Oracle® Cloud

Using Oracle Cloud Infrastructure Load Balancing Classic

E76938-09
May 2020

Oracle Cloud Using Oracle Cloud Infrastructure Load Balancing Classic,

E76938-09

Primary Authors: Peter LaQuerre, Nisha Singh

Contributing Authors: Pradeep Bhat, Kumar Dhanagopal, Amelia Han, Viktor Perelmouter, Patrice Scattolin, Archana Shukla, Parul Tiwari, Jeff Welsch

Contents

Preface

Audience v

Documentation Accessibility v

Related Resources v

Conventions v

1 Getting Started with Oracle Cloud Infrastructure Load Balancing Classic

About Oracle Cloud Infrastructure Load Balancing Classic 1-1

Oracle Cloud Infrastructure Load Balancing Classic Terminology 1-2

About the Load Balancer IP Addresses and Canonical Host Name 1-3

Interfaces to Oracle Cloud Infrastructure Load Balancing Classic 1-3

Before You Begin with Oracle Cloud Infrastructure Load Balancing Classic 1-3

How to Begin with Oracle Cloud Infrastructure Load Balancing Classic 1-4

How to Access Oracle Cloud Infrastructure Load Balancing Classic Using the Web Console 1-4

About Oracle Cloud Infrastructure Load Balancing Classic Roles 1-5

Using Load Balancer with Oracle PaaS Service Manager (PSM) 1-5

About the Components of Oracle Cloud Infrastructure Load Balancing Classic 1-6

- About Load Balancer Digital Certificates 1-6
- About Load Balancer Listeners 1-7
- About Load Balancer Server Pools 1-7
- About Load Balancer Policies 1-7
- About Load Balancer Properties 1-9

2 Creating Load Balancers

Typical Workflow for Creating a Load Balancer 2-1

Creating an IP Network 2-2

Creating Virtual NIC Sets 2-3

Creating a Load Balancer Using QuickStarts 2-3

Creating a Load Balancer 2-6

Importing a Load Balancer Digital Certificate 2-8
Creating Server Pools for a Load Balancer 2-9
Creating Listeners for a Load Balancer 2-12
Creating Policies for a Load Balancer 2-14
Configuring Load Balancer General Properties 2-19
Adding the Load Balancer IP Addresses to the IP Security List 2-21
Verifying a Load Balancer Configuration 2-21
Obtaining a Digital Certificate 2-22
Obtaining a CA-Issued Certificate 2-22
Creating a Self-Signed Certificate 2-23

3 Managing Load Balancers

Managing Load Balancer Resources Using the Visual Object Editor 3-1
Viewing and Monitoring Your Load Balancers 3-2
Viewing the Properties of a Load Balancer 3-2
Updating a Load Balancer 3-3
Deleting a Load Balancer 3-3
Managing Server Pools for a Load Balancer 3-4
Viewing Server Pools 3-4
Updating a Server Pool 3-5
Deleting a Server Pool 3-5
Managing Load Balancer Digital Certificates 3-6
Viewing a Digital Certificate 3-6
Updating a Digital Certificate 3-6
Deleting a Digital Certificate 3-6
Managing Listeners for a Load Balancer 3-7
Viewing Listeners 3-7
Updating a Listener 3-7
Deleting a Listener 3-8
Managing Policies for a Load Balancer 3-9
Viewing Policies 3-9
Updating a Policy 3-9
Deleting a Policy 3-10

4 Use Cases for Oracle Cloud Infrastructure Load Balancing Classic

Load Balancing HTTP Requests to a Pool of Web Servers 4-1
Securing Incoming Requests to Your Load Balancer 4-5
Securing Connections to the Origin Servers 4-7

Preface

Using Oracle Cloud Infrastructure Load Balancing Classic describes how to provision and manage load balancers using Oracle Cloud Infrastructure Load Balancing Classic.

Topics:

- Audience
- Documentation Accessibility
- Related Resources
- Conventions

Audience

Using Oracle Cloud Infrastructure Load Balancing Classic is intended for members of Development and Operations teams who want to set up, manage, and monitor Load Balancers.

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:

- Oracle Public Cloud

  `http://cloud.oracle.com`
- *Getting Started with Oracle Cloud*

Conventions

The following text conventions are used in this document:

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

To get started with Oracle Cloud Infrastructure Load Balancing Classic, review some basic concepts and the steps required to access the service in Oracle Public Cloud.

Topics

- About Oracle Cloud Infrastructure Load Balancing Classic
- Oracle Cloud Infrastructure Load Balancing Classic Terminology
- About the Components of Oracle Cloud Infrastructure Load Balancing Classic
- About the Load Balancer IP Addresses and Canonical Host Name
- Interfaces to Oracle Cloud Infrastructure Load Balancing Classic
- Before You Begin with Oracle Cloud Infrastructure Load Balancing Classic
- How to Begin with Oracle Cloud Infrastructure Load Balancing Classic
- How to Access Oracle Load Balancer Cloud Service Using the Web Console
- About Oracle Load Balancer Cloud Service Roles

About Oracle Cloud Infrastructure Load Balancing Classic

With Oracle Cloud Infrastructure Load Balancing Classic, you can distribute incoming traffic across multiple Oracle Cloud Infrastructure Compute Classic instances running your applications in the Oracle Public Cloud. In this way, Oracle Cloud Infrastructure Load Balancing Classic provides similar features commonly available in a dedicated hardware load balancer or a separately managed instance of Oracle Traffic Director.

You create a load balancer from the Oracle Cloud Infrastructure Compute Classic console and then associate a pool of Compute instances with each load balancer. If a Compute instance is unavailable or unreachable, the load balancer automatically reroutes the traffic to the remaining compute instances in the server pool. When a Compute instance becomes available and is reachable again, the load balancer resumes routing traffic to that instance.

Additional Compute instances can be added or removed from the server pool as the request load changes, without disrupting the overall access to the application.

The overhead of encrypting and decrypting HTTPS traffic can be offloaded to the load balancer so that Compute instances can focus on their main work.

Note:

You need to first create a load balancer on an IP network before you can use PaaS Service Manager (PSM) to associate pools of PaaS instances or IaaS Compute instances with each load balancer.

Oracle Cloud Infrastructure Load Balancing Classic Terminology

Before you begin using Oracle Cloud Infrastructure Load Balancing Classic you should be familiar with basic load balancer terminology.

Term	Definition
Load balancer	An instance of Oracle Cloud Infrastructure Load Balancing Classic. Each load balancer has at least one listener, where it receives incoming requests, and at least one origin server pool, where it routes the requests, based on the load balancer properties and policies.
Client	The source of a request that is sent to the load balancer. Typically, this a request sent from a user’s browser to your application.
Origin server	A server or host computer to which the load balancer routes requests. In the context of the Oracle Cloud Infrastructure Load Balancing Classic, an origin server is an Oracle Compute Service instance.
Server pool	A pool of origin servers to which the load balancer routes requests. The load balancer selects a specific origin server in the pool using a load balancer algorithm.
Load balancing algorithm	The method a load balancer uses to determine which origin server in the server pool should receive a specific request. The current release of Oracle Cloud Infrastructure Load Balancing Classic supports the round-robin load balancer algorithm. In other words, the load balancer sends each new request to the next origin server in the server pool, using a set order. After it reaches the last server in the list, the load balancer returns to the first server in the list.
Server certificate	A digital certificate used to secure the connection between the clients and the load balancers. See About Load Balancer Digital Certificates.
Trusted certificate	A digital certificate used to secure the connection between the load balancer and the origin servers in a server pool. See About Load Balancer Digital Certificates.

About the Load Balancer IP Addresses and Canonical Host Name

When you use Oracle Cloud Infrastructure Load Balancing Classic to create a load balancer, two IP addresses and a canonical name are assigned to the load balancer. It is important to understand the purpose of these properties and why they are created before you implement a load balancing strategy.

When you create a new load balancer, Oracle Public Cloud creates two Compute instances to host the load balancer processes. These internal Compute instances (or virtual machines) are configured in a highly available configuration to ensure your load balancer is always available.

A canonical host name is also assigned to the load balancer. You use this canonical host name as the target URL for any requests to your applications, or you can map your custom, publicly available domain name to the canonical host name, so requests to your domain URL will be routed to the load balancer.

To map your public or custom domain URL to the load balancer, you (or your domain service provider) can create a canonical name (or CNAME) record that points to the load balancer canonical host name.

Interfaces to Oracle Cloud Infrastructure Load Balancing Classic

Oracle Cloud Infrastructure Load Balancing Classic provides the following ways to access, create, and manage your load balancers.

Type of Access	Description	More Information
Oracle Cloud Infrastructure Compute Classic Web-based console	This console provides a graphical user interface to create, manage, and modify your load balancer configurations.	How to Access Oracle Cloud Infrastructure Load Balancing Classic Using the Web Console
Oracle Cloud Infrastructure Load Balancing Classic REST API	Code REST requests to call methods to programmatically create, manage, and modify the load balancers you have created on Oracle Cloud.	*REST API for Oracle Cloud Infrastructure Load Balancing Classic*

Before You Begin with Oracle Cloud Infrastructure Load Balancing Classic

Before you begin using Oracle Cloud Infrastructure Load Balancing Classic, there are several steps you can take to prepare.

1. Understand the features of Oracle Cloud Infrastructure Compute Classic.

   See About Compute Classic in *Using Oracle Cloud Infrastructure Compute Classic*

2. Understand the features of Oracle Cloud Infrastructure Load Balancing Classic.

See About Oracle Cloud Infrastructure Load Balancing Classic.

How to Begin with Oracle Cloud Infrastructure Load Balancing Classic

To begin using Oracle Cloud Infrastructure Load Balancing Classic, you must first get started with Oracle Cloud Infrastructure Compute Classic.

1. Request a trial or purchase a subscription for Oracle Cloud Service.

   See Requesting and Managing Free Oracle Cloud Promotions or Buying an Oracle Cloud Subscription in *Getting Started with Oracle Cloud*.

2. Learn about the users and roles required for Oracle Cloud Infrastructure Compute Classic.

   See About Compute Classic Roles in *Using Oracle Cloud Infrastructure Compute Classic.*

3. Learn about the users and roles required for Oracle Cloud Infrastructure Load Balancing Classic.

   See About Oracle Cloud Infrastructure Load Balancing 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.*

How to Access Oracle Cloud Infrastructure Load Balancing Classic Using the Web Console

You can access Oracle Cloud Infrastructure Load Balancing Classic from the Oracle Cloud Infrastructure Compute Classic 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. Under **Services**, click **Compute Classic**.

   The **Compute OPC** console is displayed.

3. Click the **Network** tab in the Oracle Compute Cloud Service console.

4. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.

About Oracle Cloud Infrastructure Load Balancing Classic Roles

The following table summarizes the roles you can use to administer and use Oracle Cloud Infrastructure Load Balancing Classic.

Role	Description
Oracle Load Balancer Service Administrator	Users who are assigned this role can perform all the load balancer administration tasks, such as creating, modifying, and deleting load balancers—even those load balancers that are not owned by the user.
Oracle Load Balancer Service Operations	Users who are assigned this role can view, create, update, and delete load balancers that they own. For business continuity, consider creating at least two users with the Oracle Load Balancer Service Operations role. These users must be IT system administrators in your organization.
Oracle Load Balancer Service Read Only Privileges	Users who are assigned this role can view load balancers, but they cannot create, modify, or delete load balancers. The identity domain administrator can create users with this role in Oracle Cloud My Services.
Oracle Load Balancer Service Read Write Privileges	Users who are assigned this role can view and modify the attributes of an existing load balancer. However, they cannot create or delete load balancers.
Oracle Load Balancer Service Cert Management	Users who are assigned this role can create and delete load balancer certificates. They can also view and modify the attributes of an existing load balancer. However, they cannot create or delete load balancers.

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

Using Load Balancer with Oracle PaaS Service Manager (PSM)

This guide describes how to create and manage load balancers from the Compute console for your Oracle Public Cloud Infrastructure as a Service (IaaS) environment.

Note that you can also create load balancers using Oracle PaaS Service Manager (PSM) which caters to the Oracle Cloud Platform as a Service (PaaS) environment. For example, when you are creating a Java Cloud Service instance you can also create a load balancer to use with the Java Cloud Service instance.

Notes have been added to this guide at appropriate places to indicate how the load balacers created using Oracle PaaS Service Manager (PSM) behave differently from the load balancers created from the Compute console.

About the Components of Oracle Cloud Infrastructure Load Balancing Classic

When you create a load balancer using Oracle Cloud Infrastructure Load Balancing Classic, you define various attributes and characteristics of the load balancer.

Specifically, each load balancer consists of the following components. Some of these components are mandatory and some are optional, depending upon the specific topology you are implementing.

Topics

- About Load Balancer Digital Certificates
- About Load Balancer Listeners
- About Load Balancer Server Pools
- About Load Balancer Policies
- About Load Balancer Properties

About Load Balancer Digital Certificates

You must obtain a digital certificate if you want to use a secure connection between the load balancer and the clients sending the request or between the load balancer and the origin servers in the server pool.

For example, in a production environment, you can use the HTTPS protocol to protect the HTTP requests that are sent to the load balancer from the Internet. The HTTPS protocol uses Secure Socket Layer (SSL) or Transport Layer Security (TLS) to secure the HTTP requests to the load balancer. SSL and TLS require a digital certificate.

Before you can use a digital certificate, you must obtain or create the certificate and then import the certificate so you can assign it to a load balancer.

Oracle Cloud Infrastructure Load Balancing Classic supports two types of digital certificates.

- **Server certificates**, which are used to secure the connection between Internet clients and the load balancers. Server certificates secure the URL you are using to access the load balancer; for example:

  ```
  https://mycompany.example.com
  ```

  In this scenario, the load balancer is the server and the clients are the Web browsers sending requests to the load balancer. The clients request the certificate from load balancer to ensure they are connecting to a valid site.

- **Trusted certificates**, which are used to secure the connection between the load balancer and the origin servers in the server pool.

  In this scenario, the load balancer is the client, connecting to server software (such as an application server or Web server) that is running on the origin server. The application or Web servers have been configured to accept only secure HTTPS or SSL requests, and the clients must present a specific, trusted certificate to the server.

About Load Balancer Listeners

Before you use a load balancer, you must define at least one listener. A listener defines the virtual host, port, and protocol that the load balancer will use to listen for new requests.

For example, suppose you want the load balancer to listen for all requests to the following URL:

```
http://mycompany.example.com:80
```

For this example, you define a new listener called “myFirstListener.” The listener would include the following characteristics:

- Name: `myFirstListener`
- Port: `80`
- Virtual Hosts: `mycompany.example.com`

The load balancer listens for any requests to that specific URL and then routes those requests to the server pool defined for that listener.

About Load Balancer Server Pools

When you create a load balancer with Oracle Cloud Infrastructure Load Balancing Classic, you must define one or more servers (referred to as origin servers) to which the load balancer can distribute requests. A set of origin servers is called a server pool. When a request is received on one of the load balancer listeners, the load balancer routes that request to one of the servers in the server pool.

To define a server in the server pool, enter the IP address or DNS name of the compute instances, or you can select existing Compute instances available in your environment.

You can create a single server pool for each load balancer, or you can define individual server pools and assign them to specific listeners.

About Load Balancer Policies

Oracle Cloud Infrastructure Load Balancing Classic provides advanced features that you can configure by attaching specific policies to the load balancer.

The following table describes some of the policies you can set for a specific load balancer.

Policy	Description
Application Cookie Stickiness Policy	This policy enables session stickiness (session affinity) for any request based on a given cookie name specified in the policy. The cookie name is application specific. A session is defined by the presence of the same cookie value for that given cookie name in incoming requests. All requests belonging to the same session are routed to the same origin server.

Policy	Description
CloudGate Policy	This policy provides attributes to protect resources/applications with the help of CloudGate module available in Load Balancer. The attributes in this policy are used to set CloudGate specific headers. These headers will enable CloudGate to lookup for the application and the policy present under the appropriate IDCS Tenant containing information for the protection mechanism to be enforced.
Load Balancer Cookie Stickiness Policy	This policy enables session stickiness (session affinity) for all requests for a given period of time specified in the policy. A unique cookie is generated by the load balancer with an expiration time for any new request that does not already carry the same cookie. Clients are expected to carry this cookie in subsequent requests to form a session. Until expiration time has reached, the load balancer will route all such requests to the same origin server. If expiration time specified in the policy is 0, the session will last indefinitely until the cookie is no longer present in the subsequent requests.
Load Balancing Mechanism Policy	This policy enables you to specify a load balancing mechanism for distributing client requests across multiple origin servers by using one of the following methods: • **Round Robin** • **IP Hash** • **Least Connections**
Rate Limiting Request Policy	Limits the number of requests that can be processed per second by the load balancer. This feature enables administrators to optimize the utilization of the available bandwidth by limiting the rate of incoming requests from clients.
Redirect Policy	Redirects all requests to this load balancer to a specific URI.
Resource Access Control Policy	Controls what clients have access to the load balancer, based on the IP address or the Classless Inter-Domain Routing (CIDR) range of the incoming request.
Set Request Header Policy	Configures a load balancer listener so it inserts additional information into the standard HTTP headers of the requests it forwards to its server pool. You can identify the specific headers you want to modify and actions to perform if the header is already populated with specific values.
SSL Negotiation Policy	If you are securing connections between incoming client requests and the load balancer, you can use this policy to define specific SSL protocols, ciphers, and server order preference for the secure connection.
Trusted Certificate Policy	Identifies a **trusted certificate**, which the load balancer uses when making a secure connection to the compute instances in the server pool. Before you can reference the trusted certificate, you must import it, using the Digital Certificates tab in the Compute console. You can then can reference the certificate using the name (or URI) you provided when you imported the certificate. If you are configuring a secure connection (HTTPS or SSL) between the load balancer and the origin servers, you must add this policy to the load balancer. See About Load Balancer Digital Certificates.

About Load Balancer Properties

Each load balancer you create with Oracle Cloud Infrastructure Load Balancing Classic has a set of properties. You can set these properties when you create the load balancer or edit them later.

The load balancer properties are divided into these categories:

- **Information properties**, which provide static information about the load balancer, such as the IP addresses assigned to the load balancer and the health state of the load balancer.
- **General properties**, such as the load balancer name and description. You can also use this category to optionally set additional general properties, such the list of HTTP methods supported by the load balancer (GET, PUT, POST, and so on).

2

Creating Load Balancers

You can create and manage one or more load balancers for your Oracle Public Cloud Infrastructure as a Service (IaaS) environment. The load balancer creation process ensures you have a secure, reliable, and efficient system for routing requests to your applications and services.

Topics

- Typical Workflow for Creating a Load Balancer
- Creating an IP Network
- Creating Virtual NIC Sets
- Creating a Load Balancer
- Importing a Load Balancer Digital Certificate
- Creating Server Pools for a Load Balancer
- Creating Listeners for a Load Balancer
- Creating Policies for a Load Balancer
- Verifying a Load Balancer Configuration

Typical Workflow for Creating a Load Balancer

Each time you create a new load balancer, you perform a set of steps to define the characteristics and behavior of the load balancer.

Step	Description	More Information
Create an IP network	Create an IP network by providing a name, IP address prefix, IP network exchange, and description. The address range of the IP network is determined by the IP address prefix that you specify while creating the IP network.	Creating an IP Network
Identify the servers and applications that you want to load balance.	This workflow assumes you have already created Oracle Cloud Infrastructure Compute Classic instances and have a set of servers and applications that you can assign to a load balancer.	Getting Started with Oracle Compute Cloud Service
Create a vNICset	A vNICset is a collection of one or more vNICs.	Creating Virtual NIC Sets

Step	Description	More Information
Create the load balancer and define basic properties	Provide a name and basic properties for the load balancer. When you complete this step, the new load balancer appears on the Balancers page in the Compute console.	Creating a Load Balancer
Obtain and import a digital certificate	If you plan to use a secure, Secure Socket Layer (SSL) connection between the load balancer and the host computers that connect to the load balancer, or between the load balancer and the origin servers, then you must obtain and import a valid digital certificate.	Importing a Load Balancer Digital Certificate
Add any specific policies to the load balancer	Optionally, you can assign policies to the new load balancer. Each policy defines a specific behavior or policy for specific types of requests that the load balancer receives.	Creating Policies for a Load Balancer
Create the server pools for the load balancer	Each server pool identifies a set of servers (or Compute instances). When a load balancer listener receives a request, the load balancer routes the request to the server pool.	Creating Server Pools for a Load Balancer
Create the listeners for the load balancer	The listeners define the virtual host, port, and protocol that the load balancer will use to listen for new requests.	Creating Listeners for a Load Balancer
Add the IP addresses of the load balancer to the Security IP list you created for the Compute instances in the server pool.	The security IP list identifies the IP addresses that can access the Compute instances. You should have already configured your IP network so HTTP requests can be received by the Compute instances, but this step ensures the load balancer IP is recognized by the Compute instances.	Adding the Load Balancer IP Addresses to the IP Security List
Verify the load balancer	After you complete these steps, it's important to verify that the load balancer has been configured correctly before you put into production service.	Verifying a Load Balancer Configuration

Creating 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.

To create an IP network, follow the steps provided in Creating an IP Network in *Using Oracle Cloud Infrastructure Compute Classic*.

For more information on IP network, see Managing IP Networks in *Using Oracle Cloud Infrastructure Compute Classic*.

Creating Virtual NIC Sets

A vNIC is a virtualized Network Interface Card. 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.

To create a vNICset, follow the steps provided in Creating a vNICset in *Using Oracle Cloud Infrastructure Compute Classic*.

For more information on vNICsets, see Managing vNICsets in *Using Oracle Cloud Infrastructure Compute Classic*.

Creating a Load Balancer Using QuickStarts

QuickStarts gives you the fastest, easiest way to create a load balancer.

To complete this task, you must have the **Oracle Load Balancer Service Administrator** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.

   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. To create a new load balancer using QuickStarts, click the **QuickStarts** button.

   The Create Load Balancer page is displayed.
4. Enter details for the following fields:

   Load Balancer

   - **Name -** Unique identifier for the load balancer. You must follow these conventions for the Name field:
     - It can contain only alphanumeric characters, hyphens, and underscores.
     - First and last characters cannot be hyphen or underscore.
     - It must not be more than 30 characters.
     - Period is not supported.
   - **IP Networks -** Select the IP network to be associated with the load balancer. The IP network should be pre-created as described in Creating an IP Network in *Using Oracle Cloud Infrastructure Compute Classic*.

> **Note:**
>
> You can configure your PaaS service instance and load balancer associated with it in same IP network or in the IP networks connected through an IP network exchange. You must create an IP network, create a load balancer in that IP network, and while creating the PaaS service instance choose the same IP network (or some other IP network that's connected through an IP network exchange to the IP network intended to be used for the PaaS instance).
>
> See Managing IP Network Exchanges in *Using Oracle Cloud Infrastructure Compute Classic*.

- **Scheme** - Select a scheme for the load balancer:
  - **Internet-facing** - This scheme allows you to create an internet-facing load balancer in a given IP network using Oracle Cloud Infrastructure Load Balancing Classic. This option enables you to add a load balancer to your own IP network, while assigning a internet addressable IP address to the load balancer. This allows your application to be accessible over the internet but at the same time protects the communication between the load balancer and the applications by putting both in the same IP network. In this scheme, the load balancer is typically terminating SSL as well, since the backend traffic is protected inside an IP network, no further encryption is necessary.
  - **Internal** - This scheme allows you to create an internal load balancer in a given IP network using Oracle Cloud Infrastructure Load Balancing Classic. This option enables you to add a load balancer to your own IP network for the sole consumption of other clients inside the same network. Since in this scheme, end to end communications from the client to the balancer and subsequently to the applications are all inside the same IP network, the traffic is entirely protected from the internet. In this scheme, encryption and SSL termination is no longer necessary.

Listener

- **Port** - The port on which the load balancer is listening.

  Supported port numbers are `1` to `65535`, excluding port number `22`. The port number cannot be modified after the listener is created.
- **Balancer Protocol** - The transport protocol that will be accepted for all incoming requests to the selected load balancer listener.
  - Select **HTTP** to listen for non-secure HTTP requests.
  - Select **HTTPS** to listen only for secure HTTP requests sent over SSL or TLS.
- **Security Certificates** - The server security certificate. If the balancer protocol is set to HTTPS then at least one security certificate must be specified. If you want to secure the client connections to the load balancer, then import a server security certificate. Click **Import Security Certificate** and enter details for the fields as described in Importing a Load Balancer Digital Certificate.
- **Server Protocol** - The protocol to be used for routing traffic to the origin servers in the server pool. Select an option from the drop-down list.

Server Protocol	Use this protocol to...
HTTP	Route HTTP or HTTPS requests to the origin servers using the non-secure HTTP protocol.
HTTPS	Route HTTP or HTTPS requests to the origin servers using the secure HTTPS protocol. If you select this option, you must also configure a Trusted Certificate Policy. For more information, see About Load Balancer Policies

- **Trusted Certificate** - If you want to secure the connections between the load balancer and the origin servers in the server pool, then import a trusted certificate. Click **Import Security Certificate** and enter details for the fields as described in Importing a Load Balancer Digital Certificate.
- **Virtual Hosts** - The listener accepts only URI requests that include the host names listed in this field. These host names must exist in the DNS used to reach the load balancer.

  To initially test your load balancer, enter the value of the **Canonical Host Name** load balancer property.

  Later, if you map the load balancer canonical host name to a custom domain name, you can update this property with the actual virtual host names to accept on this listener.

Server Pool

- **Servers** - You must add at least one server to the server pool. You can select a server from the list of instances provided in the drop-down list, or you can add the server details manually.

  If you are selecting a server from the drop-down list, you must first select the server instance and then enter the **Port** the server is listening on.

  If you are adding the server details manually, you must add it in the following format:

  ```
  <Host DNS Name>:Port
  ```

  or

  ```
  <Host IP Address>:Port
  ```

  > **Note:**
  >
  > After you add servers to the **Servers** field, you can double-click a server to enable or disable it, or you can right click to display a context menu of operations to perform on the servers in the field.

  Servers can be added to a server pool at any point of time. However, a server pool cannot have more than 20 servers. Servers can be removed from a server pool and can be re-assigned later to another server pool or the same server pool.

5. Click **Create**.

   A new load balancer is created. If the newly created load balancer does not appear in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

   If you selected the internal scheme for the load balancer then the newly created load balancer is enabled by default. If you selected the internet-facing scheme for the load balancer then the newly created load balancer is disabled by default. To enable the load balancer, go to the **Load Balancers** tab and click ☰ next to the load balancer that you want to enable. Select the **Enable** option.

> **Note:**
>
> - If you selected the IP networks option when creating the load balancer then two listeners (one HTTP and the other HTTPS on ports 80 and 443 respectively) are created by default.
> - If your load balancer was created by Oracle PaaS Service Manager (PSM) then certain parameters of the resources (load balancer, listener, server pool, etc) cannot be modified after the resource creation.

Creating a Load Balancer

When you create a load balancer, you provide a name and the basic properties of the load balancer. Later, you must define server pools, create at least one listener, and optionally define the policies for the load balancer.

To complete this task, you must have the **Oracle Load Balancer Service Administrator** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. To create a new load balancer, click the **Create Load Balancer** button.

   The Create Load Balancer dialog box is displayed.
4. Enter details for the following fields:
   - **Name** - Unique identifier for the load balancer.
     You must follow these conventions for the Name field:
     – It can contain only alphanumeric characters, hyphens, and underscores.
     – First and last characters cannot be hyphen or underscore.
     – It must not be more than 30 characters.

  - Period is not supported.
- **IP Networks** - Select the IP network to be associated with the load balancer. The IP network should be pre-created as described in Creating an IP Network in *Using Oracle Cloud Infrastructure Compute Classic*.

  > **Note:**
  >
  > You can configure your PaaS service instance and load balancer associated with it in same IP network or in the IP networks connected through an IP network exchange. You must create an IP network, create a load balancer in that IP network, and while creating the PaaS service instance choose the same IP network (or some other IP network that's connected through an IP network exchange to the IP network intended to be used for the PaaS instance).
  >
  > See Managing IP Network Exchanges in *Using Oracle Cloud Infrastructure Compute Classic*.

- **Description** - A short description for the load balancer. The description must not exceed 1000 characters.
- **Permitted Methods** - The permitted HTTP methods for this load balancer. You can select the predefined methods (`GET`, `POST`, `PUT`, `PATCH`, `DELETE`, or `HEAD`) or you can also create your own custom methods. Requests with methods not listed in this field will result in a `403` (unauthorized access) response.

  This option is useful if you want to limit the operations performed on the origin servers in the server pool. For example, for a typical Web server implementation, clients should only need to perform basic HTML methods, such as GET and POST. Additional methods, such as PUT and DELETE can be destructive. To take extra steps to protect your data, you can restrict the load balancer to only accept and route only GET and POST requests.
- **Scheme** - Select a scheme for the load balancer:
  - **Internet-facing** - This scheme allows you to create an internet-facing load balancer in a given IP network using Oracle Cloud Infrastructure Load Balancing Classic. This option enables you to add a load balancer to your own IP Network, while assigning a internet addressable IP address to the load balancer. This allows your application to be accessible over the internet but at the same time protects the communication between the load balancer and the applications by putting both in the same IP network. In this scheme, the load balancer is typically terminating SSL as well, since the backend traffic is protected inside an IP network, no further encryption is necessary.
  - **Internal** - This scheme allows you to create an internal load balancer in a given IP network using Oracle Cloud Infrastructure Load Balancing Classic. This option enables you to add a load balancer to your own IP network for the sole consumption of other clients inside the same network. Since in this scheme, end to end communications from the client to the balancer and subsequently to the applications are all inside the same IP network, the traffic is entirely protected from the internet. In this scheme, encryption and SSL termination is no longer necessary.

- **SSL Certificate Port Mapping** - A certificate name and port number pair which explicitly configures a certificate to be returned.
- **Enabled** - Check this option to enable the load balancer.
  Disabling the load balancer results in access getting denied to all clients. For HTTP/HTTPS listeners, disabling results in `503` responses for new requests; existing requests result in `500` responses.

5. Click **Create**.

   A new load balancer is created. If the newly created load balancer does not appear in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

> **Note:**
>
> - You cannot use a load balancer until you finish the configuration of the load balancer by adding a server pool and a listener. If you selected the IP networks option when creating the load balancer then two listeners (one HTTP and the other HTTPS on ports 80 and 443 respectively) are created by default.
> - If your load balancer was created by Oracle PaaS Service Manager (PSM) then certain parameters of the resources (load balancer, listener, server pool, etc) cannot be modified after the resource creation.

Importing a Load Balancer Digital Certificate

After you obtain a digital certificate, you must import it, so the load balancers you create can access the certificates. This operation uploads the certificate to the server, so it can be listed in the Oracle Compute Cloud Service console.

To import a digital certificate:

1. Go to the Network page in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane, and then select the **Digital Certificates**.

   The existing digital certificates are displayed.
3. Click **Import Digital Certificate**.

   The Importing Digital Certificate Dialog page is displayed.
4. Enter details for the following fields:
   - **Certificate Type** - Select the type of certificate that you want to import. You can import a Server Certificate or a Trusted Certificate:
     – If you are importing a certificate to secure the client connections to the load balancer, then select **Server Certificate**.
     – If you are importing a certificate to secure the connections between the load balancer and the origin servers in the server pool, then select **Trusted Certificate**.

     See About Load Balancer Digital Certificates.

- **Name** - Specify a name for the certificate. Name can contain only alphanumeric characters, periods, hyphens and be at most 30 characters long.
- **Certificate** - The PEM encoded body of the server certificate. A `.pem` format file begins with this line:

  ```
  ----BEGIN CERTIFICATE----
  ```

  and ends with this line:

  ```
  ----END CERTIFICATE----
  ```

  A `.pem` format file supports multiple digital certificates (for example, a certificate chain can be included). The order of certificates within the file is important.
- **Private Key** - This field displays only for server certificates. Specify the PEM encoded body of the private key.
- **Certificate Chain** - Specify the PEM encoded bodies of all certificates in the chain up to and including the CA certificate. This is not need when the certificate is self signed.

5. Click **Import**.

   A new certificate is imported. If the newly imported certificate is not appearing in the **Digital Certificates** tab, click ⟳ to refresh the list of imported digital certificates.

   > **Note:**
   >
   > A digital certificate is an immutable entity and its attributes cannot be modified once the certificate is imported. To renew a digital certificate, the listener needs to be updated with a different certificate entity which has been created with the renewed certificate. A digital certificate can be deleted only when it is not referenced by any listeners. Attempting to delete a digital certificate when it is referenced by one or more listeners will result in the `400` error code.

Creating Server Pools for a Load Balancer

Before you can use a load balancer, you must define one or more servers (also known as origin servers) to which the load balancer routes its requests. This set of origin servers is called a server pool. When a request is received on one of the load balancer listeners, the load balancer routes that request to an origin server in the pool.

Before you can add a server pool, you must create a load balancer, as described in Creating a Load Balancer.

To complete this task, you must have at least the **Oracle Load Balancer Service Read Write Privileges** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.

2. Click **Load Balancers** in the left pane, and then select **Load Balancers**.

   The console displays any existing load balancers on the Load Balancers page.

3. Click ☰ next to the load balancer you want to modify. Select the **Update** option.

   The **Overview** tab of the selected load balancer is displayed.

4. Click the **Server Pools** tab in the left pane.

   The Server Pools page lists any server pools already created for this load balancer.

5. Click **Create Server Pool**.

   The **Create Server Pool** dialog box is displayed.

6. Enter details for the following fields:

   - **Name -** Unique identifier for the server pool.

     You must follow these conventions for the Name field:

     – It can contain only alphanumeric characters, hyphens, and underscores.

     – First and last characters cannot be hyphen or underscore.

     – It must not be more than 30 characters.

     Note that the name cannot be modified after the server pool is created.

   - **Servers** - You must add at least one server to the server pool. You can select a server from the list of instances provided in the drop-down list, or you can add the server details manually.
     If you are selecting a server from the drop-down list, you must first select the server instance and then enter the **Port** the server is listening on.

     If you are adding the server details manually, you must add it in the following format:

     ```
     <Host DNS Name>:Port
     ```

     or

     ```
     <Host IP Address>:Port
     ```

     > **Note:**
     >
     > After you add servers to the **Servers** field, you can double-click a server to enable or disable it, or you can right click to display a context menu of operations to perform on the servers in the field.

     Servers can be added to a server pool at any point of time. However, a server pool cannot have more than 20 servers. Servers can be removed from a server pool and can be re-assigned later to another server pool or the same server pool.

   - **vNICSet -** Select the vNICset that has the vNICs of the servers in this pool. This is required only if the servers are attached to IP networks. This vNICSet is used to set the appropriate ACLs to allow egress traffic from the load balancer.

   - **Enabled** - Check this option to enable the server pool.
     Disabling the server pool results in no new connections being distributed to this server pool from the listener. The server pool is automatically disabled when all member servers are disabled. When at least one of the servers is re-enabled the server pool must be explicitly enabled.

7. **Health Check** - The load balancer can perform regular health checks of the origin servers and route inbound traffic to the healthy origin servers. This feature is not enabled automatically when an origin server pool is created and must be enabled explicitly either during the origin server pool creation or update.

   - **Type** - Select the health check mechanism to use to test the origin servers:

     – **HTTP** - If HTTP is selected then the load balancer will send an HTTP HEAD request to the origin servers. The HTTP request path is defined in the **Path** field. The origin server is considered healthy if the HTTP response status code matches the ones defined in the **Accept Return Codes** field.

   - **Path** - The path of the HTTP health check requests. If unspecified then / i.e., all paths is assumed. The **Path** parameter is valid only if the health check **Type** is set to **HTTP**.

   - **Accepted Return Codes** - The HTTP response status codes that indicate the origin server is healthy. The **Accepted Return Codes** field is valid only if the health check **Type** is set to **HTTP**. Accepted return codes can be one or more of the 2xx, 3xx, 4xx, or 5xx codes. If no code is specified then all 2xx and 3xx status codes are considered healthy. If the HTTP health check response status code is one of the values defined in the Accepted Return Codes field, the origin server is considered healthy.

   - **Health Check Enabled** - The health check feature is disabled by default. Check this option to enable the health check feature.

   - **Interval** - The approximate interval, in seconds, that the load balancer will wait before sending the target request to each origin server.

   - **Timeout** - The amount of time, in seconds, that the load balancer will wait without a response before identifying the origin server as unavailable. The timeout value must be less than the interval value and it should range between 2 to 60.

   - **Healthy Threshold** - The number of consecutive successful health checks required before moving the origin server to the healthy state. The value of healthy threshold ranges from 2 to 6. If no value is specified then 6 is considered as the healthy threshold value by default.

   - **Unhealthy Threshold** - The number of consecutive health check failures required before moving the origin server to the unhealthy state. The value of unhealthy threshold ranges from 2 to 10. If no value is specified then 3 is considered as the unhealthy threshold value by default.

8. Click **Create**.

   A new server pool is created. If the newly created server pool is not appearing on the Server Pools page, then click ⟳ to refresh the list of server pools.

> **Note:**
>
> You cannot use a load balancer until you finish the configuration of the load balancer by adding a server pool and a listener.

Creating Listeners for a Load Balancer

A listener defines a virtual host, port that the load balancer is listening on. It also defines the protocol accepted on the listening port. At least one enabled listener is required for a load balancer. You can configure multiple listeners on a single load balancer.

Before you can add a listener, you must create a load balancer, as described in Creating a Load Balancer.

To complete this task, you must have at least the **Oracle Load Balancer Service Operations** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. Click ≡ next to the load balancer that you want to modify. Select the **Update** option.

   The **Overview** page of the load balancer is displayed.
4. Click the **Listeners** tab in the left pane.

   The **Listeners** page with a list of existing listeners is displayed.
5. Click the **Create Listener** button.

   The **Create Listener** page is displayed.
6. Enter details for the following fields:
   - **Name** - Unique identifier for the listener.
     You must follow these conventions for the Name field:
     - It can contain only alphanumeric characters, hyphens, and underscores.
     - First and last characters cannot be hyphen or underscore.
     - It must not be more than 30 characters.

     Note that listener name must be unique within the set of load balancers for a compute region in the identity domain in which the load balancer was created.. The name cannot be modified after the listener is created.
   - **Port** - The port on which the load balancer is listening.

Supported port numbers are 1 to 65535, excluding port number 22. The port number cannot be modified after the listener is created.

- **Balancer Protocol** - The transport protocol that will be accepted for all incoming requests to the selected load balancer listener.
  - Select **HTTP** to listen for non-secure HTTP requests.
  - Select **HTTPS** to listen only for secure HTTP requests sent over SSL or TLS.
- **Server Protocol** - The protocol to be used for routing traffic to the origin servers in the server pool. Select an option from the drop-down list.

Server Protocol	Use this protocol to...
HTTP	Route HTTP or HTTPS requests to the origin servers using the non-secure HTTP protocol.
HTTPS	Route HTTP or HTTPS requests to the origin servers using the secure HTTPS protocol. If you select this option, you must also configure a Trusted Certificate Policy. For more information, see About Load Balancer Policies

- **Server Pool** - The server pool to which the load balancer distributes requests. See Creating Server Pools for a Load Balancer

  Note that you can define a single, common server pool for all listeners for a load balancer by updating the properties of the load balancer. Alternatively, you can define a server pool for each listener. If you do not specify the server pool for the listener then the server pool defined in the Load Balancer properties will be used. If the server pool is specified in the listener, then the server pool for the load balancer will be ignored for this listener.
- **Security Certificate** - The server security certificate. If the balancer protocol is set to either HTTPS or SSL then you must select a server certificate. See About Load Balancer Digital Certificates.
- **Policies** - List of the load balancer policies application to the listener.

  The policies may be applicable to the client side (for example proxy protocol behavior, SSL negotiation) or the server side interaction (for example, security policies) or behavior of certain routing capability (for example, HTTP request header injections) .

> **Note:**
>
> If you set the Server Pool protocol to HTTPS or SSL, then you must select and configure the **Trusted Certificate Policy**. See Creating Policies for a Load Balancer

- **Virtual Hosts** - The listener accepts only URI requests that include the host names listed in this field.

To initially test your load balancer, enter the value of the **Canonical Host Name** load balancer property. See Verifying a Load Balancer Configuration.

Later, if you map the load balancer canonical host name to a custom domain name/vanity URL, you can update this property with the actual virtual host names to accept on this listener.

- **Path Prefixes -** Use this field to configure the listener to accept only requests that are targeted to a specific path within the URI of the request.

  If unspecified, then the listener will accept all request URIs that meet the other criteria of the listener.
- **Tags** - Collection of tags for this listener.
- **Enabled** - Check this option to enable the listener.

  Disabling the listener results in access getting denied to all clients. For HTTP/HTTPS listeners, disabling results in `503` responses with standardized HTML content for new requests; existing requests result in `500` responses.

7. Click **Create**.

   A new listener is created. If the newly created listener is not appearing in the **Listeners** tab, click ⟳ to refresh the list of listeners.

> **Note:**
>
> You cannot use a load balancer until you finish the configuration of the load balancer by adding a server pool and a listener.

Creating Policies for a Load Balancer

Oracle Load Balancer Cloud Service provides advanced features that you can configure by attaching specific policies to the load balancer.

After you create a load balancer as described in Creating a Load Balancer, you can add policies to the load balancer.

To complete this task, you must have at least the **Oracle Load Balancer Service Read Write Privileges** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. Click ☰ next to the load balancer to which you want you modify. Select the **Update** option.

The **Overview** page of the load balancer is displayed.

4. Click the **Policies** tab in the left pane.

   The **Policies** page with a list of any existing policies is displayed.

5. Click **Create Policy**.

   The **Create Policy** dialog displays.

6. Enter details for the following fields:

   - **Policy Type -** Select a policy type from the drop-down list:

     - **Application Cookie Stickiness Policy**
     - **CloudGate Policy**
     - **Load Balancer Cookie Stickiness Policy**
     - **Load Balancing Mechanism Policy**
     - **Rate Limiting Request Policy**
     - **Redirect Policy**
     - **Resource Access Control Policy**
     - **Set Request Header Policy**
     - **SSL Negotiation Policy**
     - **Trusted Certificate Policy**

     For information about these policies, see About Load Balancer Policies.

   - **Name -** Unique identifier for the policy.
     You must follow these conventions for the Name field:

     - It can contain only alphanumeric characters, hyphens, and underscores.
     - First and last characters cannot be hyphen or underscore.
     - It must not be more than 30 characters.

     Note that you cannot change the name of a policy after you create it.

   - Depending on the policy type you select, you may need to provide additional information as follows:

     - **Application Cookie Stickiness Policy**

       * **Name -** Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.

       * **App Cookie Name -** Name of the application cookie used to control how long the load balancer will continue to route requests to the same origin server.

     - **CloudGate Policy**

       * **Name -** Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.

       * **Virtual Hostname for Policy Attribution -** Host name needed by CloudGate to enforce OAuth policies.

     - **Load Balancer Cookie Stickiness Policy**

       * **Name -** Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.

    - **Cookie Expiration Period** - The time period, in seconds, after which the cookie should be considered stale. If the value is zero or negative the stickiness session lasts for the duration of the browser session.

  - **Load Balancing Mechanism Policy**

    - **Name -** Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.

    - **Load Balancing Mechanism** - Select the type of load balancing mechanism for distributing client requests across multiple origin servers:

      - **Round Robin -** In Round Robin mechanism the load balancer forwards requests sequentially to the available origin servers—the first request to the first origin server in the pool, the second request to the next origin server, and so on. After it sends a request to the last origin server in the pool, it starts again with the first origin server.

      - **IP Hash** - In IP Hash mechanism a hash-function is used to determine which server should be selected for the next request based on the client's IP address. This can be used to achieve IP based session stickiness.

      - **Least Connections -** In Least Connections mechanism when a client request is processed, the load balancer assesses the number of connections that are currently active for each origin server, and forwards the request to the origin server with the least number of active connections.

      If no option is specified, the **Round Robin** mechanism is selected by default.

  - **Rate Limiting Request Policy**

    - **Name** - Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.

    - **Zone Name** - Name of the shared memory zone.

    - **Requests Per Second -** Maximum number of requests per second.

    - **Burst Size** - The number of requests that can be delayed until it exceeds the maximum number specified as burst size in which case the request is terminated with an error `503` (Service Temporarily Unavailable).

      > **Note:**
      >
      > Burst size should be a positive integer value between 1 and 10.

    - **Delay Excessive Requests** - Select this option if you don't want to delay excessive requests while requests are being limited.

    - **Logging Level** - Select the desired logging level for cases when the server refuses to process requests due to rate exceeding, or delays request processing:

      - **Info**

        - **Notice**
        - **Warn**
        - **Error**

        If no option is specified, the logging level is set to **Warn** by default.

    - **Rate Limiting Criteria** - Select the criteria based on which requests will be throttled:
        - **Server** - can be used to limit the requests processed by the virtual server.
        - **Remote Address** - can be used to limit the processing rate of requests coming from a single IP address.

        > **Note:**
        >
        > Rate limiting criteria is immutable. It cannot be modified once the policy is created.

    - **HTTP Error Code** - The status code to return in response to rejected requests. You can specify any status code between `405` to `599`. The HTTP error code is set to `503` by default.
    - **Zone Memory Size (MB)** - Size of the shared memory occupied by the zone. The default value for zone memory size is 10 MB.

- **Redirect Policy**
    - **Name** - Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.
    - **Redirect URI** - When this policy is attached to a listener, all requests served by that listener will be redirected to the specified URI.
    - **Response Code** - The exact 3xx response code to use when redirecting.

- **Resource Access Control Policy**
    - **Name** - Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.
    - **Disposition** - The fundamental disposition of security rules.
    - **Permitted Clients** - Set of IP address or CIDR ranges identifying clients from which requests must be accepted by the load balancer.
    - **Denied Clients** - Set of IP address or CIDR ranges identifying clients from which requests must be denied by the load balance.

- **Set Request Header Policy**
    - **Name** - Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.
    - **Header Name** - The name of the HTTP header to be added to the request before proxying to the origin servers. The header name must conform to relevant HTTP RFC guidelines. You can specify any header including standard headers like `HOST`. Header names are not case-sensitive.

    - **Value** - The header value to be added to the request. If multi-valued, multi-line, or special formatting values are used, then appropriate custom transport encoding should also be used. The value is set as-is in the header. The header value must conform to the length restrictions as per HTTP RFC guidelines.
    - **Action When Header Exists** - Select an action to be taken when a header exists in the request:
        - **NOOP -** Take no action if the header exists already.
        - **Prepend** - Add the provided header value to the existing header, but insert it before the existing header content.
        - **Append** - Add the provided header value to the existing header, but insert it after the existing header content.
        - **Overwrite** - Remove any existing value in the header and replace it with the provided header information.
        - **Clear** - Clear any existing header information from the request.

      If no action is specified, the **Overwrite** action is performed.
    - **Action When Header Value Is -** The specified action is taken only when the header exists in the request and the value of the header matches the value in this field.
    - **Action When Header Value Is Not** - The specified action is taken only when the header exists in the request and the value of the header does not match the value in this field.
  - **SSL Negotiation Policy**
    - **Name** - Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.
    - **SSL Protocol** - Click this field and select the specific security protocols supported for incoming secure client connections to the selected listener.
    - **SSL Cipher** - Click this field and select the SSL ciphers supported for incoming secure client connections to the selected listener. The server certificate you are using for this listener should have been created using a signing algorithm based on the ciphers selected in this field. See About Load Balancer Digital Certificates.
    - **Port** - The load balancer port for the the SSL protocols and the SSL ciphers. Supported port numbers are 1 to 65535, excluding port number 22.
    - **Server Order Preference -** Use this option to enable or disable the server order preference for secure connections to this listener.

      During the SSL connection negotiation process, the client and the load balancer present a list of ciphers and protocols that they each support, in order of preference. By default, the first cipher on the client's list that matches any one of the load balancer's ciphers is selected for the SSL connection.

      If **Server Order Preference** is not enabled, the order of ciphers presented by the client is used to negotiate connections between the client and the load balancer. If the **Server Order Preference** is enabled, then the load balancer selects the first cipher in its list that is in the client's list of ciphers. This ensures that the load balancer determines which cipher is used for SSL connection. The default policy has **Server Order Preference** enabled.

- **Trusted Certificate Policy**
  - **Name -** Enter a unique name for this policy, so you can easily identify it in the list of load balancer policies or reference it in a REST API call.
  - **Trusted Certificate URI -** Select a trusted certificate from the drop-down menu.

    The list in the drop-down menu contains the trusted certificates you have obtained or created and imported so they are available to the load balancer.

    This policy is required when you are configuring a secure connection between the load balancer and the origin servers in the server pool. In this scenario, you have configured the application server or Web server software on the origin servers to accept only secure HTTPS or SSL connections.

    See About Load Balancer Digital Certificates.

7. Click **Create**.

   A new policy is created. If the newly created policy is not appearing in the **Policies** tab, click ⟳ to refresh the list of policies.

Configuring Load Balancer General Properties

When you create a load balancer, you change the name of the load balancer and set some general properties of the load balancer.

To complete this task, you must have at least the **Oracle Load Balancer Service Read Write Privileges** role. See About Oracle Load Balancer Cloud Service Roles. If the required 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 My Services. See Modifying User Roles in *Managing and Monitoring Oracle Cloud*.

To configure the load balancer general properties:

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. Click ≡ next to the load balancer that you want to modify. Select the **Update** option.

   The **Overview** page of the load balancer is displayed.
4. Scroll down and expand the General section of the Overview page.

   The list of general properties appears.
5. Use the following table to update the properties:

Option	Description
Name	Unique identifier for the load balancer. This name will appear in the list of load balancers in the console, and you can use this name when referencing the load balancer in REST API calls. You must follow these conventions for field: • It can contain only alphanumeric characters, hyphens, and underscores. • First and last characters cannot be hyphen or underscore. • It must not be more than 30 characters.
Description	A short description for the load balancer. This description appears in the list of load balancers in the console. The description must not exceed 1000 characters.
Permitted Methods	The permitted HTTP methods on this load balancer. You can select the predefined methods (`GET`, `POST`, `PUT`, `PATCH`, `DELETE`, or `HEAD`) or you can also create your own custom methods. Requests with methods not listed in this field will result in a `403` (unauthorized access) response. This property is useful if you want to limit the operations performed on the origin servers in the server pool. For example, for a typical Web server implementation, clients should need to perform only basic HTML methods, such as GET and POST. Additional methods, such as PUT and DELETE can be destructive. To take extra steps to protect your data, you can restrict the load balancer to accept and route only GET and POST requests. Note that the method names are case-sensitive.
Server Pool	Specify the server pool for this listener. Note that you can also specific server pools to each load balancer listener. If you specify a listener for a listener, the server pool designation on the listener overrides this setting on the load balancer. See Creating Listeners for a Load Balancer
Permitted Clients	The list of permitted client IP addresses or CIDR ranges which can connect to this load balancer on the configured listener ports. If the list is empty all connections are permitted.
Tags	Collection of tags for this load balancer.
Enabled	Check this option to enable the load balancer. Disabling the load balancer results in access getting denied to all clients. For HTTP/HTTPS listeners, disabling results in `503` responses for new requests; existing requests result in `500` responses.

6. Click **Update**.

Adding the Load Balancer IP Addresses to the IP Security List

The security IP list identifies the IP addresses that can access the Compute instances in your server pool. Before a load balancer can route requests to the Compute instances, you must modify the security IP list for the compute instances so it includes the IP addresses of the load balancer.

You should have already configured your IP network so HTTP requests can be received by the Compute instances, but this step ensures the load balancer IP addresses are recognized by the Compute instances. See Managing Security IP Lists in *Using Oracle Compute Cloud Service (IaaS)*.

For more information about the load balancer IP addresses, see About the Load Balancer IP Addresses and Canonical Host Name.

1. Locate and note the IP addresses assigned to the Load Balancer.
   a. Go to the **Network** tab in the Oracle Compute Cloud Service console.
   b. Click **Load Balancers** in the left pane and select the Load Balancers option.

      The existing load balancers are displayed.
   c. Go to the load balancer that you want to view and click
   d. In the Information section of the page, note the IP values in the **Virtual Load Balancer IP addresses** field.
2. Add both of the Load Balancer IP addresses to the security IP list for the compute instances in the Server pool.

Verifying a Load Balancer Configuration

When you finish initially configuring a new load balancer, it is important to verify that the load balancer is working properly and routing requests to the origin servers, based on your configuration settings.

To verify a load balancer you just configured, open your browser and enter one of the virtual host URLs that you created the load balancer listeners. Verify that the URL returns the files or applications it should. If you receive an error, review the log files for specific errors.

1. Click the **Network** tab in the Oracle Compute Cloud Service console.
2. Click the **Load Balancers** tab in the left pane, and then click **Load Balancers**.

   The Load Balancers page displays any existing load balancers you have already created.
   If you created a new load balancer recently and it is not appearing on the **Load Balancers** page, click ⟳ to refresh the list of load balancers.
3. Click ≡ next to the load balancer that you want to modify. Select the **Update** option.

   The **Overview** page of the load balancer is displayed.

4. In the Information section of the page, make a note of the **Canonical Host Name** of the load balancer.
5. In a browser window, enter the Canonical Host Name of the load balancer as the URL, followed by the port or path identified in the listener.

   For example:

   ```
   http://canonical_host_name:7777/
   ```

   Based on the settings in the load balancer listener, you should see the appropriate HTML page of your Web server or the appropriate application, served from one of the Compute instances in the load balancer server pool.

Obtaining a Digital Certificate

Before you can secure the connections between Internet clients and the load balancer, or between the load balancer and the origin servers in the server pool, you must obtain a digital certificate.

For production environments, it is recommended that you use a certificate issued from a Certificate Authority (CA). For development environments, you can use either a CA-issued or self-signed certificate.

Topics:

- Obtaining a CA-Issued Certificate
- Creating a Self-Signed Certificate

Obtaining a CA-Issued Certificate

This option is typically used for production deployments. A CA-issued certificate ensures clients are connecting to a valid server and reduces the chances of a man-in-the-middle attack.

There are multiple CA vendors in the marketplace today, each offering different levels of service at varying price points. Research and choose a CA vendor that meets your service-level and budget requirements.

For a CA vendor to issue you a CA-issued SSL certificate, you need to provide the following information:

- Your custom domain name.
- Public information associated with the domain confirming you as the owner.
- Email address associated with the custom domain for verification.

To obtain a CA-issued SSL certificate, you create and submit a Certificate Signing Request (CSR). For more information, review the instructions provided by your CA vendor.

Creating a Self-Signed Certificate

This option is typically used for your development and testing environments.

There are several ways to create a self-signed certificate and several different third-party software tools you can use to accomplish this task. For example, you can use an open source utility, such as OpenSSL.

3 Managing Load Balancers

After creating your load balancers, you can view and monitor your load balancers. You can also manage the server pools, digital certificates, listeners and policies for a load balancer.

Topics

- Viewing and Monitoring Your Load Balancers
- Managing Server Pools for a Load Balancer
- Managing Load Balancer Digital Certificates
- Managing Listeners for a Load Balancer
- Managing Policies for a Load Balancer

> **Note:**
>
> Requests for modifying any resource (load balancer, listener, etc) are asynchronous. The change is not reflected in the resource representation nor effective until the state of the resource transitions to healthy. When the state of the resource is modification-in-progress, it signifies the change is not effective and is therefore not yet part of the resource representation nor effective.

Managing Load Balancer Resources Using the Visual Object Editor

Visual object editor enables you to understand how certain objects relate to each other. If you want a holistic picture of a large number of resources in your account, you might find it easier to grasp this information using the visual object editor.

About the Visual Object Editor

The visual object editor provides you a graphical layout of the load balancer and associated resources such as listeners and server pools in your account. When you view the objects in the visual object editor, you can clearly identify relationships between the load balancers and associated resources. 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. The visual object editor uses connecting lines to indicate relationships between objects.

Accessing the Visual Object Editor

To get to the visual object editor:

1. Sign in to the Compute Classic console.

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, or REST API, then you must refresh the visual object editor to display those changes.

Creating and Updating Objects Using the Visual Object Editor

Note that you cannot create or delete load balancer objects from the visualization page. However, you can create connections between listeners and server pools in the visual object editor. Load balancers, listeners, and server pools can be enabled/ disabled in the visual object editor. You can also update listeners and server pools in the visual object editor. When you update a listener or a server pool using the **Update** option from the popup menu of the object, the object under operation is locked out until the submitted operation is completed.

For more information on the visual object editor, see Managing Resources Using the Visual Object Editor in *Using Oracle Cloud Infrastructure Compute Classic*.

Viewing and Monitoring Your Load Balancers

The Load Balancers page enables you to view, create, modify and delete load balancers.

Topics

- Viewing the Properties of a Load Balancer
- Creating a Load Balancer
- Updating a Load Balancer
- Deleting a Load Balancer

Viewing the Properties of a Load Balancer

After you create a load balancer, you can verify its properties, policies, listeners, and server pools at any time.

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select **Load Balancers**.

   Any existing load balancers are displayed.

If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Go to the load balancer that you want to view and click ⊞.

   The **Overview** tab displays the current Information and General properties of the load balancer. See Configuring Load Balancer General Properties.

Updating a Load Balancer

To modify the details of a load balancer:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Click ☰ next to the load balancer that you want to modify. Select the **Update** option.

   The **Overview** page of the load balancer is displayed.
4. Based on the information that you want to modify, go to the **General** section and make the required changes. See Configuring Load Balancer General Properties.
5. Click **Update**.
6. A **Update Load Balancer** confirmation dialog box is displayed. Click **Ok**.

> **Note:**
>
> - If your load balancer was created by Oracle PaaS Service Manager (PSM) then certain parameters of the resources (load balancer, listener, server pool, etc) cannot be modified after the resource creation.
> - If your load balancer is used by Oracle PaaS Service Manager (PSM) then it is recommended that you do not modify any of the load balancer parameters as the load balancer may become inaccessible.

Deleting a Load Balancer

To delete a load balancer:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Click ≣ next to the load balancer that you want to delete and select the **Delete** option.
4. A **Delete Load Balancer** confirmation dialog box is displayed. Click **Yes**.

> **Note:**
>
> If your load balancer was created by Oracle PaaS Service Manager (PSM) then you cannot delete the load balancer directly. When the provisioned service is deleted, the load balancer will be deleted by Oracle PaaS Service Manager (PSM).

Managing Server Pools for a Load Balancer

The Server Pools page enables you to view, create, modify and delete the server pools for the selected load balancer.

Topics

- Viewing Server Pools
- Creating Server Pools for a Load Balancer
- Updating a Server Pool
- Deleting a Server Pool

Viewing Server Pools

To view server pools for the selected load balancer:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Go to the load balancer whose server pools you want to view and click .

   The **Overview** page of the load balancer is displayed.
4. Go to the **Server Pools** option in the left pane.

   The existing server pools are displayed.
5. Go to the server pool that you want to view and click .

   The **Update Server Pool** window with details of the selected server pool is displayed.
6. Use the **Update Server Pool** window to view and modify parameters for the selected server pool. See Creating Server Pools for a Load Balancer.

Updating a Server Pool

To modify the details of a server pool:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Go to the load balancer whose server pool you want to update and click .

   The **Overview** page of the load balancer is displayed.
4. Go to the **Server Pools** option in the left pane.

   The existing server pools are displayed.
5. Click ☰ next to the server pool that you want to modify. Select the **Update** option.

   The **Update Server Pool** window with details of the selected server pool is displayed.
6. Use the **Update Server Pool** window to view and modify parameters for the selected server pool. See Creating Server Pools for a Load Balancer.
7. Click **Update**.

> **Note:**
>
> If your load balancer was created by Oracle PaaS Service Manager (PSM) then certain parameters of the resources (load balancer, listener, server pool, etc) cannot be modified after the resource creation.

Deleting a Server Pool

To delete a server pool:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Go to the load balancer whose server pools you want to delete and click .

   The **Overview** page of the load balancer is displayed.
4. Go to the **Server Pools** option in the left pane.

   The existing server pools are displayed.

5. Click ≣ next to the server pool that you want to delete and select the **Delete** option.
6. A **Delete Server Pool** confirmation dialog box is displayed. Click **Ok**.

> **Note:**
>
> If your load balancer was created by Oracle PaaS Service Manager (PSM) then the server pools associated with your load balancer cannot be deleted.

Managing Load Balancer Digital Certificates

The Digital Certificates page enables you to view, import and delete digital certificates, which are required if you plan to configure your load balancers for SSL.

Topics

- Viewing a Digital Certificate
- Importing a Load Balancer Digital Certificate
- Updating a Digital Certificate
- Deleting a Digital Certificate

Viewing a Digital Certificate

To view a digital certificate:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the **Digital Certificates** option.

   The existing digital certificates are displayed.

   If you created a new digital certificate recently and it is not appearing in the **Digital Certificates** page, click ⟳ to refresh the list of digital certificates.
3. Go to the digital certificate that you want to view and click 🏅.

   Details of the selected digital certificate are displayed.
4. Click Cancel to return to the **Digital Certificates** page

   .

Updating a Digital Certificate

You can only view, import and delete digital certificates. Updating a digital certificate is not supported.

Deleting a Digital Certificate

To delete a digital certificate:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.

2. Click **Load Balancers** in the left pane and select the **Digital Certificates** option.

   The existing digital certificates are displayed.

   If you created a new digital certificate recently and it is not appearing in the **Digital Certificates** page, click ⟳ to refresh the list of digital certificates.

3. Click ≡ next to the digital certificate that you want to delete and select the **Delete** option.
4. A **Digital Certificate Deletion** confirmation dialog box is displayed. Click **Ok**.

Managing Listeners for a Load Balancer

The Listeners page enables you to view, create, modify and delete listeners for the selected load balancer.

Topics

- Viewing Listeners
- Creating Listeners for a Load Balancer
- Updating a Listener
- Deleting a Listener

Viewing Listeners

To view listeners for the selected load balancer:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Go to the load balancer whose listeners you want to view and click .

   The **Overview** page of the load balancer is displayed.

4. Go to the **Listeners** option in the left pane.

   The existing listeners are displayed.

5. Go to the listener that you want to view and click .

   The **Update Listener** window with details of the selected listener is displayed.

6. Use the **Update Listener** window to view and modify parameters for the selected listener. See Creating Listeners for a Load Balancer

Updating a Listener

To modify the details of a listener:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.

2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Go to the load balancer whose listener you want to update and click ⊞.

   The **Overview** page of the load balancer is displayed.

4. Go to the **Listeners** option in the left pane.

   The existing listeners are displayed.

5. Click ☰ next to the listener that you want to modify. Select the **Update** option.

   The **Update Listener** window with details of the selected listener is displayed.

6. Use the **Update Listener** window to view and modify parameters for the selected listener. See Creating Listeners for a Load Balancer

7. Click **Update**.

> **Note:**
>
> If your load balancer was created by Oracle PaaS Service Manager (PSM) then certain parameters of the resources (load balancer, listener, server pool, etc) cannot be modified after the resource creation.

Deleting a Listener

To delete a listener:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.

2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Go to the load balancer whose listeners you want to view and click ⊞.

   The **Overview** page of the load balancer is displayed.

4. Go to the **Listeners** option in the left pane.

   The existing listeners are displayed.

5. Click ☰ next to the listener that you want to delete and select the **Delete** option.

6. A **Delete Listener** confirmation dialog box is displayed. Click **Ok**.

> **Note:**
>
> If your load balancer was created by Oracle PaaS Service Manager (PSM) then the listeners associated with your load balancer cannot be deleted.

Managing Policies for a Load Balancer

The Policies page enables you to view, create, modify, and delete policies assigned to the selected load balancer.

Topics

- Viewing Policies
- Creating Policies for a Load Balancer
- Updating a Policy
- Deleting a Policy

Viewing Policies

To view policies for the selected load balancer:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Go to the load balancer whose policies you want to view and click .

   The **Overview** page of the load balancer is displayed.
4. Go to the **Policies** option in the left pane.

   The existing policies are displayed.
5. Go to the policy that you want to view and click .

   The **Update Policy** window with details of the selected policy is displayed.
6. Use the **Update Policy** window to view and modify parameters for the selected policy. See Creating Policies for a Load Balancer

Updating a Policy

To modify the details of a policy:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.

3. Go to the load balancer whose policy you want to update and click ⊞.

   The **Overview** page of the load balancer is displayed.
4. Go to the **Policies** option in the left pane.

   The existing policies are displayed.
5. Click ≡ next to the policy that you want to modify. Select the **Update** option.

   The **Update Policy** window with details of the selected policy is displayed.
6. Use the **Update Policy** window to view and modify parameters for the selected policy. See Creating Policies for a Load Balancer.
7. Click **Update**.

Deleting a Policy

To delete a policy:

1. Go to the **Network** tab in the Oracle Compute Cloud Service console.
2. Click **Load Balancers** in the left pane and select the Load Balancers option.

   The existing load balancers are displayed.

   If you created a new load balancer recently and it is not appearing in the **Load Balancers** tab, click ⟳ to refresh the list of load balancers.
3. Go to the load balancer whose policy you want to delete and click ⊞.

   The **Overview** page of the load balancer is displayed.
4. Go to the **Policies** option in the left pane.

   The existing policies are displayed.
5. Click ≡ next to the policy that you want to delete. Select the **Delete** option.
6. A **Delete Policy** confirmation dialog box is displayed. Click **Ok**.

4

Use Cases for Oracle Cloud Infrastructure Load Balancing Classic

To learn more about Oracle Cloud Infrastructure Load Balancing Classic, consider some specific use cases that demonstrate what you can do when you add a load balancer to your Oracle Cloud environment.

Topics

- Load Balancing HTTP Requests to a Pool of Web Servers
- Securing Incoming Requests to Your Load Balancer
- Securing Connections to the Origin Servers

Load Balancing HTTP Requests to a Pool of Web Servers

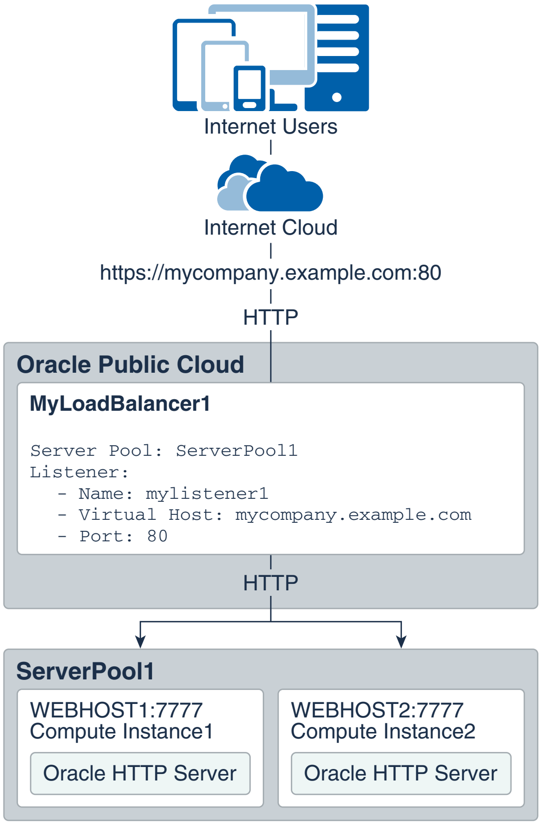
One of the typical use cases for a load balancer is to distribute HTTP requests to a pool of Web servers. The load balancer not only balances the load on the Web servers, but also provides high availability, by routing requests to the remaining Web servers if one or more of the servers is not available.

Topics

- Use Case Topology
- Use Case Prerequisites
- Summary of the Use Case Steps

Use Case Topology

The following topology diagram shows how a load balancer can route HTTP requests to a pool of Web servers in your Oracle Public Cloud environment.



Use Case Prerequisites

This use case tutorial assumes the following:

- You have subscribed to the Compute service.
- You have two existing compute instances (WEBHOST1 and WEBHOST2), each running Oracle HTTP Server.
- You have a custom domain name, defined and managed by a DNS provider. For example:

  ```
  http://www.myCompany.example.com
  ```

  This URL will be used to access the services Web sites and applications that are running on the Web servers.
- On each Web server host, the main page of your Web site or application is available at the following URL:

  ```
  http://host:7777/
  ```

  Before you begin, it is important that you are able to successfully connect to this URL on each origin server in the server pool. This step confirms that you have configured the required security list, security rules, and security applications to allow HTTP access to the Compute instances over port 7777. For more information about configuring access to a Compute instance, see Configuring the Shared Network in *Using Oracle Cloud Infrastructure Compute Classic*.

> **Note:**
>
> In this use case you are not configuring the load balancer for HTTPS connections or SSL/TLS. This means you have no requirement for secure communications between the load balancer and the clients who are submitting the requests.
>
> Similarly, this use case assumes you are not configuring security for the connections between the load balancer and the origin servers that you are load balancing.

Summary of the Use Case Steps

This use case comprises the following high-level steps.

Tutorial

Task	Description	More Information
Understand the prerequisites	This workflow assumes you have already created twoOracle Cloud Infrastructure Compute Classic instances, and you have installed Oracle HTTP Server on each instance.	Use Case Prerequisites
Create the load balancer	Provide a name and basic properties for the load balancer. When you complete this step, the new load balancer appears on the Balancers page in the Compute console.	Creating a Load Balancer
Create a server pool for the load balancer	Create a new server pool and assign the two Compute instances to the pool. For each host, identify the host name and port of the Web server instance.	Creating Server Pools for a Load Balancer

Task	Description	More Information
Create a listener for the load balancer	Create a listener for the load balancer. Set the following properties of the listener: • Name: `myListener1` • Port: `80` • Balancer Protocol: **HTTP** • Server Protocol: **HTTP** • Server Pool: **serverPool1** • Virtual Hosts: Enter the canonical host name of the server. • Enabled: Select this check box	Creating Listeners for a Load Balancer About the Load Balancer IP Addresses and Canonical Host Name
Add the IP addresses of the load balancer to the Security IP list you created for the Compute instances in the server pool.	The security IP list identifies the IP addresses that can access the Compute instances. You should have already configured your IP network so HTTP requests can be received by the Compute instances, but this step ensures the load balancer IP is recognized by the Compute instances.	Adding the Load Balancer IP Addresses to the IP Security List
Verify the load balancer using the canonical host name	In a browser window, enter the URL to the canonical host name of the load balancer. The request should be routed successfully to the application or Web server running on your origin servers.	Verifying a Load Balancer Configuration
Map your custom domain name to the canonical host name	Work with your DNS or domain provider to create a C record, which will map your custom domain name to the canonical host name of the load balancer. After you map your custom domain name to the canonical host name, modify the appropriate listener to include your custom domain name in the list of virtual hosts associated with the listener.	About the Load Balancer IP Addresses and Canonical Host Name Updating a Listener
Verify the load balancer using your custom domain name URL	In a browser window, enter the URL to your custom domain name. Verify that the request is routed to your application URL on the origin servers successfully.	Verifying a Load Balancer Configuration

Securing Incoming Requests to Your Load Balancer

When you configure a load balancer for your Oracle Public Cloud environment, you can configure the load balancer to use a digital certificate and the HTTPS (Secure HTTP) protocol to secure all incoming requests. This is also referred to as **SSL/TLS offloading**, because the load balancer performs the operations required to encrypt and decrypt requests sent over the Secure Socket Layer (SSL) and Transport Layer Security (TLS) protocols. Without a load balancer to perform these tasks, your Web servers or application servers need to perform these tasks.

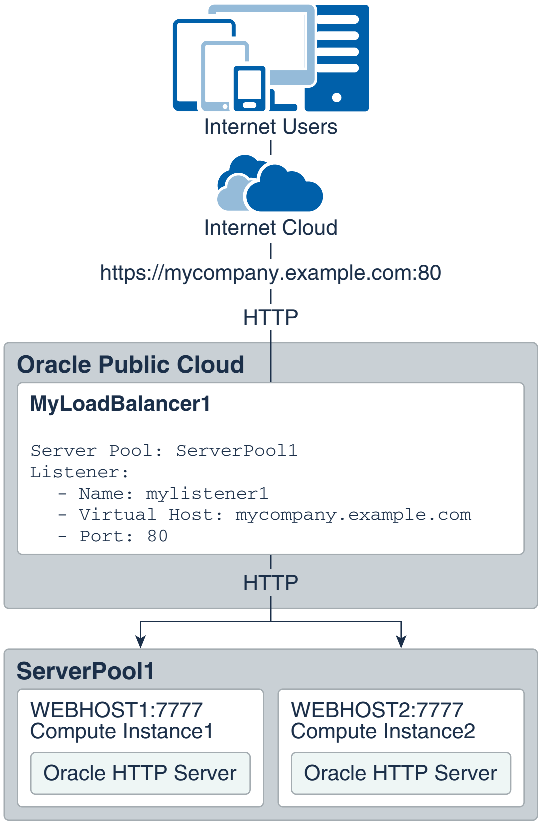
SSL/TLS offloading also makes it easier to maintain and replace security certificates, which typically have regular expiration dates. All certificates are uploaded and stored as part of the load balancer configuration, and there is no need to update them on the individual origin servers in the server pool.

Topics

- Use Case Topology
- Use Case Prerequites
- Summary of the Use Case Steps

Use Case Topology

The following diagram represents a simple load balancer topology where the load balancer secures the client connections, processes the secure connections using a certificate, and passes the requests to the server pool using the standard HTTP protocol.



Use Case Prerequites

This task assumes you have already configured a functioning load balancer that is directing traffic to an existing server pool. For example, you should already have a topology similar to the one described in Load Balancing HTTP Requests to a Pool of Web Servers.

Summary of the Use Case Steps

This use case comprises the following high-level steps.

Tutorial

Task	Description	More Information
Understand the prerequisites	This task assumes you have already configured a functioning load balancer that is directing traffic to an existing server pool. For example, you should already have a topology similar to the one described in Load Balancing HTTP Requests to a Pool of Web Servers.	Use Case Prerequisites
Obtain a digital certificate from a Certificate Authority.	You obtain a CA-issued certificate by submitting a Certificate Signing Request frome a CA vendor.	Obtaining a CA-Issued Certificate
Upload the certificate as a **server certificate** to the Oracle Public Cloud.	For this step, you use the Compute Console to import the certificate, so you can reference it from the load balancer listeners you create in the Console.	Importing a Load Balancer Digital Certificate
Configure the load balancer listener, so it listens on the HTTPS protocol and references the server certificate.	The listener determines the port and specific address that the load balancer will use to listen for new HTTP requests. The listener is also used to indicate that you want the load balancer to receive only secure requests over the HTTPS protocol. When you select the HTTPS protocol, you must reference the digital certificate that you previously imported.	Creating Listeners for a Load Balancer
Verify the secure connection to your application.	After you finish completing the SSL/TLS configuration, verify the connection to be sure it working correctly.	Verifying a Load Balancer Configuration

Securing Connections to the Origin Servers

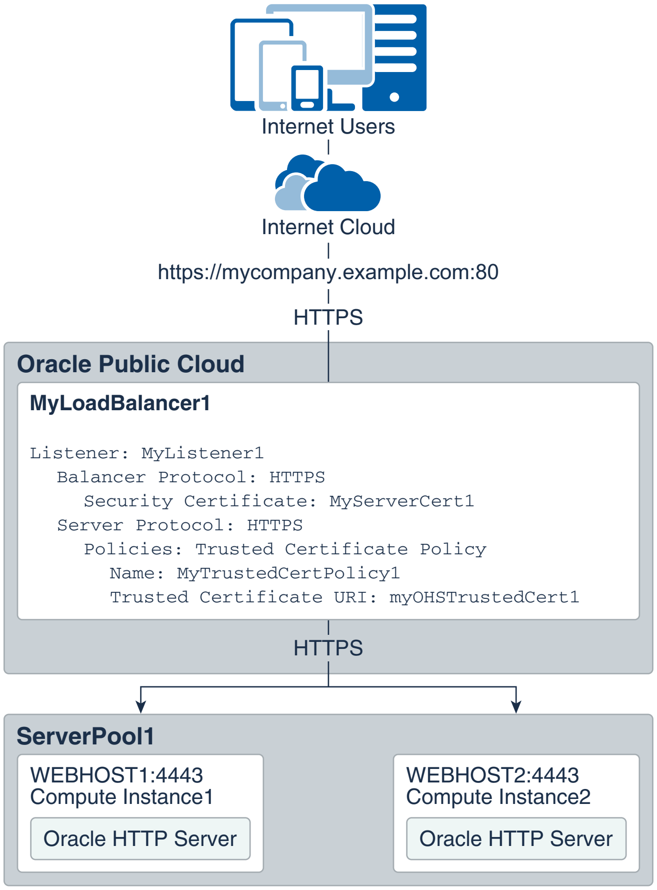
In addition to securing incoming requests to the load balancer, you can also secure the connections between the load balancer and the Web servers, application servers, and applications that are running on the origin servers in the server pool.

Topics

- Use Case Topology
- Use Case Prerequites
- Summary of the Use Case Steps

Use Case Topology

The following diagram represents a simple load balancer topology, where the load balancer secures the client connections, processes the secure connections using a certificate, and passes the requests to the server pool using the standard HTTPS protocol.



```
Listener: MyListener1
  Balancer Protocol: HTTPS
    Security Certificate: MyServerCert1
  Server Protocol: HTTPS
    Policies: Trusted Certificate Policy
      Name: MyTrustedCertPolicy1
      Trusted Certificate URI: myOHSTrustedCert1
```

Use Case Prerequites

This task assumes you have already configured a functioning load balancer that is directing traffic to an existing server pool. For example, you should already have a topology similar to the one described in Load Balancing HTTP Requests to a Pool of Web Servers

Summary of the Use Case Steps

This use case comprises the following high-level steps.

Tutorial

Task	Description	More Information
Understand the prerequisites	This task assumes you have already configured a functioning load balancer that is directing traffic to an existing server pool. For example, you should already have a topology similar to the one described in Load Balancing HTTP Requests to a Pool of Web Servers.	Use Case Prerequisites
Obtain a digital certificate from a Certificate Authority (or create a self-signed certificate).	You obtain a CA-issued certificate by submitting a Certificate Signing Request frome a CA vendor.	Obtaining a Digital Certificate
Upload the certificate as a **trusted certificate** to the Oracle Public Cloud.	For this step, you use the Compute Console to import the certificate, so you can reference it from the load balancer listeners you create in the Console.	Importing a Load Balancer Digital Certificate
Create a **trusted certificate policy** for the load balancer.	After you create a load balancer, you can add policies to the load balancer.	Creating Policies for a Load Balancer
Configure the load balancer listener, so the server protocol is set to HTTPS.	The listener determines the port and specific address that the load balancer will use to listen for new HTTP requests. The listener is also used to indicate that you want the load balancer to receive only secure requests over the HTTPS protocol. When you select the HTTPS protocol, you must reference the digital certificate that you previously imported.	Creating Listeners for a Load Balancer
Verify the secure connection to your application.	After you finish completing the SSL/TLS configuration, verify the connection to be sure it working correctly.	Verifying a Load Balancer Configuration