [Mounting a Storage Volume on an Oracle Solaris Instance](#)
- [Unmounting a Storage Volume from an Oracle Solaris Instance](#)
- [Mounting a Storage Volume on a Windows Instance](#)
- [Unmounting a Storage Volume from a Windows Instance](#)

## Mounting a Storage Volume on a Linux Instance

To access a storage volume, you must attach it to your instance and mount it.

For the steps to mount a volume on a Windows instance, see [Mounting a Storage Volume on a Windows Instance](#).

For the steps to mount a volume on an Oracle Solaris instance, see [Mounting a Storage Volume on an Oracle Solaris Instance](#).

### Note:

When an instance is deleted and re-created or shut down and restarted, storage volumes that were attached manually (that is, not attached automatically through the orchestration that was used to create the instance) must be attached again.

When an instance that's set up to boot from a nonpersistent boot disk is re-created, all the storage volumes attached to the instance must be mounted again.

### Prerequisites

- You have created the storage volume and attached it to your instance. See [Attaching a Storage Volume to an Instance](#).
- You know the disk number of the storage volume that you want to mount. See [Viewing Details of a Storage Volume](#).

### Procedure

1. Log in to the instance.
2. List the devices available on your instance:

```
ls /dev/xvd*
```

Device names start from `/dev/xvdb` and are determined by the index number that you assigned when you attached the storage volumes. For example, if you attached a storage volume at index 1, the volume gets the device name, `/dev/xvdb`. The storage volume at index 2 would be `/dev/xvdc`, the storage volume at index 3 would be `/dev/xvdd`, and so on.

3. Identify the device name corresponding to the disk number that you want to mount.

For example, if you want to mount the storage volume that you had attached at index 3, the device name would be `/dev/xvdd`.

4. When mounting a storage volume for the first time, after formatting the storage volume, use a tool such as `mkfs` to create a file system on the storage volume. For example, to create an `ext3` file system on `/dev/xvdd`, run the following command:

```
sudo mkfs -t ext3 /dev/xvdd
```



 **Note:**

If the Extended File System utilities aren't available on your instance, a message such as the following is displayed:

```
mkfs.ext3: No such file or directory
```

To install the Extended File System utilities, run the following command:

```
sudo yum install e4fsprogs
```

5. Create a mount point on your instance. For example, to create the mount point `/mnt/store`, run the following command:

```
sudo mkdir /mnt/store
```

6. Mount the storage volume on the mount point that you created on your instance. For example, to mount the device `/dev/xvdd` at the `/mnt/store` directory, run the following command:

```
sudo mount /dev/xvdd /mnt/store
```

If you prefer, you can specify the disk UUID instead of the device name in the mount command. To find out the UUID of the disks attached to your instance, run the `blkid` command.

7. To make the mount persistent across instance restarts, edit the `/etc/fstab` file and add the mount as an entry in that file.

 **Note:**

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any mount points that you defined are lost. You must create the mount points again.

## Unmounting a Storage Volume from a Linux Instance

To detach a storage volume from your instance, or to delete the instance that a storage volume is attached to, you must first unmount the storage volume.

 **Note:**

For the steps to unmount a volume from a Windows instance, see [Unmounting a Storage Volume from a Windows Instance](#).

For the steps to unmount a volume from an Oracle Solaris instance, see [Unmounting a Storage Volume from an Oracle Solaris Instance](#).

To unmount a storage volume from a Linux instance:

1. Identify the disk number of the storage volume that you want to unmount. See [Viewing Details of a Storage Volume](#).

2. Log in to the instance. See [Accessing an Oracle Linux Instance Using SSH](#).
3. List the devices available on your instance and their mount points:

```
sudo df -hT

Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/xvdb2      ext4      16G   2.9G  12G  20% /
tmpfs           tmpfs     3.7G   0    3.7G  0% /dev/shm
/dev/xvdb1      ext4      194M   90M   94M  49% /boot
/dev/mapper/vg_binaries-lv_tools
                ext4      9.9G  156M   9.2G   2% /u01/app/oracle/tools
/dev/mapper/vg_backup-lv_backup
                ext4     20G   4.0G   15G  21% /u01/data/backup
/dev/mapper/vg_domains-lv_domains
                ext4      9.9G   1.2G   8.3G  12% /u01/data/domains
/dev/mapper/vg_binaries-lv_mw
                ext4      9.9G   2.0G   7.4G  21% /u01/app/oracle/middleware
/dev/mapper/vg_binaries-lv_jdk
                ext4      2.0G   334M   1.6G  18% /u01/jdk
```

Device names start from `/dev/xvdb` and are determined by the index number that you assigned when you attached the storage volumes. For example, if you attached a storage volume at index 1, then the volume gets the device name, `/dev/xvdb`. The storage volume at index 2 would be `/dev/xvdc`, the storage volume at index 3 would be `/dev/xvdd`, and so on.

 **Note:**

For an instance that's set up to boot from a nonpersistent boot disk, `/dev/xvda` is used for the boot disk.

4. Identify the device name corresponding to the disk number that you want to unmount, and note the mount point for that device.

For example, to unmount the storage volume that is attached at index 3, you must unmount `/dev/xvdd`.

5. Run the `umount` command.

```
sudo umount mount_point
```

For example, to unmount the device mounted at `/mnt/store`, run the following command:

```
sudo umount /mnt/store
```

6. If you had defined this mount point in `/etc/fstab` file, then edit `/etc/fstab` and remove the mount.

If you no longer need the volume that you just unmounted, then you can detach it from the instance and delete it. See [Detaching a Storage Volume from an Instance](#) and [Deleting a Storage Volume](#).

## Mounting a Storage Volume on an Oracle Solaris Instance



This topic does not apply to Oracle Cloud at Customer.

After attaching a storage volume to your Oracle Solaris instance, to be able to access the new disk, you must mount it. You do this by creating a ZFS storage pool using the disk that you want to mount.

 **Note:**

When an instance is deleted and re-created or shut down and restarted, storage volumes that were attached manually (that is, not attached automatically through the orchestration that was used to create the instance) must be attached again.

When an instance that's set up to boot from a nonpersistent boot disk is re-created, all the storage volumes attached to the instance must be mounted again.

The steps to mount a storage volume on an Oracle Solaris instance vary depending on whether a ZFS storage pool exists for the volume.

- If the storage volume that you want to mount was attached previously to any Oracle Solaris instance, or if you're not sure about this, then start with the steps in [Importing a ZFS Storage Pool](#).
- If the storage volume that you want to mount has just been created, or if you're sure that it has never been attached previously to any Oracle Solaris instance, then proceed to [Creating a ZFS Pool](#).

### Importing a ZFS Storage Pool

Complete the steps in this section if the storage volume that you want to mount was attached previously to any Oracle Solaris instance or if you're not sure about that. Otherwise, go to [Creating a ZFS Pool](#).

1. Identify and make a note of the disk number of the storage volume that you want to mount.  
See [Viewing Details of a Storage Volume](#).
2. Log in to the instance on which you want to mount the storage volume.  
See [Accessing an Oracle Solaris Instance Using SSH](#).
3. Assume the `root` role, by running the following command:

```
su -
```

When prompted, enter the `root` password.

 **Note:**

If this is the first time that you're assuming the `root` role on the instance, then a prompt to change the password is displayed. Change the password as prompted and then proceed.

4. Run the following command:  

```
zpool import
```
5. Examine the output of the command:

- If the command returns the message `no pools available to import`, then proceed to [Creating a ZFS Pool](#).
- If the command lists one or more pools, then pick the pool that you want to import.

Here's an example of the output of the `zpool import` command:

```
pool: mypool2
  id: 14352758040898370875
state: ONLINE
action: The pool can be imported using its name or numeric
identifier.
config:
```

```
  mypool2    ONLINE
    c2t2d0   ONLINE
```

```
pool: mypool3
  id: 1124470769081803325
state: ONLINE
action: The pool can be imported using its name or numeric
identifier.
config:
```

```
  mypool3    ONLINE
    c2t3d0   ONLINE
```

In this example, two pools are available for importing: `mypool2` (for disk `c2t2d0`) and `mypool3` (for disk `c2t3d0`).

In the disk names—that is, `c2t2d0`, `c2t3d0`, and so on—look at the `t1`, `t2`, `t3`, ... number. This number, technically known as the *target number*, matches the index that was specified when the volume was attached to the Oracle Solaris instance. For example, `c2t3d0` is the disk that's attached to the instance at index 3.

6. Identify the disk that you want to mount, and note its pool name. For example, if you want to mount the storage volume that's attached to the instance at index 3, then the disk in this example would be `c2t3d0` in `mypool3`.

 **Note:**

If the index number of the storage volume that you want to mount doesn't match the target number of any of the disks listed by the `zpool import` command, then you must create a ZFS storage pool. See [Creating a ZFS Pool](#).

7. Import the ZFS pool that you noted earlier, by running the `zpool import` command, as shown in the following example:  

```
zpool import mypool3
```

The storage volume and the ZFS file systems defined in it, if any, are now mounted on the instance.
8. Verify that the volume is mounted. See [Verifying that the Storage Volume is Mounted](#).

## Creating a ZFS Pool

Complete the steps in this section if the storage volume that you want to mount has just been created, or if you're sure that it has never been attached previously to any Oracle Solaris instance. Otherwise, see [Importing a ZFS Storage Pool](#).

1. Identify and make a note of the disk number of the storage volume that you want to mount.  
See [Viewing Details of a Storage Volume](#).
2. Log in to the instance on which you want to mount the storage volume.  
See [Accessing an Oracle Solaris Instance Using SSH](#).
3. Assume the `root` role, by running the following command: `su -`  
When prompted, enter the `root` password.

### Note:

If this is the first time that you're assuming the `root` role on the instance, then a prompt to change the password is displayed. Change the password as prompted and then proceed.

4. Find out the names of the disks attached to your instance, by running the `format` command:

```
format
```

The following is an example of the output of this command:

```
Searching for disks...done
```

```
AVAILABLE DISK SELECTIONS:
```

- ```
0. c2t1d0 <Unknown-Unknown-0001-34.00GB>
   /xpvd/xdf@51728
1. c2t2d0 <Unknown-Unknown-0001-10.00GB>
   /xpvd/xdf@51744
2. c2t3d0 <Unknown-Unknown-0001 cyl 1024 alt 0 hd 64 sec 32>
   /xpvd/xdf@51872
```

```
Specify disk (enter its number):
```

In this example, three disks are attached to the instance: `c2t1d0`, `c2t2d0` and `c2t3d0`.

In the disk names—that is, `c2t2d0`, `c2t3d0`, and so on—look at the `t1`, `t2`, `t3`, ... number. This number, technically known as the *target number*, matches the index that was specified when the volume was attached to the Oracle Solaris instance. For example, `c2t3d0` is the disk that's attached to the instance at index 3.

5. Using the storage volume index number that you noted earlier, identify and make a note of the disk name of the storage volume that you want to mount.  
For example, if you want to mount the storage volume that was attached at index 3, then the disk name in this example would be `c2t3d0`.
6. Kill the `format` process by pressing `Ctrl+c`.
7. Create a ZFS storage pool for the disk that you want to mount:

**Command syntax:** `zpool create pool_name disk_file_name`

**Command example:** `zpool create mypool3 c2t3d0`

The storage volume is now mounted on the instance. By default, the mount point is the name of the pool.

8. If required, create ZFS file systems in the new ZFS storage pool.

**Command syntax:** `zfs create pool_name/filesystem_name`

**Command example:** `zfs create mypool3/myfs1`

The ZFS file systems are mounted automatically. By default, the mount point of each file system is its name.

9. To give the `opc` user access to the ZFS storage pool and its filesystems, make the `opc` user the owner of the mount by using the `chown` command, as shown in the following example:

```
chown -R opc /mypool
```

10. Verify that the volume is mounted.  
See [Verifying that the Storage Volume is Mounted](#).

### Verifying that the Storage Volume is Mounted

To verify that the ZFS pool and file systems are mounted, run the `zfs mount` command on the instance.

The following is an example of the output of the `zfs mount` command:

```

rpool/ROOT/solaris           /
rpool/ROOT/solaris/var      /var
rpool/VARSHARE               /var/share
rpool/export                 /export
rpool/export/home           /export/home
rpool/export/home/opc       /export/home/opc
rpool                        /rpool
rpool/VARSHARE/zones        /system/zones
rpool/VARSHARE/pkg          /var/share/pkg
rpool/VARSHARE/pkg/repositories /var/share/pkg/repositories
mypool3                     /mypool3
mypool3/myfs1              /mypool3/myfs1

```

In this example,

- The `rpool` entries are for the `root` pool that contains the boot disk of the instance.
- `mypool3` is the ZFS storage pool of the storage volume that you mounted. It is mounted at `/mypool3`.
- `mypool3/myfs1` is a filesystem in the ZFS storage pool, and it's mounted at `/mypool3/myfs1`.

 **See Also:**

- [Unmounting a Storage Volume from an Oracle Solaris Instance](#)
- [Managing ZFS File Systems in Oracle Solaris 11.3](#)

## Unmounting a Storage Volume from an Oracle Solaris Instance



This topic does not apply to Oracle Cloud at Customer.

To detach a storage volume from your instance, or to delete the instance that a storage volume is attached to, you must first unmount the storage volume.

To unmount a storage volume from an Oracle Solaris instance:

1. Identify and make a note of the disk number of the storage volume that you want to unmount. See [Viewing Details of a Storage Volume](#).
2. Log in to the instance. See [Accessing an Oracle Solaris Instance Using SSH](#).
3. Assume the `root` role, by running the following command:

```
su -
```

When prompted, enter the `root` password.

4. Find out the names of the disks mounted on your instance and the ZFS pool to which each disk belongs, by running the following command:

```
zpool status
```

The following is an example of the output of this command:

```
pool: mypool
state: ONLINE
scan: none requested
config:

    NAME      STATE    READ WRITE CKSUM
    mypool    ONLINE      0     0     0
        c2t2d0 ONLINE      0     0     0

errors: No known data errors

pool: rpool
state: ONLINE
scan: none requested
config:

    NAME      STATE    READ WRITE CKSUM
    rpool    ONLINE      0     0     0
        c2t1d0 ONLINE      0     0     0

errors: No known data errors
```

In this example, two disks are mounted on the instance: `c2t1d0` (in `rpool`) and `c2t2d0` (in `mypool`)

Focus on the `t1`, `t2`, ... number in the disk file names. This number corresponds to the index that was specified while attaching the storage volume to the instance.

5. Identify and make a note of the disk file name of the storage volume that you want to unmount.

For example, if you want to unmount the storage volume that was attached at index 2, then the disk file name in this example would be `c2t2d0`.

 **Caution:**

`rpool` is the pool that contains the boot disk. Do NOT unmount it.

6. Export the ZFS pool that contains the disk that you want to unmount:

**Command syntax:** `zpool export pool_name`

**Command example:** `zpool export mypool`

This command unmounts the ZFS pool and any file systems in it. To verify that the pool has been exported, run the `zpool import` command. The output shows that the pool that you exported is available for importing, as shown in the following example:

```
pool: mypool
  id: 1124470769081803325
 state: ONLINE
action: The pool can be imported using its name or numeric identifier.
config:
```

```
  mypool    ONLINE
    c2t2d0  ONLINE
```

If you no longer need the volume that you just unmounted, then you can detach it from the instance and delete it. See [Detaching a Storage Volume from an Instance](#) and [Deleting a Storage Volume](#).

To mount the volume again, run the `zpool import` command, as shown in the following example:

```
zpool import mypool
```

 **See Also:**

- [Exporting a ZFS Storage Pool](#) in *Managing ZFS File Systems in Oracle Solaris 11.3*.
- [Importing a ZFS Storage Pool](#) in *Managing ZFS File Systems in Oracle Solaris 11.3*.



## Mounting a Storage Volume on a Windows Instance

To access a storage volume from a Windows instance, you must attach the volume to the instance and mount it.

 **Note:**

When an instance is deleted and re-created or shut down and restarted, storage volumes that were attached manually (that is, not attached automatically through the orchestration that was used to create the instance) must be attached again.

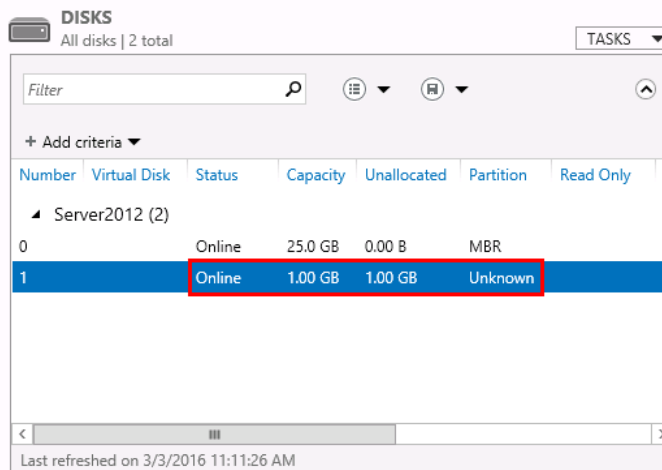
When an instance that's set up to boot from a nonpersistent boot disk is re-created, all the storage volumes attached to the instance must be mounted again.

After attaching a storage volume to a Windows instance (see [Attaching a Storage Volume to an Instance](#)), mount it as follows:

1. Log in to the Windows instance.  
See [Accessing a Windows Instance Using RDP](#).
2. From the **Start** menu, select **Server Manager**.
3. Navigate to **File and Storage Services**, and from there to **Volumes**, and then **Disks**.

The storage volumes that are attached to the instance are listed as disks. For newly attached disks, the **Partition** type would be Unknown and the **Unallocated** capacity would be equal to the total size of the disks.

See the following example:

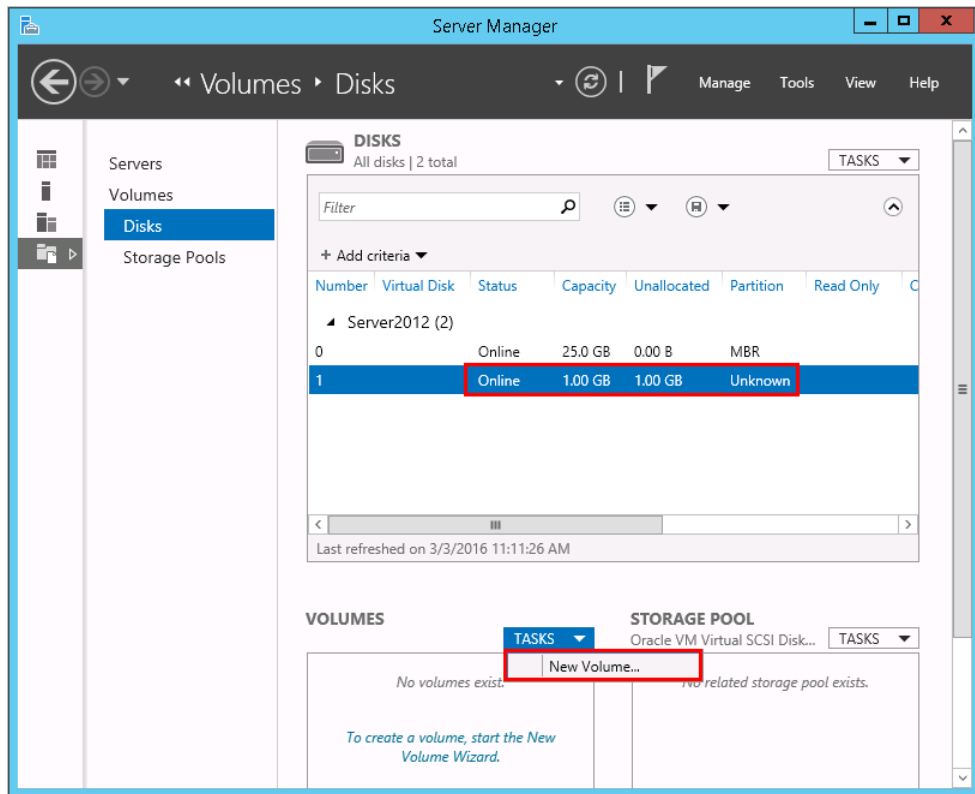


| Number         | Virtual Disk | Status | Capacity | Unallocated | Partition | Read Only |
|----------------|--------------|--------|----------|-------------|-----------|-----------|
| Server2012 (2) |              |        |          |             |           |           |
| 0              |              | Online | 25.0 GB  | 0.00 B      | MBR       |           |
| 1              |              | Online | 1.00 GB  | 1.00 GB     | Unknown   |           |

Last refreshed on 3/3/2016 11:11:26 AM

4. Select the disk that you want to mount.
5. If the **Status** of the disk is Offline, right-click and select **Bring Online**.

6. At the confirmation prompt, click **Yes**.
7. Wait until the status changes to Online.  
Refresh the page after a few seconds.  
You can now create partitions, format them, and assign each partition to a drive letter or folder.
8. In the **Volumes** pane, click **Tasks** and select **New Volume**, as shown in the following example:



**Note:**

Don't confuse the term *volume* that you see in Windows with the concept of *storage volumes* in Compute Classic.

A storage volume in Compute Classic is a virtual disk that you can attach to an instance. In the context of Windows, a volume is essentially a partition on a disk that's attached to a server. You can create multiple partitions on each storage volume that you attach to your Windows instance.

9. Follow the instructions in the **New Volume Wizard** to complete creation of the partition.  
After the partition is created, it's displayed in the **Volumes** pane.
10. Create more partitions, if required.

The new partitions are now available at the drive letters that you assigned while partitioning the disk, as shown in the following example:

The screenshot shows the Windows Server 2012 Disk Management console. The 'DISKS' section shows two disks for 'Server2012 (2)'. Disk 0 is a 25.0 GB disk with an MBR partition. Disk 1 is a 1.00 GB disk with a GPT partition. The 'VOLUMES' section shows two volumes for 'Server2012 (2)'. Volume D: is a 400 MB Fixed volume with 384 MB of free space. Volume E: is a 400 MB Fixed volume with 384 MB of free space. The 'STORAGE POOL' section shows 'No related storage pool exists.'

| Number | Virtual Disk | Status | Capacity | Unallocated | Partition | Read Only | Clustered | Subsystem | Bus Type |
|--------|--------------|--------|----------|-------------|-----------|-----------|-----------|-----------|----------|
| 0      |              | Online | 25.0 GB  | 0.00 B      | MBR       |           |           |           | SCSI     |
| 1      |              | Online | 1.00 GB  | 192 MB      | GPT       |           |           |           | SCSI     |

| Volume | Status | Provisioning | Capacity | Free Space | Deduplication |
|--------|--------|--------------|----------|------------|---------------|
| D:     | Fixed  |              | 400 MB   | 384 MB     |               |
| E:     | Fixed  |              | 400 MB   | 384 MB     |               |

In this example, on a 1-GB storage volume attached to a Windows Server 2012 Standard instance, two 400-MB partitions were created, formatted, and assigned to the drives D and E.

For detailed instructions for managing disks & partitions (changing the drive assignment, changing the file system type, extending the partition or deleting it), see the Windows Server documentation.

## Unmounting a Storage Volume from a Windows Instance

When you no longer need a storage volume for a Windows instance, you can take the disk offline and detach it from the instance.

To unmount a storage volume from a Windows instance:

1. Log in to the Windows instance.  
See [Accessing a Windows Instance Using RDP](#).
2. From the **Start** menu, select **Server Manager**.
3. Navigate to **File and Storage Services**, and from there to **Volumes**, and then **Disks**.

The storage volumes that are attached to the instance are listed as disks.

4. Select the disk that you want to unmount.
5. Right-click and select **Take Offline**.

Wait for a few seconds, until the **Status** of the disk changes to Offline.

 **Note:**

The partitions and data on the disk are intact. You can either bring the disk online later on the same instance, or detach it from this instance and attach it to another instance.

For detailed instructions for managing disks and partitions, see the Windows Server documentation.

If you no longer need the volume that you just unmounted, then you can detach it from the instance and delete it. See [Detaching a Storage Volume from an Instance](#) and [Deleting a Storage Volume](#).

## Increasing the Size of a Storage Volume

After creating a storage volume, you can increase the size of a storage volume when the storage volume is online, even if the storage volume is already attached to an instance. The procedure to increase the storage volume varies depending on whether it has been created directly in the **Storage** page of the Compute Classic console or by using an orchestration.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

 **Note:**

If you increase the size of a storage volume that was created using an orchestration, then if you stop the orchestration and start it again later, the storage volume will be destroyed and re-created with the size originally specified in the orchestration.


 **Note:**

If you increase the size of a storage volume that's attached to an Oracle Solaris or a Windows instance, you'll need to reboot the instance to make the additional storage available on the OS.

**Prerequisites**

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

To increase the size of a storage volume that you have created directly in the **Storage** page of the Compute Classic console:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Go to the storage volume that you want to increase the size of. From the  menu, select **Update**.
4. In the Update Storage Volume dialog box, enter the required size in GB and the click **Update**.

 **Note:**

You can only increase the size of the storage volume, you can't reduce it. So you must enter a value larger than the current value.

5. Make the increased storage available on the volume that is attached and mounted on an instance.
  - On an Oracle Linux instance, the increased size of the storage volume gets updated on the instance automatically. You can use the `fdisk` command to verify the updated size:
    - a. Log in to the instance. See [Accessing an Oracle Linux Instance Using SSH](#).
    - b. Run `sudo su`.
    - c. Run the `ls` command to list devices on your instance.

```
ls /dev/xvd*
```

Device names are determined by the index number that you assigned when you attached the storage volumes. For example, if you attached a storage volume at index 1, the volume gets the device name, `/dev/xvdb`. The storage volume at index 2 would be `/dev/xvdc`, the storage volume at index 3 would be `/dev/xvdd`, and so on.

- d. Identify the device name corresponding to the disk number of the storage volume that you've updated.
- e. Run the `fdisk -l` command to verify that the size of the storage volume has been updated. Here's an example of the output of this command:

```
Disk /dev/xvdc: 4294 MB, 4294967296 bytes
255 heads, 63 sectors/track, 522 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

This output shows the size of `/dev/xvdc` as 4294 MB, which corresponds to a 4 GB storage volume.

- On an Oracle Solaris instance: (Not available on Oracle Cloud at Customer)
  - a. Log in to the instance. See [Accessing an Oracle Solaris Instance Using SSH](#).
  - b. Run `su -`.
  - c. Run `zpool status` command to identify the storage pool that you want to expand. Here's an example of the output of this command:

```
opc@d7becf:~$ zpool status
pool: mypool3
state: ONLINE
scan: none requested
config:

        NAME      STATE    READ WRITE CKSUM
        mypool3    ONLINE   0     0     0
            c2t2d0  ONLINE   0     0     0

errors: No known data errors

pool: rpool
state: ONLINE
scan: none requested
config:

        NAME      STATE    READ WRITE CKSUM
        rpool      ONLINE   0     0     0
            c2t1d0  ONLINE   0     0     0

errors: No known data errors
```

This output indicates that the disk `c2t2d0` is in storage pool `mypool3`. Here the number `c2t2d0`, technically known as the *target number*, matches the index that was specified when the volume was attached to the Oracle Solaris instance. For example, `c2t2d0` is the disk that's attached to the instance at index 2.

- d. Set `autoexpand` to `on` for the ZFS storage pool.

```
zpool set autoexpand=on mypool3
```

- e. Go to the Compute Classic web console and restart the instance. See [Rebooting an Instance](#).
- f. Log in to the instance.
- g. Run the `df -k mypool3` command to view the size of the storage volume before resizing. Here's an example of the output of this command for the storage pool `mypool3`:

```
Filesystem          1024-blocks      Used  Available
Capacity Mounted on
mypool3              6177024          32    6176880
1% /mypool3
```

This output shows the size of `mypool3` as 6177024 blocks, which corresponds to a 6 GB storage volume.

- h. Run `su -`.
- i. Export and import the ZFS storage pool.

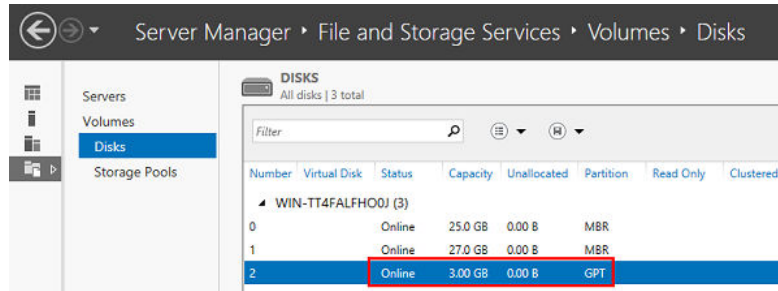
```
zpool export mypool3
zpool import mypool3
```

- j. Run the `df -k` command again to verify that the size of the storage volume has been updated. Here's the output of this command for `mypool3` after expanding the volume from 6 GB to 9 GB:

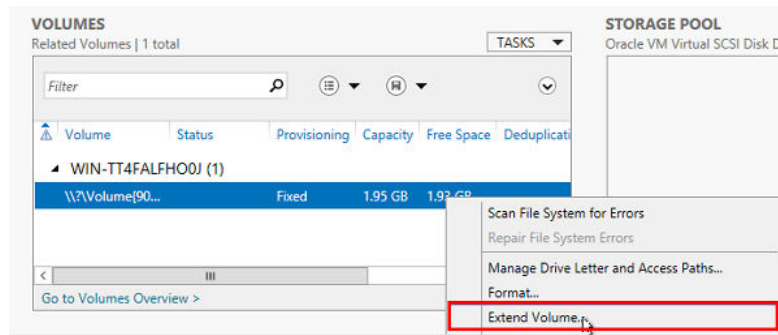
```
Filesystem          1024-blocks      Used  Available
Capacity Mounted on
mypool3              9273600          32    9273453
1% /mypool3
```

This output shows the size of `mypool3` as 9273600 blocks, which corresponds to a 9 GB storage volume.

- On a Windows instance:
  - a. Go to the Compute Classic web console and restart the instance. See [Rebooting an Instance](#).
  - b. Log in to the instance. See [Accessing a Windows Instance Using RDP](#).
  - c. If you have a Windows Server 2012 R2 instance, from the **Start** menu, select **Server Manager**.  
  
If you have a Windows Server 2008 R2 instance, from the **Start** menu, select **All Programs** and **Administrative Tools**, and then select **Server Manager**.
  - d. Navigate to **File and Storage Services**, and from there to **Volumes**, and then **Disks**. The resized storage volume shows the updated size.

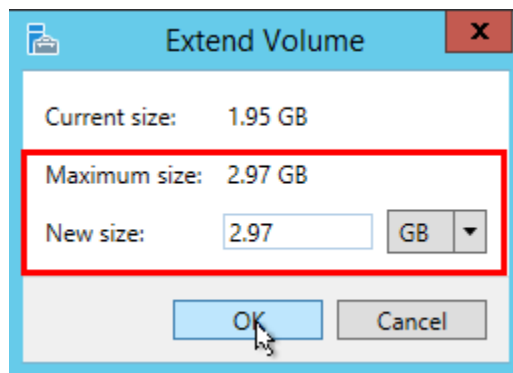


- e. To expand the volume on your Windows instance, select the volume in the Volumes pane. Click **Tasks** and select **Extend Volume**.



- f. The Extend Volume dialog box shows the updated size of the storage volume as the maximum size. Specify the required size in the **New size** field and click **OK**.

The volume is expanded to the specified size.



To update a storage volume using the CLI, use the `opc compute storage-volume update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a storage volume using the API, use the `PUT /storage/volume/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.



## Detaching a Storage Volume from an Instance

A **storage volume** is a virtual disk that provides persistent block storage space for instances in Compute Classic. When you no longer require access to a storage volume, you can unmount it and detach it from your instance.

After you detach a storage volume from an instance, you can no longer read from or write data to the storage volume, unless you attach it to any instance.

### Note:


You can't detach or delete a storage volume that was attached while creating an instance.

If you're sure that a storage volume is no longer required, then back up the data elsewhere and delete the storage volume.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- Ensure that you've unmounted the storage volume that you want to detach. See [Unmounting a Storage Volume from a Linux Instance](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Go to the storage volume that you want to detach. From the  menu, select **Detach Instance**.

You can also detach a storage volume from the Instances page. See [Detaching a Storage Volume from an Instance](#).

To detach a storage volume from an instance using the CLI, you must remove a storage attachment object by using the `opc compute storage-attachment delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To detach a storage volume from an instance using the API, you must remove a storage attachment object, by using the `DELETE /storage/attachment/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

# Deleting a Storage Volume

If you delete a storage volume, all the data and applications that were saved on that storage volume are lost. Delete a storage volume only when you're sure that you no longer need any of the data that's stored on that volume.

## Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- Ensure that the storage volume that you want to delete isn't attached to any instance. See [Detaching a Storage Volume from an Instance](#).
- Ensure that there are no colocated snapshots of the storage volume that you want to delete. See [Listing Storage Volume Snapshots](#).


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

## Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.
3. Go to the storage volume that you want to delete. From the  menu, select **Delete**.

To delete a storage volume using the CLI, use the `opc compute storage-volume delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete a storage volume using the API, use the `DELETE /storage/volume/name` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

# Backing Up and Restoring Storage Volumes Using Snapshots

## Topics

- [About Storage Volume Snapshots](#)
- [Creating a Storage Volume Snapshot](#)
- [Listing Storage Volume Snapshots](#)
- [Restoring a Storage Volume from a Colocated Snapshot](#)
- [Restoring a Storage Volume from a Remote Snapshot](#)
- [Deleting a Storage Volume Snapshot](#)

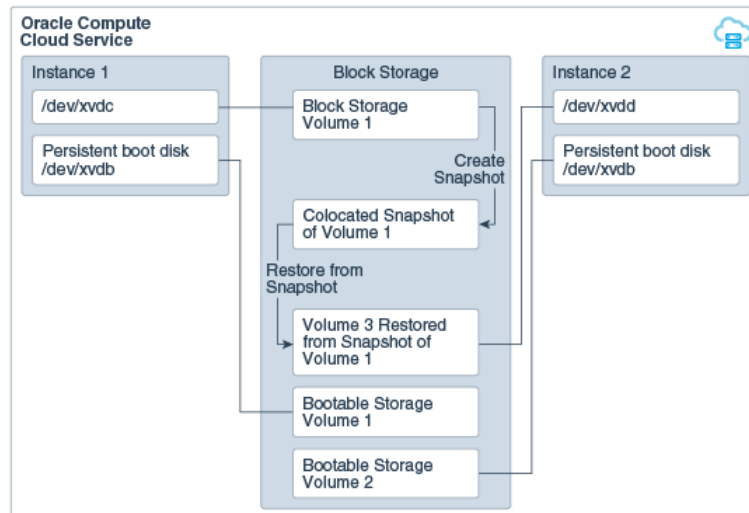
## About Storage Volume Snapshots

Creating a snapshot of a storage volume enables you to capture all the data stored on the storage volume. You can retain snapshots as a backup, or use them to create new, identical storage volumes.

If you take multiple snapshots of a storage volume, the first snapshot captures all the data stored on the storage volume at that point in time. Successive snapshots are incremental backups, which capture new or modified data since the previous snapshot. This reduces the storage requirement for your snapshots. When there are ten incremental backups for a specified volume, the next backup is a full backup, which once again captures all the data that is on the storage volume when the snapshot is created. You can restore a storage volume from any snapshot and all the data in the storage volume at that time is restored.

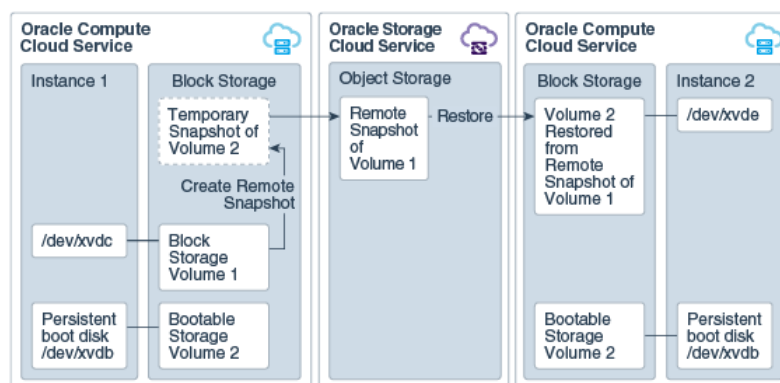
You can create either colocated or remote storage volume snapshots.

- **Colocated snapshots:** Snapshots are stored in the same physical location as the original storage volume. Colocated snapshots and volumes from colocated snapshots can be created very quickly. Colocated snapshots are useful for quickly cloning storage volumes within a site. However, you can't restore volumes across sites using colocated snapshots.



The graphic shows the process for creating a colocated snapshot of a storage volume mounted on one instance and restoring it to a storage volume mounted on another instance in the same site.

- Remote snapshots:** Snapshots aren't stored in the same location as the original storage volume. Instead, they are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. Remote snapshots are useful if your domain spans multiple sites. With remote snapshots, you can create a snapshot in one site, then switch to another site and create a copy of the storage volume on that site. Creating a remote snapshot and restoring a storage volume from a remote snapshot can take more time than for colocated snapshots, as data is written to and from the Oracle Cloud Infrastructure Object Storage Classic instance.



The graphic shows the process for creating a remote snapshot of a storage volume mounted on a Compute Classic instance. The snapshot is stored in Oracle Cloud Infrastructure Object Storage Classic and can be restored to a storage volume mounted on another Compute Classic instance, either in the same site or in another site.

## Creating a Storage Volume Snapshot

Creating a snapshot of a storage volume enables you to capture all the data that is currently stored on the storage volume.

You can create a snapshot of a storage volume when it is either attached to an instance or detached. If the storage volume is attached to an instance, then only data that has already been written to the storage volume will be captured in the snapshot. Data that is cached by the application or the operating system will be excluded from the snapshot.

### Tip:

To create a snapshot of a bootable storage volume that is currently being used by an instance, it is recommended that:

- if you have created the instance using orchestration v2, suspend the orchestration before you create the snapshot.
- if you have created the instance using orchestration v1, terminate the instance orchestration to delete the instance before you create the snapshot. Ensure that you stop only the instance orchestration and not the master orchestration or the orchestration that creates storage volumes. That way, only your instance is deleted and re-created and storage volumes or other resources defined in other orchestrations are not deleted.

This ensures that all data is written to the storage volume and no further data can be written to the disk while taking the snapshot. Data on the bootable storage volume is not deleted because the data is stored on a persistent boot disk. You can create the instance again later, after the snapshot is created.

### Prerequisites


- When you create a remote storage snapshot, the snapshot is stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. Ensure that you've set a replication policy for your Oracle Cloud Infrastructure Object Storage Classic instance. See *Selecting a Replication Policy for Oracle Cloud Infrastructure Object Storage Classic* in *Using Oracle Cloud Infrastructure Object Storage Classic*.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Procedure

To create a storage volume snapshot:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.

The Storage Volumes page is displayed.

3. Go to the storage volume that you want to create a snapshot of. From the  menu, select **Create Snapshot**.
4. In the Create Storage Snapshot dialog box, enter the required information:
  - **Name:** Enter a name for the snapshot.
  - **Colocated:** Specify if you want to create a colocated snapshot. Colocated snapshots are stored in the same physical location as the original storage volume. Colocated snapshots and clones from colocated snapshots are created very quickly. However, you can't use a colocated snapshot across sites. Colocated snapshots as well as clones created from those snapshots are always on the same site as the original storage volume that you want to clone. If you want to back up a storage volume and restore it on another site in your account, don't select this option. If you don't select this option, a remote snapshot is created.
  - **Description:** Enter a description for this snapshot, if required.
  - **Tags:** Enter tags to help identify your snapshot, if required.
5. Click **Create**.

A storage volume snapshot is generated.
6. To see a list of storage snapshots, click **Storage Snapshots** in the left pane. See [Listing Storage Volume Snapshots](#).

To create a storage volume snapshot using the CLI, use the `opc compute storage-snapshot add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To create a storage volume snapshot using the API, use the `POST /storage/snapshot/` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After creating a colocated storage volume snapshot, to use this snapshot to create a storage volume, see [Restoring a Storage Volume from a Colocated Snapshot](#).

After creating a remote storage volume snapshot, to use this snapshot to restore a storage volume, see [Restoring a Storage Volume from a Remote Snapshot](#).

## Listing Storage Volume Snapshots

Storage volume snapshots allow you to create a new storage volume from an existing storage volume. You can take multiple snapshots of a storage volume and create multiple storage volumes from a snapshot. After you've created a snapshot, you can see a list of snapshots and view information about each snapshot on the Storage Snapshots page.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Storage** tab.

The Storage Volumes page is displayed.

3. Click the **Storage Snapshots** tab in the left pane.

The Storage Snapshots page displays the list of storage volume snapshots, arranged alphabetically by storage volume name. If there are multiple snapshots of a storage volume, the most recent snapshot is at the top.

The icon associated with each snapshot listed on this page also shows the status of the snapshot. While a snapshot is being created, the status shows as **Initializing**. Colocated snapshots are completed quickly, but remote snapshots can take longer to create and the status indicates the percentage of data copied.

If you create multiple remote snapshots at the same time, the first four snapshots go to the initializing state, and the later snapshots remain in the queued state until the previous snapshots are completed. When a snapshot is completed, its status changes to **Completed**, and you can perform actions like restoring a storage volume or deleting the snapshot.

The Storage Snapshots page also displays other relevant information about your storage volume snapshots, including the date each snapshot was created, and for colocated snapshots, the new storage volumes created from each snapshot, if any.

 **Tip:**

You can filter the list of storage volume snapshots according to their category. To view storage volumes of a specific category (such as IaaS, PaaS, or Personal), click the **Category** menu and select the appropriate filter. You can also filter snapshots according to the type of snapshot. To view remote or colocated snapshots, click the **Show** menu and select the appropriate filter.

4. (Optional) If your domain spans multiple sites, you can use the **Site** menu on the Storage Snapshots page to switch to another site. You'll then be able to view snapshots from another site in your domain.

 **Note:**

The **Site** menu near the top of every page controls the site that you're logged in to. You use this option to view site resource usage and switch to another site in your domain. The **Site** menu on the Storage Snapshots page determines the list of storage snapshots displayed on this page. By default, the **Site** menu on the Storage Snapshots page is set to the current site, so you see the storage snapshots created in the site you're logged in to. To see storage snapshots created in another site, select that site from the **Site** menu on the Storage Snapshots page. This action doesn't change the site that you're logged in to.

To view a list of your storage volume snapshots using the CLI, use the `opc compute storage-snapshot list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see Preparing to Use

the Compute Classic CLI in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view a list of your storage volume snapshots using the API, use the `GET /storage/snapshot/container/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After you've created a colocated storage volume snapshot, to use this snapshot to create a clone of a storage volume, see [Restoring a Storage Volume from a Colocated Snapshot](#).

After you've created a remote storage volume snapshot, to use this snapshot to restore a storage volume, see [Restoring a Storage Volume from a Remote Snapshot](#).


## Restoring a Storage Volume from a Colocated Snapshot

You can back up and restore an existing storage volume by creating a colocated snapshot of the storage volume and using the snapshot to create a new storage volume.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- You should have already created a colocated snapshot of the storage volume that you want to clone and the process of creating the snapshot should be complete. See [Creating a Storage Volume Snapshot](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. In the **Storage** drop-down list, click **Storage Snapshots**.
4. (Optional) To see the list of colocated snapshots, from the **Show** menu, select **Colocated**.
5. Go to the snapshot that you want to create a storage volume from. From the  menu, select **Restore Volume**.
6. In the Restore Storage Volume dialog box, enter a name for the new storage volume and specify a description if required.
7. Click **Restore**.

A new storage volume is created.

To create a storage volume from a storage volume snapshot using the CLI, use the `opc compute storage-volume add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.



To create a storage volume from a storage volume snapshot using the API, use the `POST /storage/volume/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After you've created a storage volume, to view details of your storage volume, see [Viewing Details of a Storage Volume](#). To attach a storage volume to an instance, see [Attaching a Storage Volume to an Instance](#).

## Restoring a Storage Volume from a Remote Snapshot

You can back up an existing storage volume by creating a remote snapshot of the storage volume. Remote snapshots are stored in an associated Oracle Cloud Infrastructure Object Storage Classic instance. You can use the snapshot to restore the storage volume. If your domain spans multiple sites, using remote snapshots allows you to back up a storage volume in one site and restore it in another site.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- You should have already created a remote snapshot of the storage volume that you want to restore and the snapshot should be in the complete state. See [Creating a Storage Volume Snapshot](#).

### Procedure

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then ensure that you select the site where you want to create the storage volume. Also check that the site you've selected has sufficient capacity to create the required storage volume. Click **Site** near the top of the page to view resource usage across a site. If resource usage on the selected site is close to maximum, select another site.
3. Click the **Storage** tab.  
The Storage Volumes page is displayed.
4. In the **Storage** drop-down list, click **Storage Snapshots**.
5. (Optional) If your domain spans multiple sites, and if you want to restore a volume from a snapshot created on another site, you can select a site by using the **Site** menu on the Storage Snapshots page to switch to another site. You'll then be able to view the snapshots on another site in your domain and select a remote snapshot to restore from.

 **Note:**

The **Site** menu near the top of every page controls the site that you're logged in to. You use this option to view site resource usage and switch to another site in your domain. The **Site** menu on the Storage Snapshots page determines the list of storage snapshots displayed on this page. By default, the **Site** menu on the Storage Snapshots page is set to the current site, so you see the storage snapshots created in the site you're logged in to. To see storage snapshots created in another site, select that site from the **Site** menu on the Storage Snapshots page. This action doesn't change the site that you're logged in to.


 **Note:**

When you view snapshots in another site, there is no actions menu for the colocated snapshots in that site, because you can't perform any action on the colocated snapshots in another site. To use colocated snapshots that were created in another site, you must switch to the site where the colocated snapshot was created.

6. (Optional) The Storage Snapshots page displays both colocated and remote snapshots. To list only remote snapshots, from the **Show** menu, select **Remote**.

 **Note:**

If your domain spans multiple sites and you want to restore a storage volume across sites, you must use remote snapshots. You can't do this using colocated snapshots.

7. Go to the completed snapshot that you want to create a storage volume from. From the  menu, select **Restore Volume**.
8. In the Restore Storage Volume dialog box, enter a name for the new storage volume and specify a description, if required.
9. Click **Restore**.

Depending on the size of the storage volume, it can take some time to restore a storage volume from a remote snapshot. While the storage volume is being created, the status on the Storage Volume page shows **Initializing**. After the storage volume has been restored, its status changes to **Online**.

To create a storage volume from a storage volume snapshot using the CLI, use the `opc compute storage-volume add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a storage volume from a storage volume snapshot using the API, use the `POST /storage/volume/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After you've created a storage volume, to view details of your storage volume, see [Viewing Details of a Storage Volume](#). To attach a storage volume to an instance, see [Attaching a Storage Volume to an Instance](#).

## Deleting a Storage Volume Snapshot

You can restore or clone an existing storage volume by creating a snapshot of the storage volume and using the snapshot to create a new storage volume. You can create multiple snapshots of a storage volume. If a storage volume snapshot gets outdated, or if you no longer need a snapshot, you can delete it. You can delete a snapshot when it is in the initializing or online state. When you delete a remote snapshot, the snapshot file in the associated Oracle Cloud Infrastructure Object Storage Classic account is deleted.

### Note:

You can't delete a colocated snapshot if it has been used to create a new storage volume.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- If you want to delete a colocated snapshot, ensure that no storage volumes have been created using the snapshot that you want to delete. See [Listing Storage Volume Snapshots](#).
- If you want to delete a remote snapshot, ensure that no storage volumes are currently being created using the snapshot that you want to delete. If a restored storage volume is in the initializing state, the remote snapshot can't be deleted.


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. In the **Storage** drop-down list, click **Storage Snapshots**.
4. Go to the snapshot that you want to delete. From the  menu, select **Delete**.

 **Note:**

You can't delete a colocated snapshot if it has been used to create storage volumes. In this case, the **Delete** option is disabled.

To delete a storage volume snapshot using the CLI, use the `opc compute storage-snapshot delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a storage volume snapshot using the API, use the `DELETE /storage/snapshot/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Scheduling Backups of Storage Volumes and Restoring from Backups



This topic does not apply to Oracle Cloud at Customer.

You can create a schedule to back up a storage volume automatically at defined intervals. Scheduling a backup creates snapshots of the specified storage volume at the specified intervals of time. These snapshots are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance.

When you schedule a backup of a storage volume, you can also specify the number of backups to retain. For example, you can specify that a given storage volume should be backed up at hourly intervals and the two most recent completed snapshots of the storage volume should be retained. This enables you to always have your most current data backed up, without creating unnecessary copies of your data. After a snapshot is completed, if required you can restore a storage volume using the snapshot. The time taken to back up or restore storage volumes depends on the size of the storage volume.

### Topics

- [Creating a Backup Schedule](#)
- [Listing Backup Schedules](#)

- [Updating a Backup Schedule](#)
- [Deleting a Backup Schedule](#)
- [Listing Backups](#)
- [Restoring a Storage Volume from a Backup](#)
- [Deleting a Backup](#)

## Creating a Backup Schedule



This topic does not apply to Oracle Cloud at Customer.

A backup schedule allows you to specify the periodicity of backups and the number of backups to retain.

When you schedule a backup of a storage volume that's attached to an instance, all data that has been written to the storage volume is captured in the backup. However, data that's cached by an application or by the operating system is excluded from the backup.


When you create a backup schedule for the bootable storage volume of an instance, consider the steps you can take to ensure that data writes have been completed and no further modifications are made to the volume while the scheduled backup is being created. For example, you can use tools or utilities provided by the OS to pause modifications. Alternatively, when feasible, you can unmount the storage volume that you want to back up or delete the instance and re-create it later.

To schedule the backup of a storage volume:

### Prerequisites

- When you create a backup schedule, snapshots of the specified storage volume are created according to the specified schedule. These snapshots are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. Ensure that you've set a replication policy for your Oracle Cloud Infrastructure Object Storage Classic instance. See [Selecting a Replication Policy for Oracle Cloud Infrastructure Object Storage Classic](#) in *Using Oracle Cloud Infrastructure Object Storage Classic*.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles](#) in *Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. Go to the storage volume that you want to create a backup schedule for. From the  menu, select **Schedule Snapshots**.

4. In the Create Snapshot Schedule dialog box, select or enter the following information:
  - **Name:** Enter a name for the schedule.
  - **Enable:** Select this option if you want to start creating backups. Otherwise, don't select this option. You can enable the backup schedule later by updating the schedule.
  - **Retention Count:** Specify the number of backups to retain. Whenever a backup is completed, the most recent backups are retained and the oldest backup is deleted.
  - **Interval:** Specify the frequency at which you want to create backups.
  - **Every:**
    - If you selected the hourly interval for the backup schedule, specify the number of hours after which each successive backup must be taken. Select a number between 1 and 24.
    - If you selected the weekly interval for the backup schedule, specify the day of the week when you want to create each successive backup.
  - **At:** If you selected the weekly interval, specify the time when you want to create each successive backup.
  - **Description:** Enter a meaningful description, to help you identify the intended purpose of this backup schedule.
5. Click **Schedule**.

A backup schedule is created and if the schedule is enabled, the specified storage volume is backed up as per the schedule.

To create a backup schedule for a storage volume using the CLI, use the `opc compute backup-configuration add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To create a backup schedule for a storage volume using the API, use the `POST /backup-service/v1/configuration` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After creating a backup schedule, you can view the storage volume snapshots created by scheduled backups on the [Storage Snapshots](#) page. See [Listing Backups](#).

To restore a storage volume from a snapshot created by a scheduled backup, see [Restoring a Storage Volume from a Backup](#).

## Listing Backup Schedules



This topic does not apply to Oracle Cloud at Customer.

Backup schedules allow you to generate multiple backups of a storage volume at defined intervals. You can configure backup schedules to retain a specified number of backups of storage volumes. After you've created backup schedules, you can see the

list of schedules and view information about each schedule on the Snapshot Schedules page.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Storage** tab.

The Storage Volumes page is displayed.

3. Click the **Snapshot Schedules** tab in the left pane.

The Snapshot Schedules page displays a list of schedules. The icon associated with each schedule listed on this page indicates whether the schedule is enabled.

The Snapshot Schedules page also displays other information about your backup schedules, including the storage volume for which the backup schedule was created, the number of backups that will be retained, and the intervals at which the volume is backed up.

 **Tip:**

You can filter the list of backup schedules according to their category. To view backup schedules of a specific category (such as IaaS, PaaS, or Personal), click the **Category** menu and select the appropriate filter. You can also filter schedules according to the status of the schedule. To view enabled or disabled backup schedules, click the **Show** menu and select the appropriate filter.

To list backup schedules using the CLI, use the `opc compute backup-configuration list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list backup schedules using the API, use the `GET /backupservice/v1/configuration/` method. To retrieve the details of a specific schedule using the API, use the `GET /backupservice/v1/configuration/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

To update a backup schedule, see [Updating a Backup Schedule](#).

To delete a backup schedule, see [Deleting a Backup Schedule](#).


## Updating a Backup Schedule



This topic does not apply to Oracle Cloud at Customer.

After you've created a backup schedule for a storage volume, if required, you can update the schedule to change the frequency of backups or the number of backups that you want to retain. You can also enable or disable a backup schedule.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. Click the **Snapshot Schedules** tab in the left pane.  
The Snapshot Schedules page displays the list of backup schedules.
4. Go to the backup schedule that you want to update, and from the  menu, select **Update**.
5. In the Update Snapshot Schedule dialog box, update any of the following:
  - **Enable:** Enable or disable scheduled backups. If you enable a backup schedule that you'd created and disabled earlier, and if the next scheduled backup would have occurred in the past, the backup will start right away.
  - **Retention Count:** Update the number of backups to retain.
  - **Interval:** Update the schedule for creating backups.
  - **Every:**
    - If you selected the hourly interval for the backup schedule, specify the number of hours after which each successive backup must be taken. Select a number between 1 and 24.
    - If you selected the weekly interval for the backup schedule, specify the day of the week when you want to create each successive backup.
  - **At:** If you selected the weekly interval, click the `clock` icon. Specify the time when you want to create each successive backup.
  - **Description:** Enter a meaningful description, to help you identify the intended purpose of this backup schedule.
6. Click **Update**.  
The backup schedule is updated.

To update a backup schedule for a storage volume using the CLI, use the `opc compute backup-configuration update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a backup schedule using the API, use the `PUT /backupservice/v1/configuration/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.



If you no longer need a backup schedule, you can delete the schedule. See [Deleting a Backup Schedule](#).

To view the storage volume snapshots created by the backup schedule, see [Listing Backups](#).

## Deleting a Backup Schedule




This topic does not apply to Oracle Cloud at Customer.

If you no longer need scheduled backups of a storage volume, you can delete a backup schedule.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- To delete a backup schedule, you must delete all backups and restores created from that schedule.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. Click the Snapshot Schedules tab in the left pane.  
The Snapshot Schedules page displays the list of backup schedules.
4. Go to the backup schedule that you want to delete, and from the  menu, select **Delete**.

The backup schedule is deleted and no further scheduled backups are generated. To delete a backup schedule for a storage volume using the CLI, use the `opc compute backup-configuration delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a backup schedule for a storage volume using the API, use the `DELETE /backupservice/v1/configuration/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Listing Backups



This topic does not apply to Oracle Cloud at Customer.

When you back up a storage volume using scheduled backups, a remote snapshot is created for each backup. These snapshots are stored in the associated Oracle Cloud

Infrastructure Object Storage Classic instance. You can view the snapshots created by a scheduled backup on the Storage Snapshots page.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Storage** tab.

The Storage Volumes page is displayed.

3. In the **Storage** drop-down list, click **Storage Snapshots**.

The Storage Snapshots page displays a list of storage volume snapshots, which includes colocated snapshots, remote snapshots, and the remote snapshots created by scheduled backups. Snapshots are arranged alphabetically by the name of the storage volume that each snapshot is of. If there are multiple snapshots of a storage volume, the most recent snapshot is at the top.

**Tip:**

You can filter the list of backups according to their category. To view storage volumes of a specific category (such as IaaS, PaaS, or Personal), click the **Category** menu and select the appropriate filter. Click the **Show** menu and select the remote filter to view all the backups along with the remote snapshots.

The Storage Snapshots page also displays other information about your storage volume snapshots and scheduled backups, such as the date each snapshot was created, the volumes restored from a snapshot, if any, and the backup schedule associated with a snapshot.

The icon associated with each snapshot listed on this page indicates whether a backup is a colocated or a remote snapshot and also shows the status of the backup. While a snapshot is being created, the status shows **Initializing**.

When a backup is completed, its status changes to **Completed**, and you can perform actions like restoring a storage volume or deleting the backup.

To list the backups using the CLI, use the `opc compute backup list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list snapshots created by scheduled backups using the API, use the `GET /backupservice/v1/backup` method. To view the details of a specific backup using the API, use the `GET /backupservice/v1/backup/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

 **Note:**

The Storage Snapshots page displays snapshots created by using scheduled backups as well as snapshots created by using the **Create Snapshot** option.

To delete a backup, see [Deleting a Backup](#).

After a backup is created from a schedule, to use this backup to restore a storage volume, see [Restoring a Storage Volume from a Backup](#).

## Restoring a Storage Volume from a Backup




This topic does not apply to Oracle Cloud at Customer.

When you back up a storage volume using scheduled backups, a remote snapshot is created for each backup. These snapshots are stored in the associated Oracle Cloud Infrastructure Object Storage Classic instance. At any time, if required, you can use these backups to restore a storage volume.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- You should have already created a scheduled backup of the storage volume that you want to restore. See [Creating a Backup Schedule](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. In the **Storage** drop-down list, click **Storage Snapshots**.  
The Storage Snapshots page displays a list of storage volume snapshots as well as the name of the storage volume that each snapshot was created from. For snapshots created by using scheduled backups, this page also displays the name of the backup schedule.
4. Go to the backup that you want to restore a storage volume from. From the  menu, select **Restore Volume**.
5. In the Restore Storage Volume dialog box, enter a name for the new storage volume, specify a description if required and click **Restore**.

While the storage volume is being created, the status on the Storage Volume page shows **Initializing**. After the storage volume has been restored, its status changes to **Online**.

To restore a storage volume from a scheduled backup using the CLI, use the `opc compute restore add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To restore a storage volume from a scheduled backup using the API, use the `POST /backupservice/v1/restore` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

 **Note:**

You can use the `POST /backupservice/v1/restore` method to restore a storage volume from a snapshot that was created by using the `POST /backupservice/v1/configuration` method or the `POST /backupservice/v1/backup` method only. If you've created snapshots by using the `POST /storage/snapshot/` method, you can restore a volume from those snapshots by using the `POST /storage/volume` method. You can't restore a storage volume from those snapshots by using the `POST /backupservice/v1/restore` method.

After you've restored a storage volume, to view details of your storage volume, see [Viewing Details of a Storage Volume](#).

## Deleting a Backup



This topic does not apply to Oracle Cloud at Customer.

If you no longer require a backup of a storage volume, you can delete the backup. This is useful if, for example, the data on that backup is redundant or obsolete or if creating a new volume from that backup isn't required.

Deleting a specific backup has no impact on the backup schedule. The next backup will be created as scheduled. There's also no impact on storage volumes that might have already been created using the backup that you want to delete.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Storage** tab.  
The Storage Volumes page is displayed.
3. In the **Storage** drop-down list, click **Storage Snapshots**.  
The Storage Snapshots page displays the list of backups and snapshots.
4. Go to the backup that you want to delete, and from the  menu, select **Delete**.  
The snapshot is deleted.

To delete a backup of a storage volume using the CLI, use the `opc compute backup delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a backup of a storage volume using the API, use the `DELETE /backupservice/v1/backup/name` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

 **Note:**

You can use the `DELETE /backupservice/v1/backup` method to delete a backup that was created using the `POST /backupservice/v1/configuration` method or the `POST /backupservice/v1/backup` method only. If you've created snapshots by using the `POST /storage/snapshot/` method, you can delete those snapshots by using the `DELETE /storage/snapshot` method.

# 10

## Configuring the Shared Network

### Topics

- [About Network Settings](#)
- [Managing Security Lists](#)
- [Managing Security Rules](#)
- [Managing Security Applications](#)
- [Managing Security IP Lists](#)
- [Managing Public IP Addresses](#)
- [Setting Up Networking for a Sample Scenario Using the Shared Network](#)

### About Network Settings

You can implement fine-grained control over network access to your Compute Classic instances, both from other instances as well as from external hosts. When you create an instance, by default, it doesn't allow access from any other instance or external host. Network settings allow you to configure access to your instances.

You can configure network access to your instance by using the shared network as well as by setting up your own IP networks. If an instance has multiple vNICs, you can associate that instance with both the shared network and the IP network. In the shared network, access to your instances is determined by security lists and security rules, while in the IP network you create security rules and access control lists (ACLs) enable access to instances.

If you want to access an instance directly over the public Internet, you must assign a public IP address to the instance. Public IP addresses can be assigned to the instance interface on the shared network as well as to its interfaces IP networks, if any.

#### Note:

This section describes network configuration in the shared network. For information about setting up an IP network, see [Configuring IP Networks](#).

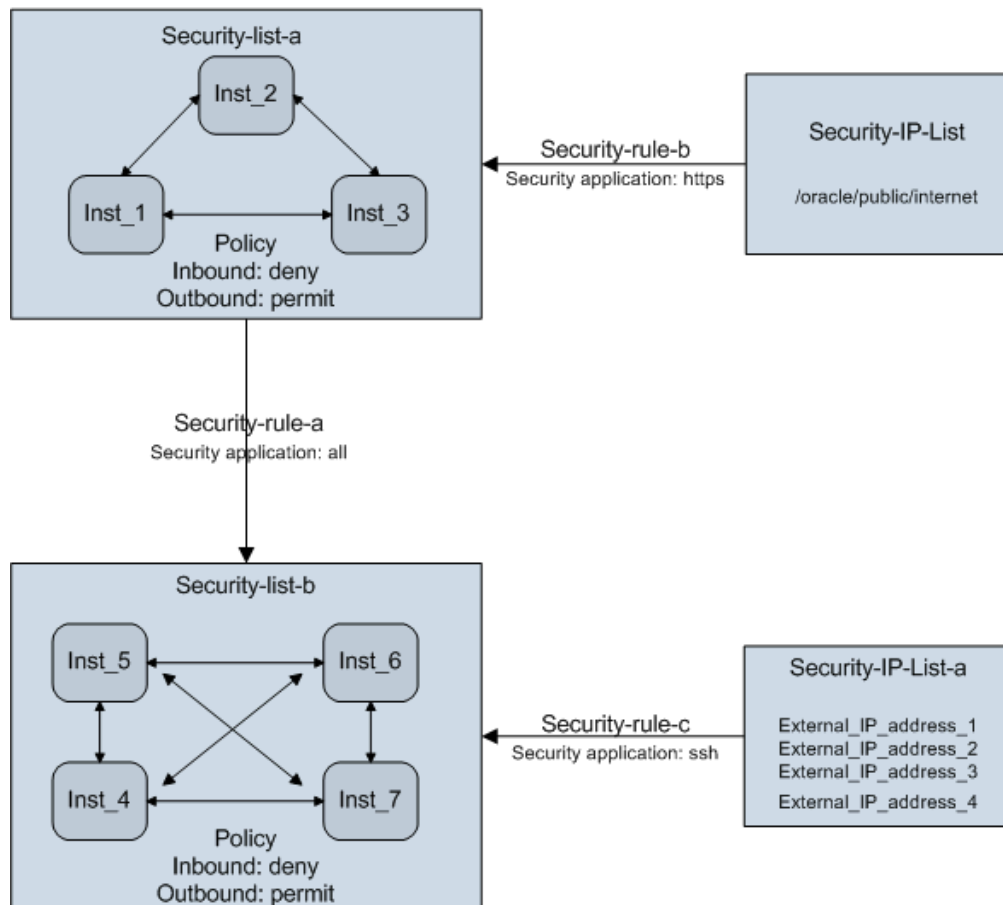
In the shared network, to enable unrestricted communication among some of your instances (for example, to enable all the instances hosting your development environment to communicate with each other), you can create a security list and add the instances to that security list. When you add an instance to a security list, the instance can communicate with all the other instances in the same security list using their private IP addresses.

By default, the instances in a security list are isolated from hosts outside the security list. You can override this default setting by creating security rules. Each security rule

defines a specific source, a destination, and a protocol-port combination over which communication is allowed. For example, you can set up a security rule to permit SSH access over port 22 from a set of external hosts (specified in a security IP list) to all the instances in a security list.

A security list can be used as a source or destination in up to 10 security rules.

The following diagram illustrates how you can use security lists and security rules to restrict traffic between your instances and control access to them.



This diagram shows the following communication paths:

- Instances in Security-list-a can send traffic to instances in Security-list-b over any protocol, as defined by Security-rule-a.
- Instances in Security-list-a can receive HTTPS traffic from any host on the public internet, as defined by Security-rule-b.
- Instances in Security-list-b can receive traffic over SSH from any of the IP addresses specified in Security-IP-list-a, as defined by Security-rule-c.





If no security rules are defined for a security list, then, by default, instances in that security list can't receive traffic from hosts outside the security list. However, instances in the security list can still access other instances in the same security list.

When you remove an instance from a security list, the instance can no longer communicate with other instances in that security list, and traffic to and from that instance is no longer controlled by the security rules defined for that security list.

A security IP list specifies a set of IP addresses that can be used as a source or a destination in security rules. See [Managing Security IP Lists](#).

An instance can be added to multiple security lists. In case of conflicts in policy, the most restrictive policy takes precedence. For example if an instance belongs to one security list with the outbound policy `permit` and the same instance is added to another security list with the outbound policy `deny`, effectively the outbound policy for that instance would be `deny`.

 **See Also:**

-  [Permitting Public TCP Traffic to Instances](#)
-  [Permitting Traffic Between Instances](#)
-  [Permitting SSH Access to Instances](#)
-  [Permitting Ping Requests to Instances](#)

## Setting Up Networking for a Sample Scenario Using the Shared Network

This section illustrates how you can use security lists and security rules to create firewalls and open ports in a sample topology where several Compute Classic Linux instances are attached to the shared network.

### Scenario

In this scenario, you'll create a topology with eight Compute Classic Linux instances: four used for development (`dev1` through `dev4`) and four for production (`prod1` through `prod4`).

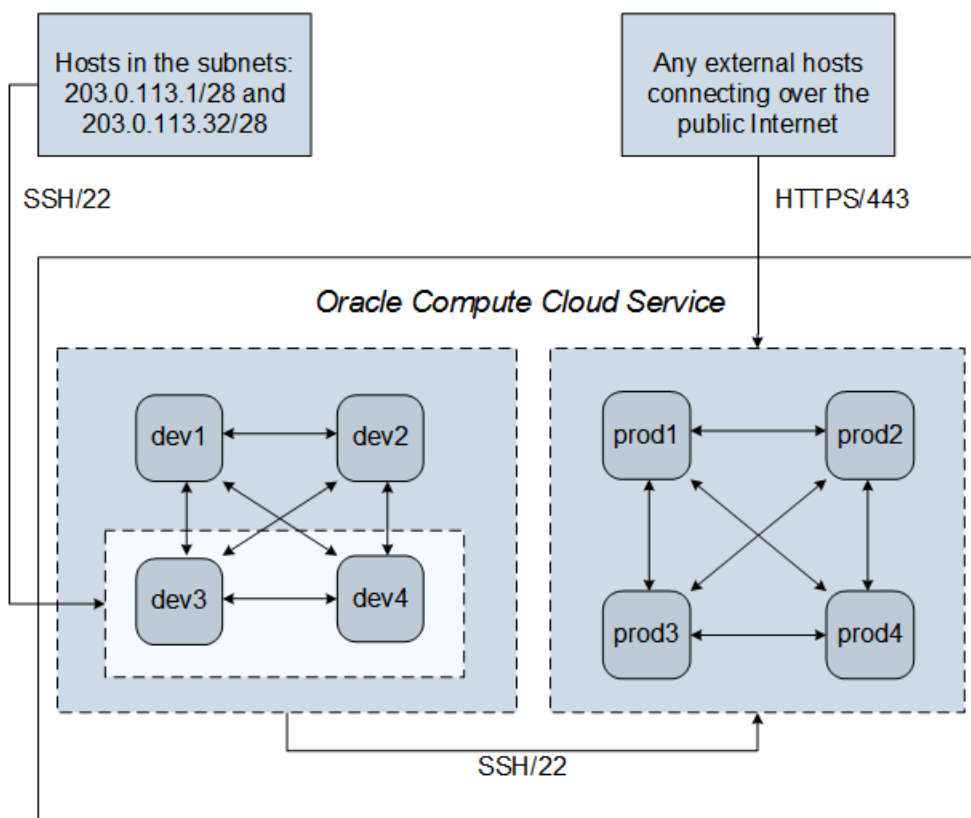
Let's assume that you have the following firewall requirements:

| Requirement | Source                   | Destination              | Protocol and Port | Policy        |
|-------------|--------------------------|--------------------------|-------------------|---------------|
| 1           | Any development instance | Any development instance | All               | Allow traffic |
| 2           | Any development instance | Any production instance  | SSH/22            | Allow traffic |
| 3           | Any production instance  | Any production instance  | All               | Allow traffic |



| Requirement | Source                                                           | Destination                | Protocol and Port | Policy        |
|-------------|------------------------------------------------------------------|----------------------------|-------------------|---------------|
| 4           | Any production instance                                          | Any development instance   | All               | Deny traffic  |
| 5           | Any development instance                                         | Internet                   | All               | Deny traffic  |
| 6           | Internet                                                         | Any development instance   | All               | Deny traffic  |
| 7           | Any host in the subnets<br>203.0.113.1/28 and<br>203.0.113.32/28 | Instances dev3 and<br>dev4 | SSH/22            | Allow traffic |
| 8           | Internet                                                         | Any production instance    | HTTPS/443         | Allow traffic |

The following graphic illustrates the required communication routes between your production and development instances and from external hosts over the public Internet.



To implement these firewall rules using the web console, see [Procedure Using the Web Console](#).

To implement these firewall rules using orchestrations, see [Procedure Using Orchestrations v1](#).

For a graphic showing the topology with the firewall rules implemented, see [Network Topology with the Required Firewall Rules Implemented](#).

### Procedure Using the Web Console

To create the required instances and set up the required security rules for this scenario, complete the following tasks:

1. Generate at least one SSH key pair and upload the SSH public key to Compute Classic. See [Generating an SSH Key Pair](#) and [Adding an SSH Public Key](#).
2. Reserve public IP addresses for the instances that will be accessed over SSH: dev3, dev4, prod1, prod2, prod3, and prod4.

See [Reserving a Public IP Address](#).

3. Create the following security lists, as described in [Creating a Security List](#).

| Security List    | Inbound Policy | Outbound Policy |
|------------------|----------------|-----------------|
| dev              | Deny           | Deny            |
| dev_allow_access | Deny           | Deny            |
| prod             | Deny           | Deny            |

4. Create a bootable storage volume for each of your instances, as described in [Creating a Bootable Storage Volume](#).
5. Create your instances. Remember to associate an SSH public key and a public IP address with each of the instances that you will access over SSH: dev3, dev4, prod1, prod2, prod3, and prod4. See [Creating Instances](#).
6. Add your instances to the required security lists as follows:
  - Add dev1 and dev2 to the dev security list.
  - Add dev3 and dev4 to the dev and the dev\_allow\_access security lists.
  - Add prod1, prod2, prod3, and prod4 to the prod security list.

See [Adding an Instance to a Security List](#).

Adding all the development instances to the dev security list enables all instances in the development environment to communicate with each other over any protocol. By default, no host outside this security list can communicate with any development instance, and no development instance can communicate with any host outside this security list. This fulfils firewall requirements 1, 4, 5, and 6.

Adding all the production instances to the prod security list enables all instances in the production environment to communicate with each other over any protocol. This fulfils firewall requirement 3.

7. Create a security IP list named ip\_list1 consisting of the subnets 203.0.113.1/28 and 203.0.113.32/28. See [Creating a Security IP List](#).
8. Create the following security rules, as described in [Creating a Security Rule](#).

| Security Rule    | Parameters                                                                                                    | Description                                                                                                                                          |
|------------------|---------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| dev-to-prod      | Security application: ssh<br>Source security list: dev<br>Destination security list: prod                     | Any development instance can communicate over SSH with any production instance.<br>This fulfils firewall requirement 2.                              |
| iplist-to-dev    | Security application: ssh<br>Source security IP list: ip_list1<br>Destination security list: dev_allow_access | Any host in the 203.0.113.1/28 and 203.0.113.32/28 subnets can connect to instances dev3 and dev4 using SSH.<br>This fulfils firewall requirement 7. |
| internet-to-prod | Security application: https<br>Source security IP list: public-internet<br>Destination security list: prod    | Any host on the Internet can send HTTPS requests to any production instance.<br>This fulfils firewall requirement 8.                                 |

### Procedure Using Orchestrations v1

1. Generate at least one SSH key pair and upload the SSH public key to Compute Classic. See [Generating an SSH Key Pair](#) and [Adding an SSH Public Key](#).
2. Reserve public IP addresses for the instances that will be accessed over SSH: dev3, dev4, prod1, prod2, prod3, and prod4 . You can use the following sample orchestration to reserve public IP addresses. This sample shows you how to reserve two public IP addresses. Use a similar JSON construct to reserve another four IP addresses.

```
{
  "name": "/Compute-acme/joe/myIPreservations",
  "opplans": [
    {
      "label": "My IP reservations",
      "obj_type": "ip/reservation",
      "objects": [
        {
          "name": "/Compute-acme/joe/ipres1",
          "parentpool": "/oracle/public/ippool",
          "permanent": true
        },
        {
          "name": "/Compute-acme/joe/ipres2",
          "parentpool": "/oracle/public/ippool",
          "permanent": true
        },
        <Add more IP reservations here.>
      ]
    }
  ]
}
```

3. Create the following security lists.

| Security List | Inbound Policy | Outbound Policy |
|---------------|----------------|-----------------|
| dev           | Deny           | Deny            |

| Security List    | Inbound Policy | Outbound Policy |
|------------------|----------------|-----------------|
| dev_allow_access | Deny           | Deny            |
| prod             | Deny           | Deny            |

You can use the following sample orchestration to create security lists. This sample shows you how to create the `dev` security list. Use a similar JSON construct to create another two security lists.

```
{
  "name": "/Compute-acme/joe/mySecurityLists",
  "opplans": [
    {
      "label": "seclists",
      "obj_type": "seclist",
      "objects": [
        {
          "name": "/Compute-acme/joe/dev",
          "outbound_cidr_policy": "deny"
        },
        <Add more security lists here.>
      ]
    }
  ]
}
```

4. Create a bootable storage volume for each of your instances. You can use the following sample orchestration to create storage volumes. This sample shows you how to create one storage volume. Use a similar JSON construct to create all the required storage volumes.

```
{
  "name": "/Compute-acme/joe/myStorageVolumes",
  "opplans": [
    {
      "label": "My storage volumes",
      "obj_type": "storage/volume",
      "objects": [
        {
          "name": "/Compute-acme/joe/boot",
          "bootable": true,
          "imagelist": "/oracle/public/OL_7.2_UEKR3_x86_64",
          "properties": ["/oracle/public/storage/default"],
          "size": "22548578304"
        },
        <Add more bootable storage volumes here.>
      ]
    }
  ]
}
```

 **Note:**

Don't define storage volumes and instances in the same orchestration. By keeping storage volumes and instances in separate orchestrations, you can shut down and start the instances when required and yet preserve the attached storage volumes. Note that the recommendation here is to define the storage *volumes* outside the instance orchestration. To ensure that the storage volumes remain attached after an instance is re-created, you must define the storage *attachments* within the instance orchestration.

5. Create your instances. Remember to associate an SSH public key and a public IP address with each of the instances that you will access over SSH: `dev3`, `dev4`, `prod1`, `prod2`, `prod3`, and `prod4`. You can also specify the security lists that you want to add each instance to. Add your instances to the required security lists as follows:

- Add `dev1` and `dev2` to the `dev` security list.
- Add `dev3` and `dev4` to the `dev` and the `dev_allow_access` security lists.
- Add `prod1`, `prod2`, `prod3`, and `prod4` to the `prod` security list.

Adding all the development instances to the `dev` security list enables all instances in the development environment to communicate with each other over any protocol. By default, no host outside this security list can communicate with any development instance, and no development instance can communicate with any host outside this security list. This fulfils firewall requirements 1, 4, 5, and 6.

Adding all the production instances to the `prod` security list enables all instances in the production environment to communicate with each other over any protocol. This fulfils firewall requirement 3.

You can use the following sample orchestration to create your instances. This sample shows you how to create the `dev3` instance, and associate an SSH public key and a public IP address with the instance. This sample orchestration also shows you how to add this instance to the required security lists, `dev` and `dev_allow_access`. Use similar JSON constructs to define each of the required instances.

```
{
  "name": "/Compute-acme/joe/myInstances",
  "opplans": [
    {
      "label": "My instances",
      "obj_type": "launchplan",
      "objects": [
        {
          "instances": [
            {
              "name": "/Compute-acme/joe/dev3",
              "shape": "oc3",
              "boot_order": [1],
              "label": "dev3",
              "networking": {
                "eth0": {
                  "seclists": ["/Compute-acme/joe/dev", "/Compute-acme/joe/
dev_allow_access"],
```



You can use the following sample orchestration to create your security rules. This sample shows you how to create the `iplist-to-dev` security rule. Use a similar JSON construct to create another two security rules.

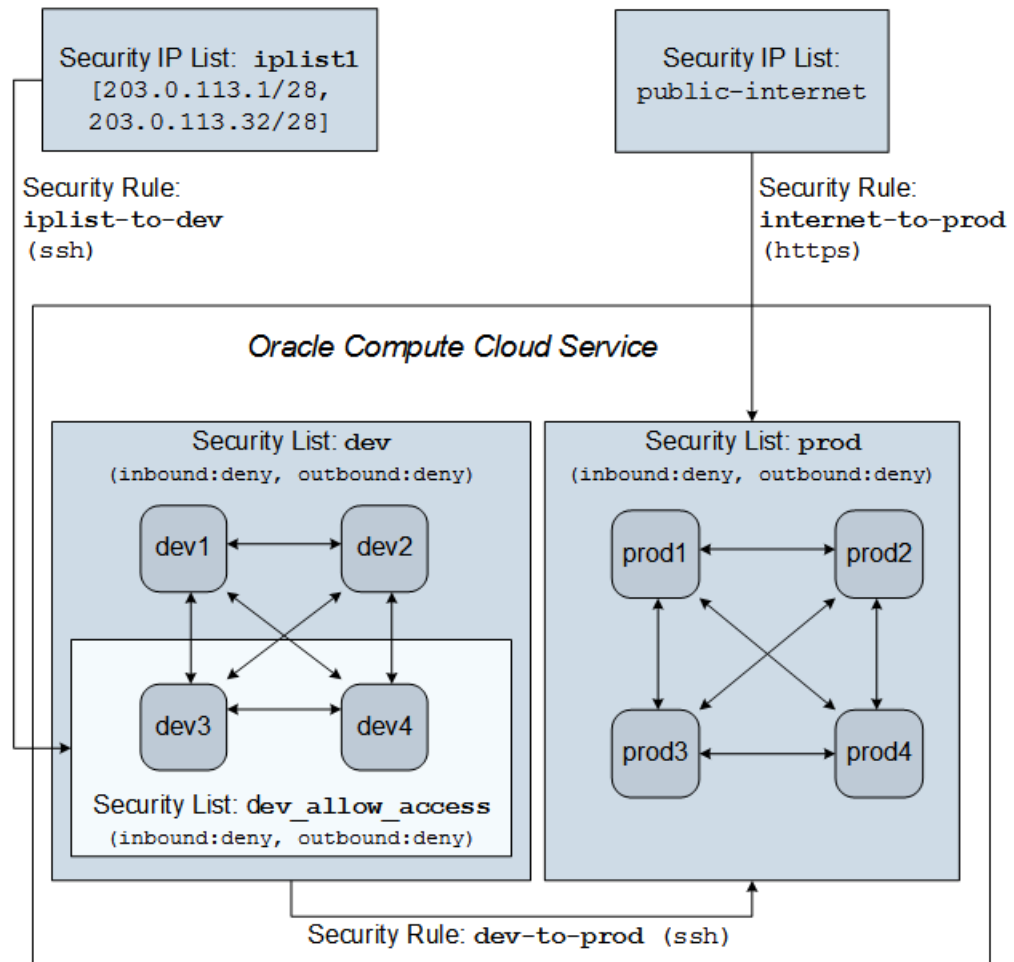
```
{
  "name": "/Compute-acme/joe/mySecRules",
  "opplans": [
    {
      "label": "My security rules",
      "obj_type": "secrule",
      "objects": [
        {
          "name": "/Compute-acme/joe/iplist-to-dev",
          "application": "/oracle/public/ssh",
          "src_list": "seciplist:/Compute-acme/joe/ip_list1",
          "dst_list": "seclist:/Compute-acme/joe/dev_allow_access",
          "action": "PERMIT"
        },
        <Add more security rules here.>
      ]
    }
  ]
}
```

After you've created all the required orchestrations, upload and start your orchestrations to create the required objects and instances. See [Uploading an Orchestration v1](#) and [Starting an Orchestration v1](#).

Remember that you must define relationships for objects referenced by another object in the same orchestration. For example, if you create IP reservations or security lists and instances in the same orchestration, you must define relationships to ensure that the required IP reservations and security lists are created before the instances that use them. Similarly, if you create security lists or security IP lists and security rules in the same orchestration, define relationships to ensure that the security lists and security IP lists are created before the security rules that use them. See [Relationships Between Object Plans](#).

### Network Topology with the Required Firewall Rules Implemented

The following graphic shows the topology with the security rules, security lists, and security IP list set up to enable the network communication required for the scenario described earlier.



## Managing Security Lists

### Topics

- [About Security Lists](#)
- [Creating a Security List](#)
- [Updating a Security List](#)
- [Adding an Instance to a Security List](#)
- [Removing an Instance from a Security List](#)
- [Deleting a Security List](#)

### About Security Lists

A **security list** is a group of Compute Classic instances that you can specify as the source or destination in one or more security rules. The instances in a security list can communicate fully, on all ports, with other instances in the same security list using their private IP addresses.



When you add an instance to a security list, the inbound and outbound policies of the security list are applicable to that instance.

- The inbound policy controls the flow of traffic into the security list. The inbound policy is always set to `deny`, so by default traffic from any source outside the security list can't access the instances that are part of the security list.
- The outbound policy controls the flow of traffic out of the security list. For example, if the outbound policy is set to `deny`, packets can't flow out of the security list. To allow instances in a security list to communicate with hosts outside the security list, set the outbound policy to `permit`.

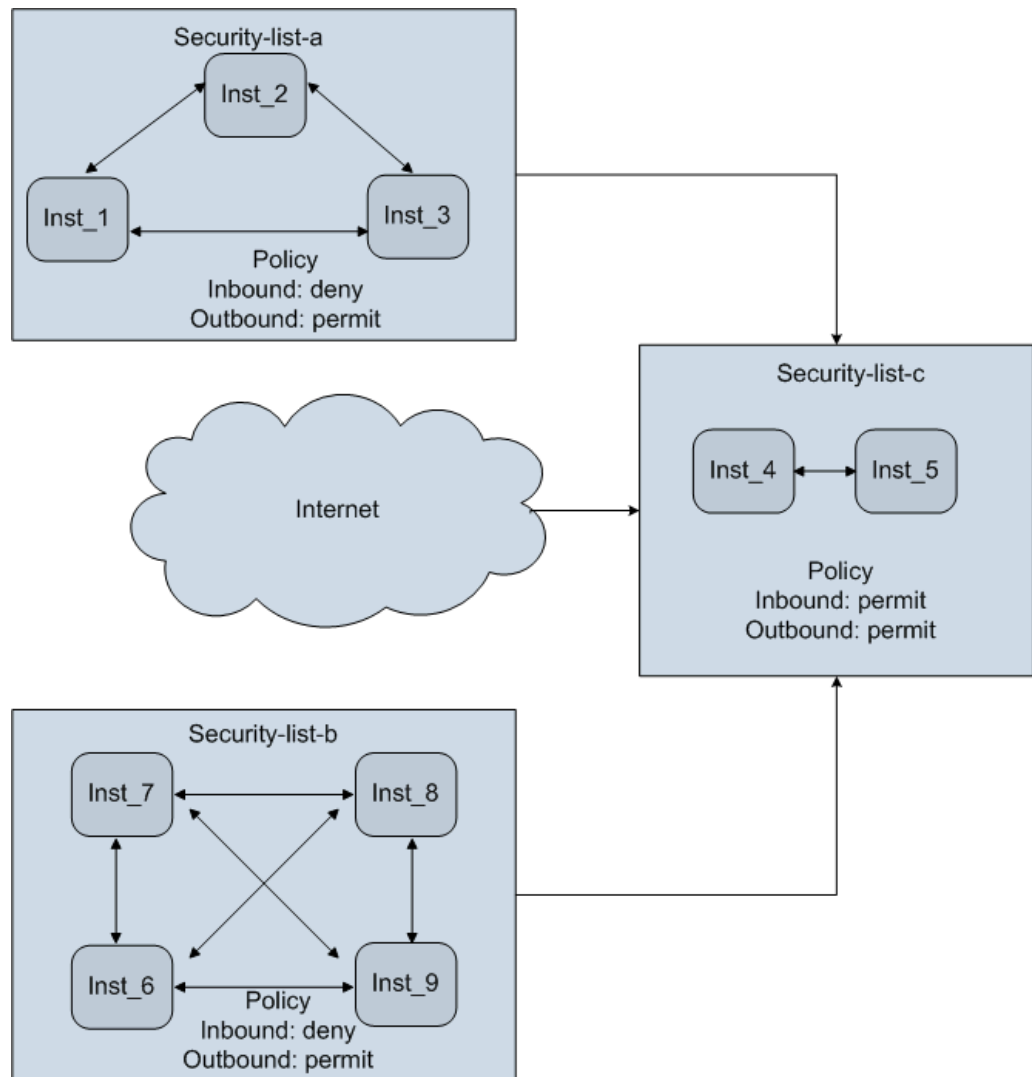
By default, a security list has its inbound policy set to `deny` and outbound policy set to `permit`. However, you can specify a different outbound policy when you create a security list. If you specify the outbound policy as `deny`, then you can set up security rules to override that policy. Similarly, you can create security rules to permit inbound traffic from specified sources, over specified protocols and ports, to the instances in that security list.

 **Note:**

A security rule acts only on a policy that is set to `deny`. If a security list has its outbound policy set to `permit` (the default), then you don't need to define security rules to enable outbound traffic from instances in that security list.

When you create a security rule, you can specify a security list as a source or destination in that security rule. A security list can be specified as the source or destination in up to 10 security rules.

The following diagram shows the relationship between instances and security lists.



In this diagram,

- Security-list-c has the inbound policy set to `permit`. So traffic from the other security lists can reach the instances in this security list, as indicated by the arrows. Traffic from the Internet can also reach the instances in this security list.

 **Note:**

The web console doesn't allow you to specify the inbound policy as `permit`. This is because setting the inbound policy to `permit` in effect disables the firewall. If you need to specify this inbound policy, use the `PUT` or `POST /seclist/` API method, or the `opc compute security-list add` or `opc compute security-list update` CLI command.

- For Security-list-a and Security-list-b, the inbound policy is `deny`. So the instances in these security lists can't receive traffic from any host outside their security lists.

You can add an instance to up to five security lists.

 **Note:**

If an instance is added to multiple security lists that have different policies, then the most restrictive policy is applicable to the instance. For example, in the previous diagram, `Inst_4` is in `Security-list-c`, which has the inbound policy `permit`. If you were to add `Inst_4` to `Security-list-b` as well (inbound policy is `deny`), then the effective inbound policy for `Inst_4` would be `deny`.

Remember, however, that all instances in a security list can communicate with each other across all protocols and ports. So in this scenario, `Inst_4` would be able to communicate with `Inst_5` in `Security-list-c`, as well as with `Inst_6`, `Inst_7`, `Inst_8`, and `Inst_9` in `Security-list-b`.

## Creating a Security List

A **security list** is a group of Compute Classic instances that you can specify as the source or destination in one or more security rules. The instances in a security list can communicate fully, on all ports, with other instances in the same security list using their private IP addresses.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Lists**.
4. Click **Create Security List**.
5. Enter or select the required details—a name and description, and the inbound and outbound policies—and click **Create**.

To create a security list using the CLI, use the `opc compute sec-list add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a security list using the API, use the `POST /seclist/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

You can also create a security list by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Updating a Security List


After creating a security list, at any time, you can update it to change its description as well the inbound and outbound policies.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Lists**.
4. Identify the security list that you want to update. From the  menu, select **Update**.
5. Make the required changes, and click **Update**.

To update a security list using the CLI, use the `opc compute sec-list update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To update a security list using the API, use the `PUT /seclist/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Adding an Instance to a Security List

You can add an instance to a security list either when you create the instance or later by updating the instance.

See [Creating an Instance from the Instances Page](#) and [Updating an Instance](#).

## Removing an Instance from a Security List

To prevent other hosts from accessing an instance, you can remove the instance from the security lists that it is attached to. This may be useful when you want to perform maintenance activities, change or upgrade applications, and so on.

See [Updating an Instance](#).

## Deleting a Security List

You can delete a security list that isn't being used by any instance or security rule.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- Ensure that no instance is attached to the security list that you want to delete.
- Ensure that no security rule uses the security list that you want to delete.


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Lists**.
4. Identify the security list that you want to delete. From the  menu, select **Delete**.

To delete a security list using the CLI, use the `opc compute sec-list delete` command. For help with that command, run the command with the `-h` option. For the

instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a security list using the API, use the `DELETE /seclist/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing Security Rules

### Topics

- [About Security Rules](#)
- [Creating a Security Rule](#)
- [Updating a Security Rule](#)
- [Deleting a Security Rule](#)

## About Security Rules

Security rules are essentially firewall rules, which you can use to permit traffic between Compute Classic instances in different security lists, as well as between instances and external hosts.

The source and destination specified in a security rule can be either a security IP list (that is, a list of external hosts) or a security list.

When you create an instance by using the web console, if you accept the default settings and don't specify any security lists that you want to add your instance to, then your instance is added to a default security list. Any security rules that specify this default security list as a source or destination automatically apply to the instance when the instance is created.

You can create security lists and add instances to those security lists either while creating an instance, or later on when the instance is running. You can then define appropriate security rules that control traffic to and from all instances in the specified security lists.

## Creating a Security Rule

A **security rule** is a firewall rule that you can define to control network access to Compute Classic instances over a specified security application.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.
- Identify (or create) the following:
  - The security application that you want to use in your security rule.
  - The security list or security IP list that you want to use as the source in the security rule.
  - The security list or security IP list that you want to use as the destination in the security rule.

 **Note:**

You can't use any of the predefined security IP lists as a destination in a security rule. If you want to use a security IP list as a destination, ensure that you've created the security IP list.

See [Creating a Security Application](#), [Creating a Security IP List](#), and [Creating a Security List](#).

 **Caution:**

Use security rules carefully and open only a minimal and essential set of ports. Keep in mind your business needs and the IT security policies of your organization.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. Click **Create Rule**.  
The Create Security Rule dialog box is displayed.
4. Enter or select the following:
  - Enter a name for the new security rule.
  - By default, new security rules are enabled. If you'd like to enable the rule later, then set **Status** to **Disabled**.
  - In the **Security Application** field, select the security application that you want to enable traffic over.
  - In the **Source** field, select the security list or security IP list from which traffic over the specified protocol should be allowed.
  - In the **Destination** field, select the security list or security IP list to which traffic should be allowed.
  - Enter a meaningful description for the rule.
5. Click **Create**.

To create a security rule using the CLI, use the `opc compute sec-rule add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create a security rule using the API, use the `POST /secrule/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create a security rule by using orchestrations. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Updating a Security Rule


You can update a security rule to enable or disable it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

#### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. Identify the security rule that you want to update. From the  menu, select **Update**.
4. In the Update Security Rule dialog box, change the **Status** as required, and click **Update**.

To update a security rule using the CLI, use the `opc compute sec-rule update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To update a security rule using the API, use the `PUT /secrule/name` method. For more information, see [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Deleting a Security Rule

If a security rule is no longer required, you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).




 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. Go to the security rule that you want to delete. From the  menu, select **Delete**.

To delete a security rule using the CLI, use the `opc compute sec-rule delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To delete a security rule using the API, use the `DELETE /secrule/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Managing Security Applications

### Topics

- [About Security Applications](#)
- [Listing Security Applications](#)
- [Creating a Security Application](#)
- [Deleting a Security Application](#)

## About Security Applications

A **security application** allows you to specify the protocol and port that you want to use to enable traffic between a source and a destination using security rules.

You can either create a security application, or use one of the predefined security applications.

## Listing Security Applications

Compute Classic provides a number of predefined security applications that you can use in security rules.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Applications**.

The Security Applications page displays a list of security applications.

### Tip:

You can filter the list of security applications according to their category. To view security applications of a specific category (such as IaaS, PaaS, or Personal), click the **Category** menu and select the appropriate filter. You can also filter security applications according to the protocol type by clicking the **Show** menu and selecting the appropriate filter.

To view a list of security applications using the CLI, use the `opc compute sec-application list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view a list of security application using the API, use the `GET /secapplication/container` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Creating a Security Application

Compute Classic provides a number of predefined security applications that you can use. You can also create your own security applications.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.

3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Applications**.
4. Click **Create Security Application**.
5. Enter or select the following information:
  - Enter a name for the security application.
  - Select the port type.
    - If you select the **tcp** or **udp** port type, then enter the port range.
    - If you select the **icmp** port type, then enter the ICMP type.
  - Enter a meaningful description.
6. Click **Create**.

To create a security application using the CLI, use the `opc compute sec-application add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a security application using the API, use the `POST /secapplication/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

You can also create a security application by using orchestrations. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Deleting a Security Application

You can delete a security application that isn't being used by any security rule.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- Ensure that no security rule is using the security application that you want to delete.


 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Applications**.
4. Identify the security application that you want to delete. From the  menu, select **Delete**.

To delete a security application using the CLI, use the `opc compute sec-application delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a security application using the API, use the `DELETE /secapplication/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

If you created a security application using an orchestration v1, then you can delete the security application by stopping the orchestration. See [Terminating an Orchestration v1](#).

If you created a security application using an orchestration v2, then you can delete the security application by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

## Managing Security IP Lists

### Topics

- [About Security IP Lists](#)
- [Creating a Security IP List](#)
- [Updating a Security IP List](#)

- [Deleting a Security IP List](#)

## About Security IP Lists

A **security IP list** is a list of IP subnets (in the CIDR format) or IP addresses that are external to instances in Compute Classic. You can use a security IP list as the source or the destination in security rules to control network access to or from Compute Classic instances.

A security IP list can contain a maximum of 100 entries.

The following table lists the predefined security IP lists that are available in Compute Classic.

| Security IP List               | Description                                                                                                        |
|--------------------------------|--------------------------------------------------------------------------------------------------------------------|
| /oracle/public/instance        | <i>Don't</i> use this security IP list as the source in any security rule.                                         |
| /oracle/public/ntp             | <i>Don't</i> use this security IP list as the source in any security rule.                                         |
| /oracle/public/powerbroker     | <i>Don't</i> use this security IP list as the source in any security rule.                                         |
| /oracle/public/public-internet | You can use this security IP list as the source in security rules to permit traffic from any host on the Internet. |
| /oracle/public/site            | <i>Don't</i> use this security IP list as the source in any security rule.                                         |

### Note:

You can use any security IP list that you create as either a source or a destination in a security rule. However, of the predefined security IP lists, you can use only /oracle/public/public-internet as a source in a security rule, and you can't use any of the predefined security IP lists as a destination in a security rule.

## Creating a Security IP List

To permit traffic from external hosts to Compute Classic instances, you must define those hosts in a Security IP List.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security IP Lists**.

4. Click **Create Security IP List**. Enter the required details and click **Create**.
5. In the Create Security IP List dialog box, enter the following details:
  - In the **Name** field, enter a name for the security IP list.
  - In the **IP List** field, enter a comma-separated list of the subnets (in CIDR format) or IPv4 addresses for which you want to create the security IP list.

For example, to create a security IP list containing the IP addresses 203.0.113.1 and 203.0.113.2, enter one of the following in the **IP List** field:

```
203.0.113.0/30
```

```
203.0.113.1, 203.0.113.2
```

You can specify up to 100 entries in a security IP list.

 **Note:**

You can specify only IP addresses that are external to Compute Classic in a security IP list. You can't specify the IP address of a Compute Classic instance.

- In the **Description** field, enter a description for the security IP list.

6. Click **Create**.

To create a security IP list using the CLI, use the `opc compute sec-ip-list add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create a security IP list using the API, use the `POST /seciplist/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create a security IP list by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Updating a Security IP List


You can update the IP addresses and description for a Security IP List.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. Identify the security IP list that you want to update. From the  menu, select **Update**.
4. In the Update Security IP List dialog box, change the **IP List** or **Description** field, as required, and click **Update**.

To update a security IP list using the CLI, use the `opc compute sec-ip-list update` command. You can use this command to *replace* the list of IP addresses and change the description. To *add* IP addresses to the list, use the `opc compute security-ip-lists add` command and specify the new IP addresses. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a security IP list using the API, use the `PUT /seciplist/name` method. You can use this method to *replace* the list of IP addresses and change the description. To *add* IP addresses to the list, use the `POST /seciplist/` method and specify the new IP addresses. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a Security IP List

If a security IP list isn't used in any security rule and if you don't plan to use the security IP list in the future, then you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.
- Ensure that no security rule is using the security list that you want to delete.


 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security IP Lists**.
4. Identify the security IP list that you want to delete. From the  menu, select **Delete**.

To delete a security IP list using the CLI, use the `opc compute sec-ip-list delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To delete a security IP list using the API, use the `DELETE /seciplist/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

If you created a security IP list using an orchestration v1, then you can delete the list by stopping the orchestration. See [Terminating an Orchestration v1](#).

If you created a security IP list using an orchestration v2, then you can delete the security IP list by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

## Managing Public IP Addresses

### Topics

- [About Public IP Addresses](#)
- [About Private IP Addresses](#)
- [Reserving a Public IP Address](#)
- [Updating an IP Reservation](#)



- [Attaching a Public IP Address to an Instance](#)
- [Removing a Public IP Address from an Instance](#)
- [Deleting an IP Reservation](#)

## About Public IP Addresses

If you want to enable access to your instance over the public Internet, you must associate a public IP address with your instance. You can associate either a temporary or a persistent public IP address with an instance when you create the instance.

Temporary public IP addresses are assigned dynamically from a pool of public IP addresses. When you associate a temporary public IP address with an instance, if the instance is restarted or is deleted and created again later, its public IP address might change. If you want to assign a persistent public IP address to your instance, you must first create an IP reservation, and then associate the IP reservation with the instance.

To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).

## About Private IP Addresses

When you create an instance, you can specify whether the instance should be added to the shared network, to IP networks, or to both. When you add an instance to the shared network, it is automatically assigned a private IP address from an Oracle-defined pool of addresses. This private IP address might change when the instance is restarted.

When you add an instance to one or more IP networks, each network interface is assigned an IP address in the subnet of the specified IP network. You can specify a fixed IP address from the specified IP network while creating the instance. If no IP address is specified, an IP address is allocated dynamically. This dynamically assigned IP address might change when the instance is restarted. To understand how to set up and use IP networks, see [About IP Networks](#).

To find out the private IP addresses of your instance, view the information on the Instances page. See [Listing Instances](#).

## Reserving a Public IP Address

An **IP reservation** is a public IP address that you can attach to a Compute Classic instance that requires access to or from the Internet. You can create an IP reservation and associate it with an instance to enable access to the instance from the public Internet.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console.
2. (Optional) If your domain spans multiple sites, then check that the site you've selected has sufficient capacity to create the required resources. Click **Site** near the top of the page to view the aggregate resource usage by all tenants on the

currently selected site. If resource usage on the selected site is close to maximum, pick another site.

If you're using the REST API to create resources, note the API end point of the site that you want to use.

3. Click the **Network** tab.
4. In the **Network** drop-down list, expand **Shared Network**, and then click the **IP Reservations**.
5. Click **Create IP Reservation**.
6. Enter a name for the IP reservation.
7. In the **For Instance** field, you can select the instance that you want to attach the IP address to.

 **Note:**

When you attach an IP reservation to a running instance, then if you delete and re-create or shut down and restart the instance, the IP reservation reverts to whatever was specified while creating the instance and any updates made to the IP reservation are lost. You must update the IP reservation again.

Alternatively, you can create the IP reservation now without attaching it to any instance, and attach it later. See [Attaching a Public IP Address to an Instance](#).

8. Click **Create**.

To create an IP reservation using the CLI, use the `opc compute ip-reservation add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create an IP reservation using the API, use the `POST /ip/reservation/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create an IP reservation by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Updating an IP Reservation

You can change the status of an IP reservation or attach it to an instance by updating the IP reservation.

- If you've created an instance without a public IP address, then updating an IP reservation allows you to attach a public IP address to an instance.
- If you've created an instance with a temporary public IP address, then updating an IP reservation allows you to change its status to permanent.


### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to

ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **IP Reservations**.
4. Identify the IP reservation that you want to update. From the  menu, select **Update**.
  - If the selected IP reservation isn't attached to an instance, then you can attach it now.

 **Note:**

When you attach an IP reservation to a running instance, then if you delete and re-create or shut down and restart the instance, the IP reservation reverts to whatever was specified while creating the instance and any updates made to the IP reservation are lost. You must update the IP reservation again.

- If the IP reservation is attached to an instance, then you can change its status to **Temporary** or **Permanent**.

To change the status of an IP reservation using the CLI, use the `opc compute ip-reservation update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To change the status of an IP reservation using the API, use the `PUT /ip/reservation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Attaching a Public IP Address to an Instance

You can attach an IP reservation to an instance either while creating the instance or when the instance is already running, by updating the IP reservation or by updating the instance.

For information about creating instances, see [Creating an Instance from the Instances Page](#). For information about attaching a public IP address to an instance while updating an instance, see [Attaching a Public IP Address to an Instance on the Shared Network](#). For information about updating an IP reservation, see [Updating an IP Reservation](#).

 **Note:**

When you attach an IP reservation to a running instance, then if you delete and re-create or shut down and restart the instance, the IP reservation reverts to whatever was specified while creating the instance and any updates made to the IP reservation are lost. You must update the IP reservation again.

You can also associate an IP reservation with an instance when you create instances using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#).

Internally, an IP reservation is associated with an instance through the instance's `vcable`. A `vcable` provides an attachment point to a specific network interface on an instance. The `vcable` of an instance is created automatically when the instance is launched and is deleted when the instance is deleted.

The process of adding a virtual link between an instance and an IP reservation is also referred to as IP association.

To find out the `vcable` ID of your instance using the CLI, use the `opc compute instance get` command. To associate an IP reservation with an instance using the CLI, use the `opc compute ip-association add` command and specify the `vcable` ID. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To find out the `vcable` ID of your instance using the API, use the `GET /instance/name` method. To associate an IP reservation with an instance using the API, use the `POST /ip/association/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Removing a Public IP Address from an Instance

If you want to change the public IP address of an instance, or if you no longer need a public IP address for the instance, then you can remove the IP reservation from the instance.


 **Note:**

If you associate a persistent public IP address with an instance while creating the instance, and later on you remove the public IP address from the instance, then if you delete and re-create or shut down and restart the instance, the public IP address reverts to whatever was specified while creating the instance. You must remove the public IP address from the instance again.

 **Note:**

You can't remove a temporary IP address from an instance. You can only remove a persistent IP address. If you created an instance with an autogenerated IP address or if you changed the status of the IP address associated with an instance to temporary, then to remove that IP address from the instance, first update it to change its status to permanent. See [Updating an IP Reservation](#).

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **IP Reservations**.
4. Identify the IP reservation that you want to detach. From the  menu, select **Remove Instance**.

Internally, an IP reservation is associated with an instance by using a `vcable`. A `vcable` provides an attachment point to a specific network interface on an instance. A `vcable` is created automatically when an instance is launched and is deleted when the instance is deleted.

The process of adding a virtual link between an instance and an IP reservation is also referred to as IP association.

To find out the `vcable` ID of your instance using the CLI, use the `opc compute instance get` command. To remove an IP reservation from an instance using the CLI, use the `opc compute ip-association delete` command. For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To find out the `vcable` ID of your instance using the API, use the `GET /instance/name` method. To remove an IP reservation from an instance using the API, use the `DELETE /ip/association/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

If you specified an IP reservation to be associated with an instance in an orchestration v1, then, when you stop the orchestration, the IP reservation is detached, and the instance is deleted. See [Terminating an Orchestration v1](#).

## Deleting an IP Reservation

When you no longer need an IP reservation, you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).
- Ensure that no instance is using the IP reservation that you want to delete.

### Caution:

If a persistent public IP address is associated with an instance during instance creation, then if required, you can remove that IP address from the instance later on. Ensure, however, that you don't delete this IP reservation. If you delete and re-create the instance, the IP reservation will be required again. If you've deleted the IP reservation, you won't be able to re-create the instance.


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **IP Reservations**.
4. Identify the IP reservation that you want to delete. From the  menu, select **Delete**.

To delete an IP reservation using the CLI, use the `opc compute ip-reservation delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an IP reservation using the API, use the `DELETE /ip/reservation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

# 11

## Configuring IP Networks

### Topics

- [About IP Networks](#)
- [About Access Control to Interfaces on IP Networks](#)
- [Workflows for Using IP Networks](#)
- [Managing IP Networks](#)
- [Managing IP Network Exchanges](#)
- [Managing vNICsets](#)
- [Managing Routes](#)
- [Managing IP Address Prefix Sets](#)
- [Managing Security Protocols for IP Networks](#)
- [Managing Security Rules for IP Networks](#)
- [Managing ACLs](#)
- [Managing Public IP Addresses](#)

### About Access Control to Interfaces on IP Networks

When you add an instance to an IP network, two factors determine whether traffic can flow to and from the instance: network reachability and access control. By default, an interface on an IP network is reachable by other instances only if those instances have an interface either on the same IP network or on an IP network connected to the same IP network exchange. An interface on an IP network is not, by default, reachable from any source that is not on the same IP network or on the same IP network exchange.

When reachability is provided, access control to an interface is determined by the vNICsets that the vNIC is added to and on the access control lists applied to those vNICsets. A **Virtual NIC**, or **vNIC**, is a virtual network interface card that enables an instance to be associated with a network. Instances created using Oracle-provided Oracle Linux or Windows images with the release version 16.3.6 or later support eight vNICs, enabling each instance to be associated with up to eight networks. You can add each vNIC to multiple vNICsets. You can then define various access control lists and apply them to each vNICset.

An **access control list (ACL)** is a collection of security rules that can be applied to a vNICset. ACLs determine whether a packet can be forwarded to or from a vNIC, based on the criteria specified in its security rules. A **security rule** permits traffic from a specified source or to a specified destination. You must specify the direction of a security rule — either ingress or egress. In addition, you can specify the source or destination of permitted traffic, and the security protocol and port used to send or receive packets. Each of the parameters that you specify in a security rule provides a criterion that the type of traffic permitted by that rule must match. Only packets that



match all of the specified criteria are permitted. If you don't specify match criteria for any parameter, all traffic for that parameter is permitted. For example, if you don't specify a security protocol, then traffic using any protocol and port is permitted.

When you create an instance, you can specify one or more vNICsets for each interface on an IP network. You can also define various ACLs and apply them to each vNICset. When the instance is created, the vNICs for those interfaces are added to the specified vNICsets and access to each vNIC is controlled by the ACLs applied to those vNICsets.

If you don't specify any vNICsets for an interface, the vNIC for that interface is added to the default vNICset. A default ACL is applied to the default vNICset. This default ACL contains a default ingress and egress security rule, to permit traffic across the vNICs in the default vNICset.

 **Note:**

Remember, however, that reachability must be ensured as well. If a vNICset contains vNICs on IP networks that aren't connected by an IP network exchange, those vNICs won't be able to communicate with each other.

The following network objects exist by default to control traffic to and from vNICs in the default vNICset.

- `/Compute-identity_domain/default`: The default vNICset. If you don't specify a vNICset for an interface while creating an instance, the vNIC for that interface is automatically added to this default vNICset.
- `/Compute-identity_domain/default`: The default ACL. This ACL is automatically applied to the default vNICset.
- `/Compute-identity_domain/default/ingress`: The default ingress security rule. This security rule specifies the default vNICset as the source and the destination. It doesn't specify any security protocol. Traffic is permitted from any vNIC within the default vNICset to any destination within the default vNICset over all protocols and ports.
- `/Compute-identity_domain/default/egress`: The default egress security rule. This security rule specifies the default vNICset as the source. No destination or security protocol are specified. Traffic from the default vNICset is permitted to all destinations over all protocols and ports.

 **Caution:**

While it is possible to delete any of the default access control objects, doing so could result in cutting off access to multiple vNICs. If you delete the default ACL, vNICset, or security rules, ensure that you create the corresponding objects required to enable and control traffic to and from the affected vNICs.

 **See Also:**[Controlling Network Access to Instances in IP Networks](#)

## About IP Networks

In the shared network each instance is assigned a private IP address from a common pool of Oracle-provided IP addresses. An **IP network** allows you to define an IP subnet in your account. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network. These IP addresses aren't part of the common pool of Oracle-provided IP addresses used by the shared network. When you add an instance to an IP network, the instance is assigned an IP address in that subnet. You can assign IP addresses to instances either statically or dynamically, depending on your business needs. So you have complete control over the IP addresses assigned to your instances.

Here are a few things that you can do with IP networks:

- **Bring your own IP addresses**

In the shared network, private IP addresses are assigned to your instances from a common pool of IP addresses. These IP addresses aren't persistent. If you stop or delete an instance, the private IP address that was assigned to that instance is returned to the shared pool of IP addresses and could be reassigned to an instance created by another tenant. So, if you're using IP addresses from the shared pool to access your instances — over a VPN connection, for example — you must ensure that whenever you create or re-create an instance, the private IP address of that instance is included in the range of reachable routes, and whenever you delete an instance, that private IP address is removed from the list of reachable routes. Any applications running in your on-premises environment that access an instance using its private IP address might also need to be updated.

When you create an IP network, you specify the IP address prefix for that network. This allows you to allocate IP addresses to your instances as per your requirements, and gives you complete control over the IP addresses that are assigned to your instances. You can specify the private IP addresses you want to use for each instance, without worrying about conflicts with private IP addresses used by other tenants in a multitenant site.

**Note:**

The first unicast IP address of any IP network is reserved for the default gateway, the DHCP server, and the DNS server of that IP network.

- **Isolate instances by creating multiple IP networks**

With an IP network you can isolate instances by creating separate IP networks and adding instances to specific networks. Traffic can flow between instances within the same IP network, but by default each network is isolated from other networks and from the public Internet.

- **Attach instances to multiple IP networks**

Oracle-provided Oracle Linux and Windows images with release version 16.3.6 or later support up to eight virtual network interfaces (vNICs). If you use a private image, you can set up the image to support multiple vNICs. When you create an instance using these images, you can add each instance to up to eight different networks, including the shared network. An instance that has a public IP address as well as one or more private IP addresses from IP networks, can be set up as a gateway, to route packets to each of the IP networks that it is added to.

 **Note:**

Only images with the release version 16.3.6 or later support eight vNICs. Instances created using older images don't support multiple vNICs.

- **Assign static or dynamic IP addresses to instances**

You can use static IP addresses to ensure that the IP address of an instance doesn't change when an instance is stopped and restarted, or deleted and re-created. Here a static IP address means that the DHCP server on the IP network always assigns a specified IP address to a specified interface of an instance.

Alternatively, you can use dynamically assigned IP addresses for your instances, without having to add or remove these IP addresses from routes when you create or delete instances. Even when a dynamically assigned IP address is not being used by your instance, it can't be assigned to another customer's instance.

- **Associate a MAC address with a network interface**

You can specify the MAC address for each network interface of your instance. You might need to do this if the MAC address of an instance is used for software licensing or other purposes.

- **Enable communication across IP networks or to external IP addresses**

After creating private IP networks and adding instances to the networks, you can connect different IP networks by creating IP network exchanges. You can specify paths to destination subnets outside your IP networks by creating routes. An **IP network exchange** enables access between IP networks that have non-overlapping addresses, so that instances on these networks can exchange packets with each other without NAT. An IP network exchange can include multiple IP networks, but an IP network can be added to only one IP network exchange.

A **route** specifies the IP address of the destination as well as the vNICset that provides the next hop for routing packets. Using routes to direct traffic allows you to specify multiple routes to various destination subnets. If each route uses a vNICset that contains multiple vNICs, then egress load balancing and high availability is also ensured.

In IP networks, while ARP and DHCP are supported only within the scope of the local interface, other types of broadcast, like ARP, are not supported between different instances. So if you try arping any IP address, the answer comes with different MAC address, not the MAC address of the vNIC associated with the IP address.

- **Set up a VPN connection to instances attached to IP networks**

You can simplify the set-up of a two-way VPN connection, which allows you to establish a secure connection between your Compute Classic instances and your data center. See [Connecting to Instances in a Multitenant Site Using VPN](#).

- **Associate multiple public IP addresses with each instance**

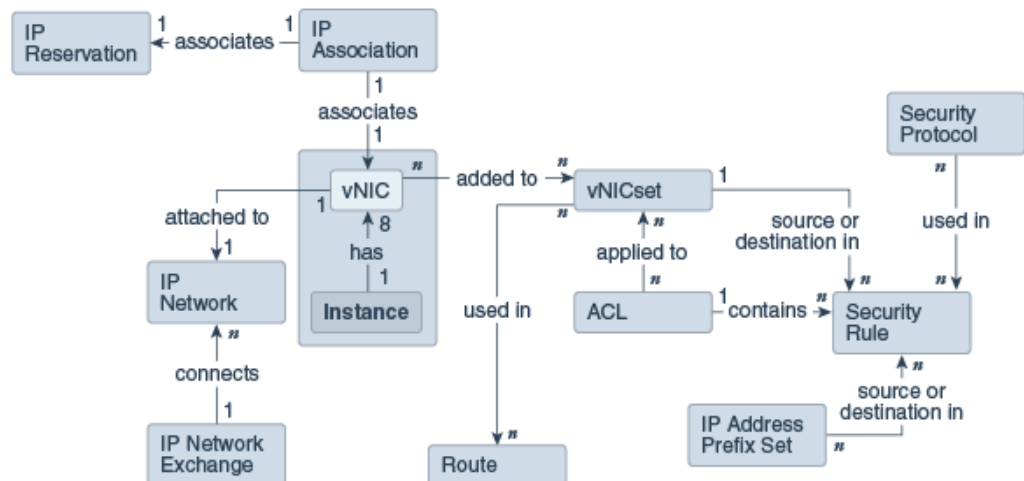
When you create an instance with multiple virtual interfaces, you can associate a public IP address with each vNIC that is added to an IP network. An instance can have up to eight vNICs. So if you create an instance and associate each of the eight vNICs with an IP network, you can associate up to eight public IP addresses with the instance.

The shared network, however, allows you to associate only one IP address with an instance. So if you create an instance with an interface on the shared network and without any interfaces on IP networks, you can associate only a single public IP address with the instance.

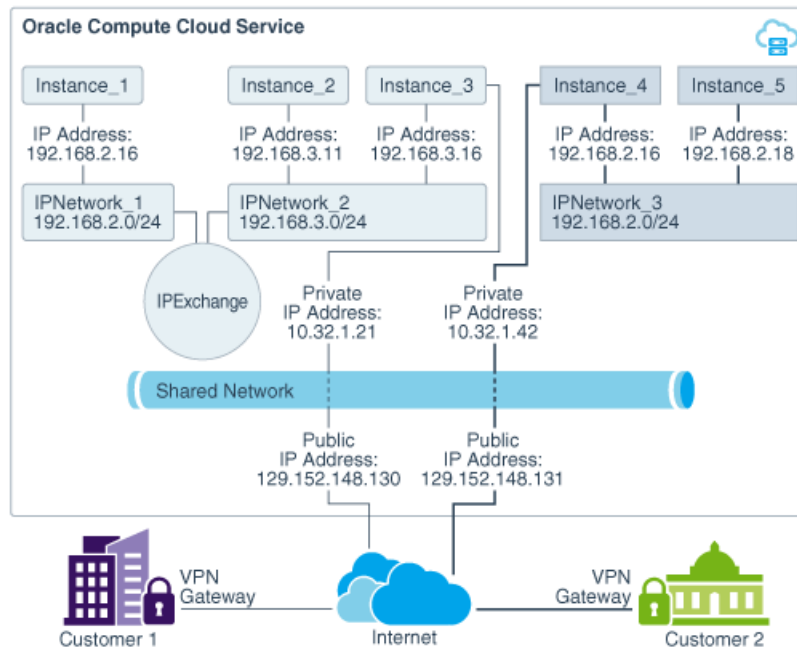
- **Create ACLs to control the flow of traffic to and from each interface on an instance**

Using IP networks enables you to create access control lists (ACLs), which control the type of traffic that is permitted to and from each interface of your instances. ACLs are a collection of security rules, where each rule can specify the direction of traffic, a list of permitted sources and destinations, and the type of packet as well as the port that can be used at the source or destination.

Essentially, with IP networks, you have complete control over your network configuration. Using IP networks allows you to create a network architecture that mirrors and extends the architecture you use in your data center.



The following figure shows the interaction between the IP networks and the shared network for two customers in a multitenant site:



The graphic shows that Customer 1 has created two IP networks, 192.168.2.0/24 and 192.168.3.0/24. Customer 2 has created one IP network, 192.168.2.0/24, which overlaps with one of the subnets specified by Customer 1. However, there is no conflict in the overlapping IP addresses, because these networks aren't connected with each other. Both Customer 1 and Customer 2 have set up a VPN tunnel to their instances. Traffic from Customer 1 is routed to Instance 3, which has the public IP address 129.152.148.130 and traffic from Customer 2 is routed to Instance 4, which has the public IP address 129.152.148.131. Customer 1 has also set up an IP network exchange to connect their two networks.

### Considerations for Configuring vNICs and IP Networks

Here are a few points to keep in mind while setting up your IP networks and assigning vNICs to IP networks:

- You can't assign multiple vNICs on a single instance to the same IP network. Each vNIC on an instance must belong to a separate IP network.
- You can attach a vNIC to either the shared network or to an IP network, not to both. Also, you can't mix attributes from both networks for a single vNIC. For example, you can't specify the `ip` attribute with the `seclist` attribute for a given vNIC.
- If you connect two IP networks using an IP network exchange, traffic across those IP networks is routed through the default gateway.
- If you want to connect IP networks by using an IP network exchange, it is recommended that you do this before creating instances with an interface on those IP networks. This ensures that routes are appropriately configured on instances by the DHCP client during instance initialization.
- If you create an instance that has interfaces on multiple IP networks, ensure that you don't also add those IP networks to the same IP network exchange. Doing so creates a loop within the network, leading to erratic network behavior.
- You can update an IP network and change the specified IP address prefix for the network after you've created the network and attached instances to it. However,

when you change an IP address prefix, it could cause the IP addresses currently assigned to existing instances to fall outside the specified IP network. If this happens, all traffic to and from those vNICs will be dropped.

If the IP address of an instance is dynamically allocated, stopping the instance orchestration and restarting it will reassign a valid IP address from the IP network to the instance.

However, if the IP address of an instance is static — that is, if the IP address is specified in the instance orchestration while creating the instance — then the IP address can't be changed by stopping the instance orchestration and restarting it. You must manually update the orchestration to assign a valid IP address to the vNIC attached to that IP network.

It is therefore recommended that if you update an IP network, you only expand the network by specifying the same IP address prefix but with a shorter prefix length. For example, you can expand 192.168.1.0/24 to 192.168.1.0/20. Don't, however, change the IP address of the network. This ensures that all IP addresses that have been currently allocated to instances remain valid in the updated IP network.

## Workflows for Using IP Networks

Here are a few workflows for using IP networks to set up your network.

### Topics

- [Workflow for Adding an Instance to an IP Network](#)
- [Workflow for Connecting IP Networks](#)
- [Workflow for Creating Routes to External Destinations](#)
- [Workflow for Applying Access Control Lists](#)
- [Workflow for Assigning a Public IP Address to a Network Interface](#)
- [Workflow for Enabling SSH Access to an Instance Using IP Networks](#)

### Workflow for Adding an Instance to an IP Network

An **IP network** allows you to define an IP subnet in your account. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network. These IP addresses aren't part of the common pool of Oracle-provided IP addresses used by the shared network. When you add an instance to an IP network, the instance is assigned an IP address in that subnet. You can assign IP addresses to instances either statically or dynamically, depending on your business needs. So you have complete control over the IP addresses assigned to your instances.

Here's a workflow to set up IP networks and add an instance to an IP network:

1. Create the required IP networks. See [Creating an IP Network](#).
2. (Optional) Create the required vNICsets. A **Virtual NIC**, or **vNIC**, is a virtual network interface card that enables an instance to be associated with a network. Instances created using Oracle-provided Oracle Linux or Windows images with the release version 16.3.6 or later support eight vNICs, enabling each instance to be associated with up to eight networks.

A **Virtual NIC Set**, or **vNICset**, is a collection of one or more vNICs. vNICsets are useful when you want to use multiple vNICs for the same action. For example, you use vNICsets to specify multiple vNICs as a source or a destination in a security rule. You can also use vNICsets in routes to specify multiple vNICs as the next hop destination for that route.

3. Create an instance with the required interfaces. While creating the instance, use the network attributes section to specify the interface that should be associated with the shared network, if any, and the interfaces that should be associated with your IP networks, as required. See [Creating Instances](#).

 **Note:**

For instances that are associated with an IP network, while creating the instance you can specify the vNICsets that you want to add each vNIC to. The vNICs are then automatically added to the specified vNICsets when the instance is created, and removed from the vNICsets when the instance is deleted.

If you create an instance by using an orchestration, then, to specify vNICsets in instance network attributes, see [Subparameters for a Network Interface on an IP Network](#).

If you create an instance using the web console, then to specify vNICsets in the Create Instance wizard, see [Creating an Instance from the Instances Page](#).

If you create an instance and don't specify any vNICsets, the vNICs that are associated with IP networks are automatically added to the default vNICset.

If you have any existing instances that have vNICs associated with IP networks, if those vNICs aren't explicitly added to a specified vNICset, they are automatically added to the default vNICset.

### Workflow for Connecting IP Networks

An **IP network exchange** enables access between IP networks that have non-overlapping addresses, so that instances on these networks can exchange packets with each other without NAT.

Connecting multiple IP networks using an IP network exchange is useful when you want a larger set of instances to be able to communicate with each other, but the currently defined IP network won't support such a large number of instances.

For example, consider a scenario where you've created an IP network with a /28 subnet mask and you find that you want to use this IP network for, let's say, 40 instances. You'd have to increase the size of the subnet by using a /26 subnet mask. However, in certain situations, you might not want to increase the size of the subnet. For example, increasing the size of the subnet might cause overlapping IP addresses with another subnet in your account or in your data center. This could be problematic if the overlapping address ranges are connected using an IP network exchange or if you have a VPN connection that includes the overlapping subnets at either end.

In such situations, rather than modifying an existing IP network, you can create another IP network and link it to one or more existing IP networks using an IP network exchange.

Here's a workflow to enable communication across IP networks using an IP network exchange.

1. Create the required IP networks. See [Creating an IP Network](#).
2. Create the required instances and specify the interfaces that should be added to the IP networks. See [Creating Instances](#).
3. Create an IP network exchange. See [Creating an IP Network Exchange](#).
4. Reference the IP network exchange in all the required IP networks. You can reference an IP network exchange in an IP network either when you create the IP network, or later, by updating the IP network.

 **Caution:**

You must ensure that there's no conflict in the range of IP addresses used in various IP networks, the IP addresses used in your on-premises network, or the range of private IP addresses used in the shared network. If IP networks with overlapping IP subnets are linked to an IP exchange, packets going to and from those IP networks are dropped.

See [Creating an IP Network](#) or [Updating an IP Network](#)

Instances on different IP networks that are part of the IP network exchange should now be able to communicate with each other using the IP addresses assigned from the participating IP networks. However, this communication is controlled by the security rules created and the ACLs applied to the relevant vNICsets.

For example, consider that you've added two instances to IP network 1 and two instances to IP network 2. You've also created vNICset 1 for the vNICs in IP network 1 and vNICset 2 for the vNICs in IP network 2. If you add both IP networks to the same IP network exchange, it establishes a channel for communication between the instances added to the two IP networks. However, whether the instances can communicate or not, and the protocol and port that they can use to communicate, depends on the ACLs that apply to each vNICset. See [Workflow for Applying Access Control Lists](#).

### Workflow for Creating Routes to External Destinations

Creating a route allows you to specify a preferred path for traffic to specified destinations outside your IP networks. A **route** specifies the IP address of the destination as well as the vNICset that provides the next hop for routing packets. A **Virtual NIC Set**, or **vNICset**, is a collection of one or more vNICs. You must specify a vNICset when you create a route. When a vNICset containing multiple vNICs is used in a route, Equal Cost Multipath (ECMP) anycast routing is implemented. Traffic routed by that route is load-balanced across all the vNICs in the vNICset. Using vNICsets with multiple vNICs also ensures high availability for traffic across the specified vNICs.

Here's a workflow for creating a vNICset and using it in a route:

1. Create the required IP networks. See [Creating an IP Network](#).
2. Create a vNICset. See [Creating a vNICset](#).
3. Create the required instances and specify interfaces to be added to the IP networks. See [Creating Instances](#).



#### 4. Create a route. See [Creating a Route](#).

Traffic to the specified destinations is now routed over the specified vNICs.

 **Note:**

A route directs traffic to the specified next hop based on the destination address in the packet. To ensure that packets can access a vNIC in either the ingress or the egress direction, you must define the required security rules and apply them to a vNICset using an access control list. See [Workflow for Applying Access Control Lists](#).

### Workflow for Applying Access Control Lists

An **access control list (ACL)** is a collection of security rules that can be applied to a vNICset. ACLs determine whether a packet can be forwarded to or from a vNIC, based on the criteria specified in its security rules.

A **security rule** permits traffic from a specified source or to a specified destination. You must specify the direction of a security rule — either ingress or egress. In addition, you can specify the source or destination of permitted traffic, and the security protocol and port used to send or receive packets. Each of the parameters that you specify in a security rule provides a criterion that the type of traffic permitted by that rule must match. Only packets that match all of the specified criteria are permitted. If you don't specify match criteria for any parameter, all traffic for that parameter is permitted. For example, if you don't specify a security protocol, then traffic using any protocol and port is permitted.

When you create a security rule, you specify the ACL that it belongs to. ACLs apply to vNICsets. Each vNICset can reference multiple ACLs and each ACL can be referenced in multiple vNICsets. When an ACL is referenced in a vNICset, every security rule that belongs to the ACL applies to every vNIC that is specified in the vNICset.

A security rule allows you to specify the following parameters:

- The flow direction — ingress or egress
- (Optional) A source vNICset
- (Optional) A list of source IP address prefix sets
- (Optional) A destination vNICset
- (Optional) A list of destination IP address prefix sets
- (Optional) A list of security protocols
- (Optional) The name of the ACL that contains this rule
- (Optional) An option to disable the security rule

When you apply a security rule to a vNICset using an ACL, packets that match all the criteria in any one security rule applied to a vNIC are allowed.

For example, consider that you've created `vnicset_a`. In this vNICset, you've referenced `acl_1` and `acl_2`. Now, consider that you've created a security rule `secrule_ingress` in which you've specified only the flow direction `ingress` and the ACL `acl_2`. Because you haven't specified any other match criteria, this security rule

allows **all** incoming traffic, and because this security rule is added to `acl_2`, it applies to every vNIC in `vnicset_a`.

In `acl_1` and `acl_2`, you might have added other ingress security rules that include specific criteria for incoming packets, such as a source IP address range or a protocol or port. However, since incoming packets of any protocol and port and from any source match the criteria specified in `secrule_ingress`, those packets will be delivered to the relevant vNICs in `vnicset_a` despite other security rules that might filter out such traffic.

So, when creating security rules and adding them to ACLs keep your security rules as specific as possible. When you apply an ACL to a vNICset, check the other ACLs that apply to the same vNICset as well. If you've created very specific security rules and added them to an ACL, those rules might not apply when another ACL contains very permissive security rules applied to the same vNICset.

Here's a workflow for creating a security rule and an ACL and applying the ACL to a vNICset:

1. Create the required IP networks. See [Creating an IP Network](#).
2. (Optional) If you want to specify a vNICset as a source or destination in a security rule, create the required vNICsets. A vNICset is a collection of one or more vNICs. See [Creating a vNICset](#).
3. Create the required instances and specify the interfaces that should be added to the IP networks and vNICsets that you've created. If you don't specify the vNICsets that you want to associate a vNIC with, the vNIC is added to the default vNICset. See [Creating Instances](#).

 **Note:**

If you created instances prior to the 16.4.6 release and you didn't add vNICs on those instances to any vNICset, then those vNICs are automatically added to the default vNICset.

4. (Optional) Create an IP address prefix set. An **IP address prefix set** contains a set of IPv4 addresses in the CIDR address prefix format. When you create a security rule, you can specify a list of IP address prefix sets as the source or destination for permitted traffic.

You can specify an IP address prefix set in multiple security rules.

To create an IP address prefix set, see [Creating an IP Address Prefix Set](#).

5. (Optional) Create a security protocol. A **security protocol** allows you to specify a transport protocol and the source and destination ports to be used with the specified protocol. When you create a security rule, you can specify the security protocols that you want to use to permit traffic. Only packets that match the transport protocol and ports in any of the specified security protocols will be permitted. If you don't specify protocols and ports in a security protocol, traffic is permitted over all protocols and ports.

You can specify a security protocol in multiple security rules. So if you have a protocol that you want to use in a number of security rules, you don't have to create the protocol multiple times.

To create a security protocol, see [Creating a Security Protocol for IP Networks](#).

6. Create an ACL. After creating the required ACL, you'll reference it in the security rule and apply it to the required vNICsets. To create an ACL, see [Creating an ACL](#).
7. Create a security rule specifying the parameters you require such as security protocols, source and destination vNICsets or IP address prefix sets, and the ACL that you want to add this security list to. Ensure that each of the objects that you reference in a security rule does exist. If you reference a security protocol or an IP address prefix set that doesn't exist, the security rule won't be used. Also, when you're ready to apply a security rule, remember to specify the ACL that you want to add the security rule to. If you don't specify an ACL, the security rule won't be used.

To create a security rule, see [Creating a Security Rule for IP Networks](#).

8. Create or update the vNICsets that you want to apply the ACL to, and specify the ACL that you want to associate with that vNICset. See [Creating a vNICset](#) or [Updating a vNICset](#).

### Workflow for Assigning a Public IP Address to a Network Interface

If you want an interface on your instances to be accessible from the public Internet, or if you want your instances to be able to communicate with other Oracle services on other IP networks, you can reserve a public IP address and associate that IP address with a vNIC on your instance.

You can reserve a public IP address from one of two IP pools:

- **/oracle/public/public-ippool:** When you attach an IP address from this pool to an instance, you enable access between the public Internet and the instance.
- **/oracle/public/cloud-ippool:** When you attach an IP address from this pool to an instance, the instance can communicate privately (that is, without traffic going over the public Internet) with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region.

A public IP address or a cloud IP address can be associated with only one vNIC at a time. A vNIC can have two NAT IP addresses, one from each IP pool.

Here's the workflow for reserving a NAT IP address and associating it with an instance.

1. Create an IP network. See [Creating an IP Network](#).
2. Create an IP address reservation. When you reserve an IP address from a specified IP pool, an IPv4 address is allocated for your use. You can associate this IP address with a vNIC on one of your instances. See [Reserving a Public IP Address for IP Networks](#)
3. Associate the IP reservation with the vNIC on an instance. See [Associating a Public IP Address with a vNIC](#).

If you selected the `/oracle/public/public-ippool` IP pool, then your instance can communicate with external hosts over the public Internet. If you selected the `/oracle/public/cloud-ippool` IP pool while creating an IP reservation, then your instance can communicate with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region, without sending traffic over the public Internet. However, the routes, security rules, and ACLs required to enable traffic must be set up as well.

## Workflow for Enabling SSH Access to an Instance Using IP Networks

When you create an instance with one or more interfaces added to IP networks, access to those interfaces depends on the vNICsets that each vNIC belongs to. Access control lists (ACLs) are applied to vNICsets to control the type of traffic that is allowed to and from vNICs in the vNICset.

Here's a workflow for creating a vNICset and configuring the security rules and ACL required to enable SSH access to the vNICs in this vNICset.

1. Create an IP network. See [Creating an IP Network](#).
2. Create an IP address reservation and specify the IP pool `/oracle/public/public-ippool`. See [Reserving a Public IP Address for IP Networks](#).
3. Create an ACL. See [Creating an ACL](#).
4. Create a vNICset and apply the ACL to the vNICset. See [Creating a vNICset](#).
5. Create a security protocol with the protocol `TCP` and the destination port set `22`. See [Creating a Security Protocol for IP Networks](#).
6. (Optional) Create one or more IP address prefix sets to specify the IP addresses from which you want to permit SSH access. See [Creating an IP Address Prefix Set](#).

If you want to permit SSH access from all sources, you don't need to create an IP address prefix set.

7. Create the following security rules:

- **Ingress security rule**

Specify the ACL and security protocol that you just created and the source IP address prefix set, if required. If no source is specified, SSH traffic from all sources is permitted. Specify the vNICset that you just created as the destination.

- **(Optional) Egress security rule**

 **Note:**

This rule is required only if you want to initiate SSH requests from the vNICset to external destinations. You don't need this security rule if the vNICs in the vNICset must respond to incoming SSH requests, but must not be allowed to initiate SSH requests.

Specify the ACL and security protocol that you just created and specify the vNICset as the source. If required, specify the IP address prefix set as the destination. If no destination is specified, traffic to all destinations is permitted.

See [Creating a Security Rule for IP Networks](#).

8. Finally, while creating an instance do the following:
  - Add one interface to the IP network that you just created.
  - Add the vNIC to the vNICset that you just created.
  - Associate the vNIC with the IP address reservation that you just created.

See [Creating Instances](#).

Subsequently, for any instance that you create, specify the same IP network and vNICset, and associate an IP reservation with the vNIC while creating the instance. When the instance is created, you can access its public IP address using SSH.

## Managing IP Networks

An **IP network** allows you to define an IP subnet in your account. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network. These IP addresses aren't part of the common pool of Oracle-provided IP addresses used by the shared network. When you add an instance to an IP network, the instance is assigned an IP address in that subnet. You can assign IP addresses to instances either statically or dynamically, depending on your business needs. So you have complete control over the IP addresses assigned to your instances.

### Topics

- [Creating an IP Network](#)
- [Listing IP Networks](#)
- [Adding an Instance to an IP Network](#)
- [Updating an IP Network](#)
- [Deleting an IP Network](#)

## Creating an IP Network

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Networks**.
4. Click **Create IP Network**.
5. Select or enter the required information:
  - **Name:** Enter a name for the IP network.
  - **IP Address Prefix:** Enter the IP address prefix for this IP network, in CIDR format. When you create instances, you can associate a vNIC on the instance with an IP network. That vNIC on the instance is then allocated an IP address from the specified IP network.

Select the IP address prefix for your IP networks carefully. Consider the number of instances that you might want to add to the network. This will help determine the size of the subnet required.

The prefix length of the IP address prefix that you specify in an IP network should be between /16 to /30.

If you create multiple IP networks and you might want to add these IP networks to the same IP network exchange, then ensure that you don't allocate overlapping address ranges to these IP networks.

Similarly, if you plan to connect to your IP networks using VPN, then ensure that the addresses you specify for your IP networks don't overlap with each other, or with the IP addresses used in your on-premises network.

 **Note:**

RFC 6598 addresses aren't supported.

- **IP Exchange:** Specify the IP network exchange that you want to add this IP network to. An IP network can belong to only one IP network exchange. Before you specify an IP network exchange for an IP network, ensure that the IP addresses in this IP network don't overlap the IP addresses in any other network in the same IP network exchange. If you don't specify an IP network exchange while creating an IP network, you can do so later, by updating an IP network.

If you want to connect IP networks by using an IP network exchange, it is recommended that you do this before creating instances with an interface on those IP networks. This ensures that routes are appropriately configured on instances by the DHCP client during instance initialization.

- **Description:** Enter a meaningful description for your IP network, if required.
- **Tags:** Enter a list of the tags that you want to associate with your IP network, if required.

6. Click **Create**.

The IP network is created and added to the specified IP network exchange.

## Other Ways of Creating an IP Network

To create an IP network using the CLI, use the `opc compute ip-network add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create an IP network using the API, use the `POST /network/v1/ipnetwork/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create an IP network by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

To add an instance to an IP network, you must specify the IP network attributes while creating the instance. See [Adding an Instance to an IP Network](#).

## Listing IP Networks

After creating IP networks, you can view a list of your IP networks along with information about the IP address prefix of the IP network, and the IP network exchange you've added the IP network to, if any.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Networks**.

The IP Networks page displays a list of IP networks, along with information about each network, such as the IP address prefix of the network, the IP network exchange it belongs to, and the description.

To list IP networks using the CLI, use the `opc compute ip-network list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To list IP networks using the API, use the `GET /network/v1/ipnetwork/container/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

Next, if you want to change the IP address prefix, IP network exchange, or description of an IP network, see [Updating an IP Network](#).

If you want to add an instance to an IP network, you must specify the IP network attributes while creating the instance. See [Adding an Instance to an IP Network](#).

## Adding an Instance to an IP Network

You can specify the IP networks that you want to add an instance to only when you create the instance. You can't add an instance to IP networks or change the IP networks that you've added an instance to after you've created the instance.

To specify the IP networks that you want to add an instance to:

- If you create an instance using the Create Instance wizard, use the **IP Network** section on the Network page. The interfaces on your instance are associated with the IP networks specified here. See [Creating an Instance from the Instances Page](#).
- If you create an instance using an orchestration, use the instance networking attributes to specify the IP networks that you want to add your instance to. See [Orchestration v1 Attributes for instances](#) or [Orchestration v2 Attributes for Instance](#).
- If you create an instance from the API using a launch plan, use the networking attributes to specify the IP networks that you want to add your instance to. See [Creating Instances Using Launch Plans](#).

While adding interfaces to IP networks, you can also specify a static IP address for each interface. If you don't specify a static IP address, then an IP address is allocated dynamically from the specified IP network. This IP address might change each time you re-create your instance.

 **Note:**

A dynamic IP address is useful if you might need to update the IP address of your IP networks. If an instance has a dynamic IP address, then, if you update the IP address of an IP network, you can restart the instance orchestration to ensure that the instance is allocated a valid IP address from the updated network. However, if an instance has a static IP address, then if you update the IP address of an IP network, the instance will go into an error state. You'll have to update the instance orchestration to specify a valid IP address in the updated IP network.

You can't remove an instance from an IP network after you've created the instance. However, if you don't require an IP network any more, you can delete the IP network. The corresponding interface on each instance that was added to that IP network then becomes unreachable. See [Deleting an IP Network](#).

## Updating an IP Network

After creating an IP network, if required, you can update the network. Updating an IP network allows you to modify all attributes of an IP network, except the name.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Networks**.



4. Go to the IP network that you want to update, and from the  menu, select **Update**.
5. You can update the following information:

- **IP Address Prefix:** The IP address prefix for this IP network, in CIDR format. You can change the specified IP address prefix for the network even after you've created the network and attached instances to it. However, when you change an IP address prefix, it could cause the IP addresses currently assigned to existing instances to fall outside the specified IP network. If this happens, all traffic to and from those vNICs will be dropped.

If the IP address of an instance is dynamically allocated, stopping the instance orchestration and restarting it will reassign a valid IP address from the IP network to the instance.

However, if the IP address of an instance is static — that is, if the IP address is specified in the instance orchestration while creating the instance — then the IP address can't be updated by stopping the instance orchestration and restarting it. You must manually update the orchestration to assign a valid IP address to the vNIC attached to that IP network.

It is therefore recommended that if you update an IP network, you only expand the network by specifying the same IP address prefix but with a shorter prefix length. For example, you can expand 192.168.1.0/24 to 192.168.1.0/20. Don't, however, change the IP address of the network. This ensures that all IP addresses that have been currently allocated to instances remain valid in the updated IP network.

The prefix length of the IP address prefix that you specify in an IP network should be between /16 to /30.

If this IP network belongs to an IP exchange, then ensure that after updating the IP network address, the IP address prefix for this IP network doesn't overlap with the IP address prefix assigned to another IP network that is part of the same IP network exchange.

Similarly, if you plan to connect to your IP networks using VPN, then ensure that the addresses you specify for your IP networks don't overlap with each other, or with the IP addresses used in your on-premises network.

If you need to increase the size of the subnet of an IP network, instead of updating the IP address prefix, consider creating a separate IP network and adding it to an IP exchange. That way, you don't risk disrupting the IP addresses already allocated to existing instances, or overlapping with IP addresses on another IP network.

 **Note:**

RFC 6598 addresses aren't supported.

- **IP Exchange:** The IP network exchange that you want to add this IP network to. An IP network can belong to only one IP network exchange. Before you specify an IP network exchange for an IP network, ensure that the IP addresses in this IP network don't overlap the IP addresses in any other network in the same IP network exchange.

If you want to connect IP networks by using an IP network exchange, it is recommended that you do this before creating instances with an interface on those IP networks. This ensures that routes are appropriately configured on instances by the DHCP client during instance initialization.

- **Description:** Update the description, if required.
- **Tags:** Update the list of tags associated with your IP network, if required.

To update an IP network using the CLI, use the `opc compute ip-network update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an IP network using the API, use the `PUT /network/v1/ipnetwork/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting an IP Network

If you no longer need an IP network, you can delete it.

Here's what happens when you delete an IP network:

- If any instances had interfaces on that network, those interfaces will be unreachable after the IP network is deleted. However, the instance itself won't show any error and can still be reached on other interfaces.
- If the IP network that you want to delete is used as a destination in a route, traffic sent over that route won't reach its destination when the IP network is deleted. However, if you create another IP network with the same IP address prefix, then the route will apply to that destination.
- If any vNICs on an IP network are used in a route, then if you delete that IP network, those vNICs will become unreachable. Any other vNICs in the same vNICset will continue to be used to route traffic. If all vNICs in a vNICset become unreachable, then any routes that use the vNICset won't work.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Networks**.
4. Identify the IP network that you want to delete. From the  menu, select **Delete**.

To delete an IP network using the CLI, use the `opc compute ip-network delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To delete an IP network using the API, use the `DELETE /network/v1/ipnetwork/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Managing IP Network Exchanges

### Topics

- [Creating an IP Network Exchange](#)
- [Listing IP Network Exchanges](#)
- [Adding an IP Network to an IP Network Exchange](#)
- [Deleting an IP Network Exchange](#)

## Creating an IP Network Exchange

An **IP network exchange** enables access between IP networks that have non-overlapping addresses, so that instances on these networks can exchange packets with each other without NAT.

 **Note:**

An IP network exchange provides a communication channel across IP networks. However, whether communication is permitted over this channel, the protocols and ports used for communication, and the source and destination IP addresses for which communication is permitted, is controlled by the security rules created and the ACLs applied to the relevant vNICsets.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Exchanges**.
4. Click **Create IP Exchange**.
5. Enter a name, description, and tags for your IP network exchange and then click **Create**.

The IP network exchange is created.

To create an IP network exchange using the CLI, use the `opc compute ip-network-exchange add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To create an IP network exchange using the API, use the `POST /network/v1/ipnetworkexchange/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create an IP network exchange by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating an IP network exchange, you can add IP networks to the exchange either while creating IP networks, or later, by updating an IP network. See [Creating an IP Network](#) and [Updating an IP Network](#).

## Listing IP Network Exchanges

After creating IP network exchanges, you can view a list of IP network exchanges and the description of each IP network exchange.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Exchanges**.

The IP Exchanges page displays a list of IP network exchanges.

To list IP network exchanges using the CLI, use the `opc compute ip-network-exchange list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To list IP network exchanges using the API, use the `GET /network/v1/ipnetworkexchange/container/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Adding an IP Network to an IP Network Exchange

If you need to enable access between instances on different IP networks, you can add those networks to an IP network exchange.

If you've already created an IP network exchange, you can add the IP network to the IP network exchange when you create the IP network. See [Creating an IP Network](#).

Otherwise, you can add the IP network to the IP network exchange later, by updating the IP network. See [Updating an IP Network](#).

As a best practice, add a maximum of 20 IP networks to an IP network exchange. Due to DHCP limitations routing is automatically configured for only for 20 IP networks in an IP network exchange. If you want to add more than 20 IP networks to an IP network network, you'll need to manage routing in each instance manually.

## Deleting an IP Network Exchange

If no IP networks are using an IP network exchange, or if you no longer want to enable traffic across the IP networks in an IP network exchange, you can delete the IP network exchange.

### Note:

If you delete an IP network exchange which is referenced in an IP network, then if you try to create an instance with an interface on that IP network, your instance will go into an error state and won't be created. The IP network exchange referenced by an IP network must exist when you create an instance with an interface on that IP network.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Exchanges**.
4. Identify the IP network exchange that you want to delete. From the  menu, select **Delete**.

To delete an IP network exchange using the CLI, use the `opc compute ip-network-exchange delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the*

Compute Classic CLI in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an IP network exchange using the API, use the `DELETE /network/v1/ipnetworkexchange/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing vNICsets

A **Virtual NIC Set**, or **vNICset**, is a collection of one or more vNICs. vNICsets are useful when you want to use multiple vNICs for the same action. For example, you use vNICsets to specify multiple vNICs as a source or a destination in a security rule. You can also use vNICsets in routes to specify multiple vNICs as the next hop destination for that route.

In a vNICset, you can specify a maximum of 32000 vNICs and 256 access control lists (ACLs).

### Topics

- [Creating a vNICset](#)
- [Listing vNICsets](#)
- [Adding an Instance Interface to a vNICset](#)
- [Updating a vNICset](#)
- [Deleting a vNICset](#)

## Creating a vNICset

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Virtual NIC Sets**.
4. Click **Create vNICset**.
5. In the Create vNICset dialog box, select or enter the following:
  - **Name**: Enter a name for the vNICset.
  - **vNICs**: Select the required vNICs.
  - **Applied Access Control Lists**: Select the access control lists (ACLs) that you want to apply to this vNICset. When you apply an ACL to a vNICset, all the

security rules in that ACL are applied to traffic to or from each of the vNICs in the vNICset.

- **Description:** Enter a meaningful description for the vNICset.
- **Tags:** Enter a list of the tags that you want to associate with this vNICset.

6. Click **Create**.

The vNICset is created. You can use this vNICset as the next hop in any routes that you create, or as the source or destination in a security rule. ACLs are also applied to vNICsets.

## Other Ways of Creating a vNICset

To create a vNICset using the CLI, use the `opc compute virtual-nic-set add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create a vNICset using the API, use the `POST /network/v1/vnicset/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create a vNICset by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating a vNICset, if you want to add or remove vNICs from the vNICset, you can update the vNICset. See [Updating a vNICset](#). To use a vNICset as the next hop in a route, see [Creating a Route](#).

## Listing vNICsets

After creating vNICsets, you can view a list of vNICsets along with information about the vNICs in each vNICset.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Virtual NIC Sets**.

The vNIC Set page displays a list of vNICsets along with the vNICs in each vNICset. To list vNICsets using the CLI, use the `opc compute virtual-nic-set list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To list vNICsets using the API, use the `GET /network/v1/vnicset/container/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).



## Adding an Instance Interface to a vNICset

You can add an instance interface to a vNICset either while creating the instance, or later, when the instance is running. When you add an interface to a vNICset while creating the instance, then if you stop and restart or delete and re-create the instance, the interface is automatically added back to the specified vNICset. However, if you add an interface to a vNICset when the instance is running, then if you stop and restart or delete and re-create the instance, you must add the interface to the required vNICsets again.

While creating an instance, you can specify a maximum of 4 vNICsets for each interface. To specify vNICsets for instance interfaces while creating an instance, see [Creating Instances](#).

To add a running instance to a vNICset, or to add an instance interface to more than 4 vNICsets, specify the required vNICs while creating or updating the vNICset. See [Creating a vNICset](#) or [Updating a vNICset](#).

## Updating a vNICset

After you've created a vNICset, you can add or remove vNICs by updating the vNICset.

### Note:

When a vNICset is used in a route, then if a vNIC in that vNICset becomes unreachable — for example, when an instance is stopped or deleted — traffic is automatically routed and load-balanced across the remaining vNICs in the vNICset.


### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Virtual NIC Sets**.
4. Go to the vNICset that you want to update, and from the  menu, select **Update**.
5. Update any of the following fields, as required:
  - **vNICs**: Add or remove vNICs from the vNICset.
  - **Applied Access Control Lists**: Add or remove access control lists (ACLs) to be applied to this vNICset. When you apply an ACL to a vNICset, all the security rules in that ACL are applied to traffic to or from each of the vNICs in the vNICset.
  - **Description**: Update the description, if required.
  - **Tags**: Updated the tags associated with this vNICset, if required.
6. Click **Update**.

To update a vNICset using the CLI, use the `opc compute virtual-nic-set update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a vNICset using the API, use the `PUT /network/v1/vnicset/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a vNICset

If you no longer need to group a set of vNICs into a vNICset, you can delete the vNICset. If you delete a vNICset that was used by any route as the next hop, then that route will no longer work. Note that deleting a vNICset doesn't delete the vNICs in the set.

### Note:

If you create an instance with one or more interfaces on IP networks and you don't specify any vNICset for an interface, the vNIC for that interface is automatically added to the default vNICset. Access to vNICs in the default vNICset is controlled by the default ingress and egress security rules, which are added to the default ACL. If you delete the default vNICset, ensure that all vNICs are added to other vNICsets with the appropriate security rules and applied ACLs. Otherwise communication to those vNICs will be blocked.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to


ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Virtual NIC Sets**.
4. Identify the vNICset that you want to delete. From the  menu, select **Delete**.

To delete a vNICset using the CLI, use the `opc compute virtual-nic-set delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete a vNICset using the API, use the `DELETE /network/v1/vnicset/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Managing Routes

### Topics

- [Creating a Route](#)
- [Listing Routes](#)
- [Updating a Route](#)
- [Deleting a Route](#)

## Creating a Route

You can use routes to specify preferred paths for traffic from your network to destinations outside your network. A **route** specifies the IP address of the destination as well as the vNICset that provides the next hop for routing packets.

For example, if you want to set up a VPN connection to your instances, you must create a route to specify that traffic to the reachable subnets specified in the VPN connection should be routed through the vNICs of the VPN gateway.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Routes**.
4. Click **Create Route**.
5. Select or enter the required information:
  - **Name:** Enter a name for the route.
  - **Administrative Distance:** Enter 0, 1, or 2 to specify the administrative distance of the route. The administrative distance indicates the priority of a route. The highest priority is 0. When multiple routes to a destination exist, the route with the highest priority is used. If multiple routes with the highest priority exist, traffic is routed and load-balanced across all those routes. In this case, traffic is routed over all vNICs specified in these routes as if they belonged to a single vNICset.
  - **IP Address Prefix:** Enter the IP address prefix, in CIDR format, of the destination network that you want to specify the route to. The destination IP address prefix must be an external network or host.
  - **Next Hop vNICset:** Select the vNICset that you want to use to route packets to the specified destination. When a vNICset containing multiple vNICs is used in a route, Equal Cost Multipath (ECMP) anycast routing is implemented. Traffic routed by that route is load balanced across all the vNICs in the vNICset. Using vNICsets with multiple vNICs also ensures high availability for traffic across the specified vNICs.
  - **Description:** Enter a meaningful description for the route.
  - **Tags:** Enter a list of the tags that you want to associate with the route, if required.
6. Click **Create**.

The route is created.

To create a route using the CLI, use the `opc compute route add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a route using the API, use the `POST /network/v1/route/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

You can also create a route by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

## Listing Routes

After creating routes, you can view a list of routes along with information about the destination of each route, the vNICset used by the route, and the administrative distance of the route.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Routes**.

The Routes page displays a list of routes along with information about each route such as the destination IP address prefix, the vNICset used by the route, and the administrative distance of the route.

To list routes using the CLI, use the `opc compute route list` command. See *For help with that command, run the command with the -h option. For the instructions to install the CLI client, see Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list routes using the API, use the `GET /network/v1/route/container/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating a Route


After you've created a route, if required, you can update the route to change the destination, the next hop vNICset, or the administrative distance of the route.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Routes**.
4. Go to the route that you want to update, and from the  menu, select **Update**.
5. Update the required information:
  - **Administrative Distance:** Enter 0, 1, or 2 to specify the administrative distance of the route. The administrative distance indicates the priority of a route. The highest priority is 0. When multiple routes to a destination exist, the route with the highest priority is used. When multiple routes with the highest priority exist, all those routes are used.
  - **IP Address Prefix:** Enter the IP address prefix, in CIDR format, of the destination network that you want to specify the route to. The destination IP address prefix can be either another IP network, or an external network or host.
  - **Next Hop vNICset:** Select the vNICset that you want to use to route packets to the specified destination. When a vNICset containing multiple vNICs is used in a route, Equal Cost Multipath (ECMP) anycast routing is implemented. Traffic routed by that route is load balanced across all the vNICs in the vNICset. Using vNICsets with multiple vNICs also ensures high availability for traffic across the specified vNICs.
  - **Description:** Update the description for the route, if required.
  - **Tags:** Update the list of tags associated with the route, if required.
6. Click **Update**.

The route is updated.

To update a route using the CLI, use the `opc compute route update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a route using the API, use the `PUT /network/v1/route/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a Route

If you no longer need to use a specified route to a destination, you can delete the route. If no routes to a destination are specified, the default route will be used.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Routes**.
4. Identify the route that you want to delete. From the  menu, select **Delete**.

To delete a route using the CLI, use the `opc compute route delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a route using the API, use the `DELETE /network/v1/route/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing IP Address Prefix Sets

### Topics

- [Creating an IP Address Prefix Set](#)
- [Listing IP Address Prefix Sets](#)

- [Updating an IP Address Prefix Set](#)
- [Deleting an IP Address Prefix Set](#)

## Creating an IP Address Prefix Set

An **IP address prefix set** contains a set of IPv4 addresses in the CIDR address prefix format. When you create a security rule, you can specify a list of IP address prefix sets as the source or destination for permitted traffic.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Address Prefix Sets**.
4. Click **Create IP Address Prefix Set**.
5. Select or enter the required information:
  - **Name:** Enter a name for the IP address prefix set.
  - **IP Address Prefixes:** Enter a set of IPv4 addresses in CIDR address prefix format.  
The maximum number of IP address prefixes that you can specify in an IP address prefix set is limited to 2047.
  - **Description:** Enter a meaningful description for the IP address prefix set.
  - **Tags:** Enter one or more tags to help you identify the IP address prefix set.
6. Click **Create**.

The IP address prefix set is created.

To create an IP address prefix set using the CLI, use the `opc compute ip-address-prefix-set add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To create an IP address prefix set using the API, use the `POST /network/v1/ipaddressprefixset/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create an IP address prefix set by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating an IP address prefix set, to update or delete the IP address prefix set, see [Updating an IP Address Prefix Set](#) or [Deleting an IP Address Prefix Set](#). To use an IP address prefix set in a security rule, see [Creating a Security Rule for IP Networks](#).



## Listing IP Address Prefix Sets

After creating IP address prefix sets, you can view a list of your IP address prefix sets along with information about the IP address prefixes in each set and the security rules that each IP address prefix set is used in.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Address Prefix Sets**.

The IP Address Prefix Sets page displays a list of IP address prefix sets, along with information about each IP address prefix set, such as its name, description, the IP address prefixes contained in this set, and the security rules that specify this IP address prefix set as a source or destination.

To list IP address prefix sets using the CLI, use the `opc compute ip-address-prefix-set list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To list IP address prefix sets using the API, use the `GET /network/v1/ipaddressprefixset/{container}/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After listing IP address prefix sets, to update or delete an IP address prefix set, see [Updating an IP Address Prefix Set](#) or [Deleting an IP Address Prefix Set](#). To use an IP address prefix set in a security rule, see [Creating a Security Rule for IP Networks](#).

## Updating an IP Address Prefix Set


After creating an IP address prefix set, if required, you can modify the IP address prefixes in that set. You can also change the description or tags of an IP address prefix set.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Address Prefix Sets**.
4. Go to the IP address prefix set that you want to update, and from the  menu, select **Update**.
5. Update the information, as required:
  - **IP Address Prefixes:** Enter the set of IPv4 addresses in CIDR address prefix format.  
The maximum number of IP address prefixes that you can specify in an IP address prefix set is limited to 2047.
  - **Description:** Update the description, if required.
  - **Tags:** Update the tags, if required.
6. Click **Update**.  
The IP address prefix set is updated.

To update an IP address prefix set using the CLI, use the `opc compute ip-address-prefix-set update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an IP address prefix set using the API, use the `PUT /network/v1/ipaddressprefixset/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After updating an IP address prefix set, to use the IP address prefix set in a security rule, see [Creating a Security Rule for IP Networks](#).

## Deleting an IP Address Prefix Set

If you no longer use an IP address prefix set as a source or destination in any security list, you can delete the IP address prefix set.

### Prerequisites

- Ensure that the IP address prefix set that you want to delete isn't referenced in any security rule. If you delete an IP address prefix set that is referenced in a security rule, that security rule won't be used.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Address Prefix Sets**.
4. Go to the IP address prefix set that you want to delete, and from the  menu, select **Delete**.

To delete an IP address prefix set using the CLI, use the `opc compute ip-address-prefix-set delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete an IP address prefix set using the API, use the `DELETE /network/v1/ipaddressprefixset/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

# Managing Security Protocols for IP Networks

## Topics

- [Creating a Security Protocol for IP Networks](#)
- [Listing Security Protocols for IP Networks](#)
- [Updating a Security Protocol for IP Networks](#)
- [Deleting a Security Protocol for IP Networks](#)

## Creating a Security Protocol for IP Networks

A **security protocol** allows you to specify a transport protocol and the source and destination ports to be used with the specified protocol. When you create a security rule, you can specify the security protocols that you want to use to permit traffic. Only packets that match the transport protocol and ports in any of the specified security protocols will be permitted.

In a security protocol, you can specify a maximum of 32 port numbers or port range strings for **Source Port Set** and **Destination Port Set**.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Protocols**.
4. Click **Create Security Protocol**.
5. Select or enter the required information:
  - **Name:** Enter a name for the security protocol.
  - **IP Protocol:** Select a protocol or enter a number in the range 0–254 to represent the protocol that you want to specify. See *Assigned Internet Protocol Numbers* (<http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>).  
Traffic is enabled by a security rule when the protocol in the packet matches the protocol specified here. If no protocol is specified, all protocols are allowed.
  - **Source Port Set:** Enter a list of port numbers or port range strings. Traffic is enabled by a security rule when a packet's source port matches the ports specified here.  
For TCP, SCTP, and UDP, each port is a source transport port, between 0 and 65535, inclusive. For ICMP, each port is an ICMP type, between 0 and 255, inclusive.  
If no source ports are specified, all source ports or ICMP types are allowed.

- **Destination Port Set:** Enter a list of port numbers or port range strings. Traffic is enabled by a security rule when a packet's destination port matches the ports specified here.  
For TCP, SCTP, and UDP, each port is a destination transport port, between 0 and 65535, inclusive. For ICMP, each port is an ICMP type, between 0 and 255, inclusive.  
  
If no destination ports are specified, all destination ports or ICMP types are allowed.
- **Description:** Enter a meaningful description for the security protocol.
- **Tags:** Enter one or more tags to help you identify the security protocol.

6. Click **Create**.

The security protocol is created.

To create a security protocol using the CLI, use the `opc compute security-protocol add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a security protocol using the API, use the `POST /network/v1/secprotocol/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

You can also create a security protocol by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating a security protocol, to update or delete a security protocol, see [Updating a Security Protocol for IP Networks](#) or [Deleting a Security Protocol for IP Networks](#). To use a security protocol in a security rule, see [Creating a Security Rule for IP Networks](#).

## Listing Security Protocols for IP Networks

After creating security protocols for IP networks, you can view a list of security protocols along with information about each security protocol, such as the specified transport protocol and source and destination ports.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles](#) in *Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Protocols**.

The Security Protocols page displays a list of security protocols, along with information about each protocol.

To list security protocols using the CLI, use the `opc compute security-protocol list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list security protocols using the API, use the `GET /network/v1/secprotocol/container/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After listing security protocols, to update or delete a security protocol, see [Updating a Security Protocol for IP Networks](#) or [Deleting a Security Protocol for IP Networks](#). To use a security protocol in a security rule, see [Creating a Security Rule for IP Networks](#).

## Updating a Security Protocol for IP Networks

After creating a security protocol, if required, you can change the transport protocol or the source and destination ports specified in the security protocol.


In a security protocol, you can specify a maximum of 32 port numbers or port range strings for **Source Port Set** and **Destination Port Set**.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

#### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Protocols**.
4. Go to the security protocol that you want to update, and from the  menu, select **Update**.
5. Update the information, as required:
  - **IP Protocol:** Select a protocol or enter a number between 0 and 254. Traffic is enabled by a security rule when the protocol in the packet matches the protocol specified here. If no protocol is specified, all protocols are allowed.
  - **Source Port Set:** Enter a list of port numbers or port range strings. Traffic is enabled by a security rule when a packet's source port matches the ports specified here.  
For TCP, SCTP, and UDP, each port is a source transport port, between 0 and 65535, inclusive. For ICMP, each port is an ICMP type, between 0 and 255, inclusive.

If no source ports are specified, all source ports or ICMP types are allowed.

- **Destination Port Set:** Enter a list of port numbers or port range strings. Traffic is enabled by a security rule when a packet's destination port matches the ports specified here.  
For TCP, SCTP, and UDP, each port is a destination transport port, between 0 and 65535, inclusive. For ICMP, each port is an ICMP type, between 0 and 255, inclusive.

If no destination ports are specified, all destination ports or ICMP types are allowed.

- **Description:** Update the description, if required.
- **Tags:** Update the tags, if required.

6. Click **Update**.

The security protocol is updated.

To update a security protocol using the CLI, use the `opc compute security-protocol update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a security protocol using the API, use the `PUT /network/v1/secprotocol/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After updating a security protocol, to use the security protocol in a security rule, see [Creating a Security Rule for IP Networks](#).

## Deleting a Security Protocol for IP Networks

If you no longer use a security protocol in any security rule, you can delete the security protocol.

### Prerequisites

- Ensure that the security protocol that you want to delete isn't referenced in any security rule. If you delete a security protocol that is referenced in a security rule, that security rule won't be used.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles](#) in *Managing and Monitoring Oracle Cloud*.


 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Protocols**.
4. Go to the security protocol that you want to delete, and from the  menu, select **Delete**.

To delete a security protocol using the CLI, use the `opc compute security-protocol delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a security protocol using the API, use the `DELETE /network/v1/secprotocol/name` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

## Managing Security Rules for IP Networks

### Topics

- [Creating a Security Rule for IP Networks](#)
- [Listing Security Rules for IP Networks](#)
- [Updating a Security Rule for IP Networks](#)
- [Deleting a Security Rule for IP Networks](#)

## Creating a Security Rule for IP Networks

A **security rule** permits traffic from a specified source or to a specified destination. You must specify the direction of a security rule — either ingress or egress. In



addition, you can specify the source or destination of permitted traffic, and the security protocol and port used to send or receive packets. Each of the parameters that you specify in a security rule provides a criterion that the type of traffic permitted by that rule must match. Only packets that match all of the specified criteria are permitted. If you don't specify match criteria for any parameter, all traffic for that parameter is permitted. For example, if you don't specify a security protocol, then traffic using any protocol and port is permitted.

In a security rule, you can specify a maximum of 32 security protocols, 32 source IP address prefix sets, and 32 destination IP address prefix sets.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Rules**.
4. Click **Create Security Rule**.
5. Select or enter the required information:
  - **Name:** Enter a name for the security rule.
  - **Status:** Security rules are enabled by default. To disable a security rule, select **Disabled**.
  - **Type:** Select the direction of flow of traffic for this security rule.
  - **Access Control List:** Select the access control list that you want to add this security rule to. Security rules are applied to vNICsets by using ACLs.
  - **Security Protocols:** Select a list of security protocols for which you want to permit traffic. Only packets that match the specified protocols and ports are permitted. When no security protocols are specified, traffic using any protocol over any port is permitted.
  - **Source IP Address Prefix Sets:** Enter a list of IP address prefix sets from which you want to permit traffic. Only packets from IP addresses in the specified IP address prefix sets are permitted. When no source IP address prefix sets are specified, traffic from any IP address is permitted.
  - **Source vNICset:** Select the vNICset from which you want to permit traffic. Only packets from vNICs in the specified vNICset are permitted. When no source vNICset is specified, traffic from any vNIC is permitted.
  - **Destination IP Address Prefix Sets:** Enter a list of IP address prefix sets to which you want to permit traffic. Only packets to IP addresses in the specified IP address prefix sets are permitted. When no destination IP address prefix sets are specified, traffic to any IP address is permitted.
  - **Destination vNICset:** Select the vNICset to which you want to permit traffic. Only packets to vNICs in the specified vNICset are permitted. When no destination vNICset is specified, traffic to any vNIC is permitted.
  - **Description:** Enter a meaningful description for the security rule.

- **Tags:** Enter one or more tags to help you identify the security rule.

To create a security rule using the CLI, use the `opc compute security-rule add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To create a security rule using the API, use the `POST /network/v1/secrule/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create a security rule by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating a security rule, to update or delete the security rule, see [Updating a Security Rule for IP Networks](#) or [Deleting a Security Rule for IP Networks](#).

## Listing Security Rules for IP Networks

After creating a security rule, you can view a list of your security rules for IP networks along with information about each security rule.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Rules**.

The Security Rules page displays a list of security rules for IP networks. You can view information about each security rule such as whether a rule is an ingress rule or an egress rule, and whether a rule is enabled or disabled. You can also see the ACL that a security rule references as well as the security protocol, source, and destination specified in each rule, if any.

To list security rules using the CLI, use the `opc compute security-rule list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

To list security rules using the API, use the `GET /network/v1/secrule/container/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

After creating a security rule, to update or delete the security rule, see [Updating a Security Rule for IP Networks](#) or [Deleting a Security Rule for IP Networks](#).

## Applying a Security Rule for IP Networks

After you've created a security rule for using with IP networks, you can apply this security rule to one or more specified vNICsets. If you don't apply a security rule, the security rule isn't used.

To apply a security rule, do the following:

1. Reference an ACL in the security rule. A security rule can reference only one ACL, so plan your security rules and ACLs carefully. You can reference an ACL in a security rule either while creating the security rule, or later, by updating the security rule. See [Creating a Security Rule for IP Networks](#) or [Updating a Security Rule for IP Networks](#).
2. Apply the ACL to the required vNICsets. You can apply an ACL to a vNICset by specifying the required ACL either while creating the vNICset, or later, by updating the vNICset. See [Creating a vNICset](#) or [Updating a vNICset](#).

## Updating a Security Rule for IP Networks

After creating a security rule, if required you can modify the security rule.


In a security rule, you can specify a maximum of 32 security protocols, 32 source IP address prefix sets, and 32 destination IP address prefix sets.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

#### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Rules**.
4. Go to the security rule that you want to update, and from the  menu, select **Update**.
5. Update the information as required:
  - **Status:** Security rules are enabled by default. To disable a security rule, select **Disabled**.
  - **Type:** Update the direction of flow of traffic for this security rule, if required.
  - **Access Control List:** Select the access control list that you want to add this security rule to. Security rules are applied to vNICsets by using ACLs.
  - **Security Protocols:** Select a list of security protocols for which you want to permit traffic. Only packets that match the specified protocols and ports are

permitted. When no security protocols are specified, traffic using any protocol over any port is permitted.

- **Source IP Address Prefix Sets:** Enter a list of IP address prefix sets from which you want to permit traffic. Only packets from IP addresses in the specified IP address prefix sets are permitted. When no source IP address prefix sets are specified, traffic from any IP address is permitted.
  - **Source vNICset:** Select the vNICset from which you want to permit traffic. Only packets from vNICs in the specified vNICset are permitted. When no source vNICset is specified, traffic from any vNIC is permitted.
  - **Destination IP Address Prefix Sets:** Enter a list of IP address prefix sets to which you want to permit traffic. Only packets to IP addresses in the specified IP address prefix sets are permitted. When no destination IP address prefix sets are specified, traffic to any IP address is permitted.
  - **Destination vNICset:** Select the vNICset to which you want to permit traffic. Only packets to vNICs in the specified vNICset are permitted. When no destination vNICset is specified, traffic to any vNIC is permitted.
  - **Description:** Update the description, if required.
  - **Tags:** Update the tags, if required.
6. Click **Update**. The security rule is updated.

To update a security rule using the CLI, use the `opc compute security-rule update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a security rule using the API, use the `PUT /network/v1/secrule/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a Security Rule for IP Networks

If you no longer need a security rule, you can delete it.

### ▲ Caution:

Default ingress and egress security rules exist to allow communication between vNICs in the default vNICset. These default security rules belong to the default ACL and are applied to the default vNICset. If you delete either of these default security rules, ensure that you have other security rules or other ACLs in place to permit communication to and from the vNICs in the default vNICset. Otherwise communication with these vNICs will be blocked.

### Prerequisites


- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Security Rules**.
4. Go to the security rule that you want to delete, and from the  menu, select **Delete**.

To delete a security rule using the CLI, use the `opc compute security-rule delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

## Managing ACLs

### Topics

- [Creating an ACL](#)
- [Listing ACLs](#)
- [Adding a Security Rule to an ACL](#)
- [Applying an ACL to a vNICset](#)
- [Updating an ACL](#)
- [Deleting an ACL](#)

## Creating an ACL

An **access control list (ACL)** is a collection of security rules that can be applied to a vNICset. ACLs determine whether a packet can be forwarded to or from a vNIC, based on the criteria specified in its security rules. When you create a security rule, you specify the ACL that it belongs to. ACLs apply to vNICsets. Each vNICset can reference multiple ACLs and each ACL can be referenced in multiple vNICsets. When

an ACL is referenced in a vNICset, every security rule that belongs to the ACL applies to every vNIC that is specified in the vNICset.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Access Control Lists**.
4. Click **Create Access Control List**.
5. Select or enter the required information:
  - **Name:** Enter a name for the ACL.
  - **Status:** ACLs are enabled by default. To disable an ACL, select **Disabled**.
  - **Description:** Enter a meaningful description for the ACL.
  - **Tags:** Enter one or more tags to help you identify the ACL.

To create an ACL using the CLI, use the `opc compute acl add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create an ACL using the API, use the `POST /network/v1/acl/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

You can also create an ACL by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After creating an ACL:

- To update or delete an ACL, see [Updating an ACL](#) or [Deleting an ACL](#).
- To use an ACL in a security rule, see [Creating a Security Rule for IP Networks](#) or [Updating a Security Rule for IP Networks](#).
- To apply an ACL to a vNICset, see [Creating a vNICset](#) or [Updating a vNICset](#).

## Listing ACLs

After creating access control lists (ACLs), you can view a list of your ACLs along with information about each ACL such as its status and the security rules it contains.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Access Control Lists**.

The Access Control Lists page displays a list of ACLs along with information about each ACL such as its status and the security rules that are added to each ACL. To list ACLs using the CLI, use the `opc compute acl list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list ACLs using the API, use the `GET /network/v1/acl/container/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After listing ACLs:

- To update or delete an ACL, see [Updating an ACL](#) or [Deleting an ACL](#).
- To use an ACL in a security rule, see [Creating a Security Rule for IP Networks](#) or [Updating a Security Rule for IP Networks](#).
- To apply an ACL to a vNICset, see [Creating a vNICset](#) or [Updating a vNICset](#).

## Adding a Security Rule to an ACL

After you've created the access control lists (ACLs) that you want to use in your IP networks, to add security rules to an ACL, reference the ACL in each of the required security rules. If you don't specify any ACL in a security rule, that security rule isn't used.

A security rule can reference only one ACL, so plan your security rules and ACLs carefully. You can reference an ACL in a security rule either while creating the security rule, or later, by updating the security rule. See [Creating a Security Rule for IP Networks](#) or [Updating a Security Rule for IP Networks](#).

## Applying an ACL to a vNICset

To apply an ACL, reference it in one or more vNICsets when you create or update vNICsets. When an ACL is referenced in a vNICset, every security rule that references that ACL is applied to every vNIC in that vNICset.

See [Creating a vNICset](#) or [Updating a vNICset](#).

## Updating an ACL


After creating an ACL, if required, you can modify the description and tags associated with it, or change its status to disabled or enabled.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Access Control Lists**.
4. Go to the ACL that you want to update, and from the  menu, select **Update**.
5. Update the required information:
  - **Status:** ACLs are enabled by default. To disable an ACL, select **Disabled**.
  - **Description:** Update the description, if required.
  - **Tags:** Update the tags, if required.
6. Click **Update**. The ACL is updated.

To update an ACL using the CLI, use the `opc compute acl update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an ACL using the API, use the `PUT /network/v1/acl/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After updating an ACL:

- To use an ACL in a security rule, see [Creating a Security Rule for IP Networks](#) or [Updating a Security Rule for IP Networks](#).
- To apply an ACL to a vNICset, see [Creating a vNICset](#) or [Updating a vNICset](#).

## Deleting an ACL

If you no longer need to use an ACL, you can delete it. Remember, however, that security rules reference ACLs and ACLs are applied to vNICsets. If you delete an ACL that is referenced in one or more security rules, those security rules can no longer be used. If you delete an ACL that is applied to a vNICset, the security rules in that ACL no longer apply to that vNICset.

Before deleting an ACL, ensure that other ACLs are in place to provide access to relevant vNICsets. If you delete all the ACLs applied to a vNICset, some vNICs in that vNICset might become unreachable.



 **Caution:**

A default ACL is applied to the default vNICset. This ACL allows communication between vNICs in the default vNICset. If you delete the default ACL, it can cause all communication to and from vNICs in the default vNICset to be blocked.

**Prerequisites**

- Ensure that the ACL that you want to delete isn't referenced in any security rule that you want to use.
- Ensure that vNICs in vNICsets that the ACL applies to don't become unreachable by deleting the ACL.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).


 **Note:**

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

**Procedure**

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Access Control Lists**.
4. Go to the ACL that you want to delete, and from the  menu, select **Delete**.

To delete an ACL using the CLI, use the `opc compute acl delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To delete an ACL using the API, use the `DELETE /network/v1/acl/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Managing Public IP Addresses

### Topics

- [Reserving a Public IP Address for IP Networks](#)
- [Listing IP Reservations for IP Networks](#)
- [Updating an IP Reservation for IP Networks](#)
- [Associating a Public IP Address with a vNIC](#)
- [Removing an IP Reservation from a vNIC](#)
- [Deleting an IP Reservation for IP Networks](#)

## Reserving a Public IP Address for IP Networks

When an instance has an interface on an IP network, you can specify a public IP address to be associated with that interface. An IP reservation allows you to reserve a public IP address from a specified IP pool. When you create an IP reservation, you can associate the public IP address with a specified vNIC on an instance.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Reservations**.
4. Click **Create IP Reservation**.
5. Select or enter the required information:
  - **Name:** Enter a name for the IP reservation.
  - **IP Pool:** Select the required IP pool.  
If you select **public-ippool**, then your instance can communicate with external hosts over the public Internet.

If you select **cloud-ippool**, then your instance can communicate with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region, without sending traffic over the public Internet. You can use this IP address to connect your instance only to service endpoints. You can't connect your instance to another instance using this IP address.

- **For Instance:** Select the instance that you want to associate this IP reservation with. An IP address is associated with a vNIC on an instance. After

you select the instance, you must also select the vNIC on that instance, that you want to associate this IP reservation with.

- **vNIC:** Select the vNIC that you want to associate this IP reservation with. If you haven't selected an instance, this list shows all available vNICs. If you've selected an instance, this field shows available vNICs on the specified instance. If you don't select a vNIC, the IP reservation isn't associated with any vNIC. You can associate the IP reservation with a vNIC later, by updating the IP reservation.
- **Description:** Enter a meaningful description for the IP reservation.
- **Tags:** Enter one or more tags to help you identify the IP reservation.

6. Click **Create**.

The IP reservation is created.

 **Note:**

You can view the public IP addresses associated with each interface of an instance on the instance details page.

To reserve a public IP address using the CLI, use the `opc compute ip-address-reservation add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To reserve a public IP address using the API, use the `POST /network/v1/ipreservation/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

You can also create an IP reservation by using an orchestration. See [Orchestration v1 Attributes Specific to Each Object Type](#) or [Orchestration v2 Attributes Specific to Each Object Type](#).

After reserving a public IP address:

- To update an IP reservation, see [Updating an IP Reservation for IP Networks](#)
- To associate an IP reservation with a vNIC, see [Associating a Public IP Address with a vNIC](#).
- To remove an IP reservation from a vNIC, see [Removing an IP Reservation from a vNIC](#).
- To delete an IP reservation, see [Deleting an IP Reservation for IP Networks](#).

## Listing IP Reservations for IP Networks

After creating IP reservations for IP networks, you can view a list of IP reservations along with information about each IP reservation such as the public IP address and the vNIC it is associated with, if any.

To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system

administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Reservations**.

The IP Reservations page displays a list of IP reservations, along with information about each IP reservation.

To list IP reservations using the CLI, use the `opc compute ip-address-reservation list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list IP reservations using the API, use the `GET /network/v1/ipreservation/container/` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

After listing IP reservations:

- To update an IP reservation, see [Updating an IP Reservation for IP Networks](#)
- To associate an IP reservation with a vNIC, see [Associating a Public IP Address with a vNIC](#).
- To remove an IP reservation from a vNIC, see [Removing an IP Reservation from a vNIC](#).
- To delete an IP reservation, see [Deleting an IP Reservation for IP Networks](#).

## Updating an IP Reservation for IP Networks


After creating an IP reservation, if required, you can add, remove, or change the vNIC associated with the public IP address. You can also modify the description of the IP reservation.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

#### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to update an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state. See [Workflows for Updating Orchestrations v2](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Reservations**.
4. Go to the IP reservation that you want to update, and from the  menu, select **Update**.
5. Update the information, as required:
  - **IP Pool:** Modify the IP pool that you've used for the IP reservation, if required. If you select **public-ippool**, then your instance can communicate with external hosts over the public Internet. If you select **cloud-ippool**, then your instance can communicate with other Oracle Cloud services, such as the REST endpoint of an Oracle Cloud Infrastructure Object Storage Classic account in the same region, without sending traffic over the public Internet.

 **Note:**

If you modify the IP pool, the IP address associated with this IP reservation will change.

- **For Instance:** Select the instance that you want to associate this IP reservation with. An IP address is associated with a vNIC on an instance. After you select the instance, you must also select the vNIC on that instance, that you want to associate this IP reservation with. If the IP reservation is already associated with an instance, you can remove the instance or select another instance to associate this IP reservation with.
- **vNIC:** Select the vNIC that you want to associate this IP reservation with. If you haven't selected an instance, this list shows all available vNICs. If you've selected an instance, this field shows available vNICs on the specified instance. If you want to remove the IP reservation from an instance, remove the vNIC that the IP reservation is associated with.
- **Description:** Update the description, if required.
- **Tags:** Update the tags, if required.

To update an IP reservation using the CLI, use the `opc compute ip-address-reservation update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update an IP reservation using the API, use the `PUT /network/v1/ipreservation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Associating a Public IP Address with a vNIC

You can associate a public IP address with a vNIC of an instance either while creating the instance or when an instance is already running.

- When you create an IP reservation using the web console, you can associate the IP reservation with a vNIC on an existing instance. See [Reserving a Public IP Address for IP Networks](#).
- If you've already created an IP reservation, then you can associate this IP reservation with an instance while creating the instance. This ensures that the vNIC is associated with the specified IP reservation whenever the instance is created or re-created. See [Creating Instances](#).
- If you've already created an IP reservation and you want to associate it with a vNIC on a running instance, you can update the IP reservation using the web console and select the required instance vNIC. See [Updating an IP Reservation for IP Networks](#).

To associate an IP reservation with a vNIC using the CLI, use the `opc compute ip-address-association add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in \*CLI Reference for Oracle Cloud Infrastructure Compute Classic\*](#).

To associate an IP reservation with a vNIC using the API, use the `POST /network/v1/ipassociation/` method. See [REST API for Oracle Cloud Infrastructure Compute Classic](#).

 **Note:**

When you attach an IP reservation to a running instance, then if you delete and re-create or shut down and restart the instance, the IP reservation reverts to whatever was specified while creating the instance and any updates made to the IP reservation are lost. You must update the IP reservation again.


## Removing an IP Reservation from a vNIC

If you associate an IP reservation with a vNIC while creating or updating the IP reservation, then you can remove the IP reservation from the vNIC by updating the IP reservation.

 **Note:**

However, if you associate an IP reservation with an instance while creating the instance, then to remove the IP reservation, update the instance orchestration. Otherwise, whenever your instance orchestration is stopped and restarted, the IP reservation will again be associated with the vNIC.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Reservations**.
4. Go to the IP reservation that you want to remove, and from the  menu, select **Remove vNIC Association**.

To remove an IP reservation from a vNIC using the CLI, use the `opc compute ip-address-association delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To remove an IP reservation from a vNIC using the API, use the `DELETE /network/v1/ipassociation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting an IP Reservation for IP Networks

If you no longer need a public IP address that you've reserved, you can delete the IP reservation.

### Prerequisites

- Ensure that the IP reservation that you want to delete isn't associated with a vNIC.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.


### Note:

You should always use your orchestrations to manage resources that you've created using orchestrations. Don't, for example, use the web console or the CLI or REST API to delete an object that you created using an orchestration. This could cause your orchestration to either attempt to re-create the object and associated resources, or to go into an error state.

If you created the object using orchestration v1, then you can delete the object by terminating the orchestration. See [Terminating an Orchestration v1](#).

If you created the object using an orchestration v2, then you can delete the object by suspending, terminating, or updating the orchestration. See [Suspending an Orchestration v2](#), [Terminating an Orchestration v2](#), or [Updating an Orchestration v2](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **IP Reservations**.
4. Go to the IP reservation that you want to delete, and from the  menu, select **Delete**.

To delete an IP reservation using the CLI, use the `opc compute ip-address-reservation delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete an IP reservation using the API, use the `DELETE /network/v1/ipreservation/name` method. See *REST API for Oracle Cloud Infrastructure Compute Classic*.



# 12

## Accessing an Oracle Linux Instance Using SSH

If you've created your instance using an Oracle-provided Oracle Linux image, then you can log in to your instance using SSH as the `opc` user.

If you've created your instance using a custom machine image, then ensure that you've added a script to copy SSH public keys to the appropriate files for default users. This script must run automatically when your instance starts. It must retrieve the SSH public keys from the metadata stored in the instance, and copy these keys to the following path for one or more default users: `/home/user/.ssh/authorized_keys`. For information about retrieving SSH public keys, see [Retrieving Instance Metadata](#).

### Topics

- [Accessing an Instance from UNIX and UNIX-Like Systems](#)
- [Accessing an Instance from Windows](#)
- [Adding Users on an Oracle Linux Instance](#)

## Accessing an Instance from UNIX and UNIX-Like Systems

You can log in to an Oracle-provided Oracle Linux instance as the default user, `opc`. The `opc` user has `sudo` privileges.

### Prerequisites

- Ensure that the SSH private key corresponding to the public key that you associated with your instance while creating it is available on the host from which you want to `ssh` to the instance.
- Ensure that the instance has a public IP address. See [Managing Public IP Addresses](#). To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).
- Ensure that a security rule exists to enable SSH access to your instance. See [Creating a Security Rule for IP Networks](#).

If your instance is on IP network, see [Workflow for Enabling SSH Access to an Instance Using IP Networks](#).

If your instance is on shared network, see  [Permitting SSH Access to Compute Classic Instances](#).

### Procedure

You can use SSH to log in to your instance as the default user, `opc`, by using the following command:

```
ssh opc@ip_address -i private_key
```

In this command, *ip\_address* is the public IP address of the instance, and *private\_key* is the full path and name of the file that contains the private key corresponding to the public key associated with the instance that you want to access.

 **Note:**

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

If an error occurs, see [Can't connect to an instance using SSH](#).

When you're logged in as the default user, `opc`, use the `sudo` command to run administrative tasks.

## Accessing an Instance from Windows

You can log in to an Oracle-provided Oracle Linux instance as the default user, `opc`. The `opc` user has `sudo` privileges. If you're using a Windows host, you can use PuTTY or any other similar client to connect to your instance using SSH.

### Prerequisites

- This procedure assumes you're using PuTTY to connect to your instance. Ensure that you have PuTTY installed on your Windows host. To download PuTTY, go to <http://www.putty.org/>.
- Ensure that the SSH private key corresponding to the public key that you associated with your instance while creating it is available on the Windows host from which you want to `ssh` to the instance.
- Ensure that the instance has a public IP address. See [Managing Public IP Addresses](#). To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).
- Ensure that a security rule exists to enable SSH access to your instance. See [Creating a Security Rule](#).

See  [Permitting SSH Access to Compute Classic Instances](#).

### Procedure

1. Run the PuTTY program.  
The PuTTY Configuration window is displayed, showing the Session panel.
2. In **Host Name (or IP address)** box, enter the public IP address of your instance.

 **Note:**

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

3. Confirm that the **Connection type** option is set to **SSH**.
4. In the Category tree, expand **Connection** if necessary and then click **Data**.  
The Data panel is displayed.
5. In **Auto-login username** box, enter `opc`.
6. Confirm that the **When username is not specified** option is set to **Prompt**.
7. In the Category tree, expand **SSH** and then click **Auth**.  
The Auth panel is displayed.
8. Click the **Browse** button next to the **Private key file for authentication** box. Navigate to and open the private key file that matches the public key that is associated with your instance.
9. In the Category tree, click **Session**.  
The Session panel is displayed.
10. In the **Saved Sessions** box, enter a name for this connection configuration and click **Save**.
11. Click **Open** to open the connection.  
The PuTTY Configuration window is closed and the PuTTY window is displayed.
12. If this is the first time you are connecting to an instance, the PuTTY Security Alert window is displayed, prompting you to confirm the public key. Click **Yes** to continue connecting.

If an error occurs, see [Can't connect to an instance using SSH](#).

When you're logged in as the default user, `opc`, use the `sudo` command to run administrative tasks.

## Adding Users on an Oracle Linux Instance

If you've created your instance using an Oracle-provided Oracle Linux image, then you can use SSH to access your Oracle-provided Oracle Linux instance from a remote host as the `opc` user. After logging in, you can add users on your instance.

### Note:

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any users that were added manually (that is, users that weren't defined in the machine image) must be added again.

1. Generate an SSH key pair for the new user. See [Generating an SSH Key Pair on UNIX and UNIX-Like Systems](#).
2. Copy the public key value to a text file. You'll use this key later in this procedure.
3. Log in to your instance. See [Accessing an Instance from UNIX and UNIX-Like Systems](#).

4. Become the `root` user.

```
sudo su
```

5. Create the new user:

```
useradd new_user
```

6. Create a `.ssh` directory in the new user's home directory.

```
mkdir /home/new_user/.ssh
```

7. Copy the SSH public key that you noted earlier to the `/home/new_user/.ssh/authorized_keys` file.

```
echo "key" > /home/new_user/.ssh/authorized_keys
```

Here, `key` is the SSH public key value from the key pair that you generated earlier, enclosed in double quotation marks.

8. Add the new user to the list of allowed users in the `/etc/ssh/sshd_config` file on your instance, by editing the `AllowUsers` parameter, as shown in the following example:

```
AllowUsers opc myadmin
```

In this example, the `AllowUsers` parameter already had the `opc` user. The `myadmin` user has now been added.

9. Change the owner and group of the `/home/username/.ssh` directory to the new user:

```
chown -R new_user:group /home/new_user/.ssh
```

10. Restart the SSH daemon on your instance.

```
/sbin/service sshd restart
```

11. To enable `sudo` privileges for the new user, edit the `/etc/sudoers` file by running the `visudo` command.

In `/etc/sudoers`, look for the following line:

```
%opc ALL=(ALL) NOPASSWD: ALL
```

Add the following line right after the preceding line:

```
%group_of_new_user ALL=(ALL) NOPASSWD: ALL
```

You can now log in as the new user:

```
ssh new_user@ip_address -i private_key
```

In this command, *ip\_address* is the public IP address of the instance, and *private\_key* is the full path and name of the file that contains the private key corresponding to the public key that you added to the `authorized_keys` file earlier in this procedure.

#### Note:

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

If an error occurs, see [Can't connect to an instance using SSH](#).

Use the `sudo` command to run administrative tasks.

#### See Also:

 [Creating an SSH-Enabled User on an Oracle Cloud Infrastructure Compute Classic Instance](#)

# 13

## Accessing an Oracle Solaris Instance Using SSH



This topic does not apply to Oracle Cloud at Customer.

In instances created by using any of the Oracle-provided Oracle Solaris images, a user named `opc` is preconfigured. The `opc` user is assigned the System Administrator profile and can perform basic administration tasks without entering a password by using `pfexec`.

### Prerequisites

- Ensure that the SSH private key corresponding to the public key that you associated with your instance while creating it is available on the host from which you want to `ssh` to the instance.
- Ensure that the instance has a public IP address. To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).
- Ensure that a security rule exists to enable SSH access to your instance. See [Creating a Security Rule](#).

### Procedure

You can use SSH to log in to your instance as the default user, `opc`, by using the following command:

```
ssh opc@ip_address -i private_key
```

- `ip_address` is the public IP address of the instance.  
If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)
- `private_key` is the full path and name of the file that contains the private key corresponding to the public key associated with the instance that you want to access.


If an error occurs, see [Can't connect to an instance using SSH](#).

When you're logged in as the `opc` user, you can use the `pfexec` command to run administrative tasks.

 **Note:**

Direct login as `root` is disabled. You can assume the `root` role by running `su -`. The password is `solaris_opc` and is marked as expired. You must change the password the first time that you assume the `root` role.

 **Tip:**

For instructions to add SSH-enabled users to the instance, see  [Creating an SSH-Enabled User on an Oracle Solaris Instance](#).

# Accessing a Windows Instance Using RDP

Remote desktop protocol (RDP) allows you to securely access your Windows instance from a remote host. To access a Windows instance from a Windows host, you can use the default RDP client, Remote Desktop Connection.

 **Note:**

This procedure assumes that your local host runs a Windows operating system and that you're using the Remote Desktop Connection client to access your Windows instance. If your local host has another operating system, use an appropriate RDP client to access your Windows instance.

## Topics

- [Accessing a Windows Instance on IP Network Using RDP](#)
- [Accessing a Windows Instance on Shared Network Using RDP](#)

## Accessing a Windows Instance on IP Network Using RDP

RDP access to your Windows instance on IP network is not enabled by default. Before accessing your Windows instance using RDP, you must create the following networking components: an ACL, a vNICset which contains your instance's vNIC and the created ACL is applied to it, ingress and egress security rules for IP network to enable RDP access. You don't need to perform this task if you have created your Windows instance using QuickStarts. When you use QuickStarts to create your Windows instance, it also creates all the networking objects that's required to access the instance over remote desktop protocol (RDP). You'll need to set up the required security rules and ACLs if you have created the Windows instance by using orchestrations or by using the Create Instance wizard.

## Prerequisites

- Ensure that you've created your Windows instance with the following configuration:
  - Has one interface on an IP network and is added to a vNICset. This interface should be specified as the default gateway for the instance.
  - Has one IP address from the `/oracle/public/public-ippool` IP address pool.
  - Doesn't have an interface on the shared network.
  - Has the required `userdata` attributes. See [Creating an Instance from the Instances Page](#) for information about the required attributes. See [Retrieving Instance Metadata](#) to find out how to view the metadata associated with your instance. If you're using an orchestration to manage your instance, you can



view the orchestration to check the specified attributes. See [Monitoring Orchestrations v1](#).

- The instance is in the running state.

#### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **IP Network**, and then click **Access Control Lists**.
4. Click **Create Access Control List**.
5. Select or enter the required information, and then click **Create**.
  - **Name:** Enter a name for the ACL.
  - **Status:** Select **Enabled** to enable the ACL.
  - **Description:** Enter a meaningful description for the ACL.
  - **Tags:** Enter one or more tags to help you identify the ACL.
6. In the **Network** drop-down list, expand **IP Network**, and then click **Virtual NIC Sets**.
7. Click **Create vNICset** to create a vNICset which contains the instance vNIC and to apply the ACL that you have created to this vNICset.
8. In the Create vNICset dialog box, select or enter the following, and then click **Create**.
  - **Name:** Enter a name for the vNICset.
  - **vNICs:** Select the vNIC of the instance that you want to access using RDP.
  - **Applied Access Control Lists:** Select the access control list that you have created. The selected ACL is applied to this vNICset. When you apply an ACL to a vNICset, all the security rules in that ACL are applied to traffic to or from each of the vNICs in the vNICset.
  - **Description:** Enter a meaningful description for the vNICset.
  - **Tags:** Enter a list of the tags that you want to associate with this vNICset.
9. In the **Network** drop-down list, expand **IP Network**, and then click **Security Rules**.
10. Click **Create Security Rule** to create an ingress security rule for IP network.
11. Select or enter the required information:
  - **Name:** Enter a name for the security rule.
  - **Status:** Security rules are enabled by default.
  - **Type:** Select **Ingress** as the direction of flow of traffic for this security rule.
  - **Access Control List:** Select the access control list that you have created. This security rule is added to the specified access control list. Security rules are applied to vNICsets by using ACLs.

- **Security Protocols:** Select **rdp** as the security protocol for which you want to permit traffic. Only packets that match the specified protocols and ports are permitted. When no security protocols are specified, traffic using any protocol over any port is permitted.
- **Destination vNICset:** Select the vNICset that you have created to permit traffic to this vNICset. Only packets to vNICs in the specified vNICset are permitted. When no destination vNICset is specified, traffic to any vNIC is permitted.

You must provide values for the specified fields while creating the security rule. It is optional to provide values for other fields that appear in the **Create Security Rule** dialog box.

12. Click **Create Security Rule** to create an egress security rule for IP network.

13. Select or enter the required information:

- **Name:** Enter a name for the security rule.
- **Status:** Security rules are enabled by default.
- **Type:** Select **Egress** as the direction of flow of traffic for this security rule.
- **Access Control List:** Select the access control list that you have created. This security rule is added to the specified access control list. Security rules are applied to vNICsets by using ACLs.

You must provide values for the specified fields while creating the security rule. It is optional to provide values for other fields that appear in the **Create Security Rule** dialog box.

The security rules that you have created are applied to the running instance.

14. Next, on your Windows local host, start Remote Desktop Connection.

- To start Remote Desktop Connection from the GUI:
  - Click the **Start** button and type `Remote Desktop` in the search field.
  - In the search result, click **Remote Desktop Connection**.
  - In the **Computer** field, enter the public IP address of your Windows instance and then click **Connect**.
- To start Remote Desktop Connection from the command line, enter:
  - `mstsc /v:public-IP-address-of-your-instance`

 **Note:**

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

The Remote Desktop Connection client starts.

15. In the Windows security dialog box, enter the user name and password that you specified in `userdata` attributes while creating the instance.

 **Note:**

The first time you log in to your Windows instance, you must log in as Administrator using the `administrator_password` that you specified while creating the instance. After logging in, you can specify a list of users who are allowed to access the Windows instance remotely using RDP. Subsequently, you can log in as one of the new users. Alternatively, you can provide `userdata` attributes while creating the instance, to add users with RDP access enabled. For more information, see [User Data Attributes Used on Windows Instances](#).

 **Note:**

You should change the `Administrator` password when you log in to your instance the first time. You can also add additional administrators and users who are enabled for remote access, so that even if you lose or forget the Administrator password, you don't get locked out of your instance. If your instance uses a persistent boot disk, any instance configuration, including tasks such as adding users or changing passwords, will be retained as long as the boot disk isn't deleted. However, if you're using a nonpersistent boot disk with your Windows instance, then if you terminate the orchestration and start it again later, the `Administrator` password will be reset to the password that you specified in the orchestration. This is true for any user password that you specify in an orchestration.

After you've logged in to your Windows instance, to change the administrator password, add users, enhance security, or perform other customization and configuration tasks, see the Windows Server documentation.

## Accessing a Windows Instance on Shared Network Using RDP

RDP access to your Windows instance on shared network is not enabled by default. Before accessing your Windows instance using RDP, you must add your instance to a security list and create a security rule to enable RDP access.

### Prerequisites


- Ensure that you've created your Windows instance with the required `userdata` attributes. See [Creating an Instance from the Instances Page](#) for information about the required attributes. See [Retrieving Instance Metadata](#) to find out how to view the metadata associated with your instance.

If you're using an orchestration to manage your instance, you can view the orchestration to check the specified attributes. If you have created your instance

using orchestration v2, see [Monitoring Orchestration v2](#). If you have created your instance using orchestration v1, see [Monitoring Orchestration v1](#).

- Ensure that your instance has a public IP address. See [Managing Public IP Addresses](#). To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **Shared Network**, and then click the **Security Lists**.
4. Click **Create Security List**.
5. Enter or select the required details and then click **Create**.
  - Name: Enter `Enable RDP access`
  - Description: Enter an appropriate description
  - Inbound policy: Retain the default setting, **deny**.
  - Outbound policy: Retain the default setting, **permit**.The `Enable RDP access` security list is created.
6. Click the **Instances** tab.
7. On the Instances page, identify the instance that you want to update. From the  menu, select **View**.
8. On the instance details page, click **Add to Security List**.
9. Select the **Enable RDP access** security list and click **Attach**.

The instance is added to the **Enable RDP access** security list.
10. Click the **Network** tab.
11. Click **Create Security Rule**.
12. Enter or select the required detail and then click **Create**.
  - Name: Enter an appropriate name.
  - Status: Retain the default setting, **Enabled**.
  - Security application: Select the predefined security application, **rdp**.
  - Source: From the **Security IP Lists** drop down list, select **public-internet**, or select any other security IP list as the source.
  - Destination: Select the **Enable RDP access** security list that you just created.
  - Description: Enter an appropriate description.
13. Next, on your Windows local host, start Remote Desktop Connection.
  - To start Remote Desktop Connection from the GUI:
    - Click the **Start** button and type `Remote Desktop` in the search field.
    - In the search result, click **Remote Desktop Connection**.

- In the **Computer** field, enter the public IP address of your Windows instance and then click **Connect**.
- To start Remote Desktop Connection from the command line, enter:
  - `mstsc /v:public-IP-address-of-your-instance`

 **Note:**

If you've enabled a VPN tunnel to your Compute Classic instances, you can use the private IP address of your instance to connect to the instance. To set up a VPN tunnel, see [Connecting to Instances in a Multitenant Site Using VPN](#), [Setting Up VPN Using VPNaaS](#), or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#). (Not available on Oracle Cloud at Customer)

The Remote Desktop Connection client starts.

14. In the Windows security dialog box, enter the user name and password that you specified in `userdata` attributes while creating the instance.

 **Note:**

The first time you log in to your Windows instance, you must log in as Administrator using the `administrator_password` that you specified while creating the instance. After logging in, you can specify a list of users who are allowed to access the Windows instance remotely using RDP. Subsequently, you can log in as one of the new users. Alternatively, you can provide `userdata` attributes while creating the instance, to add users with RDP access enabled. For more information, see [User Data Attributes Used on Windows Instances](#).

 **Note:**

You should change the Administrator password when you log in to your instance the first time. You can also add additional administrators and users who are enabled for remote access, so that even if you lose or forget the Administrator password, you don't get locked out of your instance. If your instance uses a persistent boot disk, any instance configuration, including tasks such as adding users or changing passwords, will be retained as long as the boot disk isn't deleted. However, if you're using a nonpersistent boot disk with your Windows instance, then if you terminate the orchestration and start it again later, the Administrator password will be reset to the password that you specified in the orchestration. This is true for any user password that you specify in an orchestration.

After you've logged in to your Windows instance, to change the administrator password, add users, enhance security, or perform other customization and configuration tasks, see the Windows Server documentation.

# Connecting to Instances in a Multitenant Site Using VPN



This topic does not apply to Oracle Cloud at Customer.

You can set up a VPN connection to establish a secure communication channel between your data center and your Compute Classic instances.

You can set up VPN access to your instances by using Corente Services Gateway on a Compute Classic instance. Corente is an Oracle-provided IPsec solution. Corente Services Gateway acts as a proxy to facilitate secure access and data transfer to your instances. All VPN connections to your multitenant Compute Classic site use a Corente Services Gateway instance in the cloud.

In your data center, you can use either a supported third-party VPN device, or Corente Services Gateway installed on a host.

## Note:

In Compute Classic accounts provisioned from 17.4.2 onwards, Corente VPN solutions might not be available, as these solutions are being deprecated. Use VPN as a Service (VPNaaS) instead. See [Setting Up a VPN Connection Using VPNaaS](#).

For Compute Classic accounts provisioned prior to 17.4.2, the following table provides the relevant documentation resource for each of the Corente VPN scenarios.

| Cloud Network  | Data Center Gateway      | Document                                                                               |
|----------------|--------------------------|----------------------------------------------------------------------------------------|
| IP network     | Third-party device       | <i>Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud</i>      |
| IP network     | Corente Services Gateway | <i>Setting Up VPN from a Corente Services Gateway to an IP Network in Oracle Cloud</i> |
| Shared network | Third-party device       | <i>Setting Up VPN From a Third-Party Gateway On-Premises to the Shared Network</i>     |
| Shared network | Corente Services Gateway | <i>Setting Up VPN from Corente Services Gateway On-Premises to the Shared Network</i>  |

To use the web console to set up or manage a VPN connection using a Corente Services Gateway in the cloud and a third-party device in your data center, see:

- [Setting Up VPN](#)
- [Managing VPN](#)

 **Note:**

If you've subscribed to Oracle Cloud Infrastructure Dedicated Compute Classic, then you can use the Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic service. See [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#).

## Setting Up VPN



This topic does not apply to Oracle Cloud at Customer.

### Topics

- [About Setting Up VPN](#)
- [Creating a Cloud Gateway](#)
- [Registering a Third-Party VPN Device](#)
- [Connecting the Cloud Gateway with the Third-Party Device](#)

 **Note:**

You must have the `Compute_Operations` role to access the pages under the **VPN** tab. If you don't have this role, you won't be able to view these pages.

## About Setting Up VPN



This topic does not apply to Oracle Cloud at Customer.

You can set up VPN access to Compute Classic instances by creating a Corente Services Gateway instance and connecting it with a certified third-party VPN device in your data center.

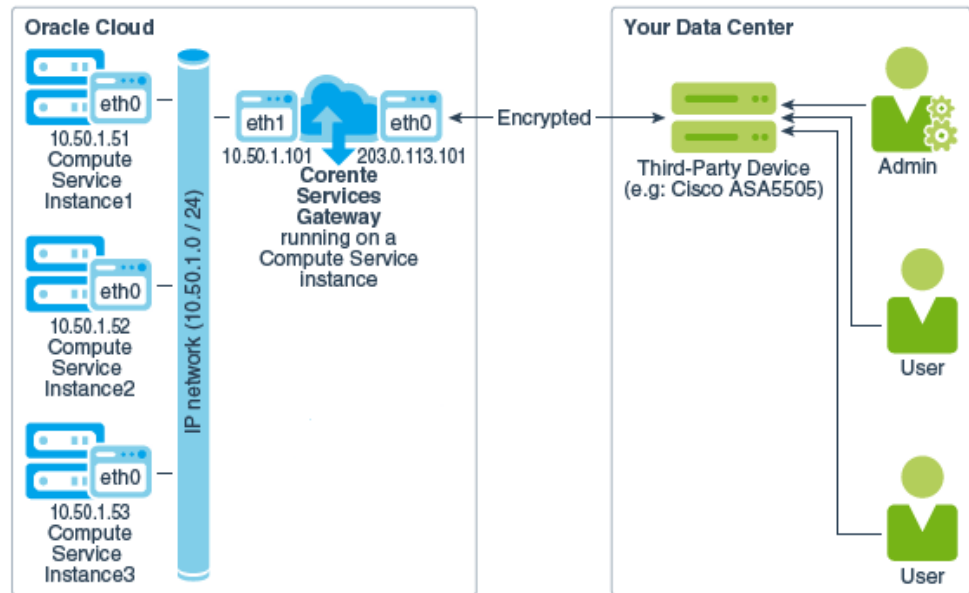
### Considerations for Setting Up a Single-Homed or Dual-Homed VPN Gateway

While setting up a VPN connection to your Compute Classic instances, consider whether the instances that you want to access will be on IP networks or on the shared network.

- Using IP networks allows you to define IP subnets in your account and isolate or enable traffic between subnets. By adding instances to IP networks, you can control the IP address assigned to each instance and you can also assign static IP addresses to each instance. See [About IP Networks](#).

If you want to access instances that are added to IP networks, you can create a dual-homed VPN gateway, which has one interface on the shared network and one interface on an IP network. With this gateway, you can use VPN to access all instances that are on the same IP network as the gateway instance.

The following figure shows a VPN connection between a third-party VPN device and a dual-homed cloud gateway. This gateway allows VPN access to instances on the same IP network.



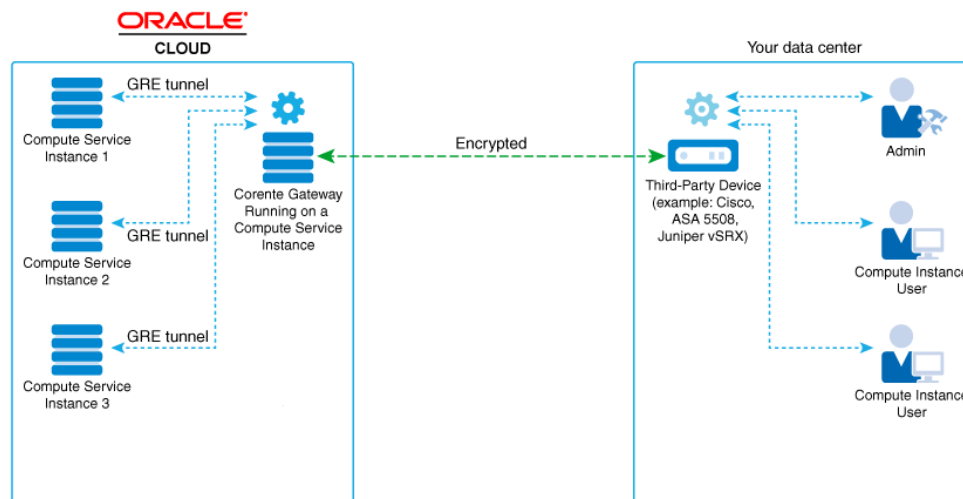
#### Note:

You can also extend VPN access to instances on other IP networks. However, if you want access to a large number of instances, it is recommended that you avoid setting up numerous IP networks with a /32 subnet. Instead, use a smaller number of IP networks with larger subnets. If you create a very large number of IP networks, a large number of IPSec security associations are required, which could cause performance degradation on some third-party devices. See [Workflow for Adding IP Networks to an Existing VPN Connection](#).

- If you don't need to set up IP networks and the instances that you want to access over VPN all have an interface on the shared network, then you can create a single-homed VPN gateway. After you've set up the VPN connection, you must configure a Generic Routing Encapsulation (GRE) tunnel from each instance to the gateway.

The following figure shows a VPN connection between a third-party VPN device and a single-homed cloud gateway. This gateway allows VPN access to instances on the shared network with a GRE tunnel between each instance and the gateway.





### VPN Scenarios Not Supported by the Compute Classic Web Console

You can use the web console to set up a VPN connection between your Corente Services Gateway instance and the third-party device in your data center. However, you *can't* use the web console to do the following:

- Connect a Corente Services Gateway instance in the cloud with a Corente Services Gateway instance in your data center. To do this, see *About Setting Up VPN Using Corente Services Gateway in Setting Up VPN from Corente Services Gateway On-Premises to the Shared Network* or *Solution Overview in Setting Up VPN from a Corente Services Gateway to an IP Network in Oracle Cloud*.
- Configure failover between two Corente Services Gateway instances to provide high availability. To do this, see *Configuring Active-Active HA in Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud*.
- If you want to add an IP network to an existing VPN connection, you can create the IP network and add it to an IP network exchange using the web console. However, you can't complete the steps to update user groups for your Corente Services Gateway and add a route on the gateway to the subnet of the newly added IP network using the web console. To complete these steps, you must use App Net Manager. See *Adding IP Networks to an Existing VPN Connection in Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud*.

### Workflow for Setting Up VPN

1. Configure a supported third-party VPN device at your data center. Device configuration varies depending on the type and model of your device. For supported configurations, see [Third-Party VPN Device Configuration](#).
2. Create a Corente Services Gateway instance in Compute Classic. See [Creating a Cloud Gateway](#).
3. Add information about your third-party VPN device. See [Registering a Third-Party VPN Device](#).
4. Create a connection between your Corente Services Gateway and your third-party device. See [Connecting the Cloud Gateway with the Third-Party Device](#).
5. If you created a single-homed VPN gateway instance, on each instance that you want to access, configure a GRE tunnel to the gateway. See [Configuring a GRE](#)

Tunnel on a Guest Instance in Oracle Cloud in *Setting Up VPN From a Third-Party Gateway On-Premises to the Shared Network*.

### Third-Party VPN Device Configuration

You can set up a VPN connection to any certified third-party device that allows interoperability with Corente Services Gateway. Devices must be configured for policy-based VPN.

The following table lists the certified third-party VPN device configurations.

| Certified Configurations                                                                                                                                                                                | Devices                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-256</li> <li>• DH phase 1 group 14</li> <li>• No Perfect Forward Secrecy (PFS); so no Diffie-Hellman (DH) phase 2 group</li> </ul> | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-256</li> <li>• DH phase 1 group 14; DH phase 2 group 14</li> </ul>                                                                 | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |
| <ul style="list-style-type: none"> <li>• Encryption AES128; Hash SHA-256</li> <li>• DH phase 1 group 14; no PFS</li> </ul>                                                                              | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |
| <ul style="list-style-type: none"> <li>• Encryption AES192; Hash SHA-1</li> <li>• DH phase 1 group 2, DH phase 2 group 2</li> </ul>                                                                     | Cisco ASA5505                                                                       |
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-1</li> <li>• DH phase 1 group 5; no PFS</li> </ul>                                                                                 | Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D               |

Other devices may work if they are configured with the certified configurations. Consider the following information while configuring your third-party device for a VPN connection.

- **Configuration Information**
  - The Corente Services Gateway uses IPsec and is behind a NAT, so network address translation traversal (NAT-T) is required. Ensure that the third-party device in your data center supports NAT-T. NAT-T requires UDP port 4500 to be open.
  - Devices must support and be configured for policy-based VPN.
  - Authentication: Pre-shared keys
  - Encryption: 3DES, AES-128, AES-192, AES-256
  - Hash: MD5, SHA-1, SHA-2

- Policy Group: Diffie-Hellman groups supported are 2, 5, 14, 15, 16, 17, 18, 22, 23, 24
  - ISAKMP: IKEv1 only. If IKEv2 is enabled by default, turn it off.
  - Exchange type: Main Mode (The cloud gateway uses **main mode** in phase one negotiations)
  - IPsec protocol: ESP, tunnel-mode
  - PFS: Enabled
  - It is highly recommended that the third-party device be configured to be responder-only.
  - Ensure the IKE and IPsec timeouts on the Corente Services Gateway and the third-party device are the same.
  - For Phase 1, ensure that the IKE ID on the Corente Services Gateway and the third-party device match.
- **HA Information**
    - When HA is configured, Dead Peer Detection (DPD) must be enabled to detect when a tunnel is down.
    - When HA is configured, asymmetric routing across the tunnels that make up the VPN connection will occur. Ensure that your firewall is configured to support this. If not, traffic will not be routed reliably.
    - Switching tunnels might take 30–40 seconds.

## Creating a Cloud Gateway



This topic does not apply to Oracle Cloud at Customer.

If you want to establish a VPN connection to your Compute Classic instances, start by creating a Corente Services Gateway instance.

### Prerequisites

- You must have already reserved the public IP address that you want to use with your gateway instance. See [Reserving a Public IP Address](#).
- If you want to add your VPN gateway instance to an IP network, you must create the IP network first. See [Creating an IP Network](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **VPN Gateways**.

4. Click **Create VPN Gateway**.
5. Select or enter the required information:
  - **Name:** Enter a name for the Corente Services Gateway instance.
  - **IP Reservation:** Select the IP reservation that you want to use with this instance. This is the public IP address of your VPN gateway.
  - **Image:** Select the machine image that you want to use to create the instance. You must select the most recent Corente Gateway image.
  - **Interface Type:** Select **Dual-homed** if you want to use this VPN gateway to connect to instances on an IP network. If you haven't set up IP networks or if you want to use this gateway to connect to instances on the shared network only, then select **Single-homed**.

If you select **Single-homed**, you must configure GRE tunnels between the Corente Services Gateway instance and each Compute Classic instance that you want to access using VPN. See *Configuring a GRE Tunnel on a Guest Instance in Oracle Cloud in Setting Up VPN From a Third-Party Gateway On-Premises to the Shared Network*.

If you select **Dual-homed**, all instances that are on the same IP network as the Corente Services Gateway instance can be accessed using VPN.

- **IP Network:** This field is displayed when you select the Dual-homed interface type. Select the IP network that you want to add the Corente Services Gateway instance to.
- **IP Network Address:** This field is displayed when you select the Dual-homed interface type. Select the IP address for your gateway instance. The IP address you specify must belong to the subnet of the specified IP network. An available IP address is allocated by default. You can specify a different LAN IP address, if required.
- **Subnets:** Enter a comma-separated list of subnets (in CIDR format) that should be reachable using this gateway. If you selected the Dual-homed interface type, you can enter the subnets of your IP networks. Ensure that all the IP networks you specify here belong to the same IP network exchange. The subnet of the IP network specified in the **IP Network** field is added by default. Don't modify or delete this subnet in this field.
- **Add reachable IP networks:** (Optional) This field is displayed when you select the Dual-homed interface type. You can select additional IP networks that should be reachable using this gateway. Ensure that the IP networks that you specify here, and the IP network that the Corente Services Gateway is added to, all belong to the same IP network exchange. See [Adding an IP Network to an IP Network Exchange](#).

You must also add a route on the gateway to the subnet of each additional IP network. You can't do this using the web console. Use App Net Manager to add this route. See *Adding IP Networks to an Existing VPN Connection in Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud*.

 **Note:**

You must also add the subnets that you specify here to the list of destination IP addresses that you specify in your third-party device.

## 6. Click **Create**.

A Corente Services Gateway instance is created. The required orchestrations are created and started automatically. For example, if you specified the name of the Corente Gateway instance as **CSG1**, then the following orchestrations are created:

- **vpn-CSG1-launchplan:** This orchestration creates the instance using the specified image, and associates the instance interfaces with the shared network and, for a dual-homed gateway, with the specified IP network.
- **vpn-CSG1-bootvol:** This orchestration creates the persistent bootable storage volume.
- **vpn-CSG1-secrules:** This orchestration creates the required security list, security applications, and security rules.
- **vpn-CSG1-master:** This orchestration specifies relationships between each of the nested orchestrations and starts each orchestration in the appropriate sequence.

While the Corente Services Gateway instance is being created, the instance status displayed in the **Instance** column on the VPN Gateways page is **Starting**. When the instance is created, its status changes to **Ready**.

To use this gateway in a VPN connection, add a third-party device and then create a connection. See [Registering a Third-Party VPN Device](#) and [Connecting the Cloud Gateway with the Third-Party Device](#).

You can also update the gateway instance to modify the reachable routes, or delete the gateway instance if you no longer require this gateway. See [Modifying the Reachable Subnets for a VPN Gateway](#) or [Deleting a VPN Gateway](#).



### Note:

You can list the gateway instance and view details on the Instances page, or view the corresponding orchestrations on the Orchestrations page. However, it is recommended that you always use the VPN Gateways page to manage your gateway instances.

## Registering a Third-Party VPN Device



This topic does not apply to Oracle Cloud at Customer.

To establish a VPN connection to your Compute Classic instances, after creating a Corente Services Gateway instance, register a VPN device to provide information about the third-party VPN gateway used in your data center.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Customer Devices**.
4. Click **Create VPN Device**.
5. Select or enter the required information:
  - **Name:** Enter a name for the third-party VPN device.
  - **Type:** Select a supported third-party VPN device from the list.
  - **Model:** Enter the model of your third-party VPN device.
  - **WAN IP Address:** Enter the IP address of the WAN interface of your third-party VPN device.
  - **Visible IP Address:** Enter the public IP address of your third-party VPN device that the Corente Services Gateway should connect to. If you use network address translation (NAT), then this IP address would be different from the WAN IP address. Otherwise, the visible IP address would be the same as the WAN IP Address.
  - **Subnets:** Enter (in CIDR format) a comma-separated list of subnets in your data center that should be reachable using this third-party device.
  - **PFS:** This option is selected by default. If your third-party device supports Perfect Forward Secrecy (PFS), retain this setting to require PFS.
  - **DPD:** This option is selected by default. If your third-party device supports Dead Peer Detection (DPD), retain this setting to require DPD.
6. Click **Create**.

A record of your third-party VPN device is created. Next, to use this VPN device to establish a VPN connection between your data center and your Compute Classic instances, create a VPN connection. See [Connecting the Cloud Gateway with the Third-Party Device](#).

## Connecting the Cloud Gateway with the Third-Party Device



This topic does not apply to Oracle Cloud at Customer.

After you've created a Corente Services Gateway instance and added a third-party device, to establish a VPN connection between your data center and your Compute Classic instances you must connect the cloud gateway with the third-party VPN device.

### Prerequisites

- You must have already created the cloud gateway that you want to use. See [Creating a Cloud Gateway](#).
- You must have already configured your third-party VPN device in your data center. See [Third-Party VPN Device Configuration](#).
- You must have already added the third-party VPN device that you want to connect to in your data center. See [Registering a Third-Party VPN Device](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to

ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Connections**.
4. Click **Create VPN Connection**.
5. Select or enter the required information:
  - **Gateway:** Select the Corente Services Gateway that you want to use. Each Corente Services Gateway can be used in multiple connections. However, each connection must reach distinct destination subnets.
  - **Device:** Select the third-party device that you want to use. Each device can be used in multiple connections. However, each connection must reach distinct destination subnets.
  - **IKE ID:** The Internet Key Exchange (IKE) ID. Only IKE v1 in Main Mode is supported. The IKE ID can be the name or IP address used to identify the Corente Services Gateway on the third-party device. Alternatively, you can specify a string that you want to use as the IKE ID.

Select one of the following:

#### Note:

The third-party device that you use might not support all of the following options for IKE ID. Select the appropriate option for your device.

- **Gateway Name:** The name of the Corente Services Gateway instance in the format `Corente_Domain_name.Corente_Services_Gateway_instance_name`. The name is auto-populated when you select this option.
- **Gateway IP Address:** The private IP address (on the shared network) of the instance hosting the Corente Services Gateway. The IP address is auto-populated when you select this option. Note, however, that this address will change each time the instance is re-created.
- **User-Defined IKE ID:** Enter text that you want to use as the IKE ID. You can specify either an alternative IP address, or any text string. If you specify a text string, you must prefix the string with @. For example, if you want to specify the text `IKEID-for-VPN1`, enter `@IKEID-for-VPN1`. If you specify an IP address, don't prefix it with @. The IKE ID is case sensitive and can contain a maximum of 255 ASCII alphanumeric characters including special characters, period (.), hyphen (-), and underscore (\_). The IKE ID can't contain embedded space characters.

 **Note:**

If you specify the IKE ID, ensure that you specify the Peer ID type as **Domain Name** on the third-party device in your data center. Other Peer ID types, such as email address, firewall identifier or key identifier, aren't supported.

- **Shared Secret:** The shared secret, also called the pre-shared key (PSK) on some devices, is used while setting up the VPN connection to establish the authenticity of the Corente Services Gateway that is requesting the VPN connection. You must enter the same shared secret here and on your third-party device. The shared secret must contain only alphanumeric characters.

The VPN connection is created.

If this connection uses a dual-homed VPN gateway, then an IP route is created automatically. The destination address of this route is the subnet address of the local side of the third-party device that will participate in the VPN connection. This route uses the vNIC of the Corente Services Gateway instance as the next hop vNICset, to route traffic from the IP network to the on-premises VPN device. This allows devices in the on-premises subnet to communicate with devices in the IP network over VPN.

An orchestration is created automatically to manage this vNICset and IP route and you can view this orchestration on the Orchestrations page of the web console. The name of the orchestration indicates the name of the Corente Services Gateway instance as well as the name of the third-party device used in the connection. For example, if you create a VPN connection between a Corente Services Gateway **CSG1** and a third-party device **TPD1**, the name of the route and the corresponding orchestration would be: **vpn-CSG1-to-TPD1**.

## Managing VPN



This topic does not apply to Oracle Cloud at Customer.

### Topics

- [Listing VPN Gateways](#)
- [Modifying the Reachable Subnets for a VPN Gateway](#)
- [Workflow for Adding IP Networks to an Existing VPN Connection](#)
- [Deleting a VPN Gateway](#)
- [Listing Third-Party VPN Devices](#)
- [Updating a Third-Party Device](#)
- [Deleting a Third-Party Device](#)
- [Listing VPN Connections](#)
- [Updating a VPN Connection](#)
- [Stopping, Restarting, and Deleting a VPN Connection](#)



 **Note:**

You must have the `Compute_Operations` role to access the pages under the **VPN** tab. If you don't have this role, you won't be able to view these pages.

## Listing VPN Gateways



This topic does not apply to Oracle Cloud at Customer.

After you've created one or more VPN gateways, you can see information about all your VPN gateways by using the web console.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **VPN Gateways**.

The VPN Gateways page displays a list of all your Corente Services Gateways, along with information about each gateway such as the interface type and status of the gateway.

 **Note:**

This page also displays Corente Services Gateways deployed on hosts outside of Compute Classic.

Each gateway can have any of the following statuses:

| Status   | Description                                                                                                                                                                                                                                              |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Active   | The Corente Services Gateway instance is running.                                                                                                                                                                                                        |
| Inactive | The Corente Services Gateway instance has been shut down or is being restarted.<br><b>Action:</b> If the instance is restarting, wait for it to return to the running state. If the instance has been shut down, start it to return to the Active state. |

| Status       | Description                                                                                                                                                                                                                                                                                                                                                                |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Download     | The configuration file for the Corente Services Gateway is available to download, but hasn't been downloaded to the gateway instance.<br><b>Action:</b> Check that the required security rules or ACLs are in place and enabled, to allow the gateway instance to download the configuration file.                                                                         |
| Downloaded   | The configuration file for the Corente Services Gateway has been downloaded but not activated. This status usually indicates that the Corente Services Gateway is not yet installed or started.<br><b>Action:</b> Check that the gateway instance is running or restart the instance if required. Check that the required security rules or ACLs are in place and enabled. |
| Upgrade      | A software upgrade is available for the Corente Services Gateway.<br><b>Action:</b> Schedule a maintenance time for the Corente Services Gateway in App Net Manager. The upgrade will occur automatically during the scheduled maintenance time. See the App Net Manager online help for more information.                                                                 |
| Disconnected | The Corente Services Gateway has lost connectivity, without being powered off safely.<br><b>Action:</b> Check your network configuration to see if outbound connectivity has been blocked by firewall rules.                                                                                                                                                               |
| Denied       | The Corente Services Gateway connection has been denied.<br><b>Action:</b> Contact Oracle Support.                                                                                                                                                                                                                                                                         |
| New          | A new Corente Services Gateway instance has been created using App Net Manager, but the configuration of this new gateway instance hasn't been completed.<br><b>Action:</b> Complete and save the configuration of the new gateway using App Net Manager. The new configuration will then be downloaded.                                                                   |
| Unknown      | The Corente Services Gateway is in an unknown state.<br><b>Action:</b> Check the status again after some time, or contact Oracle Support.                                                                                                                                                                                                                                  |

## Modifying the Reachable Subnets for a VPN Gateway




This topic does not apply to Oracle Cloud at Customer.

You must specify the list of reachable subnets while creating a VPN gateway. If required, you can modify this list of subnets at any time after creating a VPN gateway.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **VPN Gateways**.
4. Go to the VPN gateway for which you want to modify the set of subnets. From the  menu, select **Update**.
5. Modify the list of subnets as required, and then click **Update**. If you selected the dual-homed interface type, you can also modify the list of IP networks that should be reachable using this gateway.

 **Note:**

You can't modify or delete the subnet of the IP network to which your gateway belongs.

The list of subnets or IP networks reachable by the VPN gateway is updated. If you added IP networks, ensure that the IP networks that you specify here, and the IP network that the Corente Services Gateway is added to, all belong to the same IP network exchange. See [Adding an IP Network to an IP Network Exchange](#). You must also add a route on the gateway to the subnet of each additional IP network. You can't do this using the web console. Use App Net Manager to add this route. See *Adding IP Networks to an Existing VPN Connection in [Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud](#)*.

 **Note:**

You must also add the subnets that you specify here to the list of destination IP addresses that you specify in your third-party device.

## Workflow for Adding IP Networks to an Existing VPN Connection



This topic does not apply to Oracle Cloud at Customer.

When you set up a VPN connection using a dual-homed Corente Services Gateway, all instances that have an interface on the same IP network as the gateway instance are reachable over the VPN connection. You can expand the network of reachable instances by creating other IP networks and adding all the IP networks to an IP network exchange.

### Prerequisites

- You've configured a supported third-party device at your data center.
- You've created an IP network in your Compute Classic account.
- You've created a dual-homed Corente Services Gateway instance in Compute Classic.
- You've registered your third-party device.

- You've created a connection between the registered third-party VPN device in your data center and the dual-homed Corente Services Gateway in Compute Classic.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

Here's an overview of the process for adding IP networks to an existing VPN connection.

1. Create another IP network, if you haven't created it yet. See [Creating an IP Network](#).
2. Create an IP network exchange, if you haven't created it yet. See [Creating an IP Network Exchange](#).
3. Update both IP networks to add them to the IP network exchange. See [Updating an IP Network](#).
4. In App Net Manager, update user groups for your Corente Services Gateway to add the new IP network. See *Adding IP Networks to an Existing VPN Connection in Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud*.
5. In App Net Manager, add a route to the subnet of the new IP network. See *Adding IP Networks to an Existing VPN Connection in Setting Up VPN from a Third-Party Gateway to an IP Network in Oracle Cloud*.

#### Note:

You must also add the subnets that you specify here to the list of destination IP addresses that you specify in your third-party device.

## Deleting a VPN Gateway



This topic does not apply to Oracle Cloud at Customer.



If you no longer require a VPN connection, you can stop the connection and delete the VPN gateway instance. Each VPN gateway instance is managed by a master orchestration that can be used to start or stop several nested orchestrations. To delete a VPN gateway instance, go to the VPN Gateways page in the web console and stop the master orchestration.

### Prerequisites

- If you want to delete a dual-homed VPN gateway instance, the VPN gateway must not be connected to any device. If the gateway is used in a VPN connection, stop the connection first. See [Stopping, Restarting, and Deleting a VPN Connection](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to

ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.


### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **VPN Gateways**.
4. Go to the Corente Services Gateway instance that you want to delete.
  - If you want to delete only the gateway instance, from the  menu, select **Stop**. The orchestration that controls the gateway instance is stopped. This deletes the Corente Services Gateway instance.
  - If you want to delete the gateway instance as well as other associated resources, from the  menu, select **Stop All**. The master orchestration that controls the gateway instance and its associated resources is stopped. This deletes the gateway instance as well as resources created by the nested orchestrations, such as the bootable storage volume and networking objects.

#### Note:

Resources created outside the master orchestration, such as the public IP address reservation or IP networks, aren't deleted when you stop the master orchestration for the gateway instance. If you no longer need those resources, remember to delete them after you've stopped the master orchestration.

After you've deleted a gateway instance, it continues to be listed on the VPN Gateways page, with the status **Stopped**. At any time, you can restart the master orchestration to re-create the cloud gateway instance and its associated resources.

5. If you want to delete the orchestrations associated with your gateway instance, go to the gateway instance and from the  menu, select **Delete**.

The master orchestration and the associated orchestrations for the instance, storage volumes, and security rules are deleted. The VPN gateway is no longer listed on the VPN Gateways page.

## Listing Third-Party VPN Devices



This topic does not apply to Oracle Cloud at Customer.

After you've added third-party devices, you can see information about all your third-party devices by using the web console.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Customer Devices**.

The Customer Devices page displays a list of all the third-party devices that you've added, along with information about each device such as its model and type and its IP address.


## Updating a Third-Party Device



This topic does not apply to Oracle Cloud at Customer.

After you've added a third-party device, if required, you can modify the information associated with a third-party devices by using the web console.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Customer Devices**.
4. Go to the device that you want to update. From the  menu, select **Update**.
5. In the Update VPN Device dialog box, modify the information as required. Note that you can't change the device name or type. If you need to modify that information, add a new device. You can modify the following device information:
  - **Model:** The model of your third-party VPN device.
  - **WAN IP Address:** The IP address of the WAN interface of your third-party VPN device.
  - **Visible IP Address:** The public IP address of your third-party VPN device that the Corente Services Gateway should connect to. If you use network address translation (NAT), then this IP address would be different from the WAN IP address. Otherwise, the visible IP address would be the same as the WAN IP Address.

- **Subnets:** A list of IP addresses or subnets in your data center that should be reachable by this third-party device.
  - **PFS:** Perfect Forward Secrecy.
  - **DPD:** Dead Peer Detection.
6. Click **Update**. The device information is updated.

## Deleting a Third-Party Device




This topic does not apply to Oracle Cloud at Customer.

After you've added a third-party device, if you no longer want to use the device in a VPN connection, you can delete the device information by using the web console.

### Prerequisites

- The device that you want to delete must not be used in a connection with a dual-homed VPN gateway. If the device is used in a VPN connection with a dual-homed VPN gateway, stop the connection first. See [Stopping, Restarting, and Deleting a VPN Connection](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Customer Devices**.
4. Go to the device that you want to delete. From the  menu, select **Delete**.

The information about the selected device is deleted and the device is no longer displayed on the Customer Devices page.

## Listing VPN Connections



This topic does not apply to Oracle Cloud at Customer.

After you've created a connection between your VPN gateway and your third-party device, you can see a list of connections by using the web console.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in Managing and Monitoring Oracle Cloud](#).

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Connections**.

When a dual-homed gateway is used in a connection — that is, the gateway instance has one virtual network interface on an IP network and one interface on the shared network — then an IP route is created with the subnet of the third-party device as the destination. This IP route uses the vNIC of the cloud gateway as the next hop vNICset, to route traffic from the IP network to the third-party VPN device. An orchestration is created to manage the required vNICset and IP route and the **IP Route** column displays the status of the route. When a single-homed gateway is used, this column is blank.

The Connections page also shows the status of each of your VPN connections. If a VPN connection has any status other than **Up**, check the status again after some time. If the status doesn't change to **Up**, then contact Oracle Support.

## Updating a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

After you've created a connection between a VPN gateway and a third-party device, if required, you can modify the IKE ID or the shared secret by updating the VPN connection.

The IKE ID and shared secret that you enter here must match the corresponding entries on the third-party device used in this connection. If you make any changes to these fields, ensure that the corresponding changes are made on the connected third-party device.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Connections**.
4. Go to the connection that you want to modify. From the  menu, select **Update**.
5. Update the IKE ID or modify the shared secret as required, and then click **Update**.  
The IKE ID or shared secret is updated.



 **Note:**

The IKE ID and shared secret are used to identify and authenticate the Corente Services Gateway on the third-party device. If you modify these fields, ensure that the information you enter here matches the corresponding entries on the third-party device used in this connection.




## Stopping, Restarting, and Deleting a VPN Connection



This topic does not apply to Oracle Cloud at Customer.

After you've created a connection between a VPN gateway and a third-party device, if you no longer want to use this VPN connection, you can stop the connection. You can then restart the VPN connection later, or delete it.

To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Connections**.
4. If your VPN connection uses a dual-homed Corente Services Gateway, then you can stop and restart the connection by stopping and starting the orchestration that controls the vNICset and route.
  - To stop a connection that uses a dual-homed Corente Services Gateway instance, you can delete the route between the IP network and the destination subnet. This effectively prevents traffic from the IP network from accessing the VPN connection. To stop the route orchestration, go to the connection that you want to stop. From the  menu, select **Stop**. The route orchestration is stopped.
  - To restart a VPN connection that uses a dual-homed Corente Services Gateway instance, you can restart the route orchestration. Go to the connection that you want to restart. From the  menu, select **Start**. The route orchestration is started, and traffic from the IP network can once again access the VPN connection.
5. To delete a VPN connection, go to the connection that you want to delete. From the  menu, select **Delete**.

This ends the partnership between the specified VPN gateway and the third-party device and deletes the route orchestration. The VPN connection is no longer listed on the Connections page.

After stopping or deleting a VPN connection, you can also delete the gateway instance or delete the information about the third-party device used in this connection. See [Deleting a VPN Gateway](#) or [Deleting a Third-Party Device](#).

# 16

## Setting Up a VPN Connection Using VPNaaS



This topic does not apply to Oracle Cloud at Customer.

### Topics

- [Setting Up VPN Using VPNaaS](#)
- [Creating a VPN Connection Using VPNaaS](#)
- [Viewing the Event Log for a VPN Connection](#)
- [Listing VPNaaS Connections](#)
- [Updating a VPNaaS Connection](#)
- [Deleting a VPNaaS Connection](#)
- [VPNaaS Connection to other Environments](#)

## Setting Up VPN Using VPNaaS

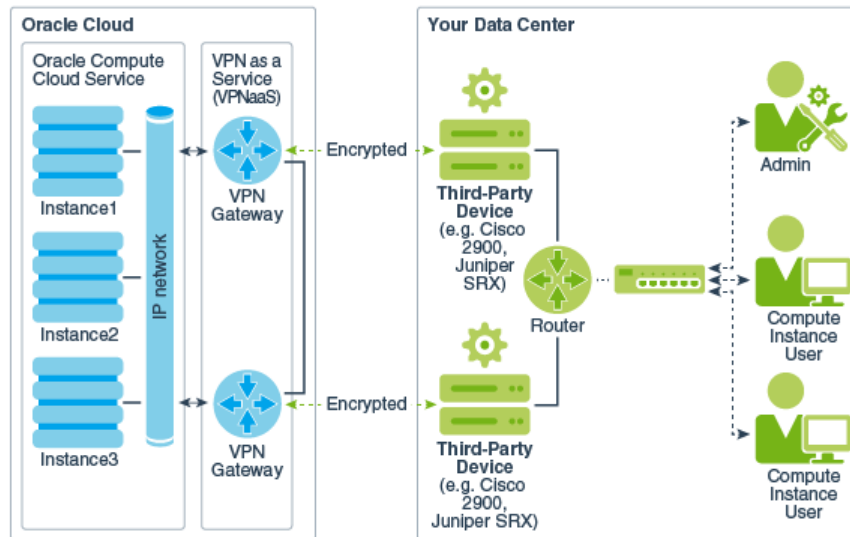


This topic does not apply to Oracle Cloud at Customer.

You can set up a VPN connection between your data center and IP networks in your Compute Classic site using VPN as a Service (VPNaaS). This provides a secure communication channel between your data center and instances that are added to your IP networks.

While you can continue to access your instances in Compute Classic over the public internet securely using SSH or RDP, a VPN connection provides enhanced security. IPsec-based tunnels carry encrypted traffic between your data center and your instances in Compute Classic. Your data can't be stolen or intercepted. By using a VPN connection, you effectively extend your data center network to include instances in Compute Classic.

The following figure shows two VPN connections between your data center and your Compute Classic site using VPN as a Service. In this scenario, both connections could be configured as failover partners to ensure active-active high availability.



You can configure any supported third-party device in your data center to participate in the VPN connection.

**Note:**

You can use VPNaaS to set up a tunnel to instances that are on IP networks. VPNaaS doesn't support VPN connections to instances that **don't** have an interface on IP networks.

**Topics**

- [Third-Party VPN Device Configurations](#)
- [Workflow for Setting Up a VPN Connection](#)

**Third-Party VPN Device Configurations**

The following table lists the certified third-party VPN device configurations.

| Certified Configurations                                                                                                                                                                                | Devices                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-256</li> <li>• DH phase 1 group 14</li> <li>• No Perfect Forward Secrecy (PFS); so no Diffie-Hellman (DH) phase 2 group</li> </ul> | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-256</li> <li>• DH phase 1 group 14; DH phase 2 group 14</li> </ul>                                                                 | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |

| Certified Configurations                                                                                                            | Devices                                                                             |
|-------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• Encryption AES128; Hash SHA-256</li> <li>• DH phase 1 group 14; no PFS</li> </ul>          | Cisco 2921<br>Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D |
| <ul style="list-style-type: none"> <li>• Encryption AES192; Hash SHA-1</li> <li>• DH phase 1 group 2, DH phase 2 group 2</li> </ul> | Cisco ASA5505                                                                       |
| <ul style="list-style-type: none"> <li>• Encryption AES256; Hash SHA-1</li> <li>• DH phase 1 group 5; no PFS</li> </ul>             | Cisco ISR 4331<br>Checkpoint 3200<br>Palo Alto 3020<br>FortiGate-200D               |

Other devices may work if they are configured with the certified configurations. Consider the following information while configuring your third-party device for a VPN connection.

- **Configuration Information**
  - The cloud gateway used by VPNaaS uses IPsec and is behind a NAT, so network address translation traversal (NAT-T) is required. Ensure that the third-party device in your data center supports NAT-T. NAT-T requires UDP port 4500 to be open.
  - Devices must support and be configured for policy-based VPN.
  - Ensure that the same subnets are defined on the third-party device and cloud gateway.
  - **IPsec configuration information**
    - \* IPsec protocol: ESP, tunnel-mode
    - \* Authentication: Pre-shared keys
    - \* Encryption: AES-128, AES-192, AES-256
    - \* Hash: MD5, SHA-1, SHA-2
    - \* Policy Group: Diffie-Hellman groups supported are 2, 5, 14, 22, 23, 24
    - \* Ensure the IPsec lifetime on the cloud gateway and the third-party device are the same.
    - \* Set life size as unlimited. Set reasonable traffic volume limits **only** if traffic limits are required by the third-party device.
    - \* Remove idle timeout.
  - ISAKMP: IKEv1 only. If IKEv2 is enabled by default, turn it off.
  - Exchange type: Main Mode (The cloud gateway uses **main mode** in phase one negotiations)
  - It is highly recommended that the third-party device be configured to be responder-only as the cloud gateway ensure that the VPN tunnel is up.
  - 
  - **Phase 1 IKE configuration information**

- \* Ensure that the IKE ID on the cloud gateway and the third-party device match.
- \* Ensure the IKE lifetime on the cloud gateway and the third-party device are the same.
- PFS: Enabled
- **HA Information**
  - When HA is configured, Dead Peer Detection (DPD) must be enabled to detect when a tunnel is down.
  - When HA is configured, asymmetric routing across the tunnels that make up the VPN connection will occur. Ensure that your firewall is configured to support this. If not, traffic will not be routed reliably.
  - Switching tunnels might take 30–40 seconds.

### Workflow for Setting Up a VPN Connection

Here's the workflow to set up your VPN connection using VPNaaS:

1. Create the IP network that you want to use for the VPN connection. See [Creating an IP Network](#). Make a note of the name of this IP network. You'll need to specify this IP network when you create the VPN connection.
2. (Optional) You can access multiple IP networks over a single VPN connection, as long as all the IP networks belong to the same IP network exchange. To do this, create an IP network exchange and add all the required IP networks to the IP network exchange.

#### Tip:

If you want access to a large number of instances, it is recommended that you avoid setting up numerous IP networks with a /32 subnet. Instead, use a smaller number of IP networks with larger subnets. If you create a very large number of IP networks, a large number of IPSec security associations are required, which could cause performance degradation on some third-party devices.

See [Creating an IP Network Exchange](#) and [Adding an IP Network to an IP Network Exchange](#).

3. Create a vNICset. When you create instances, specify this vNICset for each vNIC that is added to an IP network that will be reachable over the VPN connection. You'll use this vNICset later, when you create the VPN connection. See [Creating a vNICset](#).
4. Create the instances that you want to access using VPN. While creating instances, add them to the IP network that will be reachable over the VPN connection. If you've created multiple IP networks and added them to an IP network exchange, you can add an instance to any of those IP networks. For each vNIC that is added to an IP network to be accessed over the VPN connection, specify the vNICset that you created in the previous step. See [Creating Instances](#).

 **Note:**

You can add an instance to one or more IP networks only while creating the instance. If you've already created an instance that doesn't have an interface on any IP network, you won't be able to access it over a VPN connection created by using VPNaaS. To access these instances over VPN, set up a VPN connection to the shared network using the web console. See [Setting Up VPN](#).

 **Note:**

With VPNaaS, you access instances using their private IP addresses. So you don't have to associate a public IP address with the instance,

5. Configure a supported third-party VPN device in your data center and make a note of the public IP address and the domain name of this gateway. You'll need this information when you create the VPN connection.

Device configuration varies depending on the type and model of your device. For supported configurations, see [Third-Party VPN Device Configurations](#).

6. Ensure that you have the pre-shared key (PSK) that you want to use for this VPN connection. You'll need to specify the PSK when you create the VPN connection.
7. Create the VPN connection. See [Creating a VPN Connection Using VPNaaS](#).

The VPN connection is listed on the VPN Connections page.


8. You can monitor the provisioning status of your VPN connection by looking at the value of **Life Cycle Status** on the VPN Connections page. While your VPN connection is being configured, its **Life Cycle Status** is **Provisioning**.

When your VPN connection is fully provisioned and configured, its **Life Cycle Status** changes to **Ready**. When your cloud VPN gateway has been created, you will see the public IP address of the gateway. Make a note of this public IP address. See [Listing VPNaaS Connections](#).

 **Note:**

It can take some time for your VPN gateway to be created and for the public IP address of the gateway to become available.

If the VPN connection was not provisioned and the **Life Cycle Status** of the VPN connection remains **Provisioning**, you can retry to provision the VPN connection within an hour. You can also retry establishing a VPN connection that is in the **Error** state.

- Go to the VPN connection. From the  menu, select **Retry** to retry establishing the VPN connection.
9. Update the third-party VPN device in your data center with the public IP address of your cloud VPN gateway.

10. When the IPsec tunnel between the cloud gateway and the third-party device in your data center is established, the **Tunnel** status changes to **Up**.

 **WARNING:**

The security rules and access control list required to enable traffic to the cloud VPN gateway are created automatically. The routes required to enable traffic from your cloud gateway to your IP networks are also created automatically. Ensure that you don't accidentally modify or delete these objects, as it could break the VPN connection.

If the IPsec tunnel is not established, view the error logs and troubleshoot the issue:

On the VPN Connections page, from the  menu, select **View Event Log**. See [Viewing the Event Log for a VPN Connection](#).

11. After the first VPN connection is created and the status shows that the connection is up, you can set up a second VPN connection to provide high availability (HA). When you create a second VPN connection with exactly the same IP network and reachable routes as an existing VPN connection, the second VPN connection is automatically recognized as the failover partner for the first VPN connection. The paired connections are also automatically configured for load balancing.

 **Note:**

When you create a second VPN connection to be used as a failover partner, you don't have to specify the same gateway in your data center for this connection. Only the IP network used by this VPN connection and the reachable routes specified must be the same. You can use either the same gateway or a different gateway in your data center. If you specify the same gateway in your data center, then active-passive HA is implemented. To implement active-active HA, specify a different gateway for each of the VPN connections.

12. If you want to modify the list of subnets in your data center that you can access over this VPN connection, update the VPN connection.

If you want to modify the list of IP networks in your Compute Classic site that you can access over this VPN connection, add the IP networks to or remove the IP networks from the IP network exchange, and then update the VPN connection. See [Updating a VPNaaS Connection](#).

## Creating VPN Connections Using VPNaaS



This topic does not apply to Oracle Cloud at Customer.

You can set up a VPN connection between your data center and IP networks in your Compute Classic site to provide a secure communication channel between your data center and instances that are added to IP networks in your Compute Classic account.



- [Prerequisites](#)
- [Creating a VPN Connection Using VPNaaS](#)

## Prerequisites

Before you create a VPN connection, ensure that:

- You've created the IP network that you want to access using this VPN connection and you've added it to an IP network exchange, if required.
- You've created a vNICset for the vNICs that you want to access using this VPN connection, if required. Alternatively, you can use the default vNICset.
- You've set up a VPN gateway in your data center and you know the public IP address, pre-shared key, and IKE ID of your gateway, as well as the subnets in your data center that you want to reach.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

## Creating a VPN Connection Using VPNaaS

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Click **Create VPN Connection**.
5. Select or enter the required information:
  - **Name:** Enter a name for the VPN connection.
  - **IP Network:** Select the IP network that you want to access over this VPN connection.
  - **Connected IP Networks:** This field displays the IP networks that will be reachable over this VPN connection. The VPN connection allows you to access all IP networks that are added to the same IP network exchange as the specified IP networks.
  - **vNICsets:** Select the vNICsets that contain the vNICs that you want to access over this VPN connection. A vNIC must belong to one of the specified vNICsets and it must be part of one of the connected IP networks, to be reachable over this VPN connection.
  - **Customer Gateway:** Enter the public IP address of the VPN device in your data center that you want to connect to.
  - **Customer Reachable Routes:** Enter (in CIDR format) a comma-separated list of subnets in your data center that should be reachable using this VPN connection.
  - **Pre-shared Key:** The pre-shared key (PSK), is used while setting up the VPN connection to establish the authenticity of the gateway that is requesting the

connection. You must enter the same key here and on the gateway in your data center. The PSK must contain only alphanumeric characters.

- **IKE ID:** The Internet Key Exchange (IKE) ID is used to identify the cloud gateway on the gateway in your data center. Only IKE v1 in Main Mode is supported. The IKE ID can be the name or IP address of your cloud gateway. If you don't specify the IKE ID, then the IP address of your cloud gateway is used by default. Alternatively, you can specify a text string that you want to use as the IKE ID. The IKE ID is case sensitive and can contain a maximum of 255 ASCII alphanumeric characters including special characters, period (.), hyphen (-), and underscore (\_). The IKE ID can't contain embedded space characters.

 **Note:**

If you specify the IKE ID, ensure that you specify the Peer ID type as Domain Name on the gateway in your data center. Other Peer ID types, such as email address, firewall identifier or key identifier, aren't supported.

- **Specify Phase 1 IKE Proposal:** Select this option to specify Phase 1 IKE v1 options, if required. You can specify the following values:
  - **IKE Encryption:** Select the IKE encryption algorithm.
  - **IKE Hash:** Select the IKE hash algorithm.
  - **IKE DH group:** Select the Diffie Hellman (DH) group.
  - **IKE Lifetime:** Specify a value between 600 seconds to 9999999 seconds. The default value is 28800 seconds.

If no values are specified, all possible values are permitted.
- **Specify Phase 2 ESP Proposal:** Select this option to specify Phase 2 Encapsulating Security Payload (ESP) options, if required. You can specify the following values:
  - **ESP Encryption:** Select the ESP encryption algorithm.
  - **ESP Hash:** Select the ESP hash algorithm.
  - **IPSEC Lifetime:** Specify a value between 600 seconds to 9999999 seconds. The default value is 3600 seconds.

If no values are specified, all possible values are permitted.
- **Require Perfect Forward Secrecy:** This option is selected by default. If the gateway in your data center supports Perfect Forward Secrecy (PFS), retain this setting to require PFS.
- **Specify Outbound NAT:** Select this option to map or NAT a local subnet of IP addresses to another subnet. The local IP address of instance is not exposed to remote applications as traffic always appears to flow through the mapped IP address. This is a useful way to organize an entire application network. An administrator can map every local subnet to a distinct set of address ranges so that there are no address conflicts. The traffic from each site can then be identified by the range into which it was mapped. It is the responsibility of the system administrator in your data center to ensure that there are no conflicts between the addresses that are specified for each subnet. The maximum

number of /32 mapping entries is limited to 50 and other IP prefix mapping entries is limited 30.

- **From IP Prefix:** Enter the IP address prefix, in CIDR format, of the IP network that you want to add as a subnet of the VPN connection. You can't enter a wider range of the IP address prefix that's associated with an IP network, but you can enter a narrower or granular range of the IP address prefix. For example, if the IP address prefix of an IP network is 10.1.1.0/24, then you can enter 10.1.1.0/24 or a narrower range 10.1.1.0/28 but you can't enter a wider range such as 10.1.1.0/16.

If you enter a narrower range of the IP address prefix that's associated with an IP network, only the specified IP address prefix is added to the VPN subnet with NAT enabled and the corresponding IP network is not added as a subnet of the VPN.

- **To IP Address:** Enter an IP address to which you want to translate or map the specified IP address prefix. To remote applications traffic always appears to flow through the IP address you specify.

Although you don't enter the subnet mask for this IP address, it is assigned the same subnet mask that you have specified in the **From IP Prefix** text box.

To add a narrower range of an IP address prefix of an IP network as a VPN subnet without enabling NAT for this subnet, add the same values in the **From IP Prefix** text box and the **To IP Address** text box. For example, if you enter 10.1.2.0/24 in the **From IP Prefix** text box, then enter 10.1.2.0 in the **To IP Address** text box.

 **Note:**

If you've entered narrower range of IP address prefixes for the IP network that is directly attached to the VPN connection (that's the IP network you've selected in the **IP Network** drop-down list), check if the first IP address of the directly attached IP network is included in one of the narrower range prefixes. If the first IP address of the IP network is not included, you must create a separate entry under **Specify Outbound NAT** to NAT this IP address. For example, let's consider that the IP address prefix of an IP network is 10.1.1.0/24. To NAT the first IP address of the IP network, enter 10.1.1.1/32 for **From IP Prefix** and 10.1.1.1 for **To IP Address**. This setting manages traffic between cloud gateway and cloud hypervisor.

You must also add the first IP address, for example 10.1.1.1/32, to the remote encryption subnet list of the third-party device in your data center to establish the VPN tunnel.


- **Description:** Enter a description.
- **Tags:** Specify one or more tags to help you identify and categorize the VPN connection.

6. Click **Create**.

The VPN connection is listed on the VPN Connections page. You can monitor the provisioning status of your VPN connection by looking at the value of **Life Cycle**

**Status** on the VPN Connections page. While your VPN connection is being configured, its **Life Cycle Status** is **Provisioning**.

When your VPN connection is fully provisioned and configured, its **Life Cycle Status** changes to **Ready**. When your cloud VPN gateway has been created, you will see the public IP address of the gateway.

If the VPN connection was not provisioned and the VPN connection remains in the **Provisioning** state, you can retry to provision the VPN connection within an hour. On the VPN Connections page, go to the VPN connection that is in the **Provisioning** state. From the  menu, select **Retry**. You can also retry establishing a VPN connection that is in the **Error** state.

## Other Ways of Creating a VPN Connection Using VPNaaS

To create a VPN connection using the CLI, use the `opc compute vpn-endpoint-v2 add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a VPN connection using the API, use the `POST /vpnendpoint/v2/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Viewing the Event Log for a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

If the VPN connection between the cloud gateway and the third-party device in your data center is not established or if the IPsec tunnel goes down, you can view the event log to identify and troubleshoot the issues. This log provides information about the tunnel events.

### Prerequisites

- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Go to the VPN connection that you want to update. From the  menu, select **View Event Log**.

The event log for the last 72 hours is displayed, with the most recent events listed first.

5. Click **Download** to download the event log to identify and troubleshoot the issues.

## Listing VPNaaS Connections



This topic does not apply to Oracle Cloud at Customer.

After you've created a VPN connection using VPNaaS, you can view your existing connections and see information about the status of the connection and connected IP networks, as well as the gateway in your data center and the on-premises subnets that you are connecting to.

### Prerequisites

- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.

The VPN Connections page shows a list of VPN connections, along with information about each connection and its current status.

To list VPN connections using the CLI, use the `opc compute vpn-endpoint-v2 list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list VPN connections using the API, use the `GET /vpnendpoint/v2/container` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating a VPNaaS Connection



This topic does not apply to Oracle Cloud at Customer.


After creating a VPN connection using VPNaaS, you can update the subnets in your data center that you want to access using this VPN connection. You can also modify the public IP address of your network gateway, pre-shared key, description, and tags. If you want to modify the list of IP networks in your Compute Classic site that you can

access over this VPN connection, first add or remove the IP networks from the IP network exchange, and then update the VPN connection.

### Prerequisites

- If you want to add IP networks that you can access using an existing VPN connection, you must have created the new IP networks and added them to the same IP network exchange. See [Creating an IP Network Exchange](#) and [Creating an IP Network](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Go to the VPN connection that you want to update. From the  menu, select **Update**.
5. Update the information as required:
  - **Customer Reachable Routes:** Update the list of subnets in your data center that should be reachable using this VPN connection.
  - **Pre-shared Key:** Update the pre-shared key (PSK), used while setting up the VPN connection. You must enter the same key here and on the gateway in your data center. The PSK must contain only alphanumeric characters.
  - **IKE ID:** Update the Internet Key Exchange (IKE) ID, used to identify the cloud gateway on the gateway in your data center. Only IKE v1 in Main Mode is supported. The IKE ID can be the name or IP address of your cloud gateway. If you don't specify the IKE ID, then the IP address of your cloud gateway is used by default. Alternatively, you can specify a text string that you want to use as the IKE ID. The IKE ID is case sensitive and can contain a maximum of 255 ASCII alphanumeric characters including special characters, period (.), hyphen (-), and underscore (\_). The IKE ID can't contain embedded space characters.

 **Note:**

If you specify the IKE ID, ensure that you specify the Peer ID type as Domain Name on the gateway in your data center. Other Peer ID types, such as email address, firewall identifier or key identifier, aren't supported.

- **Specify Phase 1 IKE Proposal:** Update this option to specify Phase 1 IKE v1 options, if required. You can specify the following values:

- **IKE Encryption:** The IKE encryption algorithm.
- **IKE Hash:** The IKE hash algorithm.
- **IKE DH group:** The Diffie Hellman (DH) group.
- **IKE Lifetime:** Update to a value between 600 seconds to 9999999 seconds. The default value is 28800 seconds.

If no values are specified, all possible values are permitted.

- **Specify Phase 2 ESP Proposal:** Update this option to specify Phase 2 Encapsulating Security Payload (ESP) options, if required. You can specify the following values:
  - **ESP Encryption:** The ESP encryption algorithm.
  - **ESP Hash:** The ESP hash algorithm.
  - **IPSEC Lifetime:** Update to a value between 600 seconds to 9999999 seconds. The default value is 3600 seconds.

If no values are specified, all possible values are permitted.

- **Require Perfect Forward Secrecy:** This option is selected by default. If the gateway in your data center supports Perfect Forward Secrecy (PFS), retain this setting to require PFS.
- **Specify Outbound NAT:** Select this option to map or NAT a local subnet of IP addresses to another subnet. The local IP address of instance is not exposed to remote applications as traffic always appears to flow through the mapped IP address. This is a useful way to organize an entire application network. An administrator can map every local subnet to a distinct set of address ranges so that there are no address conflicts. The traffic from each site can then be identified by the range into which it was mapped. It is the responsibility of the system administrator in your data center to ensure that there are no conflicts between the addresses that are specified for each subnet. The maximum number of /32 mapping entries is limited to 50 and other IP prefix mapping entries is limited 30.
  - **From IP Prefix:** Enter the IP address prefix, in CIDR format, of the IP network that you want to add as a subnet of the VPN connection. You can't enter a wider range of the IP address prefix that's associated with an IP network, but you can enter a narrower or granular range of the IP address prefix. For example, if the IP address prefix of an IP network is 10.1.1.0/24, then you can enter 10.1.1.0/24 or a narrower range 10.1.1.0/28 but you can't enter a wider range such as 10.1.1.0/16.

If you enter a narrower range of the IP address prefix that's associated with an IP network, only the specified IP address prefix is added to the VPN subnet with NAT enabled and the corresponding IP network is not added as a subnet of the VPN.

- **To IP Address:** Enter an IP address to which you want to translate or map the specified IP address prefix. To remote applications traffic always appears to flow through the IP address you specify.

Although you don't enter the subnet mask for this IP address, it is assigned the same subnet mask that you have specified in the **From IP Prefix** text box.

To add a narrower range of an IP address prefix of an IP network as a VPN subnet without enabling NAT for this subnet, add the same values in the **From**

**IP Prefix** text box and the **To IP Address** text box. For example, if you enter 10.1.2.0/24 in the **From IP Prefix** text box, then enter 10.1.2.0 in the **To IP Address** text box.

 **Note:**

If you've entered narrower range of IP address prefixes for the IP network that is directly attached to the VPN connection or the IP network you've selected in the **IP Network** drop-down box, check if the first IP address of the directly attached IP network is included in one of the narrower range prefixes. If the first IP address of the IP network is not included, you must create a separate entry under **Specify Outbound NAT** to NAT this IP address. For example, let's consider that the IP address prefix of an IP network is 10.1.1.0/24. To NAT the first IP address of the IP network, enter 10.1.1.1/32 for **From IP Prefix** and 10.1.1.1 for **To IP Address**. This setting manages traffic between cloud gateway and cloud hypervisor.

You must also add the first IP address, for example 10.1.1.1/32, to the remote encryption subnet list of the third-party device in your data center to establish the VPN tunnel.

- **Description:** Enter a description.
- **Tags:** Specify one or more tags to help you identify and categorize the VPN connection.

 **Note:**

If you've added the specified IP network to an IP network exchange, or if you've added IP networks to the IP network exchange or removed IP networks from the IP network exchange, the Update dialog box displays these changes automatically.

6. Click **Update**.

When an update operation performed by VPNaaS is in progress, the lifecycle status changes to **Updating**. After the update is complete, the status transitions to **Ready** or **Error**.

To update a VPN connection using the CLI, use the `opc compute vpn-endpoint-v2 update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a VPN connection using the API, use the `PUT /vpnendpoint/v2/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a VPNaaS Connection



This topic does not apply to Oracle Cloud at Customer.




If you no longer need a VPN connection, you can delete it. Deleting a VPN connection is useful if, for example, you want to modify the list of vNICsets that should be reachable over the VPN connection.

### Prerequisites

- If you have created two VPN connections to provide failover, the second connection must be deleted before deleting the first connection.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Go to the VPN connection that you want to delete. From the  menu, select **Delete**.

To delete a VPN connection using the CLI, use the `opc compute vpn-endpoint-v2 delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a VPN connection using the API, use the `DELETE /vpnendpoint/v2/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

When a delete operation performed by VPNaaS is in progress, the lifecycle status changes to **Deleting**. After the delete operation is complete, the status transitions to **Ready** or **Error**.

 **Note:**

It might take about 15 minutes for the delete operation to complete.

## VPNaaS Connection to other Environments



This topic does not apply to Oracle Cloud at Customer.

Use VPNaaS to connect to your Compute Classic instances from another Oracle Cloud account or to connect an Oracle cloud environment to other cloud services.

### Procedure

1. Create a VPN connection (VPN-1) and wait till the status changes to **Ready**. For instructions to create a VPN connection, see [Creating VPN Connections Using VPNaaS](#).

 **Note:**

Enter a temporary IP in the **Customer Gateway** field.

2. Note down the Public IP address (Public-IP-1) of VPN-1.
3. Create the remote end VPN connection (VPN-2) and wait till the status changes to **Ready**. For instructions to create a VPN connection, see [Creating VPN Connections Using VPNaaS](#).

 **Note:**

In the **Customer Gateway** field, enter the Public IP Address (Public-IP-1) of VPN-1, which you noted down in [step 2](#).

4. Note down the Public IP address (Public-IP-2) of VPN-2.
5. Go to VPN-1 connection that you created in [step 1](#) and update the **Customer Gateway** field with the Public IP Address (Public-IP-2), which you noted down in [step 4](#).

# 17

## Set Up VPN Connection to Oracle Cloud Infrastructure

Use IPsec VPN to set up a connection between the Compute Classic environment and the Oracle Cloud Infrastructure.

### Topics

- [Set Up VPNaaS Connection between an IP Network and Oracle Cloud Infrastructure](#)
- [Set Up VPN Connection between Shared Network and Oracle Cloud Infrastructure](#)

## Set Up VPNaaS Connection between an IP Network and Oracle Cloud Infrastructure



This topic does not apply to Oracle Cloud at Customer.

Use VPN as a Service (VPNaaS) to set up a secure, private connection between an IP network in Compute Classic and a subnet in the virtual cloud network (VCN) in Oracle Cloud Infrastructure.

## About Setting Up VPN Connection between Compute Classic and Oracle Cloud Infrastructure

Use VPN as a Service (VPNaaS) to set up a VPN connection between an IP network in your Compute Classic site and a single subnet in the VCN in your Oracle Cloud Infrastructure site. This provides a secure communication channel between your Compute Classic site and your Oracle Cloud Infrastructure site.

### Workflow for Setting Up a VPNaaS Connection to Oracle Cloud Infrastructure

1. Create an IP network in Compute Classic site or use an existing IP network. See [Creating an IP Network](#). Note down the name of the IP network as you'll have to provide this information while creating the VPNaaS connection.
2. Create a vNICset. When you create instances, specify this vNICset for each vNIC that is added to an IP network that will be reachable over the VPN connection. See [Creating a vNICset](#). Note down the name of the vNICset as you'll have to provide this information while creating the VPNaaS connection.
3. Create a VPN connection using VPNaaS in the Compute Classic site. See [Create a VPN Connection in Compute Classic](#).
4. Create the required networking components in Oracle Cloud Infrastructure to set up IPsec VPN. See [Setting Up an IPsec VPN in Oracle Cloud Infrastructure documentation](#).

5. Update the VPN connection that you have created in the Compute Classic site with the pre-shared key and IP address of the IPsec VPN tunnel that you have created in Oracle Cloud Infrastructure. See [Update the VPNaaS Connection in Compute Classic](#).
6. Validate connectivity between your hosts in the Compute Classic site and Oracle Cloud Infrastructure.

Test the connection before you start using it. Depending on how you've set up your IP network's security rules and security lists in VCN, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

This sets up a VPN connection between a **single** IP network in your Compute Classic site and a **single** subnet in the VCN in your Oracle Cloud Infrastructure site. If you want to establish a VPN connection between another IP network and another subnet in Oracle Cloud Infrastructure VCN, you'll have to create another VPNaaS connection.

## Create a VPN Connection in Compute Classic

Create a VPN connection using VPNaaS in the Compute Classic site.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Click **Create VPN Connection**.
5. Select or enter the required information:
  - **Name:** Enter a name for the VPN connection.
  - **IP Network:** Select the IP network that you want to access over this VPN connection. This is the IP network that you can access from Oracle Cloud Infrastructure when the VPN connection is provisioned.
  - **Connected IP Networks:** The information displayed in this field does not apply to this procedure. You can only access a single IP network in the Compute Classic site from Oracle Cloud Infrastructure over the VPN connection.
  - **vNICsets:** Select the vNICsets that contain the vNICs that you want to access over this VPN connection. A vNIC must belong to one of the specified vNICsets and it must be part of the connected IP network, to be reachable over this VPN connection.
  - **Customer Gateway:** Enter a temporary IP address in this field. You'll replace this value later with the IP address of the IPsec VPN tunnel that you create in Oracle Cloud Infrastructure.
  - **Customer Reachable Routes:** Enter (in CIDR format) a single Oracle Cloud Infrastructure VCN subnet that should be reachable using this VPN connection. You can retrieve this information from the Oracle Cloud Infrastructure VCN details page. If you have not yet created a VCN subnet,

enter a temporary value and then replace this value after creating a subnet in Oracle Cloud Infrastructure VCN.

- **Pre-shared Key:** Enter a temporary PSK. You'll replace this value later with the PSK of the IPsec VPN tunnel that you create in Oracle Cloud Infrastructure.
- **Specify Phase 1 IKE Proposal:** Select this option to specify Phase 1 IKE v1 options, if required. You can specify the following values:
  - **IKE Encryption:** Select **AES256**.
  - **IKE Hash:** Select **SHA2 256**.
  - **IKE DH group:** Select **5**.
  - **IKE Lifetime:** Specify **28800**.
- **Specify Phase 2 ESP Proposal:** Select this option to specify Phase 2 Encapsulating Security Payload (ESP) options, if required. You can specify the following values:
  - **ESP Encryption:** Select **AES256** as the ESP encryption algorithm.
  - **ESP Hash:** Select **SHA1** as the ESP hash algorithm.
  - **IPSEC Lifetime:** Specify **1800**.
- **Require Perfect Forward Secrecy:** This option is selected by default. Retain this setting to require PFS.

6. Click **Create**.

The VPN connection is listed on the VPN Connections page. You can monitor the provisioning status of your VPN connection by looking at the value of **Life Cycle Status** on the VPN Connections page. While your VPN connection is being configured, its **Life Cycle Status** is **Provisioning**. When your VPN connection is fully provisioned and configured, its **Life Cycle Status** changes to **Ready**. When your cloud VPN gateway has been created, you will see the public IP address of the gateway.

7. Note down the public IP address of the gateway as you'll have to provide this later.

After noting down the public IP address of the VPN gateway, create the required networking components in Oracle Cloud Infrastructure. See [Setting Up an IPsec VPN](#) in *Oracle Cloud Infrastructure documentation*.


Keep the following points in mind while creating the required networking components in Oracle Cloud Infrastructure:

- After creating a dynamic routing gateway (DRG) and attaching the DRG to VCN, create a route table and route rule for the DRG. The routes should include a route to your IP network in the Compute Classic site. This is the IP network in the Compute Classic site that points to the DRG.
- While creating the Customer-Premises Equipment (CPE) object, in the **IP Address** field specify the public IP address of the VPN gateway that you have created in the Compute Classic site.
- While creating the IPsec connection from the DRG to the CPE object, in the **Static Route CIDR** field specify the CIDR block of the IP network in the Compute Classic site. You can specify the CIDR block of only one IP network.

## Update the VPNaaS Connection in Compute Classic

After setting up the networking components in Oracle Cloud Infrastructure, note down the public IP address and the pre-shared key of the IPsec VPN tunnel. Update the VPNaaS connection in your Compute Classic site to provide the correct IP address and pre-shared key that you have retrieved from the Oracle Cloud Infrastructure environment.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **VPNaaS**, and then click **VPN Connections**.
4. Go to the VPN connection that you want to update. From the  menu, select **Update**.
5. Update the information as required:
  - **Customer Gateway:** Enter the public IP address of the IPsec VPN tunnel in the Oracle Cloud Infrastructure site that you want to connect to.
  - **Customer Reachable Routes:** Enter (in CIDR format) a single VCN subnet in Oracle Cloud Infrastructure that should be reachable using this VPN connection. You can retrieve this information from the Oracle Cloud Infrastructure VCN details page.
  - **Pre-shared Key:** Enter the pre-shared key (PSK) that which was used while setting up the IPsec VPN tunnel in the Oracle Cloud Infrastructure site. The pre-shared key (PSK), is used while setting up the VPN connection to establish the authenticity of the gateway that is requesting the connection. The PSK must contain only alphanumeric characters.
6. Click **Update**.

When the update operation is in progress, the **Life Cycle Status** of the VPN connection changes to **Updating**. After the update is complete and when the VPN connection is fully provisioned and configured, the **Life Cycle Status** changes to **Ready**.

When the VPN connection in the Compute Classic site is updated and provisioned, the IPsec VPN tunnel becomes available on Oracle Cloud Infrastructure. This might take a few minutes.

Validate connectivity between your hosts in the Compute Classic site and Oracle Cloud Infrastructure. Depending on how you've set up your IP network's security rules and VCN security lists, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

# Set Up VPN Connection between Shared Network and Oracle Cloud Infrastructure



This topic does not apply to Oracle Cloud at Customer.

Using an IPsec VPN, you can set up a secure, private connection between the shared network in Compute Classic and the virtual cloud network (VCN) in Oracle Cloud Infrastructure.

## Workflow for Setting Up VPN Connection between the Shared Network and the Oracle Cloud Infrastructure

1. Complete the prerequisites. See [Before You Begin](#).
2. Create a Corente Services Gateway instance in Compute Classic. See [Create a Cloud Gateway](#).
3. Add information about your VPN device in Oracle Cloud Infrastructure. See [Register the Third-Party VPN Device](#).
4. Create a connection between your Corente Services Gateway and the Oracle Cloud Infrastructure DRG. See [Connect the Cloud Gateway with the Oracle Cloud Infrastructure VPN](#).
5. On each instance that you want to access, configure a GRE tunnel to the gateway. See [Configuring a GRE Tunnel on a Guest Instance in Oracle Cloud in Setting Up VPN From a Third-Party Gateway On-Premises to the Shared Network](#).
6. Update the timeout for the VPN connection. See [Update the Timeout](#).
7. Test the connection after the status of the VPN connection changes to **Up**. Depending on how you've set up your IP network's security rules and VCN security lists, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

## Before You Begin

Before you begin creating an IPsec VPN connection to Oracle Cloud Infrastructure, complete the following tasks.

- Create an IP reservation in the shared network. While reserving the IP address, ensure that you don't attach this IP address to any instance. See [Reserving a Public IP Address](#).

Note down the value of the public IP address that you have reserved as you will have to provide this information while creating the VPN gateway.

- Create networking components in Oracle Cloud Infrastructure. See [Setting Up an IPsec VPN in Oracle Cloud Infrastructure documentation](#).

Keep the following points in mind while creating the required networking components in Oracle Cloud Infrastructure:

- After creating a dynamic routing gateway (DRG) and attaching the DRG to VCN, create a route table and route rule for the DRG. The routes should include a route to your shared network in the Compute Classic site.
  - While creating the Customer-Premises Equipment (CPE) object, in the **IP Address** field specify the public IP address of the VPN gateway that you have created in the Compute Classic site. While creating the cloud gateway in Compute Classic site, you specify an IP reservation to assign a public IP address to the VPN gateway. Specify this IP address.
  - While creating the IPSec connection from the DRG to the CPE object, in the **Static Route CIDR** field enter `172.16.1.0/24`. This is the subnet that contains the local address of the GRE tunnel to the Corente Services Gateway instance in the Compute Classic environment.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

## Create a Cloud Gateway

If you want to establish a VPN connection to your Compute Classic instances, start by creating a Corente Services Gateway instance.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **VPN Gateways**.
4. Click **Create VPN Gateway**.
5. Select or enter the required information:
  - **Name:** Enter a name for the Corente Services Gateway instance.
  - **IP Reservation:** Select the IP reservation that you want to use with this instance. This is the public IP address of your VPN gateway.
  - **Image:** Select the machine image that you want to use to create the instance. You must select the most recent Corente Gateway image, such as `corente_gateway_images-9.4.1062`.
  - **Interface Type:** Select **Single-homed**.
  - **Subnets:** Enter `172.16.1.0/24`. This is the subnet that contains the local address of the GRE tunnel to Corente Services Gateway instance on the Cloud.
6. Click **Create**.

A Corente Services Gateway instance is created. The required orchestrations are created and started automatically. For example, if you specified the name of the Corente Gateway instance as **CSG1**, then the following orchestrations are created:

- **vpn-CSG1-launchplan:** This orchestration creates the instance using the specified image, and associates the instance with the shared network.



- **vpn-CSG1-bootvol:** This orchestration creates the persistent bootable storage volume.
- **vpn-CSG1-secrules:** This orchestration creates the required security list, security applications, and security rules.
- **vpn-CSG1-master:** This orchestration specifies relationships between each of the nested orchestrations and starts each orchestration in the appropriate sequence.

While the Corente Services Gateway instance is being created, the instance status displayed in the **Instance** column on the VPN Gateways page is **Starting**. When the instance is created, its status changes to **Ready**.

 **Note:**

You can list the gateway instance and view details on the Instances page, or view the corresponding orchestrations on the Orchestrations page. However, it is recommended that you always use the VPN Gateways page to manage your gateway instances.

## Register the Third-Party VPN Device

To establish a VPN connection to your Compute Classic instances, after creating a Corente Services Gateway instance, register a VPN device to provide information about the Dynamic Routing Gateway (DRG) in Oracle Cloud Infrastructure.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Customer Devices**.
4. Click **Add VPN Device**.
5. Select or enter the required information:
  - **Name:** Enter a name for the VPN device used in Oracle Cloud Infrastructure.
  - **Type:** Select **other**.
  - **WAN IP Address:** Enter the IP address of the WAN interface of your Oracle Cloud Infrastructure DRG.
  - **Visible IP Address:** Enter the IP address of the WAN interface of your Oracle Cloud Infrastructure DRG.
  - **Subnets:** Enter (in CIDR format) the Oracle Cloud Infrastructure VCN that should be reachable using this VPN connection. You can retrieve this information from the Oracle Cloud Infrastructure VCN details page. If you specify the VCN CIDR, then all the subnets in VCN can communicate with the shared network.
  - **PFS:** This option is selected by default. Retain this setting to require PFS.

- **DPD:** This option is selected by default. If your third-party device supports Dead Peer Detection (DPD), retain this setting to require DPD.
6. Click **Create**.

A record of your DRG in Oracle Cloud Infrastructure is created.

Next, to use this VPN device to establish a VPN connection between Oracle Cloud Infrastructure and your Compute Classic instances, create a VPN connection.

## Connect the Cloud Gateway with the Oracle Cloud Infrastructure VPN

After you've created a Corente Services Gateway instance and added a third-party device, to establish a VPN connection between your data center and your Compute Classic instances you must connect the cloud gateway with the Oracle Cloud Infrastructure VPN.

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **VPN**, expand **Corente**, and then click **Connections**.
4. Click **Create VPN Connection**.
5. Select or enter the required information:
  - **Gateway:** Select the Corente Services Gateway that you want to use.
  - **Device:** Select the device in Oracle Cloud Infrastructure that you want to use.
  - **IKE ID:** The Internet Key Exchange (IKE) ID is used in Oracle Cloud Infrastructure to identify the Corente Services Gateway. Specify the public IP address of the Corente Services Gateway that you have created in Compute Classic.
  - **Shared Secret:** The shared secret, also called the pre-shared key (PSK) on some devices, is used while setting up the VPN connection to establish the authenticity of the Corente Services Gateway that is requesting the VPN connection. You must enter the same shared secret here and in Oracle Cloud Infrastructure. The shared secret must contain only alphanumeric characters.

The VPN connection is created.

To complete the VPN setup, you must configure GRE tunnels between the Corente Services Gateway instance and each Compute Classic instance that you want to access using VPN. See *Configuring a GRE Tunnel on a Guest Instance in Oracle Cloud in Setting Up VPN From a Third-Party Gateway On-Premises to the Shared Network*.

## Update the Timeout

App Net Manager is a secure web portal that you use to modify and monitor the components of your IPsec VPN network in Compute Classic.

To update the timeout for the VPN connection:

1. Download App Net Manager from <https://www.oracle.com/technetwork/server-storage/corente/downloads/index.html>.

2. Log in to App Net Manager using the Corente credentials that you received in an email when you subscribed to Compute Classic.
3. In App Net Manager, in the **Domains** pane, click **Locations** to expand and show all of your gateways.
4. Right-click your Oracle Cloud Infrastructure Classic gateway instance, and then select **Edit**.
5. In the Edit dialog box, select the **Partners** tab, and click the **Add** button.
6. Select 3rd-Party Device and then select the Oracle Cloud Infrastructure VPN device name that you had configured in the earlier task.
7. Under **Timeouts**, enter 28800 seconds as the **IKE Lifetime**.
8. Under **Timeouts**, enter 1800 seconds as the **IPSEC Lifetime**.
9. Click **OK** to close the dialog box.
10. Click **Save** at the top of the App Net Manager screen.

After the status of the VPN Connection in Compute Classic is **UP** or when the IPsec tunnel state changes to **Available** in Oracle Cloud Infrastructure, test the connection. Depending on how you've set up your IP network's security rules and VCN security lists, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

# 18

## Connecting to Instances Using Oracle Cloud Infrastructure FastConnect Classic

### Topics

- [About FastConnect Classic](#)
- [Managing Cross Connects](#)
- [Managing Virtual Circuits](#)
- [Managing Private Gateways](#)

### About FastConnect Classic



This topic does not apply to Oracle Cloud at Customer.

Oracle Cloud Infrastructure FastConnect Classic allows you to access your Oracle Cloud services using a direct connection from your premises or colocation facilities. With FastConnect Classic, your network traffic is routed over a direct and deterministic path from your on-premises network to instances in your Compute Classic account. This ensures consistent performance with dedicated bandwidth and controlled or reduced latency in your network traffic.

Use FastConnect Classic to access your Oracle Cloud Infrastructure - Classic services. To access your Oracle Cloud Infrastructure services use Oracle Cloud Infrastructure FastConnect. See the [Oracle Cloud Infrastructure FastConnect guide](#).

Using FastConnect Classic to establish a connection from your premises to Oracle Cloud allows you to:

- Access your instances through an overlay-based direct connection. Transferring data over a direct connection provides better privacy.
- Transfer large volumes of data using high WAN bandwidth. Migrating your Internet traffic to a dedicated path improves the overall performance of your network and significantly reduces the time taken to complete data transfers.

### Features

- **Standard routing:** Leverages the Border Gateway Protocol (BGP) to manage the exchange of routes between Oracle Cloud and your DMZ or public-facing network.
- **Public and private peering:** Advertise public and private IP prefixes over the peering established with FastConnect Classic edge routers. You can establish both private and public peering sessions over a single FastConnect Classic connection by adding virtual circuits.
- **Dedicated bandwidth:** Access your Oracle Cloud services, as well as transfer large volumes of data between your private clouds and the Oracle Cloud, over a dedicated data connection. The entire assigned bandwidth is available exclusively for your use.

Use FastConnect Classic to establish a dedicated, high-speed connection between your premises and your Compute Classic instances. If you want to establish a secure connection using IPsec tunneling, you can set up a VPN connection instead. If you need a high speed connection along with the security provided by IPsec tunneling, you can set up a VPN tunnel over your FastConnect Classic connection. For information about VPN, see [Connecting to Instances in a Multitenant Site Using VPN](#) or [Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN](#).

### Connecting to Instances on the Shared Network

You can use either of the following models to connect to instances on the shared network:

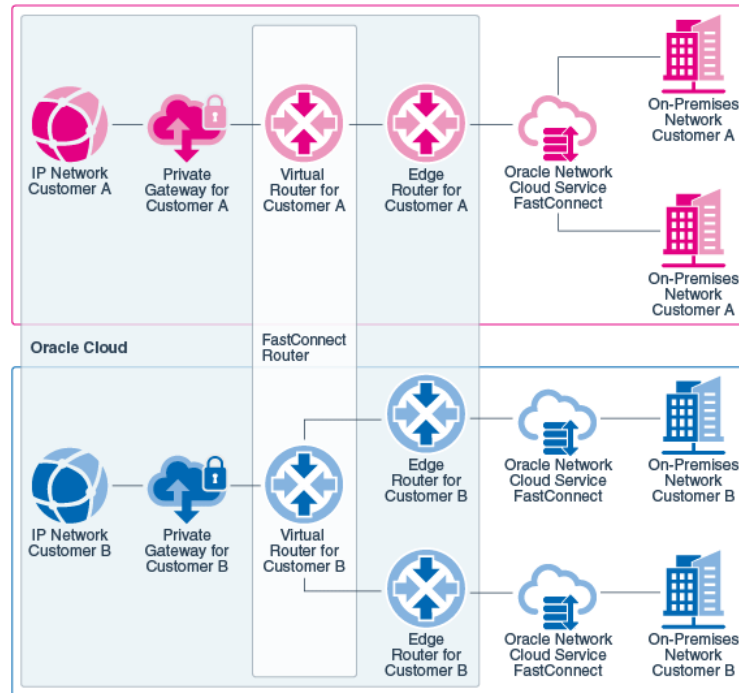
- By using Direct Cross Connects. Previously called Standard Edition service model, you can provision cross connects to the Oracle edge routers if your data center is colocated with the Oracle data center at a FastConnect location or if you can connect to the Oracle data center using a private link.
- By connecting through an Oracle FastConnect Partner. Previously called Partner Edition service model, you can connect through an Oracle FastConnect Partner such as an IP VPN or MPLS VPN network service provider or data center exchange provider.

For information about using FastConnect Classic, see *Getting Started With the Service in Using Oracle Cloud Infrastructure FastConnect Classic*.

### Connecting to Instances on IP Networks

You can also use FastConnect Classic to set up a connection between subnets in your premises and IP networks in your Compute Classic account.

To do this, create a private gateway object in Compute Classic and attach your IP networks to this private gateway. Then set up a private peering connection by using FastConnect Classic. With this connection, you can access instances on your IP networks using their private IP addresses from your on-premises private networks. You don't need to associate public IP addresses with instances on IP networks.



This figure shows private peering connections for two customers from networks in different data centers to IP networks in their Compute Classic account. Customer A uses a single FastConnect Classic connection from their data center to Oracle Cloud, while Customer B has multiple FastConnect Classic connections. Within Oracle Cloud, a private peering connection is established for each customer to connect instances on their IP networks to their private gateway.

#### Note:

Not all Oracle FastConnect Partners support private peering to enable connections using the private IP addresses of instances on IP networks. If you're connecting through an Oracle FastConnect Partner, check if your partner supports private peering. See *Getting Started With the Service in Using Oracle Cloud Infrastructure FastConnect Classic*.

Access to instances on IP networks is controlled by access control lists (ACLs). You must have the required ingress and egress security rules in place and apply those rules using ACLs that are applied to the appropriate vNICset, to enable access to instances on IP networks. You can update these ACLs at any time.

#### Workflow for Connecting to Instances on IP Networks

Here's an overview of the procedure for setting up FastConnect Classic to connect to instances on IP networks:

1. Create the required IP networks. Ensure that the subnets you specify while creating IP networks don't overlap with subnets in your on-premises networks. See [Creating an IP Network](#).

2. Create the required ACLs. An ACL is a collection of security rules. After you've created the ACL, you can reference it in each security rule that you create. When an ACL is applied, traffic is permitted only if it meets all the criteria of any one security rule in the ACL. See [Creating an ACL](#).
3. Create the required ingress and egress security rules. Each security rule defines the type of traffic permitted to or from a specified source or destination. As you create each security rule, add it to the appropriate ACL. See [Creating a Security Rule for IP Networks](#).
4. Create a vNICset. This vNICset will contain the vNICs of the instances on IP networks that you want to access using the FastConnect Classic connection. While creating the vNICset, specify the ACLs that you want to apply to this vNICset. See [Creating a vNICset](#).
5. Create the instances that you want to access over the FastConnect Classic connection and specify the appropriate IP networks and vNICsets for each instance.
6. Create a private gateway. See [Creating a Private Gateway](#).

When you create the private gateway object, specify the list of IP networks that you want to access over the FastConnect Classic connection.

 **Note:**

Note the name of your private gateway. You'll need to provide this name when you set up your FastConnect Classic connection.

7. Follow the procedure to set up a connection using Direct Cross Connects or through an Oracle FastConnect Partner. See *Getting Started With the Service in Using Oracle Cloud Infrastructure FastConnect Classic*.

After you've established a connection from your premises to instances on your IP networks, you can add IP networks to the list of accessible IP networks at any time. These additional IP network routes are automatically advertised to your remote data center or on-premises location over the FastConnect Classic private peering connection.

## Managing Cross Connects

### Topics

- [Creating a Cross Connect](#)
- [Forwarding Letter of Authorization](#)
- [Listing Cross Connects](#)
- [Updating a Cross Connect](#)
- [Deleting a Cross Connect](#)

## Creating a Cross Connect



This topic does not apply to Oracle Cloud at Customer.

Create a cross connect to establish a physical connection between the routers in your network and Oracle routers. You should create only one cross connect for every router in your network.

### Prerequisites

- FastConnect Classic must be available in the site you have selected. If FastConnect Classic isn't available in the selected site, the Cross Connect page isn't displayed in the web console.
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Cross Connects**.
4. Click **Create Cross Connect**.
5. Select or enter the following information:
  - **Name:** Enter a name for the cross connect.
  - **Port speed:** Select the appropriate port speed based on your business requirement.
  - **Make this Cross Connect redundant from:** If you are creating a secondary cross connect, select the primary cross connect that you have created. This ensures that the primary and secondary cross connects are mapped to unique routers. If you don't specify a value while creating a secondary cross connect, both primary and secondary cross connects are mapped to the same router and redundancy is not achieved at the router level. If you are creating a primary cross connect, leave this field empty.
  - **Your Circuit ID:** Enter the circuit ID, if you have assigned one for your network router. This helps you to identify the cross connect that maps to a router in your network. This is useful when you want to troubleshoot.
6. Click **Create**.

The **Confirm Create Cross Connect Details** page appears.
7. Check if the details provided are correct, make a note of the next steps that you have to perform, and then click **Confirm**.

After creating a cross connect to the primary port, repeat this task to create a redundant cross connect to a redundant router in your network to achieve end-to-end redundancy from your network edge. After creating both the cross connects, you can forward the letter of authorization. See [Forwarding Letter of Authorization](#).



## Forwarding Letter of Authorization



This topic does not apply to Oracle Cloud at Customer.

After creating a cross connect, you can view the letter of authorization which provides details that are required to set up a physical connection between your network routers and Oracle routers. You have to forward this letter to activate the cross connect by setting up the physical connection.

### Prerequisites


Before forwarding the letter of authorization, ensure that you meet the following prerequisites.

- If your data center is not colocated with the Oracle data center where your Oracle Cloud services are provisioned, contact your network service provider to order a private line from your premises to the Oracle data center. Identify a network service provider or carrier that can deliver an Ethernet private line from your premises to the Oracle edge routers. The service provider must be capable of terminating the circuit using single-mode fiber with LC connectors. The ports on Oracle routers are 1000Base-LX (1 Gbps) and 10GBASE-LR (10 Gbps). It is your responsibility to work with any network service provider or carrier of your choice to get the point-to-point Ethernet private line provisioned to the demarcation point as specified in the letter of authorization. If the network carrier is located in a different demarcation point in the cage, you must work with the network carrier to get the connection established from their demarcation point to the Oracle demarcation point.

To achieve end-to-end redundancy from your network edge, you must order redundant circuits or lines from your network service provider along with separate cross connects to Oracle edge routers from redundant routers in your network.

- Ensure that the life cycle status of the cross connect is **Provisioned**.
- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Cross Connects**.
4. Go to the cross connect for which you want to retrieve the letter of authorization. From the  menu, select **Letter of Authorization**.
5. Copy the contents of the letter of authorization and forward the letter of authorization to activate the cross connect.

- If your data center is colocated with the Oracle data center, forward the letter of authorization to the data center provider.
- If your data center is not colocated with the Oracle data center where your Oracle Cloud services are provisioned, forward the letter of authorization to the network carrier of your choice.

 **Note:**

The LOA is valid only for a limited period. If you don't establish a physical connection between your network routers and Oracle routers within this period, the LOA is revoked.

Oracle configures the routers. When the physical connection between your network routers and Oracle routers is set up and the configuration is complete, the **Physical Status** of the cross connect changes to **Up** and the **Life Cycle Status** is **Provisioned**. You can now create a virtual circuit. See [Creating a Virtual Circuit](#).

## Listing Cross Connects



This topic does not apply to Oracle Cloud at Customer.

After you've created a cross connect, you can see a list of cross connects in your account.

### Prerequisites

- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Cross Connects**.

A list of cross connects is displayed.

## Updating a Cross Connect




This topic does not apply to Oracle Cloud at Customer.

After you've created a cross connect, you can modify the circuit ID specified in the cross connect.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Cross Connects**.
4. Go to the cross connect that you want to modify. From the  menu, select **Update**.
5. Update the information as required:
  - **Your Circuit ID:** Enter the circuit ID, if you have assigned one for your network router. This helps you to identify the cross connect that maps to a router in your network. This is useful when you want to troubleshoot.
6. Click **Update**. The cross connect is updated.

## Deleting a Cross Connect




This topic does not apply to Oracle Cloud at Customer.

If you don't need a cross connect any more, you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Cross Connects**.
4. Go to the cross connect that you want to delete. From the  menu, select **Delete**.

# Managing Virtual Circuits

## Topics

- [Creating a Virtual Circuit](#)
- [Listing Virtual Circuits](#)
- [Updating a Virtual Circuit](#)
- [Deleting a Virtual Circuit](#)

## Creating a Virtual Circuit



This topic does not apply to Oracle Cloud at Customer.

A virtual circuit is a layer-2 or layer-3 Ethernet VLAN that isolates network traffic between customers. It is an isolated network path that runs over one or more physical network connections to provide a single, logical connection between the router on the edge of your network and the Oracle router. Each virtual circuit is made up of information shared between you and Oracle, as well as an Oracle FastConnect Partner (if you're connecting through an Oracle FastConnect Partner).

### Prerequisites

- FastConnect Classic must be available in the site you have selected. If FastConnect Classic isn't available in the selected site, the Virtual Circuit page isn't displayed in the web console.
- Ensure that you meet one of the following prerequisites based on whether you are setting up a public or private virtual circuit.
  - Public virtual circuit: You can advertise only public IPv4 prefixes over this connection. RFC 1918 addresses, if any, are dropped if they are advertised over the connection. Ensure that the public IPv4 prefixes that you want to advertise are registered to you in an Internet Routing Registry (IRR) or Regional Internet Registry (RIR). The BGP session is brought down or disabled if you cross the specified prefix-limit of advertising 200 IPv4 prefixes over public peering or 2000 private (RFC1918) IPv4 prefixes over private peering.
  - Private virtual circuit: You can advertise private IP addresses (RFC1918) and extend your remote data center resources that use private IP address without the need to use IPsec VPN or Network Address Translation (NAT). Your IP address are not translated, but forwarded to Oracle as is. The BGP session is brought down or disabled if you cross the specified prefix-limit of advertising 2000 private (RFC1918) IPv4 prefixes over private peering.

You must create a private gateway to use private peering. A private gateway allows you to connect from your on-premises data center to instances on IP networks using their private IP addresses. While creating a private gateway, specify the names of the IP networks that you want to associate with this private gateway. See [Creating a Private Gateway](#).

Note down the name of your private gateway. You'll need to select this private gateway while setting up your connection.

- If you are not connecting through an Oracle FastConnect Partner and connecting through Direct Cross Connects, ensure that you have created the cross connects. See [Creating a Cross Connect](#).
- You must have already created the IP networks that you want to connect to, along with the required security rules, vNICsets, and ACLs that you need to enable access to instances using FastConnect Classic. See [Connecting to Instances Using Oracle Cloud Infrastructure FastConnect Classic](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Virtual Circuits**.
4. Click **Create Virtual Circuit**.
5. Select or enter the following information.
  - **Name:** Enter a unique name for the virtual circuit.
  - **Connection Type:** Select **Direct Cross Connect** to establish a direct connection to an Oracle data center using cross connects. If you want to establish a connection through an Oracle FastConnect partner, select a partner based on their availability at the location from which you want to connect to FastConnect Classic.
  - **Circuit Type:** Select **Public** or **Private** based on whether you want to advertise public or private IP addresses over the connection. For example, select **Public** if you want to access Oracle Cloud services through Compute Classic by using public IP prefixes. Select **Private** when you want to extend your on-premise private networks to the Oracle Cloud. A private virtual circuit will enable you to connect to Oracle Cloud resources from your on-premise private (RFC1918) networks. When you use a private virtual circuit, it eliminates the need for IPSec VPN and Network Address Translation (NAT) to extend your private routing domain.
    - **Private Gateway:** If you select **Private**, you must select the private gateway that you have created. A private gateway allows you to connect from your on-premises data center to instances on IP networks using their private IP addresses.
    - **Public IP Prefixes:** If you select **Public**, specify the public IPv4 prefixes (in CIDR notation) that you want to advertise over the connection. You can also specify reverse NAT IP addresses. These IPv4 prefixes must be registered to you in an IRR or RIR.
6. Different fields appear on the console depending on the **Connection Type** that you select. If any of the following fields do not appear in the console, it indicates that those fields are not relevant for the selected **Connection Type**. Select or enter the following information based on the fields that appear.

- **AT&T NetBond Service Key:** This field appears only if you have selected AT&T NetBond as the **Connection Type**. Enter the service key that you have received from AT&T.

If you haven't ordered AT&T NetBond for Oracle FastConnect from AT&T NetBond, you can order it after creating the virtual circuit. Once you receive the service key from AT&T, you can update the virtual circuit to specify the key.

- Enter the following information to map the link that you have created to a router in Oracle Cloud. If you have created primary and secondary links to ensure high availability, specify the following information for both the links. When you set up two links between your network and Oracle Cloud, a redundant connection is established to FastConnect edge router which ensures high availability.
  - **Your Router Interface IP:** Enter the IP address of your network edge router for this VLAN in CIDR format.
  - **Oracle Router Interface IP:** Enter the IP address of the Oracle router for this VLAN in CIDR format.
  - **VLAN:** The VLAN ID that you want to use for this virtual circuit. Enter an integer between 100 and 4000. The VLAN ID must be unique.
  - **Cross Connect:** Select a cross connect that you have created.
- Enter the following information for your BGP session:
  - **Your BGP ASN:** The public or private ASN for your network. If you use a public ASN, ensure that the ASN is registered to you. You can work with your Internet service provider or one of the registries to obtain an ASN.

 **Note:**

If you do not have a registered public ASN, you can use private ASNs or use the fixed ASN provided by Oracle for configuring the network.

- **BGP MD5 Password:** Enter the BGP MD5 password.

#### 7. Click **Create**.

The virtual circuit is created.

After creating a virtual circuit, follow the procedure to set up a connection using FastConnect Classic. See *Getting Started With the Service in Using Oracle Cloud Infrastructure FastConnect Classic*.

## Listing Virtual Circuits



This topic does not apply to Oracle Cloud at Customer.

After you've created a virtual circuit, you can see a list of virtual circuits in your account.

### Prerequisites

- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Virtual Circuits**.

A list of virtual circuits is displayed.

## Updating a Virtual Circuit




This topic does not apply to Oracle Cloud at Customer.

After you've created a virtual circuit and the virtual circuit is provisioned, you can modify the parameters of the virtual circuit.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Virtual Circuits**.
4. Go to the virtual circuit that you want to modify. From the  menu, select **Update**.
5. Update the information as required. Different fields appear on the console depending on the **Connection Type** that you have selected. If any of the following fields do not appear in the console, it indicates that those fields are not relevant for the selected **Connection Type**. Select or enter the following information based on the fields that appear.
  - **Public IP prefixes:** This field appears only if you have selected the **Circuit Type** as **Public**. You can add or remove public IP prefixes. If you add

a new prefix, Oracle first verifies your company's ownership before advertising it across the connection. These IPv4 prefixes must be registered to you in an IRR or RIR. If you remove a prefix, Oracle stops advertising the prefix within a few minutes of your editing the virtual circuit.

- **AT&T NetBond Service Key:** This field appears only if you have selected AT&T Netbond as the **Connection Type**. Enter the service key that you have received from AT&T.

When you order AT&T NetBond for Oracle FastConnect, AT&T generates a service key. Enter the value of this key.

- Enter the following information to map the link that you have created to a router in Oracle Cloud. If you have created primary and secondary links to ensure high availability, specify the following information for both the links. When you set up two links between your network and Oracle Cloud, a redundant connection is established to FastConnect edge router which ensures high availability.
  - **Your Router Interface IP:** Enter the IP address of your network edge router for this VLAN in CIDR format.
  - **Oracle Router Interface IP:** Enter the IP address of the Oracle router for this VLAN in CIDR format.
  - **VLAN:** The VLAN ID that you want to use for this virtual circuit. Enter an integer between 100 and 4000. The VLAN ID must be unique.
  - **Cross Connect:** Select a cross connect that you have created.
- Enter the following information for your BGP session:
  - **Your BGP ASN:** The public or private ASN for your network. If you use a public ASN, ensure that the ASN is registered to you. You can work with your Internet service provider or one of the registries to obtain an ASN.

 **Note:**

If you do not have a registered public ASN, you can use private ASNs or use the fixed ASN provided by Oracle for configuring the network.

- **BGP MD5 Password:** Enter the BGP MD5 password.
- Only if you have selected Verizon as your Oracle FastConnect Partner, enter the following information that you have receive from Verizon. After you order Verizon SCI Connectivity for Oracle FastConnect and the service is provisioned, you'll can see the network configuration information on the Verizon SCI portal. Provide the following information that's available on the Verizon SCI portal.
  - **Verizon BGPAS Number:** Enter the Verizon BGP ASN.
  - **Primary Verizon IPAddress:** Enter the Verizon IP Address of Primary Router.
  - **Secondary Verizon IPAddress:** Enter the Verizon IP Address of Secondary Router.
  - **Primary Verizon VLANID:** Enter the Verizon VLAN ID of Primary Router.



- **Secondary Verizon VLANID:** Enter the Verizon VLAN ID of Secondary Router.
- **Cloud Service Name:** Enter `Oracle FastConnect`.

6. Click **Update**. The virtual circuit is updated.

## Deleting a Virtual Circuit




This topic does not apply to Oracle Cloud at Customer.

If you don't need a virtual circuit any more, you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Virtual Circuits**.
4. Go to the virtual circuit that you want to delete. From the  menu, select **Delete**.

## Managing Private Gateways

Setting up a private gateway allows you to connect from your on-premises data center to instances on IP networks using their private IP addresses.

### Topics

- [Creating a Private Gateway](#)
- [Listing Private Gateways](#)
- [Updating a Private Gateway](#)
- [Deleting a Private Gateway](#)

## Creating a Private Gateway



This topic does not apply to Oracle Cloud at Customer.

Setting up a private gateway allows you to connect from your on-premises data center to instances on IP networks using their private IP addresses. You can use this private gateway to establish a FastConnect Classic private peering connection.

### Prerequisites

- If you're using FastConnect Classic Partner Edition, your FastConnect Classic partner must support private peering. To find out if your partner supports FastConnect Classic private peering, see *Getting Started With the Service in Using Oracle Cloud Infrastructure FastConnect Classic*.
- You must have already created the IP networks that you want to connect to, along with the required security rules, vNICsets, and ACLs that you need to enable access to instances using FastConnect Classic. See [Connecting to Instances Using Oracle Cloud Infrastructure FastConnect Classic](#).
- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Private Gateways**.
4. Click **Create Private Gateway**.
5. Select or enter the following information:
  - **Name:** Enter a name for the private gateway.
  - **IP Networks:** Select the IP networks that you want to connect to. An IP network can be used in only one private peering connection. You can't select an IP network that has been used in another private peering connection. Also, the IP networks that you select must not have overlapping subnets.
  - **Description:** Enter a meaningful description for the private gateway.
  - **Tags:** Enter one or more tags to help you identify the private gateway.
6. Click **Create**.

The private gateway is created. Make a note of the three-part name of the private gateway and the API end point of your site. You'll need this information when you set up your FastConnect Classic connection.

To create a private gateway using the CLI, use the `opc compute private-gateway add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To create a private gateway using the API, use the `POST /network/v1/privategateway/` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After creating a private gateway, follow the procedure to set up a connection using FastConnect Classic Standard Edition or FastConnect Classic Partner Edition. See *Getting Started With the Service* in *Using Oracle Cloud Infrastructure FastConnect Classic*.

## Listing Private Gateways



This topic does not apply to Oracle Cloud at Customer.

After you've created a private gateway, you can see a list of private gateways in your account.

### Prerequisites

- To complete this task, you must have the `Compute_Monitor` or `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles* in *Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Private Gateways**.

A list of private gateways is displayed.

To list private gateways using the CLI, use the `opc compute private-gateway list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list private gateways using the API, use the `GET /network/v1/privategateway/container` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating a Private Gateway



This topic does not apply to Oracle Cloud at Customer.


After you've created a private gateway to use in a private peering connection using FastConnect Classic if required, you can modify the IP networks specified in the private gateway. You can also update the description or tags associated with the private gateway.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to

ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Private Gateways**.
4. Go to the private gateway that you want to modify. From the  menu, select **Update**.
5. Update the information as required:
  - **IP Networks:** Add or remove IP networks that you want to connect to. An IP network can be used in only one private peering connection. You can't select an IP network that has been used in another private peering connection. Also, the IP networks that you select must not have overlapping subnets.
  - **Description:** Update the description, if required.
  - **Tags:** Update the tags, if required.
6. Click **Update**. The private gateway is updated. If you modified the list of IP networks, your changes are automatically advertised to your remote data center or on-premises location.

To update a private gateway using the CLI, use the `opc compute private-gateway update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a private gateway using the API, use the `PUT /network/v1/privategateway/ name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Deleting a Private Gateway



This topic does not apply to Oracle Cloud at Customer.


If you don't need a private gateway any more, you can delete it.

### Prerequisites

- To complete this task, you must have the `Compute_Operations` role. If this role isn't assigned to you or you're not sure, then ask your system administrator to ensure that the role is assigned to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Procedure

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.

2. Click the **Network** tab.
3. In the **Network** drop-down list, expand **FastConnect**, and then click **Private Gateways**.
4. Go to the private gateway that you want to delete. From the  menu, select **Delete**.

To delete a private gateway using the CLI, use the `opc compute private-gateway delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a private gateway using the API, use the `DELETE /network/v1/privategateway/ name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

# 19

## Connecting to Oracle Cloud Infrastructure Dedicated Compute Classic Instances Using VPN



This topic does not apply to Oracle Cloud at Customer.

If you have an Oracle Cloud Infrastructure Dedicated Compute Classic account, you can use the Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic service to establish a secure communication channel between your data center and the instances in your Compute Classic site.

After this service is provisioned, you can configure your VPN gateway device to connect to the Oracle Cloud VPN gateway. See [About Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic](#).

Alternatively, you can set up a VPN connection to your site using VPN as a Service (VPNaaS). See [Setting Up VPN Using VPNaaS](#).

### Note:

If you don't have an Oracle Cloud Infrastructure Dedicated Compute Classic account, to configure VPN access to your instances see [Connecting to Instances in a Multitenant Site Using VPN](#).

### Topics

- [About Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic](#)
- [Requesting Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic](#)
- [Configuring Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Gateway](#)
- [Managing Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Connections](#)
- [Accessing Your Instances Using VPN](#)

## About Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic



This topic does not apply to Oracle Cloud at Customer.

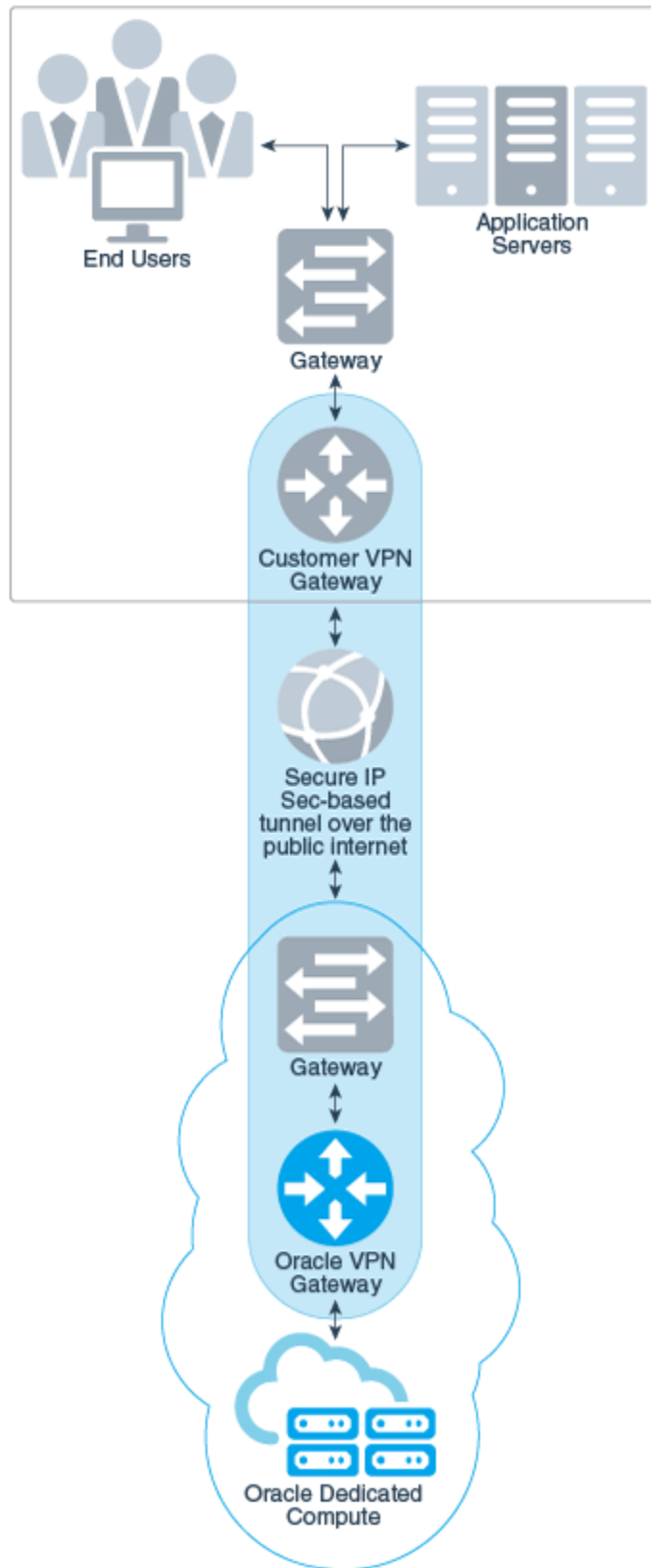
With Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic you can configure a site-to-site VPN connection to access your instances. While you can continue to access your instances over the public internet securely using SSH or RDP, using a site-to-site VPN connection enhances security by creating secure IPSec-based tunnels between your data center and the instances in your Oracle Cloud Infrastructure Dedicated Compute Classic site.

**Note:**

Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic is not available by default with Oracle Cloud Infrastructure Dedicated Compute Classic. It must be requested separately. See [Requesting Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic](#).

Using Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic, you can create up to 20 VPN tunnels to your Oracle Cloud Infrastructure Dedicated Compute Classic site. You can use any internet service provider to access your Oracle Cloud Infrastructure Dedicated Compute Classic site, provided you have a VPN device to terminate an IPSec VPN tunnel.

IPSec is a suite of protocols designed to authenticate and encrypt all IP traffic between two locations. This allows sensitive data to pass securely over networks that would otherwise be considered insecure. Traffic between your data center and your Oracle Cloud Infrastructure Dedicated Compute Classic site is encrypted and transmitted through this secure tunnel. So your data can't be stolen or intercepted. In other words, by using a site-to-site VPN connection, you're effectively extending your data center network to include instances in your Oracle Cloud Infrastructure Dedicated Compute Classic site.





## Requesting Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic



This topic does not apply to Oracle Cloud at Customer.

To set up your VPN connection, you must first request the Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic service.

You can request this service either while subscribing to Compute Classic, or later on. To request the Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic service, work with your Oracle sales representative to raise a Service Request (SR). You'll receive a form asking you to provide detailed information. Use this form to provide the following information:

- A preshared key (PSK) in the 128-bit/SHA1 format.
- (Optional) A range of 8000 private IP addresses. These should be provided as network prefixes in the CIDR format (for example, n.n/19). When you create instances or restart existing instances, the private IP address of each instance is dynamically assigned from this range of IP addresses. Note that when your Compute Classic account is provisioned, a range of private IP addresses is assigned from the 100.64/10 address range. You can either use this assigned range or specify another range of private IP addresses. An 8000-address block can meet the IP address requirements of up to 2000 instances.

### Note:

Ensure that the range of IP addresses that you provide doesn't overlap with the private IP addresses used by other devices on your on-premises network.

Also check that the private IP addresses of existing Compute Classic instances do not conflict with private IP addresses used by any of your on-premises devices. Such a conflict becomes relevant only when you configure a VPN tunnel and your Compute Classic instances become an extension of your on-premises network.

It can take up to two weeks to process your request. After your SR is processed, Oracle provides you the encoded PSK along with the name and public IP address of the Oracle Cloud VPN gateway. Use these to configure your VPN gateway to connect to the Oracle Cloud VPN gateway. See [Configuring Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Gateway](#).

## Accessing Your Instances Using VPN



This topic does not apply to Oracle Cloud at Customer.

After you've configured your VPN gateway and started a VPN connection, you can securely access your Oracle Cloud Infrastructure Dedicated Compute Classic site by using the private IP address of each instance.

 **Note:**

The private IP address of an instance might be assigned dynamically. When an instance is restarted, this dynamically assigned private IP address might change.

Do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Go to the instance that you want to access. Make a note of the private IP address of the instance.
3. After you've enabled a VPN tunnel, the instances in your Compute Classic site appear as an extension of the network in your site. You can use the private IP address of a Compute Classic instance to connect to the instance as you would connect to any host in your data center.

 **Note:**

After you've enabled a VPN tunnel, you can also continue to access your instances over the public Internet, as you did earlier. Any security rules that you might have defined for your instances continue to apply.

## Configuring Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Gateway



This topic does not apply to Oracle Cloud at Customer.

After the Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic service is provisioned, you must configure your VPN gateway to connect to the Oracle Cloud Infrastructure VPN gateway.

Do the following:

1. Configure Internet Key Exchange (IKE)
2. Configure IPSec
3. Configure a tunnel interface
4. Configure a static route

For a sample configuration of a VPN gateway, see [Example Configuration of a VPN Gateway](#).

After configuring your VPN gateway, to start a VPN connection, see [Managing Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Connections](#).

## Example Configuration of a VPN Gateway



This topic does not apply to Oracle Cloud at Customer.

This example provides the a sample configuration for your VPN gateway. You must perform this configuration for each VPN tunnel that you create.

This example is specific to Junos SRX series VPN devices. However, the IKE & IPSec parameters should generally be applicable to any device complying to IPSec VPN. As long as your VPN device is compatible with the IPSec VPN standards, and your VPN device is set up according to IKE and IPSec parameters specified in this example, you should be able to configure your VPN connection.

```
#
# VPN identifier in the e.g. below is tagged as, "vpn-dcz-site-1", to
# represent vpn connection to
# Oracle "dcz" from a customer site "site-1". Customers can create VPN
# connections from other sites as well. Each zone
# supports up to five different VPN tunnels.
# VPN Connection ID : vpn-dcz-site-1
#
#
#
-----
-----
# IPSec Tunnel #1
#
-----
-----
# #1: Internet Key Exchange (IKE) Configuration
#
# A proposal is established for the supported IKE encryption,
# authentication, Diffie-Hellman, and lifetime parameters.
#
set security ike proposal pre-g2-aes128-sha authentication-method pre-
shared-keys
set security ike proposal pre-g2-aes128-sha dh-group group2
set security ike proposal pre-g2-aes128-sha authentication-algorithm sha1
set security ike proposal pre-g2-aes128-sha encryption-algorithm aes-128-
cbc
set security ike proposal pre-g2-aes128-sha lifetime-seconds 86400

# An IKE policy is established to associate a Pre Shared Key with the
# defined proposal.Customer can have different sites where they are
# connecting from.
# Replace the the ike policy names appropriately for the site in the
# statements below.
# "dcz" below refers to "Dedicated Compute Zone".
```

```

#
set security ike policy dcz-site-1-ike-policy mode main
set security ike policy dcz-site-1-ike-policy proposals pre-g2-aes128-sha
set security ike policy dcz-site-1-ike-policy pre-shared-key ascii-text
"Use_pre_shared_key_received_from_Oracle"

# The IKE gateway is defined to be the Virtual Private Gateway. The
gateway
# configuration associates a local interface, remote IP address, and
# IKE policy.
#
# This example shows the outside of the tunnel as interface ge-0/0/0.0.
# This should be set to the interface that IP address 192.168.111.3 is
# associated with.
# This address is configured with the setup for your Customer Gateway.
#
# If the address changes, the Customer Gateway and VPN Connection must be
recreated.
#
set security ike gateway gw-vpn-site-1 ike-policy dcz-site-1-ike-policy
set security ike gateway gw-vpn-site-1 external-interface ge-0/0/0.0
set security ike gateway gw-vpn-site-1 address 192.168.111.3

# Troubleshooting IKE connectivity can be aided by enabling IKE tracing.
# The configuration below will cause the router to log IKE messages to
# the 'kmd' log. Run 'show messages kmd' to retrieve these logs.
# set security ike traceoptions file kmd
# set security ike traceoptions file size 1024768
# set security ike traceoptions file files 10
# set security ike traceoptions flag all

# #2: IPsec Configuration
#
# The IPsec proposal defines the protocol, authentication, encryption, and
# lifetime parameters for our IPsec security association.
#
set security ipsec proposal ipsec-phase2-proposal protocol esp
set security ipsec proposal ipsec-phase2-proposal authentication-algorithm
hmac-shal-96
set security ipsec proposal ipsec-phase2-proposal encryption-algorithm
aes-128-cbc

# The IPsec policy incorporates the Diffie-Hellman group and the IPsec
# proposal.
#
set security ipsec policy ipsec-phase2-policy perfect-forward-secrecy keys
group2
set security ipsec policy ipsec-phase2-policy proposals ipsec-phase2-
proposal

# A security association is defined here. The IPsec Policy and IKE gateways
# are associated with a tunnel interface (st0.0).
# The tunnel interface ID is assumed; if other tunnels are defined on
# your router, you will need to specify a unique interface name
# (for example, st0.10).

```

```

#
set security ipsec vpn vpn-dcz-site-1 bind-interface st0.0
set security ipsec vpn vpn-dcz-site-1 ike gateway gw-vpn-site-1
set security ipsec vpn vpn-dcz-site-1 ike ipsec-policy ipsec-phase2-policy
set security ipsec vpn vpn-dcz-site-1 establish-tunnels-immediately

# #3: Tunnel Interface Configuration
#

# The tunnel interface is configured with the internal IP address &
# recommended that IP address in the same subnet as the remote end IP
# address.
# This IP will be conveyed to the customer.
set interfaces st0.0 family inet
set interfaces st0.0 family inet mtu 1436 -- (Actual value needs to
# investigated)
set security zones security-zone trust interfaces st0.0

# The security zone protecting external interfaces of the router must be
# configured to allow IKE traffic inbound.
#
set security zones security-zone untrust host-inbound-traffic system-
services ike

# This option causes the router to reduce the Maximum Segment Size of
# TCP packets to prevent packet fragmentation.
#
set security flow tcp-mss ipsec-vpn mss 1350

#
-----
-----
# #4: Static Route Configuration
#

# Your Customer Gateway needs to set a static route for the prefix
# corresponding to your VPC on the tunnel.
# An example for a VPC with the prefix 10.0.0.0/16 is provided below
# set routing-options static route 10.0.0.0/16 next-hop st0.0
#

```

## Managing Your Oracle Cloud Infrastructure Networking Classic – VPN for Dedicated Compute Classic Connections



This topic does not apply to Oracle Cloud at Customer.

After you've configured your VPN gateway device, you can manage your VPN connections using the REST API or the Compute Classic web console.

### Topics

- [Starting a VPN Connection](#)
- [Listing Your VPN Connections](#)
- [Viewing Details of a VPN Connection](#)
- [Updating a VPN Connection](#)
- [Disabling a VPN Connection](#)
- [Deleting a VPN Connection](#)

## Starting a VPN Connection



This topic does not apply to Oracle Cloud at Customer.

After you've configured your VPN gateway to connect to the Oracle Cloud VPN gateway, you can start a VPN connection. You can create up to 20 VPN tunnels between your data center and your Oracle Cloud Infrastructure Dedicated Compute Classic site.

To start a VPN connection using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

3. To create a VPN tunnel, click **Create VPN Endpoint**.
4. Enter the following:
  - **Name:** Specify a name for the VPN tunnel.
  - **VPN Gateway IP:** Enter the IP address of the VPN gateway in your data center through which you want to connect to the Oracle Cloud VPN gateway. Your gateway device must support route-based VPN and IKE (Internet Key Exchange) configuration using pre-shared keys.
  - **Pre-shared Key:** Enter the 128-bit/SHA1 pre-shared key. This must be the same key that you provided when you requested the service.
  - **Reachable Routes:** Enter a list of routes (network prefixes in CIDR notation) that are reachable through this VPN tunnel.
5. To start the VPN connection as soon as the tunnel is created, click **Enabled**.

To start a VPN connection using the CLI, use the `opc compute vpn-endpoint add` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To start a VPN connection using the API, use the `POST /vpnendpoint/` method with the `enabled` parameter. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After you've established a VPN connection to your Oracle Cloud Infrastructure Dedicated Compute Classic site, if you want to end the VPN connection, see [Disabling a VPN Connection](#).

## Listing Your VPN Connections



This topic does not apply to Oracle Cloud at Customer.

To list your VPN connections using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

On this page, you can see all the VPN endpoints that you've created, and you can start, stop, view, update, or delete your VPN endpoints.

To list your VPN connections using the CLI, use the `opc compute vpn-endpoint list` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To list your VPN connections using the API, use the `GET /vpnendpoint/container` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Viewing Details of a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

To view details of a VPN connection using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

3. Go to the VPN endpoint that you want to view. From the  menu, select **Update**. The Edit VPN Endpoint page shows the details of the VPN endpoint.

To view details of a VPN connection using the CLI, use the `opc compute vpn-endpoint get` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To view details of a VPN connection using the API, use the `GET /vpnendpoint/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Updating a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

After you've configured your VPN connection, you can update the connection to enable or disable the VPN tunnel or to change other connection details.

To update a VPN connection using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

3. Go to the VPN endpoint that you want to update. From the  menu, select **Update**. Enter the details that you want to change and then click **Update**. You can update any of the details, except **name**.

To update a VPN connection using the CLI, use the `opc compute vpn-endpoint update` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To update a VPN connection using the API, use the `PUT /vpnendpoint/name` method. You can update any of the parameters, except `name`. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Disabling a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

To disable or end a VPN connection using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

3. Go to the VPN endpoint that you want to disable. From the  menu, select **Update**.
4. In the Edit VPN Endpoint page, deselect the **Enabled** check box, and then click **Update**.



To disable or end a VPN connection using the CLI, use the `opc compute vpn-endpoint update` command without the `--enabled` option. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To disable or end a VPN connection using the API, use the `PUT /vpnendpoint/name` method without the `enabled` parameter. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After disabling a VPN connection, you can start it again later on. See [Starting a VPN Connection](#).

## Deleting a VPN Connection




This topic does not apply to Oracle Cloud at Customer.

To delete a VPN connection using the web console, do the following:

1. Sign in to the Compute Classic console. If your domain spans multiple sites, select the appropriate site. To change the site, click the **Site** menu near the top of the page.
2. Click the **Network** tab, and then in the **Network** drop-down list, click **VPN Endpoints**.

The VPN Endpoints page is displayed.

3. Go to the VPN endpoint that you want to delete. From the  menu, select **Delete**.

To delete a VPN connection using the CLI, use the `opc compute vpn-endpoint delete` command. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

To delete a VPN connection using the API, use the `DELETE /vpnendpoint/name` method. For more information, see *REST API for Oracle Cloud Infrastructure Compute Classic*.

After deleting a VPN connection, you can create it again later on. See [Starting a VPN Connection](#).

# Automating Instance Initialization Using `opc-init`

## Topics

- [About `opc-init`](#)
- [Prerequisites for Using `opc-init`](#)
- [Defining Instance Initialization Attributes](#)
- [User Data Attributes](#)
- [Using `opc-init` in a Private Machine Image](#)

## About `opc-init`

When you create an instance in Compute Classic, you get a virtual machine running the operating system specified by the image that you had selected while creating the instance. Before you start using the instance, you may want to customize it based on your business needs. For example, you may want to create users, install additional packages, add SSH keys, run certain scripts, and so on. Instead of doing all of this initial configuration manually every time an instance starts, you can use the `opc-init` package to set up these steps to be performed automatically when an instance starts.

The **`opc-init`** package contains scripts provided by Oracle that allow you to perform specified instance initialization tasks automatically every time an instance is created. You can specify instance initialization tasks in the form of user data when you create an instance. The `opc-init` scripts query the metadata service on the instance for this user data. The specified user data is then used by the `opc-init` scripts to perform the required prebootstrapping tasks. In addition, `opc-init` adds the SSH public keys specified during instance creation to the `authorized_keys` file of the default `opc` user.

The `opc-init` scripts are included by default in Oracle-provided Linux and Windows machine images. If you want to use `opc-init` with private Oracle Linux images, you can install the `opc-init` scripts while creating the image. See [Using `opc-init` in a Private Machine Image](#).

### Note:

Solaris machine images don't include the `opc-init` scripts. So you can't use `opc-init` to automate instance initialization of Solaris instances.

If you specify user data while creating an instance, the `opc-init` package retrieves this data from the metadata service and uses it to do the following:

| On a Linux Instance                                                                                                            | On a Windows Instance                       |
|--------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------|
| Set the specified http and https proxy for the opc-init process                                                                | Set the specified administrator password    |
| Run the specified prebootstrap scripts                                                                                         | Add the specified users                     |
| Execute the specified operations for yum elements, including installing yum repositories, and installing or upgrading packages | Run the specified prebootstrap scripts      |
| Process chef elements and install, configure, and launch chef                                                                  | Enable Windows remote management            |
| -                                                                                                                              | Enable Remote Desktop Protocol (RDP) access |

For user data attributes and templates, see [User Data Attributes](#).

## Prerequisites for Using opc-init

You can use the opc-init package to automate instance initialization. To effectively use this tool, you must:

- Be familiar with chef-solo
- Be familiar with orchestrations
- Know JSON
- Be aware of the Ruby syntax
- Have root access to the instance you want to configure
- Know the public IP address of the instance
- Have the required permissions and licences for installing Chef, Ruby, and the associated Ruby gems

In addition, if you want to write your own scripts, you must be familiar with Python.

## Defining Instance Initialization Attributes

You can automate instance initialization by providing scripts that install applications or perform other prebootstrap tasks when you create an instance. These scripts are specified as user data when you create an instance. If you specify user data while creating an instance, the opc-init package retrieves this data and uses it to perform the specified prebootstrap tasks.

You can specify user data in the following ways:

- If you create an instance using the web console, use the **Custom Attributes** field to specify user data. The text you enter in this field must be in JSON format.

Custom Attributes

```
{
  "pre-bootstrap": {
    "scriptURL": "scriptURL",
    "failonerror": true
  },
  "chef": {
```

- If you create an instance using an orchestration or a launch plan, use the `attributes` parameter of the `instance` object type to enter user data.

```
{
  "name": "/acme/jones/my_orchestration",
  "description": "sample orchestration with user-defined data",
  "opplans": [
    {
      "label": "my_instance_1",
      "obj_type": "launchplan",
      "objects": [
        {
          "instances": [
            {
              "imagelist": "/oracle/public/oe1_6.4_2GB",
              "shape": "oc3",
              "name": "/acme/jones/primary_webserver",
              "sshkeys": ["/acme/jones/my_sshkey"],
              "networking": {},
              "attributes": {
                "userdata": {
                  {
                    "pre-bootstrap": {},
                    "chef": {},
                    "yum_repos": {},
                    "packages": [ ]
                  }
                }
              }
            }
          ]
        }
      ]
    }
  ]
}
```

- If you use the API to add a custom machine image to Compute Classic, then you can use the `attributes` parameter of the `POST /machineimage/name` method to enter user data.

```
{
  "account": "/Compute-acme/cloud_storage",
  "name": "/Compute-acme/jack.jones@example.com/oraclelinux-x64",
  "no_upload": true,
  "file": "oraclelinux-x64.img.tar.gz",
  "sizes": {"total":0}
  "attributes": {
    "userdata": {
      {
        "pre-bootstrap": {},
        "chef": {},
        "yum_repos": {},
        "packages": [ ]
      }
    }
  }
}
```

This user data is then added to all instances created using this machine image. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

- If you use the API to create an image list entry, then you can use the `attributes` parameter of the `POST /imagelist/name/entry/` method to enter user data.

```

{
  "machineimages": ["/Compute-acme/jack.jones@example.com/o166_40GB"],
  "attributes": {
    "type": "Oracle Linux 6.6",
    "userdata":
      {
        "pre-bootstrap": {},
        "chef": {},
        "yum_repos": {},
        "packages": [ ]
      }
  }
}
"version": 2
}

```

This user data is then added to all instances created using this machine image. See *REST API for Oracle Cloud Infrastructure Compute Classic*.

#### Note:

- If you want identical user data to be available to a set of instances, use a machine image or image list entry to add user data. For example, you might require a particular prebootstrap script to be executed or specific applications to be installed on all instances that use a particular image. By adding the user data in the machine image or the image list entry, you ensure that each time you use that image, your instance is created with the specified user data.
- If each instance should have unique user data, use an orchestration or the web console to provide specific user data for each instance. This is useful if, for example, you want to specify a unique user name and password, or inject a unique SSH public key into each instance.

If you specify identical attributes in a machine image, an image list entry, and while creating an instance, then the values specified in the image list entry override the values specified in the machine image, and the values specified while creating the instance override the values specified in the image list entry and in the machine image. Attributes with unique keys are appended. For example, consider that in the machine image attributes, you specify the following key-value pairs:

- {"key1": "value1"}
- {"key2": "value2"}

In the image list entry attributes, you specify the following key-value pairs:

- {"key1": "value1-a"}
- {"key3": "value3"}

And in the attributes entered while creating an instance using the web console or an orchestration, you specify the following key-value pairs:

- {"key1": "value1-b"}
- {"key4": "value4"}

Then, when your instance is created, key1 will contain the value specified while creating the instance, while the other attributes specified in the machine image, image

list entry, and while creating the instance will get appended. When you view user data on the instance, you'll see the following attributes:

- `{"key1": "value1-b"}`
- `{"key2": "value2"}`
- `{"key3": "value3"}`
- `{"key4": "value4"}`

Although you can use custom attributes to enter any custom data that you require, if you want the `opc-init` package to use this information, you must use the `userdata` attribute of the `attributes` parameter. You can use `userdata` to specify the following instance initialization instructions:

- The `http` and `https` proxy.
- Prebootstrap scripts. You can either provide the script inline, or point to a URL where the script is available. This URL must be accessible from the instance.
- Chef attributes. You can provide instructions for the instance to be configured either using `chef solo`, or as a `chef client`.
- Yum repositories. If you specify a yum repository, that repository is used to download and install the specified packages.
- A list of packages to be installed on the instance.

The information that you enter as `userdata` is stored on the instance at the location: `http://192.0.0.192/latest/user-data`. You can view the user data on your instance at this location using `curl`. For example:

```
curl http://192.0.0.192/latest/user-data
```

For information about the specific nested attributes that you can use in the `userdata` attribute, see [User Data Attributes](#). For more details about retrieving user data, see [Retrieving User-Defined Instance Attributes](#).

## User Data Attributes

You can automate instance initialization by providing scripts or other instructions to perform prebootstrap tasks or install applications when you create an instance. These instance initialization instructions are provided as user-defined data using the `userdata` attribute when you create an instance.

### User Data Attributes Used on Oracle Linux Instances

The following attributes of the `userdata` attribute are used by the `opc-init` package to perform instance initialization tasks in Oracle Linux instances.

- [http and https Proxy Attributes](#)
- [Prebootstrap Attributes](#)
- [Chef Solo Attributes](#)
- [Chef Client Attributes](#)
- [Yum Repository Attributes](#)
- [Packages and Package Upgrade Attributes](#)

- [OPC Agent Attributes](#)

The sequence for executing attributes on an Oracle Linux instance is as follows:

- http-proxy
- https-proxy
- prebootstrap
- yum-repos
- packages
- package\_upgrades
- chef

### http and https Proxy Attributes

The following sample JSON shows the http-proxy and https-proxy attribute of the userdata attribute.

```
"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "http-proxy": "your-http-proxy"
        "https-proxy": "your-https-proxy"
      }
      <Specify other userdata attributes here, if required.>
    }
    <Specify other attributes here, if required.>
  }
]
```

### Prebootstrap Attributes

The following sample JSON shows the prebootstrap attribute of the userdata attribute, with the script specified in a URL.

```
"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "pre-bootstrap": {
          "scriptURL": "http://location_of_script",
          "failonerror": true
        }
        <Specify other userdata attributes here, if required.>
      }
      <Specify other attributes here, if required.>
    }
  }
]
```

The following sample JSON shows the prebootstrap attribute of the userdata attribute, with the script specified inline.

```

"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "pre-bootstrap": {
          "failonerror": true,
          "script": [
            "line1_ofscript",
            "line2_ofscript",
            ...,
            "lineN_ofscript"
          ]
        }
      }
      <Specify other userdata attributes here, if required.>
    }
    <Specify other attributes here, if required.>
  }
]

```

The following example shows a script to install the Chef client using the `prebootstrap` attribute and the `chef` attribute. For information about Chef attributes, see [Chef Solo Attributes](#) and [Chef Client Attributes](#).

```

{
  "attributes": {
    "userdata": {
      "pre-bootstrap": {
        "script": [
          "curl -o chef-client.rpm https://download_location/
chef-12.3.0-1.el6.x86_64.rpm",
          "rpm -Uvh chef-client.rpm"
        ]
      },
      "chef": {
        "mode": "solo",
        "version": "12.3.0",
        "run_list": ["recipe[learn_chef_httpd]"],
        "cookbooks_url": ["https://download_location/cookbooks.zip"],
        "initial_attributes": {
          "wlss": {
            "key1": "a b c",
            "key2": "/u01/app/oracle",
            "key3": ""
          }
        }
      }
    }
  }
}

```

A description of the `prebootstrap` attributes is provided in the following table. Nested attributes are indented in the `Attributes` column to indicate their hierarchy.



| Attribute     | Required or Optional | Description                                                                                                                                                                                                                                                                                                                                                           |
|---------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| pre-bootstrap | Optional             | This attribute allows you to specify a script that must be run prior to any other instance initialization that is performed by the opc-init package. You can either enter the script here, or point to a URL. This attribute contains the following nested attributes:                                                                                                |
| script        | Optional             | Enter the lines of the prebootstrap script, formatted as a JSON array with each line of the script represented as one element of the array. The metadata service presents this array to the instance as text, with each array element separated by a line break.<br><br>Don't enter a script if you provide a script URL. You can specify either script or scriptURL. |
| scriptURL     | Optional             | Enter the script location. This location must be accessible to the instance.<br><br>Don't provide a script URL if you provide a script. You can specify either script or scriptURL.                                                                                                                                                                                   |
| failonerror   | Optional             | Specifies whether the prebootstrap process should stop if the script encounters an error. To stop bootstrapping, specify failonerror as true. The default is false. If set to false, the bootstrapping operations continue and any errors encountered by the script are logged in the /var/log/opc-compute/opc-init.log file.                                         |

### Yum Repository Attributes

The following sample JSON shows the yum\_repos attribute of the userdata attribute.

```
"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "yum_repos": {
          "repol": {
            "baseurl": "http://location_of_yum_repo",
            "enabled": "0",
            "failovermethod": "priority",
            "gpgcheck": "true",
            "gpgkey": "file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL",
            "name": "Extra Packages for Enterprise Linux 5 - Testing",
            "proxy": "http://proxy_server:80"
          }
        }
      }
    }
    <Specify packages and package_upgrades here.>
    <Specify other userdata attributes here, if required.>
  }
  <Specify other attributes here, if required.>
}
```

A description of the yum\_repos attributes is provided in the following table.

| Attribute                  | Required or Optional | Description                                                                                                                                                           |
|----------------------------|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| yum_repos                  | Optional             | This attribute allows you to specify the desired .repo file and the name of the yum repository. This attribute contains the following nested attributes:              |
| Name of the yum repository | Required             | This attribute has no name. You must specify the name of the yum repository to be added. This is used as the repository filename in the format <i>filename</i> .repo. |
| baseurl                    | Required             | The URL of the yum repository.                                                                                                                                        |
| Other optional attributes  | Optional             | You can add all other repository file configuration options as nested attributes under the repository name.                                                           |

### Packages and Package Upgrade Attributes

The following sample JSON shows the `packages` and `package_upgrades` attributes of the `userdata` attribute. These attributes are used to specify packages for yum install and yum upgrade.

```
"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        <Specify yum attributes here.>
        "packages": ["git-core",["sysstat", "v1"]],
        "package_upgrades": true
        <Specify other userdata attributes here, if required.>
      }
    }
    <Specify other attributes here, if required.>
  }
]
```

A description of the `packages` and `package_upgrades` attributes is provided in the following table.

| Attribute        | Required or Optional | Description                                                                                                                                                                                                                                                                |
|------------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| packages         | Optional             | A JSON array of the packages you want yum to install from the repositories. Each list entry consists of a single package. If you want to specify a package version, then the list entry is represented as a two-element array of the format <code>[name, version]</code> . |
| package_upgrades | Optional             | A boolean value indicating whether you want to run yum update on the instance.                                                                                                                                                                                             |

### Chef Solo Attributes

To install Chef client on an instance, either the image used for creating the instance must include the Chef client installation package, or you must install the Chef client in either of the following ways:

- By using the `yum_repos` and `packages` attributes.
- By specifying a script using the `prebootstrap` attribute. See [Prebootstrap Attributes](#).

The following sample JSON shows the `chef` attribute of the `userdata` attribute, for a Chef solo configuration.

```
"instances": [
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "chef": {
          "run_list": ["recipe[apache2]"],
          "cookbooks_url": ["http://location_of_cookbooks/cookbooks.zip"],
          "version": "11.4.2",
          "mode": "solo",
          "initial_attributes": {
            "apache": {
              "prefork": {
                "maxclients": 100,
                "keepalive": "off"
              }
            }
          }
        }
      }
    }
    <Specify other userdata attributes here, if required.>
  }
  <Specify other attributes here, if required.>
}
]
```

A description of the `chef` attributes for a Chef solo configuration is provided in the following table. Nested attributes are indented in the Attributes column to indicate their hierarchy.

| Attribute                       | Required or Optional | Description                                                                                                                                    |
|---------------------------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>chef</code>               | Required             | This attribute allows you to specify data used by Chef for a Chef solo configuration. This attribute contains the following nested attributes: |
| <code>cookbooks_url</code>      | Required             | A JSON array of publicly accessible URLs containing the cookbook archives that you want to use, in zip, tar, or gz format.                     |
| <code>run_list</code>           | Required             | A JSON array of recipes that Chef runs to configure your instance.                                                                             |
| <code>initial_attributes</code> | Optional             | Data in JSON format (elements, arrays, or values) that is translated into node-level attributes to be consumed by Chef recipes.                |
| <code>version</code>            | Required             | The Chef version to install. The default is 11.4.2.                                                                                            |

| Attribute     | Required or Optional | Description                                                                                                                                                                                                                        |
|---------------|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| mode          | Required             | The mode must be set to solo.                                                                                                                                                                                                      |
| ruby_version  | Optional             | The Ruby version to install when using gem files. The default is 1.8.                                                                                                                                                              |
| install_type  | Optional             | The install type. You can specify gems, packages, or omnibus. The default is gems.                                                                                                                                                 |
| force_install | Optional             | This attribute allows you to specify whether Chef should be installed even if it has been installed earlier. The default is false.                                                                                                 |
| omnibus_url   | Optional             | The location to download the omnibus Chef installer if the install_type attribute is specified as omnibus. The default location is <a href="https://www.opscode.com/chef/install.sh">https://www.opscode.com/chef/install.sh</a> . |
| databag_file  | Optional             | A comma-separated list of publicly accessible anonymous HTTP URLs pointing to Chef databag files.                                                                                                                                  |

### Chef Client Attributes

The following sample JSON shows the `chef` attribute of the `userdata` attribute, for a Chef client configuration. You can use these attributes to configure your instance using a Chef server, if you've set up a Chef server in your account.

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "chef": {
          "run_list": ["recipe[oui]"],
          "chef_server_url": "https://server IP",
          "mode": "client",
          "chef_node_name": "testnode1",
          "chef_validator_location": "download (http:// or file://) location with
filename" ,
          "version": "11.4.2",
          "initial_attributes": {
            "oui": {
              "response_file_url": "responsefile url/filename.rsp",
              "installer_url": "installer url/filename.zip",
              "ignoreSysPrereqs": true
            }
          }
        }
      }
    }
    <Specify other userdata attributes here, if required.>
  }
  <Specify other attributes here, if required.>
}
]
```

A description of the `chef` attributes for a Chef client configuration is provided in the following table. Nested attributes are indented in the Attribute column to indicate their hierarchy.

| Attribute                            | Required or Optional | Description                                                                                                                                                                                                          |
|--------------------------------------|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>chef</code>                    | Required             | This attribute allows you to specify data used by Chef for a Chef client configuration. This attribute contains the following nested attributes:                                                                     |
| <code>run_list</code>                | Optional/Required    | A JSON array of recipes that Chef runs to configure your instance. If the node is predefined on your Chef server with appropriate roles, this attribute is optional. Otherwise, it is required.                      |
| <code>initial_attributes</code>      | Optional             | Data in JSON format (elements, arrays, or values) that is translated into node-level attributes to be consumed by Chef recipes.                                                                                      |
| <code>version</code>                 | Optional             | The Chef version to install. The default is 11.4.2.                                                                                                                                                                  |
| <code>mode</code>                    | Required             | The mode must be set to <code>client</code> .                                                                                                                                                                        |
| <code>chef_server_url</code>         | Required             | The URL where the Chef client can access the Chef server.                                                                                                                                                            |
| <code>chef_node_name</code>          | Required             | A unique name used by the Chef client to register with the Chef sever.                                                                                                                                               |
| <code>chef_validator_location</code> | Required             | The URL from where <code>validation.pem</code> is downloaded to <code>/etc/chef</code> .                                                                                                                             |
| <code>validation_client_name</code>  | Optional             | The name used by the validator to communicate with the Chef server. Default is <code>chef-validator</code> .                                                                                                         |
| <code>ruby_version</code>            | Optional             | The Ruby version to install when using gem files. The default is 1.8.                                                                                                                                                |
| <code>install_type</code>            | Optional             | The install type. You can specify <code>gems</code> , <code>packages</code> , or <code>omnibus</code> . The default is <code>gems</code> .                                                                           |
| <code>force_install</code>           | Optional             | This attribute allows you to specify whether Chef should be installed even if it has been installed earlier. The default is <code>false</code> .                                                                     |
| <code>omnibus_url</code>             | Optional             | The location to download the omnibus Chef installer if the <code>install_type</code> attribute is specified as <code>omnibus</code> . The default location is <code>https://www.opscode.com/chef/install.sh</code> . |

| Attribute                 | Required or Optional | Description                                                                                       |
|---------------------------|----------------------|---------------------------------------------------------------------------------------------------|
| <code>databag_file</code> | Optional             | A comma-separated list of publicly accessible anonymous HTTP URLs pointing to Chef databag files. |

### OPC Agent Attributes

You can use the `opc_guest_agent_enabled` attribute of the `userdata` attribute to disable the OPC agent. If you create an instances using an Oracle-provided Oracle Linux image with the release version 16.4.6 or later, the OPC agent is installed and enabled by default. Disabling the OPC agent prevents memory utilization metrics from being collected on your instance.

When you create an instance with this attribute set to `false`, the OPC agent is disabled whenever the instance is created or re-created. You won't be able to manually start the agent later on. If you want to reserve the ability to stop or restart the agent at any time, you can disable or enable the OPC agent after creating an instance by logging in to the instance and stopping or starting the agent. See [Enabling and Disabling the OPC Agent](#).

The following sample JSON shows the `opc_guest_agent_enabled` attribute of the `userdata` attribute.

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "opc_guest_agent_enabled": "false"
      }
      <Specify other userdata attributes here, if required.>
    }
    <Specify other attributes here, if required.>
  }
]
```

### User Data Attributes Used on Windows Instances

The following attributes of the `userdata` attribute are used by the `opc-init` package to perform instance initialization tasks in Windows instances.

- [Enabling RDP Access and Specifying the Password for the Administrator User](#)
- [Creating Users](#)
- [Specifying a Script Using Prebootstrap Attributes](#)
- [Enabling Windows Remote Management \(WinRM\)](#)

#### Note:

Chef, yum, and proxy attributes aren't supported on Windows instances.

The sequence for executing attributes on a Windows instance is as follows:

- administrator\_password
- users
- pre\_bootstrap
- winrm
- enable\_rdp

### Enabling RDP Access and Specifying the Password for the Administrator User

To enable the Administrator user to connect to the instance by using a remote desktop protocol (RDP) connection, you must specify the password for the user and also enable RDP access. You can do this by specifying the `administrator_password` and `enable_rdp` attributes in the `userdata` section of your orchestration, as shown in the following example:

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "administrator_password": "somePassword",
        "enable_rdp": true
      }
      <Specify other attributes here, if required.>
    }
  }
]
```

### Creating Users

You can also specify a list of users that must be created automatically after the Windows instance is launched, by specifying the required users and their passwords in the `users` attribute, as shown in the following example:

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "administrator_password": "somePassword",
        "enable_rdp": true,
        "users":[
          {
            "name": "john",
            "password": "somePassword"
          }
          {
            "name": "amelia",
            "password": "somePassword"
          }
        ]
      }
      <Specify other attributes here, if required.>
    }
  }
]
```

```
}
]
```

### Specifying a Script Using Prebootstrap Attributes

The following sample JSON shows the `prebootstrap` attribute of the `userdata` attribute, with the script specified in a URL.

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "pre-bootstrap":
          {
            "scriptURL": "http://location_of_script",
            "failonerror": true
          }
        <Specify other userdata attributes here, if required.>
      }
      <Specify other attributes here, if required.>
    }
  }
]
```

The following sample JSON shows the `prebootstrap` attribute of the `userdata` attribute, with the script specified inline.

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "pre-bootstrap":
          {
            "failonerror": true,
            "script": [
              "line1_ofscript",
              "line2_ofscript",
              ...,
              "lineN_ofscript"
            ]
          }
        <Specify other userdata attributes here, if required.>
      }
      <Specify other attributes here, if required.>
    }
  }
]
```

A description of the `prebootstrap` attributes is provided in the following table. Nested attributes are indented in the Attributes column to indicate their hierarchy.



| Attribute                  | Required or Optional | Description                                                                                                                                                                                                                                                                                                                                                                                       |
|----------------------------|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>pre-bootstrap</code> | Optional             | This attribute allows you to specify a script that must be run prior to any other instance initialization that is performed by the <code>opc-init</code> package. You can either enter the script here, or point to a URL. This attribute contains the following nested attributes:                                                                                                               |
| <code>script</code>        | Optional             | Enter the lines of the prebootstrap script, formatted as a JSON array with each line of the script represented as one element of the array. The metadata service presents this array to the instance as text, with each array element separated by a line break.<br><br>Don't enter a script if you provide a script URL. You can specify either <code>script</code> or <code>scriptURL</code> .  |
| <code>scriptURL</code>     | Optional             | Enter the script location. This location must be accessible to the instance.<br><br>Don't provide a script URL if you provide a script. You can specify either <code>script</code> or <code>scriptURL</code> .                                                                                                                                                                                    |
| <code>failonerror</code>   | Optional             | Specifies whether the prebootstrap process should stop if the script encounters an error. To stop bootstrapping, specify <code>failonerror</code> as <code>true</code> . The default is <code>false</code> . If set to <code>false</code> , the bootstrapping operations continue and any errors encountered by the script are logged in the <code>/var/log/opc-compute/opc-init.log</code> file. |

### Enabling Windows Remote Management (WinRM)

You can allow users from other hosts to access the Windows instance by specifying those *trusted* hosts, as shown in the following example:

```
"instances":
[
  {
    <Specify other instance attributes.
    "attributes": {
      "userdata": {
        "winrm": {
          "trustedhosts": "appl.prod.example.com,*.dev.example.com,203.0.113.25"
        }
      }
    }
    <Specify other attributes here, if required.>
  }
]
```

If you want to allow all hosts, specify `"winrm": {}` or `"winrm": {"trusted hosts": "*"}`. This is also the default setting when `winrm` isn't specified.

## Using `opc-init` in a Private Machine Image

If you want to use `opc-init` in private Oracle Linux machine images, install `opc-init` while creating the image. You can install and use `opc-init` on Oracle Linux 6.7 images.

Here's an overview of the process for using `opc-init` in a private machine image:

1. On the Oracle Linux 6.7 VM that you use to create the image, do the following **before** you create the final image file:
  - Download the `opc-init` package from <http://www.oracle.com/technetwork/topics/cloud/downloads/opc-init-3096035.html>.
  - Extract the files in the package and use `yum` to install `opc-init`.
  - Add the `opc` user. The `opc-init` script copies the SSH keys provided during instance creation to the `/home/opc/authorized_keys` file of the `opc` user, enabling SSH access to instances as the `opc` user.
  - For instance initialization that you want to perform on all instances that use this machine image, add the required scripts and user data attributes in the machine image. See [User Data Attributes](#).

 **Note:**

For instance-specific initialization — that is, initialization tasks that must be performed only on a specific instance, not on all instances created using this machine image — you can provide the user data attributes later, while creating the instance, not in the machine image.

- Add the line `/usr/bin/opc-linux-init` to the `/etc/rc.local` file.

For detailed instructions on building an Oracle Linux machine image, see the tutorial [Building a Custom Oracle Linux Machine Image with the LAMP Stack](#).

2. After you've created your machine image file, upload it to Oracle Cloud Infrastructure Object Storage Classic. See [Uploading Image Files to Oracle Cloud Infrastructure Object Storage Classic](#).
3. Register your machine image in Compute Classic. See [Registering a Machine Image in Compute Classic](#).
4. Use your machine image to create instances. See [Creating an Instance Using a Private Image](#).
5. While creating instances, provide instance-specific configuration using user data attributes that are understood by `opc-init`. See [User Data Attributes](#).
6. If you added the `opc` user while creating the machine image, you can log in to your instance as the `opc` user using SSH. See [Accessing an Oracle Linux Instance Using SSH](#).
7. See the `opc-init` log to view error messages, if any. The `opc-init` log is found at: `/var/log/opc-init/opc-init.log`

# Best Practices for Using Compute Classic

As you create and manage instances and the associated resources in Compute Classic, consider the following guidelines and recommendations to get the best out of the service in terms of cost, manageability, and performance.

## Topics

- [Managing Service Users and Roles](#)
- [Building Private Images](#)
- [Securing the Operating System on Your Instances](#)
- [Naming Objects](#)
- [Selecting Shapes](#)
- [Using Orchestrations to Automate Resource Provisioning](#)
- [Managing Storage](#)
- [Configuring Network Settings](#)
- [Ensuring Secure Access to Instances](#)
- [Configuring Third-Party Devices When Setting Up a VPN Connection](#)

## Managing Service Users and Roles

- Only users with the `Compute_Operations` role can perform write operations (that is, create, update, and delete resources) in Compute Classic. When you create users in Oracle Cloud Infrastructure Classic Console, assign the `Compute_Operations` role to only those users who'll be responsible for creating, updating, and deleting instances and the associated storage and networking resources.
- For business continuity, consider creating at least two users with the `Compute_Operations` role. These users must be IT system administrators in your organization.

## Building Private Images

- The operating system and software that you use to build private images must have the required licenses. You're responsible for purchasing the required licenses and ensuring support for any third-party operating systems and software that you run on Compute Classic instances.
- Plan the packages that you want to include in your images keeping in mind the workload that you want to deploy.
- Before creating the image file, plan ahead and provision any users that you'd like to be available when instances are created using the image.

 **Note:**

While creating instances, you can specify one or more SSH public keys.

The keys that you specify are stored as metadata on the instance. This metadata can be accessed from within the instance at `http://192.0.0.192/{version}/meta-data/public-keys/{index}/openssh-key`.

- Oracle-provided images include a script that runs automatically when the instance starts, retrieves the keys, and adds them to the `authorized_keys` file of the `opc` user.
  - In images that you build, you can write and include a script that runs automatically when the instance starts, retrieves the SSH public keys, and adds the keys to the `authorized_keys` file of the appropriate users.
- Before creating the final image file, apply the necessary security patches and review the security configuration.
  - Keep your image disk size just as small as is essential. A large image requires more time to be uploaded to Oracle Cloud Infrastructure Object Storage Classic, and costs more to store. In addition, creating instances and bootable storage volumes from a large image requires more time. Before uploading image files to Oracle Cloud Infrastructure Object Storage Classic, make them *sparse* files. On Linux, you can convert a file to the sparse format by running the command, `cp --sparse=always original_file sparse_file`. And when creating the `tar` archive, to ensure that the `tar` utility stores the sparse file appropriately, specify the `-S` option.
  - Choose a `tar.gz` file name that you can use later to easily identify the key characteristics of the image, such as the OS name, OS version, and the disk size. For example, for a root-disabled, Oracle Linux 6.6 image with a 20-GB disk, consider using a file name such as `OL66_20GB_RD.tar.gz`.

### Securing the Operating System on Your Instances

To ensure that Compute Classic instances provide a resilient platform for your workloads, make sure that the latest security patches are applied to the operating system running on the instances. In addition, before deploying applications on an instance, review the security configuration of the operating system and verify that it complies with your security policies and standards.

- For private images (that is, images that you create and use), apply the necessary security patches and review the security configuration *before* creating the image file.
- For Oracle-provided images, apply the necessary security patches and review the security configuration right after you create the instances, *before* deploying any applications.

For security and patching-related guidelines, see the documentation for your operating system. The following are a few useful references:

- Oracle Linux

- [Installing and Using the Yum Security Plugin](#) in *Oracle Linux Administrator's Solutions Guide for Release 6*
- [Oracle Linux Security Guide for Release 6](#)
- Oracle Solaris
  - [Administering CVE Updates in Oracle Solaris](#) in *Oracle Solaris 11.3 Security Compliance Guide*
  - [Oracle Solaris 11 Security and Hardening Guidelines](#)
- Microsoft Windows
  - [Windows Server 2008 R2 Security Baseline](#)
  - [Windows Server 2012 Security Baseline](#)

### Naming Objects

When you create instances, storage volumes, security lists, and so on, select the name of the object carefully. Pick a name that helps you quickly identify the key characteristics of the object later. For example, when creating a bootable storage volume, consider including the operating system name and the image disk size in the name of the storage volume.

### Selecting Shapes

- While selecting the shape for an instance, consider the nature of the applications that you plan to deploy on the instance, the number of users that you expect to use the applications, and also how you expect the load to scale in the future. Remember to also factor in the CPU and memory resources that are necessary for the operating system.
- Select a shape that meets the requirements of your workload with a sufficient buffer for intermittent spikes in the load. If you're not sure what shape is appropriate for an instance, then start small, experiment with a representative workload, and then settle on a shape. This approach may help you achieve an optimal trade-off between resource allocation and performance.

### Using Orchestrations to Automate Resource Provisioning

- When building orchestrations to create and manage instances, set the high-availability policy to `active`, to ensure minimal disruption to your operations.
- Using orchestrations, you can control the placement of instances. You can opt to have instances placed on the same or on different physical nodes. When you use the instance placement feature, consider your requirements for application isolation and affinity. See [Relationships Between Objects Within a Launch Plan Object](#).
- If you want to shut down an instance but let other instances in the same orchestration run, then don't terminate the orchestration but update the instance and specify the `desired state as shutdown` for just the required instance.
- Don't define storage volumes and instances in the same orchestration. By keeping storage volumes and instances in separate orchestrations, you can shut down and start the instances when required and yet preserve the attached storage volumes. Note that the recommendation here is to define the storage *volumes* outside the instance orchestration. To ensure that the storage volumes remain attached after an instance is re-created, you must define the storage *attachments* within the instance orchestration.

- When you create an instance using the Create Instance wizard, a single orchestration v2 is created automatically to manage the instance and its associated resources. Storage volumes and networking objects used by the instance are created in the same orchestration. Instances are nonpersistent by default. However, storage volumes and other objects are created with persistence set to true, so that if you suspend the orchestration, instances are shut down, but storage volumes aren't deleted. Terminating the orchestration, however, will cause all objects to be deleted and any data on storage volumes will be lost.
- Earlier, when you created an instance using the Create Instance wizard, one or more orchestrations v1 were created automatically to manage the instance and its associated resources. For example, if you used the Create Instance wizard to create an instance and attach a new storage volume to it, then two separate orchestrations were created, one for the instance and the other for the storage volume. A master orchestration was also created, and the instance and storage volume orchestrations were referenced as objects in the master orchestration.

Starting or terminating a master orchestration allows you to start or terminate all the nested orchestrations. This is an easy way to handle dependencies across orchestrations. Remember, though, that if your master orchestration references any orchestration that creates storage volumes, then terminating the master orchestration will delete the storage volumes and all the data on them.

If you want to delete an instance but retain the storage volumes that were created while creating the instance, then terminate only the instance orchestration and let the storage volume orchestration remain in the **Ready** state.

### Managing Storage

- When you decide the number and size of your storage volumes, consider the limits: minimum 1 GB, maximum 2 TB, one-GB increments, and 10 volumes per instance.
  - If you attach too many small storage volumes to an instance, then you may not be able to scale block storage for the instance up to the full limit of 20 TB.
  - If you attach many large volumes to an instance, then the opportunities to spread and isolate storage are limited. In addition, too many large volumes may result in lower overall utilization of block storage space, particularly if data isolation is also critical for your business.

You can increase the size of a storage volume after creating it, even if the storage volume is attached to an instance. See [Increasing the Size of a Storage Volume](#). However, you can't reduce the size of a storage volume after you've created it. So ensure that you don't overestimate your storage requirement.

Consider the storage capacity needs of the applications that you plan to deploy on the instance, and leave some room for attaching more storage volumes in the future. This approach helps you use the available block storage capacity efficiently in the long run.

- To provide highly scalable and shared storage in the cloud over NFSv4 for your instances, consider using Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution. This appliance is provisioned on a Compute Classic instance and plays the role of a file server in the cloud. It provides shared, highly scalable, low-cost, and reliable storage capacity in Oracle Cloud Infrastructure Object Storage Classic for your Compute Classic instances running Oracle Linux. For information about the use cases that the appliance is best suited for, see About

Oracle Cloud Infrastructure Storage Software Appliance— Cloud Distribution in *Using Oracle Cloud Infrastructure Storage Software Appliance*.

- Create and use separate storage volumes for your applications, data, and the operating system. Use a configuration management framework such as Chef or Puppet for managing the configuration of the operating system and applications.
- To ensure that storage volumes remain attached *and* mounted after instances are deleted and re-created, do *both* of the following:
  - Define the storage attachments within the orchestration that you use to create instances. Note that the recommendation here is to define the storage *attachments*, and not the storage *volumes*, in the orchestration that you use to create instances.
  - Set up the instance to boot from a bootable storage volume.
- If you're sure that a storage volume is no longer required, then back up the data elsewhere and delete the storage volume.

### Configuring Network Settings

- When you create an instance, if you opt for an autogenerated public IP address, then the IP address so allocated persists only during the life of the instance. If the instance is deleted and re-created by terminating and starting its orchestration, then the instance gets a new public IP address. To assign a fixed public IP address to an instance, reserve a public IP address, and attach it to the instance—either when you create the instance or, later, by updating the IP reservation.
- If you've created an IP reservation and you no longer need it, delete it.
- If a persistent public IP address is associated with an instance during instance creation, then if required, you can remove that IP address from the instance later on. Ensure, however, that you don't delete this IP reservation. If you delete and re-create the instance, the IP reservation will be required again. If you've deleted the IP reservation, you won't be able to re-create the instance.
- You can attach an instance to a maximum of five security lists, and you can use a security list as the source or destination in up to 10 security rules. Plan your security lists and security rules keeping these overall limits in mind.

#### Note:

If an instance is added to multiple security lists that have different policies, then the most restrictive policy is applicable to the instance.

- Plan your IP network keeping the following overall limits in mind.
  - As a best practice, add a maximum of 20 IP networks to an IP network exchange. Due to DHCP limitations routing is automatically configured for only for 20 IP networks in an IP network exchange. If you want to add more than 20 IP networks to an IP network network, you'll need to manage routing in each instance manually.
  - The prefix length of the IP address prefix that you specify in an IP network should be between /16 to /30.
  - The maximum number of IP address prefixes that you can specify in an IP address prefix set is limited to 2047.

- In a security rule, you can specify a maximum of 32 security protocols, 32 source IP address prefix sets, and 32 destination IP address prefix sets.
- In a security protocol, you can specify a maximum of 32 port numbers or port range strings for **Source Port Set** and **Destination Port Set**.
- In a vNICset, you can specify a maximum of 32000 vNICs and 256 access control lists (ACLs).

### Ensuring Secure Access to Instances

- Ensure instance isolation by creating security lists and adding instances to the appropriate security lists. Instances within a security list can inter-communicate freely over any protocol. To allow incoming traffic to all the instances in a security list, set up a security rule with the security list as the destination and with the required source and protocol settings.
- Use security rules carefully and open only a minimal and essential set of ports. Keep in mind your business needs and the IT security policies of your organization.
- When you add an instance to a security list, all the security rules that use that security list—as either the source or destination—are applicable to the instance. Consider a security list that is the destination in two security rules, one rule that allows SSH access from the public Internet and another rule permitting HTTPS traffic from the public Internet. When you add an instance to this security list, the instance is accessible from the public Internet over both SSH *and* HTTPS. Keep this in mind when you decide the security lists that you want to add an instance to.
- If you're creating a Linux or Oracle Solaris instance, then try to determine, up front, how many users you expect to access the instance and plan for a separate SSH key pair for each user.
- Using the Web Console, you can associate a maximum of 10 SSH keys with your instance.
- Keep your SSH keys secure. Lay down policies to ensure that the keys aren't lost or compromised when employees leave the organization or move to other departments. If you lose your private key, then you can't access your instances. For business continuity, ensure that the SSH keys of at least two IT system administrators are added to your instances.
- If you need to edit the `~/.ssh/authorized_keys` file of a user on your instance, then before you make any changes to the file, start a second `ssh` session and ensure that it remains connected while you edit the `authorized_keys` file. This second `ssh` session serves as a backup. If the `authorized_keys` file gets corrupted or you inadvertently make changes that result in your getting locked out of the instance, then you can use the backup `ssh` session to fix or revert the changes. Before closing the backup `ssh` session, test the changes you made in the `authorized_keys` file by logging in with the new or updated SSH key.

### Configuring Third-Party Devices When Setting Up a VPN Connection

It is recommended that you consider the following suggestions while configuring your third-party device for a VPN connection.

- **Configuration Information**  
Use the following IPsec configuration for policy-based VPN:
  - Authentication: Pre-shared keys



- Encryption: 3DES, AES-128, AES-192, AES-256
- Hash: MD5, SHA-1, SHA-2
- Policy Group: Diffie-Hellman groups supported are 2, 5, 14, 15, 16, 17, 18, 22, 23, 24
- ISAKMP: IKEv1 only. If IKEv2 is enabled by default, turn it off.
- Exchange type: Main Mode (The Cloud gateway uses **main mode** in phase one negotiations)
- IPsec protocol: ESP, tunnel-mode
- PFS: Enabled
- IPsec SA session key lifetime default: 28,800 seconds (8 hours); 3,600 seconds (1 hour) on Cisco devices
- IKE session key lifetime default: 3,600 seconds (1 hour); 86400 seconds (24 hours) on Cisco devices
- **General and Debug Information**
  - It is highly recommended that the third-party device be configured to be responder-only.
  - The third-party device must support and be configured for policy-based VPN.
  - The Cloud gateway uses IPsec and is behind a NAT, so network address translation traversal (NAT-T) is required. The third-party device must support NAT-T. NAT-T requires UDP port 4500 to be open.
  - Avoid setting up numerous IP networks with a /32 subnet. Instead, use a smaller number of IP networks with larger subnets. If you create a very large number of IP networks, a large number of IPsec security associations are required, which could cause performance degradation on some third-party devices.
  - Ensure the IKE and IPsec timeouts on the Cloud gateway and the third-party device are the same.
  - For Phase 1, ensure that the IKE ID on the Cloud gateway and the third-party device match.
  - Check each security application on the third-party device to ensure that idle timeouts and traffic volume limits are reasonable.
  - After a VPN connection is set up, if you can connect to instances on some subnets but not on others, check that both gateways have the correct set of subnets configured. If the third-party device has some subnets configured that aren't on the Cloud gateway, the Cloud gateway won't report an error. However, if the Cloud gateway has some subnets configured that aren't on the third-party device, it might result in a flapping tunnel.
- **HA Information**
  - When HA is configured, Dead Peer Detection (DPD) must be enabled to detect when a tunnel is down.
  - When HA is configured, asymmetric routing across the tunnels that make up the VPN connection will occur. Ensure that your firewall is configured to support this. If not, traffic will not be routed reliably.
  - Switching tunnels might take 30–40 seconds.

# Frequently Asked Questions for Compute Classic

This section provides answers to frequently asked questions about Compute Classic.

## Topics

- [Machine Images](#)
- [Interfaces](#)
- [Instance Properties](#)
- [Instance Usage](#)
- [Windows Instances](#)
- [Shared Network Settings](#)
- [Storage Management](#)
- [Orchestrations v1](#)
- [Using SSH Keys](#)
- [Connecting to Instances](#)
- [Support](#)

## Machine Images

### **What base images can I use to create instances?**

You can use Oracle-provided or your own images to create instances. See [Managing Machine Images](#).

### **After creating instances using Oracle-provided images, if I update the operating system and kernel with additional packages, will the updated operating system and kernel continue to be supported?**

The operating system and kernel will continue to be supported as long as they are updated using Oracle public or support repositories.

### **Can I select an image from Oracle Cloud Marketplace and use it to create an instance?**

Yes, you can go to Oracle Cloud Marketplace and select the image you want to use. Click **Get App** and accept the terms of use. Then follow the instructions to launch the Create Instance wizard in the Compute Classic web console.

Alternatively, you can launch the Create Instance wizard in the web console and then select the required machine image from the Marketplace tab. See [Creating an Instance from the Instances Page](#).

## Interfaces

### What user interfaces does this service provide?

You can access Compute Classic through the web console, or by using the REST API, or the command line interface (CLI). See [Accessing Compute Classic Using the Web Console](#), [Accessing Compute Classic Using REST API](#), and [Accessing Compute Classic Using the Command Line Interface](#).

### Why does the web console time out frequently?

For security, the web console times out automatically after 15 minutes of inactivity. Log in again to continue using the web console.

### How do I connect to the service using the API?

See Quick Start in *REST API for Oracle Cloud Infrastructure Compute Classic*.

## Instance Properties

### How much CPU and memory can I assign to an instance?

The number of CPUs and RAM allocated to an instance are determined by the shape that you select while creating the instance. See [About Shapes](#).

### What's the maximum amount of memory that I can allocate across all my instances?

The memory allocated to each instance is determined by the shape that you select while creating the instance. So the maximum amount of memory that you can use across all your instances is the total amount of RAM associated with the shape that you select for each of your instances. There's no separate upper limit on memory allocation. For the amount of RAM associated with each shape, see [About Shapes](#).

### How do I provide persistent storage for my instances?

You can provide block storage space by creating storage volumes and attaching them to the instances. See [Managing Storage Volumes](#).

### Why do some of my instances have three-part names (Compute-identity\_domainuserid), while others have four-part names (Compute-identity\_domainuserlnamelid)?

The *id* in the instance name is generated automatically when the instance is created. If you specify a *name* (an optional parameter) while creating the instance, then the *name* that you specify precedes the *id* in the four-part name.

## Instance Usage

### What can I install on the Compute Classic instances?

You can deploy any application—Oracle or third-party—that's supported on the operating system included in the machine image that you used to create the instance, subject to the licensing and support terms of the vendor of that application. Oracle

doesn't provide support or indemnification for any third-party applications and software.

### How can I stop an instance?

If you no longer need the instance, you can stop it or delete it. You can stop an instance only if it boots from a persistent bootable storage volume. When you stop an instance, you can start it again later. If you delete an instance by stopping the instance orchestration, you can re-create it later on, if required. See [Managing Instance Lifecycle Operations](#).

### How can I delete an instance? The web console doesn't have a Delete option.

After mid-May 2016, when you create an instance by using the web console, an orchestration is automatically created to manage the instance and its associated resources. To delete an instance, suspend or stop the relevant orchestration. Alternatively, if your instance uses a persistent bootable storage volume, you can select the **Stop** option on the Instances page to stop the instance without deleting it. See [Managing Instance Lifecycle Operations](#).

#### See Also:

- [Terminating an Orchestration v1](#)
- [Suspending an Orchestration v2](#)
- [Terminating an Orchestration v2](#)

### How can I access the boot log for my instance?

For instances that use Oracle-provided images and are created after mid-May 2016, you can view the boot log by using the web console. See [Viewing the Boot Log of an Instance](#).

For any other instances, you might not be able to view the boot log by default. The Logs page displays a message stating that no log was found. In these cases, to view the boot log, you can log in to your instance and direct the console output to the serial port.

To view the boot log on an Oracle Linux instance:

1. Access your instance using SSH. See [Accessing an Oracle Linux Instance Using SSH](#).
2. Run the following commands:

```
sudo su
vi /etc/grub.conf
```

In the `grub.conf` file, for each kernel description, add the entry `console=ttyS0` and save the file.

3. Reboot the instance. See [Rebooting an Instance](#).

After your instance restarts, you can view the boot log by using the web console. See [Viewing the Boot Log of an Instance](#).

### Can I use an instance as a template to create another identical instance?

Yes, you can do that by creating an instance snapshot. Here's how you would do it:

1. Create an instance that uses a **nonpersistent** boot disk.

#### Note:

If you use the web console to create the instance, remember to explicitly **remove** the persistent boot disk option that is selected by default. You can't create an instance snapshot of an instance that uses a persistent boot disk.

2. Customize your instance as required, by adding users, or installing and configuring applications.
3. Create an instance snapshot.
4. Register the image created by the instance snapshot.
5. Use the image to create new instances.

For more detailed information, see [Cloning an Instance by Using Instance Snapshots](#).

## Windows Instances

For information about Windows licensing, see the section Windows Licensing Information in the Compute Classic [FAQ](#).

## Shared Network Settings

### Are the public IP addresses of instances fixed or dynamic?

While creating instances, you can choose whether the public IP address must be fixed or assigned dynamically from a pool.

#### See Also:

- [Reserving a Public IP Address](#)
- [Creating an Instance from the Instances Page](#)
- [Orchestration v1 Attributes for instances](#)
- [Orchestration v2 Attributes for Instance](#)
- [Updating an IP Reservation](#)

### Are the private IP addresses of instances fixed or dynamic?

Private IP addresses in the shared network are assigned dynamically to each instance from a pool of Oracle-provided IP addresses. If an instance is restarted, its private IP address might change.

If you create an instance using an Oracle-provided machine image with release version 16.3.6 or later, you can specify up to eight network interfaces for each instance. While creating an instance, you can add your instance to one or more IP networks that you create, as well as to the shared network. While the IP address allocated by the shared network continues to be assigned dynamically, you can specify whether you want the IP address on the IP network to be fixed or dynamic.

#### **How can I find out the IP address of my instance?**

The Instances page displays both the public and the private IP address of the instance. You can also see these and other details of an instance on the instance details page.

#### **How can I restrict and isolate traffic between my instances?**

Add the instances that should be able to communicate with each other to the same security lists. To isolate instances from other instances, add them to different security lists. By default, instances in different security lists can't communicate with each other. You can use security rules to override the default policies of security lists. See [Configuring the Shared Network](#).

## Storage Management

#### **How can I add block storage to my instance after I've created the instance?**

If you've already created the storage volume that you want to attach to a running instance, see [Attaching a Storage Volume to an Instance](#). If you want to create a storage volume and then attach it to an instance, see [Creating a Storage Volume](#).

#### **How many storage volumes can I attach to an instance?**

You can attach up to 10 block storage volumes to an instance.

#### **What is the allowed size for a storage volume?**

The allowed range is from 1 GB to 2 TB, in increments of 1 GB. You can specify the size of a storage volume when you create the volume.

#### **What if my instances need access to more storage?**

If your instance needs to access more storage than you can attach to a single instance, you can consider using Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution to provide access to infinitely scalable, shared file storage capacity. See *About Oracle Cloud Infrastructure Storage Software Appliance - Cloud Distribution* in *Using Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution*.

#### **I've already attached a storage volume to an instance. Can I resize the storage volume?**

Yes, you can resize a storage volume after creating it, even if it is attached to an instance. Note, however, that you can only increase the size of a storage volume; you can't reduce it. See [Increasing the Size of a Storage Volume](#).

### Can I attach a storage volume to multiple instances at the same time? If not, how can I implement shared storage?

No, you can't attach a storage volume to multiple instances at the same time.

To provide highly scalable and shared storage in the cloud over NFSv4 for your instances, consider using Oracle Cloud Infrastructure Storage Software Appliance – Cloud Distribution. This appliance is provisioned on a Compute Classic instance and plays the role of a file server in the cloud. It provides shared, highly scalable, low-cost, and reliable storage capacity in Oracle Cloud Infrastructure Object Storage Classic for your Compute Classic instances running Oracle Linux. For information about the use cases that the appliance is best suited for, see *About Oracle Cloud Infrastructure Storage Software Appliance– Cloud Distribution* in *Using Oracle Cloud Infrastructure Storage Software Appliance*.

## Orchestrations v1

### What kinds of resources can I create using an orchestration?

You can use orchestrations to create instances, storage volumes, or networking objects such as security rules or security lists.



#### See Also:

- [Object Types in an Orchestration](#)
- [Attributes in Orchestrations v1](#)

### I added an orchestration and started it, but nothing seems to be happening.

When you start an orchestration, its status changes to **Starting**. Depending on the number and type of objects defined in the orchestration, it can take quite a while for all the objects to be created. While the objects are being created, the orchestration continues to show the status **Starting**. After all the objects are created, the status of the orchestration changes to **Ready**. If any of the objects can't be created, then the state of the orchestration changes to **Error**. If the status of your orchestration doesn't change to **Error**, then the objects are being created. Wait till the status changes to either **Ready** or **Error**.

### Can I update an orchestration v1?

To update an orchestration, you can download and modify it and then upload it. Note that you can't upload the modified orchestration with the same name. Either give the modified orchestration a new name, or delete the existing orchestration in Compute Classic and then upload the modified orchestration. Alternatively, if your orchestration is in the `stopped` state, you can update the orchestration using the web console.

 **See Also:**

- [Updating an Orchestration v1](#)
- [Deleting an Orchestration v1](#)
- [Uploading an Orchestration v1](#)

**I defined the attributes in my orchestration in a certain order. But when I view the orchestration in the web console or download it, the attributes are in a different order. Why?**

When you build an orchestration file (in JSON format), you can arrange the attributes in any order as long as the attribute hierarchy is as described in [Attributes in Orchestrations v1](#). After you upload the orchestration to Compute Classic, the attributes are stored in a different order. For example, you may have defined the `ha_policy` attribute at the beginning of the orchestration, but when you view or download the orchestration, the `ha_policy` attribute is at the very end of the orchestration. These changes don't affect the orchestration and the attributes defined in it.

**I created five instances by using an orchestration. Now I want to delete three instances but keep the other two. How can I do that?**

You can use the REST API, CLI, or the web console to stop instances that you don't need. When you stop an instance, you can start it again later, if required. See [Managing Instance Lifecycle Operations](#).

**How are orchestrations different from launch plans?**

| Capability                                            | Launch Plans                                                                         | Orchestrations                                                                                                                                                                            |
|-------------------------------------------------------|--------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lets you create multiple instances?                   | Yes.                                                                                 | Yes.                                                                                                                                                                                      |
| Lets you specify the HA policy for an instance?       | No.<br>Instances don't persist. If you delete an instance, you must create it again. | Yes.<br>The HA policy can be specified for each instance.<br>See <a href="#">About High-Availability Policies in an Orchestration</a> .                                                   |
| Lets you create other object types?                   | No.                                                                                  | Yes.<br>See <a href="#">Object Types in an Orchestration</a> .                                                                                                                            |
| Lets you stop and re-create multiple objects at once? | No.<br>You can use the API or web console to manage individual instances.            | Yes.<br>You can stop or start an orchestration to create or destroy all the resources specified in the orchestration.<br>See <a href="#">Managing Resources Using Orchestrations v1</a> . |



| Capability                    | Launch Plans | Orchestrations                                                                                                                                        |
|-------------------------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| Is stored on Compute Classic? | No.          | Yes.<br>You can view, monitor, download, start, stop, or delete an orchestration.<br>See <a href="#">Managing Resources Using Orchestrations v1</a> . |

## Using SSH Keys

### Can I associate multiple SSH public keys with an instance?

Yes, you can associate multiple SSH public keys with your instance when you create the instance. To do this, you must upload all the required SSH public keys to Compute Classic before you start creating the instance.

Additionally, after creating a Linux or Oracle Solaris instance, you can inject more SSH public keys into the instance by logging in to the instance and editing the `~/.ssh/authorized_keys` file of the user.

If you need to edit the `~/.ssh/authorized_keys` file of a user on your instance, then before you make any changes to the file, start a second `ssh` session and ensure that it remains connected while you edit the `authorized_keys` file. This second `ssh` session serves as a backup. If the `authorized_keys` file gets corrupted or you inadvertently make changes that result in your getting locked out of the instance, then you can use the backup `ssh` session to fix or revert the changes. Before closing the backup `ssh` session, test the changes you made in the `authorized_keys` file by logging in with the new or updated SSH key.

#### Note:

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any SSH public keys that you added or edited manually (that is, not during instance creation) must be added or edited again. To do this, you must log in to the instance by using the original SSH private key. So retain and safeguard your original SSH private key.

### Can I associate a single SSH public key with more than one instance?

Yes, you can associate an SSH public key with multiple instances.

### I've lost access to my SSH private key. What do I do now?

A private SSH key is the only way you can access your Linux instances. If you don't have the private key, then you can't access your instances. Always back up an encrypted copy of your private SSH keys, and keep the keys secure.

**My SSH private key has been compromised. I've generated a new SSH key pair and I want to update the SSH public key on my running instances. How can I do that?**

To modify an SSH public key on a running instance, log in to the instance, and edit the `~/.ssh/authorized_keys` file of the user. Remove the existing SSH public key in this file and replace it with the new key.

 **Note:**

You don't need to do this if you're creating a Windows instance, because you can't log in to a Windows instance using SSH. To log in to your Windows instance using RDP, see [Accessing a Windows Instance Using RDP](#).

If you need to edit the `~/.ssh/authorized_keys` file of a user on your instance, then before you make any changes to the file, start a second `ssh` session and ensure that it remains connected while you edit the `authorized_keys` file. This second `ssh` session serves as a backup. If the `authorized_keys` file gets corrupted or you inadvertently make changes that result in your getting locked out of the instance, then you can use the backup `ssh` session to fix or revert the changes. Before closing the backup `ssh` session, test the changes you made in the `authorized_keys` file by logging in with the new or updated SSH key.

 **Note:**

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any SSH public keys that you added or edited manually (that is, not during instance creation) must be added or edited again. To do this, you must log in to the instance by using the original SSH private key. So retain and safeguard your original SSH private key.

**I want to give other users access to my instance, but I don't want to share my SSH private key. What should I do?**

You can create new local users on your instance, generate SSH key pairs for these users offline, and append the new public keys in the `~/.ssh/authorized_keys` file of the new users. These users can then `ssh` to the instance by using the appropriate private keys. See [Adding Users on an Oracle Linux Instance](#).

 **Note:**

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any users that were added manually (that is, users that weren't defined in the machine image) must be added again.

When an instance that's set up to boot from a nonpersistent boot disk is deleted and re-created, any SSH public keys that you added or edited manually (that is, not during instance creation) must be added or edited again. To do this, you must log in to the instance by using the original SSH private key. So retain and safeguard your original SSH private key.

## Connecting to Instances

### How can I connect (log in) to an instance?

- For instances created using Oracle Linux images, see [Accessing an Oracle Linux Instance Using SSH](#)
- For instances created using Oracle Solaris images, see [Accessing an Oracle Solaris Instance Using SSH](#)
- For Windows instances, see [Accessing a Windows Instance Using RDP](#)
- For images from Oracle Cloud Marketplace, see the login instructions provided by the image provider.

### How can I log in to an Oracle Linux instance as a non-opc user?

See [Adding Users on an Oracle Linux Instance](#).

## Support

### To what extent will Oracle support the applications and services deployed on Compute Classic instances?

- Support for Oracle applications that you deploy on Compute Classic instances will be provided according to the prevailing support policies for those applications.
- Oracle won't provide support for any third-party or open-source applications deployed on Compute Classic instances.

# Troubleshooting Compute Classic

This section describes common problems that you might encounter when using Compute Classic and explains how to solve them.

## Topics

- [Web Console Problems](#)
- [Networking Problems](#)
- [SSH Key Problems](#)
- [Storage Volume Problems](#)
- [Orchestration Problems](#)
- [opc-init Problems](#)
- [Launch Plan Problems](#)

## Web Console Problems

This section lists problems that you might encounter while using the Compute Classic web console.

### Can't access the web console

#### Description

When I try to log in to the web console, the following error message is displayed:

```
You are not authorized to access the Compute Classic (0706_043942.887).\
If the problem persists, contact Oracle Support.
```

#### Solution

This error indicates that you are not assigned any Compute Classic role. See [About Compute Classic Roles](#).

Ask your service administrator to assign the appropriate roles to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

### Can't create, update, or delete objects

#### Description

When I try to create, update, or delete any object, an error message similar to the following is displayed:

```
Unable to create security rule. [jack.jones@example.com_0706_045758.332] :
User /Compute-acme/jack.jones@example.com is not permitted to perform "secrule.add"
```

```
on
secrule:/Compute-acme/jack.jones@example.com/mysecrule

Unable to add IP reservation. [jack.jones@example.com_0706_050517.177] :
  User /Compute-acme/jack.jones@example.com is not permitted to perform
  "ipreservation.add" on
  ipreservation:/Compute-acme/jack.jones@example.com/myipres

Unable to update SSH key "mykey". [jack.jones@example.com_0706_052418.475] :
  User /Compute-acme/jack.jones@example.com is not permitted to perform
  "sshkey.update" on
  sshkey:/Compute-acme/jack.jones@example.com/mykey

Unable to delete SSH key "mykey". [jack.jones@example.com_0706_052437.025] -
  User /Compute-acme/jack.jones@example.com is not permitted to perform
  "sshkey.delete" on
  sshkey:/Compute-acme/jack.jones@example.com/mykey

Unable to detach storage volume "myvoll" from this instance.
[jack.jones@example.com_0706_052512.984] -
  User /Compute-acme/jack.jones@example.com is not permitted to perform
  "attachment.delete" on
  storage/attachment:/Compute-acme/jack.jones@example.com/vm-1/3b515fae.../...
  55a31eae6b5
```

### Solution

This error indicates that you're not authorized to create, update, or delete resources in Compute Classic. Ask your service administrator to assign the `Compute_Operations` role to you in Oracle Cloud Infrastructure Classic Console. See [Modifying User Roles in \*Managing and Monitoring Oracle Cloud\*](#).

## Can't upload an orchestration

### Description

When I try to upload my orchestration file, I get the following error: "Unable to create an orchestration from the JSON file."

### Solution

This error indicates that there are errors in the syntax of your orchestration JSON file. Open the JSON file in a text editor to identify and fix the problems. You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice.



#### Note:

Oracle doesn't support or endorse any third-party JSON-validation tool.

## My orchestration hasn't created any instances

### Description

I've uploaded my orchestration file but I don't see my instances. What should I do?

## Solution

After uploading your orchestration, the status of your orchestration is automatically set to `Stopped`. To create the resources defined in your orchestration, start your orchestration. If you have created orchestration v2, see [Starting an Orchestration v2](#). If you have created orchestration v1, see [Starting an Orchestration v1](#).

# Error while starting an orchestration: Specify imagelist or bootorder

## Description

I've uploaded my orchestration, but when I start it, the following error occurs:

Specify either an `ImageList` or `boot_order` and `StorageVolume`.

## Solution

This error indicates that your orchestration doesn't specify either an image or a bootable storage volume for your instance.

- To set up the instance to boot from a persistent disk, you must attach a bootable storage volume by using the `storage_attachment` instance attribute, and then specify the index number of the attached storage volume as the boot disk by using the `boot_order` instance attribute.

```
{
  "objects": [
    {
      "instances": [
        {
          "boot_order": [
            1
          ],
          "storage_attachments": [
            {
              "index": 1,
              "volume": "/Compute-acme/joe/bootable-vol1"
            }
          ]
        }
      ]
    }
  ]
}
```

- To set up the instance to boot from a nonpersistent disk, specify the image that you want to use by using the `imagelist` attribute.

```
{
  "objects": [
    {
      "instances": [
        {
          "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64"
        }
      ]
    }
  ]
}
```

 **Note:**

If you specify both `boot_order` and `imagelist` for an instance in an orchestration, the `imagelist` attribute is ignored and the instance is booted using the bootable storage volume specified by the `boot_order` attribute. See [Orchestration v1 Attributes for instances](#).

## Can't attach a storage volume to an instance

### Description

When I try to attach my storage volume to an instance, the following error occurs:

```
APIConflictError: Attachment index 1 is already in use on instance /Compute-acmecorp/acmeadmin/dev2/6073c806-f7da-47eb-9678-6e618931b29a
```

### Solution

The index number that you're trying to assign to this storage volume is already used for another storage volume. Select a different index number and try again.

 **Note:**

The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on.

To view the set of index numbers that are already in use for an instance using CLI, use the `opc compute storage-attachment list` command.

For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

## Can't detach a storage volume from an instance

### Description

I've attached three storage volumes to my instance. Now I want to delete the instance. So I started to detach the storage volumes. I detached two of the storage volumes, but can't detach the third one.

### Solution

You can detach storage volumes that were attached to an instance *after* the instance was created. You can't detach storage volumes that were attached *during* instance creation.

## Can't delete a storage volume

### Description

I want to delete a storage volume that I no longer need, but the web console doesn't show the delete option for the storage volume.

### Solution

You can't delete a storage volume if it's attached to an instance. To find out whether a storage volume is attached to an instance, view the storage volume information in the web console. Click the **Storage** tab, scroll down to the storage volume that you want to delete, and check the displayed details. If the storage volume that you want to delete is attached to an instance, then you must detach it first. See [Detaching a Storage Volume from an Instance](#).

You can also use CLI commands to find out if a storage volumes is attached to an instance. Use the `opc compute storage-attachment list` CLI command with the `--storage_volume_name` option to view the details of storage attachments for a specified storage volume. Then use the `opc compute storage-attachment delete` CLI command to detach a storage volume from an instance.

For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

Also, you can't delete a storage volume if you've created any colocated snapshots of the storage volume. See [Backing Up and Restoring Storage Volumes Using Snapshots](#).

## Can't delete a storage volume snapshot

### Description

I want to delete a storage snapshot that I no longer need, but the **Delete** option for that snapshot is disabled in the web console.

### Solution

You can't delete a colocated storage volume snapshot if you've created a clone from that snapshot. The Storage Snapshots page shows you information about each colocated storage snapshot including storage volumes cloned from a snapshot, if any. You can make a note of the clones and then view them on the Storage Volumes page. To delete a colocated snapshot, first delete all the clones created from the storage snapshot. See [Deleting a Storage Volume](#).

## Can't remove an IP address from an instance

### Description

I associated a temporary IP address with my instance while creating the instance using the Create Instance wizard. Now I want to remove the temporary IP address and use an IP address reservation instead. How can I remove the temporary IP address from my instance? The **Remove Instance** option in the web console is disabled.



### Solution

You can't remove a temporary IP address from an instance. You can only remove a persistent IP address. If you created an instance with an autogenerated IP address or if you changed the status of the IP address associated with an instance to temporary, then to remove that IP address from the instance, first update it to change its status to permanent. See [Updating an IP Reservation](#).

## Can't delete a security application

### Description


When I tried to delete the security application `/oracle/public/snmp-trap-udp`, the following error message was displayed:

```
APIUnauthorizedError: User /Compute-acmecorp/acmeadmin is not permitted to perform "secapplication.delete" on secapplication: /oracle/public/snmp-trap-udp
```

### Solution

Compute Classic has a set of predefined security applications. The names of these security applications start with `/oracle/public` container. You can't delete these predefined security applications.

#### Tip:

To view a list of predefined security applications from the web console, click the **Network** tab and then the **Security Applications** tab in the left pane. The list of available security applications is displayed. In the search field, enter `/oracle/public`, and click . A list of all the predefined security applications is displayed.

To get a list of all the predefined security applications from the CLI, use the `opc compute sec-application list` command and specify `/oracle/public` as the container. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI](#) in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

## Can't delete a private image

### Description

How can I delete a private image listed on the Images page? The **Delete** option in the web console isn't available.

### Solution

You can't delete a private image if it is used in an orchestration. To delete a private image, ensure that you've stopped and deleted any instance orchestrations that used that private image.

# Networking Problems

This section lists problems that you might encounter while setting up security rules to implement firewalls for your instances.

## Can't connect to an instance using SSH

### Description

I've created an instance but can't connect to it using SSH.

### Solution

Check for each of the following possible causes:

1. Is your instance configured for SSH access?

All Oracle-provided Oracle Linux and Solaris instances are by default configured to allow SSH access. However, if you're creating an instance using a private image or an image from Oracle Cloud Marketplace, the instance might not be configured to allow SSH access. Check with the owner of the machine image.

Remember, also, that you can't access Windows instances using SSH. If you're trying to log in to a Windows instance, use RDP. See [Accessing a Windows Instance Using RDP](#).

2. Did you use the correct user?

- To log in to an instance that was created by using an Oracle-provided Oracle Linux machine image, use the `opc` user.

For instances created by using other machine images, find out which SSH-enabled users are defined in that machine image, and log in as one of those users.

- To log in to an instance as a user that was created after the instance was provisioned, you must generate an SSH key pair for the new user and copy the public key to the `~/.ssh/authorized_keys` file of the user. You must also add the new user to the list of allowed users in the `/etc/ssh/sshd_config` file on the instance. See [Adding Users on an Oracle Linux Instance](#).

3. Did you specify the correct public IP address of the instance?

To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).

If no public IP address is associated with the instance, reserve and associate a public IP address. See [Reserving a Public IP Address](#) and [Updating an IP Reservation](#).

4. Did you specify the correct private key?

The private key that you specify must correspond to one of the public keys associated with the instance.

5. Does the instance belong to a security list with the inbound policy set to `deny`?

An instance can be associated with multiple security lists. You can find out which security lists an instance is attached to by viewing the details of the instance. See [Monitoring Instances](#).

You can see the policies used by each security list by viewing the details of the security list from the web console.

If there's a conflict between the policies of the various security lists, then the most restrictive policy is applicable. This means that if even one of the security lists that your instance is attached to has the inbound policy set to `deny`, then your instance can't receive traffic.

If this is the case, then create a security rule to explicitly allow traffic to a security list that your instance is attached to.

6. Does the error message contain the following warning?

```
WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!
```

If yes, then see [RSA key fingerprint error while connecting to an instance](#).

## RSA key fingerprint error while connecting to an instance

### Description

When I try to SSH to my Compute Classic instance, I get a warning message like the following:

```
@@
@   WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!   @
@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
d2:aa:50:d4:ff:dc:76:1d:16:95:4a:77:c4:12:87:0f.
Please contact your system administrator.
Add correct host key in /home/joe/.ssh/known_hosts to get rid of this
message.
Offending key in /home/joe/.ssh/known_hosts:63
RSA host key for 11.12.13.14 has changed and you have requested strict
checking.
Host key verification failed.
```

### Solution

This error occurs when you use SSH to connect to an Oracle-provided Oracle Linux instance that has a new RSA key fingerprint.

The RSA key fingerprint of a Compute Classic instance changes when, for example, an instance that isn't set up to boot from a persistent disk is re-created. When you first connected to your Compute Classic instance, the original RSA key fingerprint was stored on your local host. Subsequently, whenever you use SSH to connect to your instance, the instance sends its current fingerprint. The SSH client compares the received fingerprint with the locally stored fingerprint. If the fingerprints don't match, then this error occurs, and the `ssh` command fails.

Note that this warning message is returned by the OpenSSH client on an Oracle Linux host. If you're using a different SSH client or a different operating system, then the error message may be different.

To solve this error, you must remove the old (and now invalid) RSA fingerprint of the instance from the local host.

- In Linux, the RSA key fingerprints are usually stored in the `/home/user/.ssh/known_hosts` file on the host from which you are trying to `ssh` to the instance. Each line in this file starts with the IP address or host name of a remote host. Open the file in a text editor, identify the line corresponding to the IP address of the instance that you're trying to access, and delete that line.
- In Windows, by default, PuTTY stores keys for known hosts in the `HKEY_CURRENT_USER\Software\SimonTatham\PuTTY\SshHostKeys` registry. Each key has a name in the format, `rsa2@22:ip_address`. Using the Registry Editor, identify the key corresponding to the IP address of the instance that you're trying to access, and delete it.

**▲ Caution:**

Improper use of the Windows Registry Editor can cause serious problems. Before you do this, make sure that you're aware of the associated risks. See the documentation accompanying the operating system of your local host.

The next time you use SSH to connect to the Compute Classic instance, a message is displayed indicating that the authenticity of the host can't be established. At the prompt to continue connecting, enter `yes`. The new fingerprint is added to the local host, and the connection goes through.

## Can't get instances to communicate with each other

### Description

I've created multiple instances, but am unable to configure them to communicate with each other.

### Solution

By default, instances can communicate with each other only if they're part of the same security list. If your instances aren't part of the same security list, then you can add them to a security list, as described in [Adding an Instance to a Security List](#). Alternatively, if you want to keep your instances in separate security lists, then you can define security rules that enable all instances in a specified security list to communicate with all instances in another security list. See [Managing Security Rules](#).

## My instance can't connect to instances in another IP network using an IP network exchange

### Description

I created an instance using an Oracle Linux 5.11 image and added multiple interfaces of the instance to different IP networks. Some of the IP networks are connected to other IP networks through an IP network exchange. However, my instance can't send

or receive traffic from instances on any of the IP networks that are connected over an IP network exchange.

### Solution

On instances created using Oracle Linux 5.x images or any image that doesn't support DHCP option 121, when you add the instance to multiple networks, the routes to the IP network exchanges aren't added automatically in the instance route table. This prevents the instance from sending or receiving traffic to and from instances on other IP networks using IP network exchanges. However, the instance can send and receive traffic from other instances on IP networks that it is added to directly (not through an IP network exchange).

You can manually add the required routes in the routing table. Add one route for each IP network connected through an IP network exchange.

For example, consider an instance with two network interfaces. One network interface, `eth0`, is configured as the default gateway on either the shared network or on an IP network. Another network interface, `eth1`, is added to **IPNetwork1** which has the CIDR `10.32.1.0/24`. This **IPNetwork1** belongs to an IP network exchange, say **IPX1**. If another IP network, say **IPNetwork2** with CIDR `10.32.2.0/24`, also belongs to the same IP network exchange, **IPX1**, then to add a route to **IPNetwork2**, run the following command on the instance:

```
ip route add 10.32.2.0/24 via 10.32.1.1
```

In this example, `10.32.2.0/24` describes **IPNetwork2** in CIDR format, that is, the IP network that you want to exchange traffic with.

`10.32.1.1` is the default gateway of **IPNetwork1**, that is, the IP network that the instance is added to. Here both IP networks must belong to the same IP network exchange.

On a Windows instance, for the same scenario, run the following command:

```
route add 10.32.2.0/24 10.32.1.1
```

## Can't access my instance even though it has a public IP address

### Description

I created an instance and associated a public IP address with it. I had earlier created an instance that doesn't have a public IP address. I tried to access the second instance from the first instance, but `ssh` times out without connecting.

### Solution

An instance that doesn't have a public IP address can connect to any other instance only over the private IP address of the destination instance. If you attempt to connect to the public IP address of the newer instance, it will fail.

For example, let's say you created `Inst1` without a public IP address. You subsequently created `Inst2` and associated a public IP address with `Inst2`. Now `Inst1` can connect to `Inst2` using the private IP address of `Inst2`. However, `Inst1` can't connect to `Inst2` using the public IP address of `Inst2`.

To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).

## Can't remove an IP address from an instance

### Description

I associated a temporary IP address with my instance while creating the instance using the Create Instance wizard. Now I want to remove the temporary IP address and use an IP address reservation instead. How can I remove the temporary IP address from my instance? The **Remove Instance** option in the web console is disabled.

### Solution

You can't remove a temporary IP address from an instance. You can only remove a persistent IP address. If you created an instance with an autogenerated IP address or if you changed the status of the IP address associated with an instance to temporary, then to remove that IP address from the instance, first update it to change its status to permanent. See [Updating an IP Reservation](#).

## Can't delete a security application

### Description


When I tried to delete the security application `/oracle/public/snmp-trap-udp`, the following error message was displayed:

```
APIUnauthorizedError: User /Compute-acmecorp/acmeadmin is not permitted to perform "secapplication.delete" on secapplication: /oracle/public/snmp-trap-udp
```

### Solution

Compute Classic has a set of predefined security applications. The names of these security applications start with `/oracle/public` container. You can't delete these predefined security applications.

#### Tip:

To view a list of predefined security applications from the web console, click the **Network** tab and then the **Security Applications** tab in the left pane. The list of available security applications is displayed. In the search field, enter `/oracle/public`, and click . A list of all the predefined security applications is displayed.

To get a list of all the predefined security applications from the CLI, use the `opc compute sec-application list` command and specify `/oracle/public` as the container. For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

## SSH Key Problems

This section lists problems you might encounter while using SSH public keys to securely access your Compute Classic Linux instances.

### My SSH public key doesn't show up in the Create Instance wizard

#### Description

I've uploaded my SSH public key and I can see it on the SSH Public Keys page, but when I try to create an instance, the SSH key isn't listed in the Create Instance wizard. So I'm unable to associate this key with any instance.

#### Solution

Check whether your SSH key has been disabled. If an SSH public key is disabled, it isn't listed in the Create Instance wizard and you can't associate it with your instance. Go to the SSH Public Keys page and update the SSH key to enable it, and then launch the Create Instance wizard again. See [Enabling an SSH Public Key](#).

### Can't connect to an instance using SSH

#### Description

I've created an instance but can't connect to it using SSH.

#### Solution

Check for each of the following possible causes:

1. Is your instance configured for SSH access?

All Oracle-provided Oracle Linux and Solaris instances are by default configured to allow SSH access. However, if you're creating an instance using a private image or an image from Oracle Cloud Marketplace, the instance might not be configured to allow SSH access. Check with the owner of the machine image.

Remember, also, that you can't access Windows instances using SSH. If you're trying to log in to a Windows instance, use RDP. See [Accessing a Windows Instance Using RDP](#).

2. Did you use the correct user?

- To log in to an instance that was created by using an Oracle-provided Oracle Linux machine image, use the `opc` user.

For instances created by using other machine images, find out which SSH-enabled users are defined in that machine image, and log in as one of those users.

- To log in to an instance as a user that was created after the instance was provisioned, you must generate an SSH key pair for the new user and copy the public key to the `~/.ssh/authorized_keys` file of the user. You must also add the new user to the list of allowed users in the `/etc/ssh/sshd_config` file on the instance. See [Adding Users on an Oracle Linux Instance](#).

3. Did you specify the correct public IP address of the instance?

To find out the public IP address of your instance, view the information on the Instances page. See [Listing Instances](#).

If no public IP address is associated with the instance, reserve and associate a public IP address. See [Reserving a Public IP Address](#) and [Updating an IP Reservation](#).

4. Did you specify the correct private key?

The private key that you specify must correspond to one of the public keys associated with the instance.

5. Does the instance belong to a security list with the inbound policy set to `deny`?

An instance can be associated with multiple security lists. You can find out which security lists an instance is attached to by viewing the details of the instance. See [Monitoring Instances](#).

You can see the policies used by each security list by viewing the details of the security list from the web console.

If there's a conflict between the policies of the various security lists, then the most restrictive policy is applicable. This means that if even one of the security lists that your instance is attached to has the inbound policy set to `deny`, then your instance can't receive traffic.

If this is the case, then create a security rule to explicitly allow traffic to a security list that your instance is attached to.

6. Does the error message contain the following warning?

```
WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!
```

If yes, then see [RSA key fingerprint error while connecting to an instance](#).

## Can't access an instance as a local user over SSH

### Description

I created a local user on an instance by using the `useradd` command, but I can't access the instance over SSH as that user.

### Solution

To SSH into an instance using a local user account created with `useradd`, you must generate an SSH key pair for the new user and copy the SSH public key to the appropriate path for the new user. You must also add the new user to the list of allowed users in the `/etc/ssh/sshd_config` file on the instance. See [Adding Users on an Oracle Linux Instance](#).

## RSA key fingerprint error while connecting to an instance

### Description

When I try to SSH to my Compute Classic instance, I get a warning message like the following:

```
@@
@      WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!      @
@@
```



```
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!  
Someone could be eavesdropping on you right now (man-in-the-middle attack)!  
It is also possible that the RSA host key has just been changed.  
The fingerprint for the RSA key sent by the remote host is  
d2:aa:50:d4:ff:dc:76:1d:16:95:4a:77:c4:12:87:0f.  
Please contact your system administrator.  
Add correct host key in /home/joe/.ssh/known_hosts to get rid of this  
message.  
Offending key in /home/joe/.ssh/known_hosts:63  
RSA host key for 11.12.13.14 has changed and you have requested strict  
checking.  
Host key verification failed.
```

## Solution

This error occurs when you use SSH to connect to an Oracle-provided Oracle Linux instance that has a new RSA key fingerprint.

The RSA key fingerprint of a Compute Classic instance changes when, for example, an instance that isn't set up to boot from a persistent disk is re-created. When you first connected to your Compute Classic instance, the original RSA key fingerprint was stored on your local host. Subsequently, whenever you use SSH to connect to your instance, the instance sends its current fingerprint. The SSH client compares the received fingerprint with the locally stored fingerprint. If the fingerprints don't match, then this error occurs, and the `ssh` command fails.

Note that this warning message is returned by the OpenSSH client on an Oracle Linux host. If you're using a different SSH client or a different operating system, then the error message may be different.

To solve this error, you must remove the old (and now invalid) RSA fingerprint of the instance from the local host.

- In Linux, the RSA key fingerprints are usually stored in the `/home/user/.ssh/known_hosts` file on the host from which you are trying to `ssh` to the instance. Each line in this file starts with the IP address or host name of a remote host. Open the file in a text editor, identify the line corresponding to the IP address of the instance that you're trying to access, and delete that line.
- In Windows, by default, PuTTY stores keys for known hosts in the `HKEY_CURRENT_USER\Software\SimonTatham\PuTTY\SshHostKeys` registry. Each key has a name in the format, `rsa2@22:ip_address`. Using the Registry Editor, identify the key corresponding to the IP address of the instance that you're trying to access, and delete it.

### ▲ Caution:

Improper use of the Windows Registry Editor can cause serious problems. Before you do this, make sure that you're aware of the associated risks. See the documentation accompanying the operating system of your local host.

The next time you use SSH to connect to the Compute Classic instance, a message is displayed indicating that the authenticity of the host can't be established. At the prompt

to continue connecting, enter `yes`. The new fingerprint is added to the local host, and the connection goes through.

## Storage Volume Problems

This section lists problems you might encounter while creating and using storage volumes.

### Can't attach a storage volume to an instance

#### Description

When I try to attach my storage volume to an instance, the following error occurs:

```
APIConflictError: Attachment index 1 is already in use on instance /Compute-acmecorp/acmeadmin/dev2/6073c806-f7da-47eb-9678-6e618931b29a
```

#### Solution

The index number that you're trying to assign to this storage volume is already used for another storage volume. Select a different index number and try again.

#### Note:

The disk number that you specify here determines the device name. The disk attached at index 1 is named `/dev/xvdb`, the disk at index 2 is `/dev/xvdc`, the disk at index 3 is `/dev/xvdd`, and so on.

To view the set of index numbers that are already in use for an instance using CLI, use the `opc compute storage-attachment list` command.

For help with that command, run the command with the `-h` option. For the instructions to install the CLI client, see [Preparing to Use the Compute Classic CLI in CLI Reference for Oracle Cloud Infrastructure Compute Classic](#).

### Can't access a storage volume on my instance

#### Description

I successfully created a storage volume by using the web console, but I can't see that disk when I log in to my instance.

#### Solution

After creating a storage volume, you must attach it to your instance. Then you must format the volume and mount it on your instance. See [Attaching a Storage Volume to an Instance](#) and [Mounting and Unmounting a Storage Volume](#).

## I can no longer access my storage volume from my instance

### Description

I had mounted a storage volume on my instance some time ago, but I don't see it in the list of devices mounted on the instance today.

### Solution

In certain circumstances, storage volumes that were attached to and mounted on your instance might need to be attached and mounted again. This happens if your instance stopped and was re-created automatically, or if you deleted your instance and re-created it. Consider the following:

- Is your instance set up to boot from a nonpersistent disk?  
If yes, then when the instance is re-created, all the attached storage volumes must be mounted again.
- Did you attach the storage volume to the instance *after* creating the instance?  
If yes, then when the instance is re-created, you must attach the storage volume again.

Note that, though you might need to attach and mount a storage volume again after an instance is re-created, the data stored on the storage volume isn't lost.

## Can't detach a storage volume from an instance

### Description

I've attached three storage volumes to my instance. Now I want to delete the instance. So I started to detach the storage volumes. I detached two of the storage volumes, but can't detach the third one.

### Solution

You can detach storage volumes that were attached to an instance *after* the instance was created. You can't detach storage volumes that were attached *during* instance creation.

## Can't delete a storage volume

### Description

I want to delete a storage volume that I no longer need, but the web console doesn't show the delete option for the storage volume.

### Solution

You can't delete a storage volume if it's attached to an instance. To find out whether a storage volume is attached to an instance, view the storage volume information in the web console. Click the **Storage** tab, scroll down to the storage volume that you want to delete, and check the displayed details. If the storage volume that you want to delete is attached to an instance, then you must detach it first. See [Detaching a Storage Volume from an Instance](#).

You can also use CLI commands to find out if a storage volumes is attached to an instance. Use the `opc compute storage-attachment list` CLI command with the `--storage_volume_name` option to view the details of storage attachments for a specified storage volume. Then use the `opc compute storage-attachment delete` CLI command to detach a storage volume from an instance.

For help with these commands, run each command with the `-h` option. For the instructions to install the CLI client, see *Preparing to Use the Compute Classic CLI* in *CLI Reference for Oracle Cloud Infrastructure Compute Classic*.

Also, you can't delete a storage volume if you've created any colocated snapshots of the storage volume. See [Backing Up and Restoring Storage Volumes Using Snapshots](#).

## Can't delete a storage volume snapshot

### Description

I want to delete a storage snapshot that I no longer need, but the **Delete** option for that snapshot is disabled in the web console.

### Solution

You can't delete a colocated storage volume snapshot if you've created a clone from that snapshot. The Storage Snapshots page shows you information about each colocated storage snapshot including storage volumes cloned from a snapshot, if any. You can make a note of the clones and then view them on the Storage Volumes page. To delete a colocated snapshot, first delete all the clones created from the storage snapshot. See [Deleting a Storage Volume](#).

## Orchestration Problems

This section lists issues that you might encounter while using orchestrations to create and manage objects.

## Can't upload an orchestration

### Description

When I try to upload my orchestration file, I get the following error: "Unable to create an orchestration from the JSON file."

### Solution

This error indicates that there are errors in the syntax of your orchestration JSON file. Open the JSON file in a text editor to identify and fix the problems. You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice.

#### Note:

Oracle doesn't support or endorse any third-party JSON-validation tool.

## Error while uploading an orchestration: User is not permitted...

### Description

When I try to upload my orchestration file, I get an error similar to the following:

```
User /Compute-acme/jack.jones@example.com is not permitted to perform
"orchestration.add" on
orchestration:/Compute-acme/jack.jones@example.com/orchestration-1
```

### Solution

This error indicates that you're not authorized to create an orchestration in Compute Classic. Check the orchestration JSON file to ensure that you've entered the correct account and user name in the orchestration name. For example, the term `/Compute-acme` in this example should be replaced with `/Compute-your_account_name` and the user name, `jack.jones@example.com` in this example, should be replaced with your user name.

Also, check that you have the appropriate role to create an orchestration. You must have the `Compute_Operations` role assigned to you in Oracle Cloud Infrastructure Classic Console. If you don't have this role, ask your service administrator to assign the `Compute_Operations` role to you in Oracle Cloud Infrastructure Classic Console. See *Modifying User Roles in Managing and Monitoring Oracle Cloud*.

## My orchestration hasn't created any instances

### Description

I've uploaded my orchestration file but I don't see my instances. What should I do?

### Solution


After uploading your orchestration, the status of your orchestration is automatically set to `Stopped`. To create the resources defined in your orchestration, start your orchestration. If you have created orchestration v2, see [Starting an Orchestration v2](#). If you have created orchestration v1, see [Starting an Orchestration v1](#).

## Error while starting an orchestration: object not found

### Description

I'm trying to start an orchestration that I've used before to create an instance. In the past it has started successfully, but this time it is showing an error. How can I find out what's wrong?

### Solution

You can view an orchestration to see more information about its status. On the Orchestration page, go to the orchestration that you're trying to start and, from the  menu, select **View**. The orchestration might show an error message similar to the following:

```
"info": {
  "errors": {
    "0": "{u'instances[vm-1].nat': [u'ipreservation Compute-acme/jack/IP-res-1
not found']}"
```

or

```
"info": {
  "errors": {
    "0": "{u'instances[vm-1].nat': [u'ipreservation Compute-acme/jack/IP-res-1 is
already in use']}"
```

These error messages indicate that an object that is referenced in the orchestration is not available. In this example, the first message indicates that the IP reservation `IP-res-1` has been deleted and the second message indicates that the IP reservation is associated with a running instance. Similar errors would be displayed in the orchestration if an SSH key, a security list, a storage volume, or any other object referenced in the orchestration has been deleted, disabled, or assigned to another resource after this orchestration was stopped.

When you've identified the object that's preventing your orchestration from starting, make that object available to your orchestration. Then stop the orchestration and start it again.

## Error while starting an orchestration: Specify imagelist or bootorder

### Description

I've uploaded my orchestration, but when I start it, the following error occurs:

Specify either an `ImageList` or `boot_order` and `StorageVolume`.

### Solution

This error indicates that your orchestration doesn't specify either an image or a bootable storage volume for your instance.

- To set up the instance to boot from a persistent disk, you must attach a bootable storage volume by using the `storage_attachment` instance attribute, and then specify the index number of the attached storage volume as the boot disk by using the `boot_order` instance attribute.

```
{
  "objects": [
    {
      "instances": [
        {
          "boot_order": [
            1
          ],
          "storage_attachments": [
            {
              "index": 1,
              "volume": "/Compute-acme/joe/bootable-vol1"
            }
          ]
        }
      ]
    }
  ]
}
```

```
]
}
```

- To set up the instance to boot from a nonpersistent disk, specify the image that you want to use by using the `imagelist` attribute.

```
{
  "objects": [
    {
      "instances": [
        {
          "imagelist": "/oracle/public/OL_6.7_UEKR4_x86_64"
        }
      ]
    }
  ]
}
```

 **Note:**

If you specify both `boot_order` and `imagelist` for an instance in an orchestration, the `imagelist` attribute is ignored and the instance is booted using the bootable storage volume specified by the `boot_order` attribute. See [Orchestration v1 Attributes for instances](#).

## My instance was created using a wrong image

### Description

I created an instance using an orchestration. I specified an image in the orchestration file, but my instance was created using a different image.

### Solution

Check your orchestration file. In the instance attributes, did you specify a bootable storage volume using the `storage_attachment` attribute? Did you also specify an image in the `imagelist` attribute?

If you want to use a bootable storage volume to boot your instance, use the `boot_order` instance attribute to specify the appropriate storage volume index number. If you've not specified the appropriate index number in the `boot_order` attribute, then your instance will be booted using the image you've specified in the `imagelist` attribute.

If you want to boot your instance using a nonpersistent storage volume, ensure that you've *not* specified the `boot_order` attribute and that you've specified a valid image for the instance using the `imagelist` attribute instead. Remember, if you specify a valid value for both `boot_order` and `imagelist`, the `imagelist` attribute is ignored and the instance is booted using the image stored on the bootable storage volume specified by the `boot_order` attribute.

For more information about instance attributes, see [Orchestration v1 Attributes for instances](#).

## My orchestration is stuck in the stopping state

### Description

I tried to stop an orchestration but it's been stuck in the **Stopping** state for a long time and the objects defined in that orchestration haven't been deleted. Why did this happen and what should I do?

### Solution

An orchestration can get stuck in the **Stopping** state if any of the objects defined in the orchestration are used or referenced by other objects. While stopping an orchestration, ensure that none of the objects in that orchestration are used or referenced by any other object.

For example, let's say you've created an orchestration, `seclist_orch`, which defines a set of security lists. If any security list in this orchestration is used in a security rule, or has any running instances added to it, then that security list can't be deleted. So the `seclist_orch` orchestration can't be stopped. In this example, you'd have to delete any security rules that use any of the security lists in the `seclist_orch` orchestration. You'd also have to detach any instances that have been added to any of the security lists in the `seclist_orch` orchestration.

When you've cleared all existing dependencies, the orchestration that's in the **Stopping** state will automatically transition to the **Stopped** state.

## opc-init Problems

This section lists problems that you might encounter while using `opc-init` in Compute Classic instances.

### Error using `opc-init`: No chef attributes passed – nothing to do

#### Description

I used an orchestration to create an instance using an Oracle-provided Oracle Linux image. In the orchestration, I provided user data attributes to deploy Chef on the instance. However, when the instance is created, Chef is not deployed on it. The `opc-init` log file provides the following error:

```
No chef attributes passed - nothing to do
```

What does this mean?

#### Solution

This error indicates that `opc-init` was unable to locate the chef attributes in the user data you provided while creating your instance. Check your orchestration file. Ensure that your user data attributes are part of instance attributes, as shown in the following example:

```
"instances": [  
  {
```



```
<Specify other instance attributes here.>
"attributes": {
  "userdata": {
    "chef": {
      <Specify chef attributes here.>
    }
  }
}
]
```

## Error using opc-init: location not provided, exiting...

### Description

While creating an instance, I provided user data attributes to deploy Chef on the instance. However, when the instance is created, I don't see Chef deployed on it. The opc-init log file provides the following error:

```
opc-init location not provided, exiting...
```

What does this mean?

### Solution

This error indicates that opc-init wasn't found on the instance. When you create an instance, select a machine image that includes opc-init. You can either use an Oracle-provided Oracle Linux or Windows image or use a private image that has opc-init installed.

## Configuring opc-init automation in launch plan doesn't work

### Description

I tried to configure opc-init automation in my launch plan, but it doesn't seem to be working. What should I do?

### Solution

Error messages during automation steps are not captured via the API. Check the `/var/log/opc-compute/opc-init.log` file on your instance for error messages.

## Launch Plan Problems

This section lists issues that you might encounter when you use launch plans to create instances.

## Can't create an instance using a launch plan. Error: Unable to open file

### Description

When I try to create an instance using a launch plan, I get the error, "Unable to open file."

### Solution

This error indicates that you might have entered the name or path to your JSON file incorrectly. Check the filename and location of the JSON file that you want to use and then run the `launch` command again.

## Can't create an instance using a launch plan. Error displayed: Data is invalid JSON

### Description

When I try to create an instance using a launch plan, I get the error, "Data is invalid JSON."

### Solution

There may be an error in the JSON file that you specified with the `launch` command. To identify the error in the JSON file, look at the text displayed on the console immediately before this error message appeared. You might see a message similar to the following:

```
Expecting delimiter: line 10 column 13 (char 314).
```

Open your JSON file in a text editor and use the information in the error message to identify and fix the problem. You should also validate your JSON file. You can do this by using a third-party tool, such as [JSONLint](#), or any other validation tool of your choice. Then run the `launch` command again.

 **Note:**

Oracle doesn't support or endorse any third-party JSON-validation tool.

## Configuring opc-init automation in launch plan doesn't work

### Description

I tried to configure opc-init automation in my launch plan, but it doesn't seem to be working. What should I do?

### **Solution**

Error messages during automation steps are not captured via the API. Check the `/var/log/opc-compute/opc-init.log` file on your instance for error messages.