ORACLE University

Oracle Cloud Infrastructure Foundations Associate: Hands-on Workshop

Activity Guide – OU Tenancy

D1103971GC20

Learn more from Oracle University at ***education.oracle.com***

Table of Contents

Networking—Virtual Cloud Network: Create and Configure a Virtual Cloud Network

Lab 2-1 Practices

Get Started

Overview

In this practice, you will configure and deploy a Virtual Cloud Network (VCN).

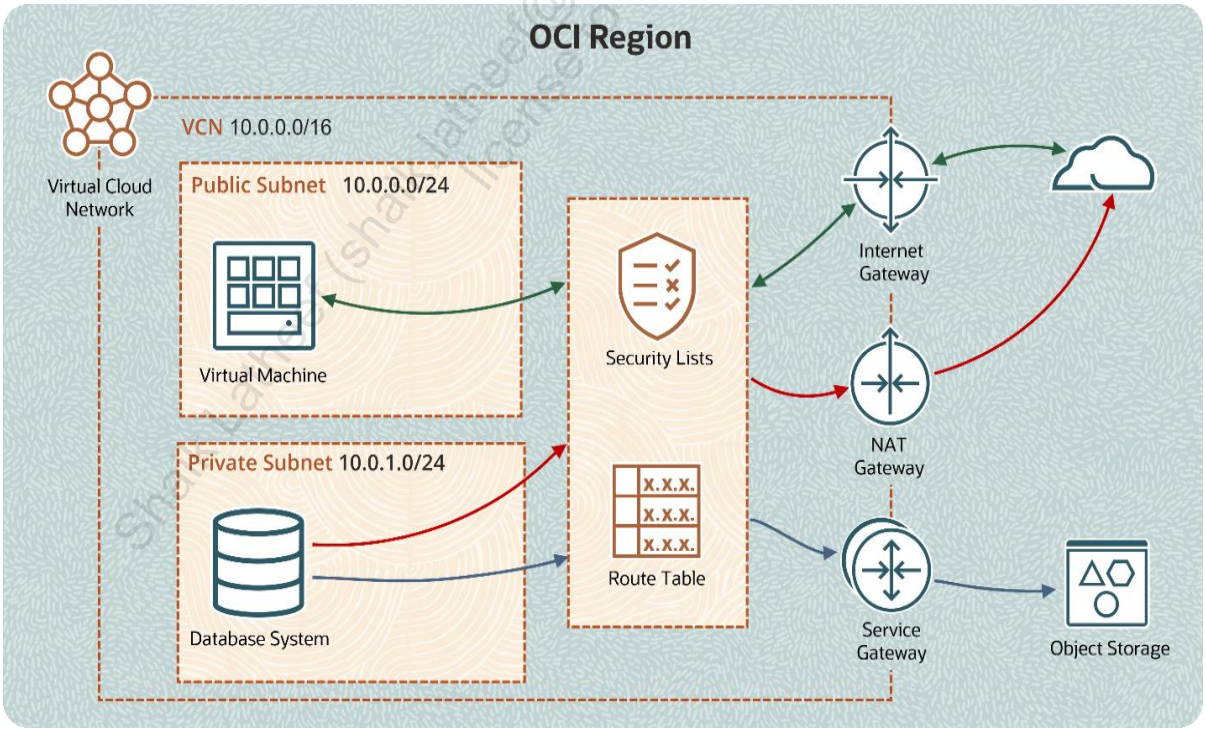
A VCN is a software-defined network specific to your OCI tenancy or a compartment in a specified region.

Upon creation, a VCN automatically includes route tables, security lists (with default security rules), and a set of DHCP options. The VCN also has access to a DNS resolver.

A VCN that is launched with the OCI VCN Wizard tool automatically creates the following:

- Public and Private Subnets
- Internet Gateway (IG)
- NAT Gateway (NAT)
- Service Gateway (SG)
- Two Route Tables (RT)
- Two Security Lists (SL)
- One CIDR Blocks/Prefixes
- One DHCP Option

For more information about VCNs, see the [OCI Networking Documentation](): https://docs.oracle.com/en-us/iaas/Content/Network/Concepts/landing.htm



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Create a Virtual Cloud Network

In this lab, you will create a VCN and associated resources by using the VCN Wizard.

Steps

1. Log in to the OCI Console.
2. In the Console ribbon at the top of the screen, click the **Region** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.
3. Click the Main Menu, click Networking, and then click Virtual Cloud Networks.
4. Click Start VCN Wizard.
5. Select the Create VCN with Internet Connectivity option, and then click Start VCN Wizard.
6. Enter the following values:

   Name: `IAD-FA-LAB02-VCN-01`

   Compartment: Select your <*assigned compartment*>.
7. Leave the default values for the remaining fields. Click **Next**.
8. Review and understand the list of resources that the OCI VCN Wizard will create. Notice that the wizard will configure CIDR block ranges for VCN IP addresses, and for the public and private subnets. It will also set up security list rules and route table rules to enable basic access to the VCN.
9. Click **Create** to start creating the VCN and its resources and wait for the VCN Wizard to successfully complete the VCN creation.
10. Click **View Virtual Cloud Network** to verify the creation of the VCN and its resources.

You can see that the VCN is successfully created with the following components:

- VCN
- Public Subnet
- Private Subnet
- CIDR Blocks/Prefixes
- Route Tables
- Internet Gateway
- Security Lists
- DHCP Options
- NAT Gateway
- Service Gateway

Networking: OCI Load Balancer

Lab 3-1 Practice

Get Started

Overview

In this practice, you will configure a Public Load Balancer, including a set of two backend compute instances.

Load Balancer

The OCI Load Balancer provides automated traffic distribution from one entry point to multiple backend servers in your VCN. It operates at the connection level and balances incoming client connections to healthy backend servers. The service offers a load balancer with your choice of a regional public or private IP address and provisioned bandwidth.

Summary of Components for OCI Load Balancer Used in This Lab

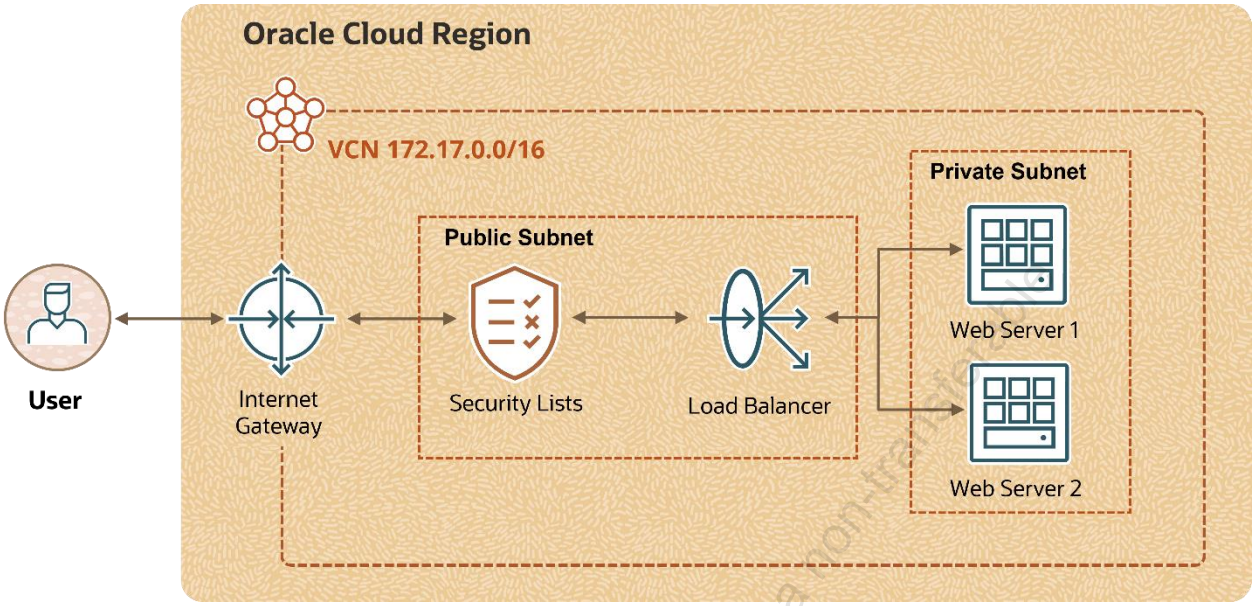
- **Listener:** A logical entity that checks for incoming traffic on the load balancer's IP address
- **Backend server:** An application server responsible for generating content in reply to the incoming traffic
- **Backend set:** A logical entity defined by a list of backend servers
- **Load-balancing policy:** Tells the load balancer how to distribute incoming traffic to the backend servers
- **Health check:** A test to confirm the availability of backend servers
- **Shape:** The bandwidth capacity of the load balancer

In this lab, you will:

a. Create a Virtual Cloud Network

b. Create two compute instances

c. Create a load balancer



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Create a Virtual Cloud Network

In this practice, you will create a VCN and associated resources using the VCN Wizard.

Tasks

1. In the Console ribbon at the top of the screen, click the **Regions** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

2. From the **Main Menu**, select **Networking**, and then click **Virtual Cloud Networks**.

3. Click **Start VCN Wizard**.

4. Select the **Create VCN with Internet Connectivity** option, and then click **Start VCN Wizard.**

5. Enter the following values:
   - **Name:** `IAD-FA-LAB03-VCN-01`
   - **Compartment:** Select your assigned *<compartment name>*.
   - **VCN CIDR Block:** `172.17.0.0/16`
   - **Public Subnet CIDR Block:** `172.17.0.0/24`
   - **Private Subnet CIDR Block:** `172.17.1.0/24`

6. Leave the default values for the remaining fields. Click **Next**.

7. Review and understand the list of resources that the OCI VCN Wizard will create. Notice that the wizard will configure CIDR block ranges for VCN IP addresses, and for the public and private subnets. It will also set up security list rules and route table rules to enable basic access to the VCN.

8. Click **Create**.

9. When complete, click **View Virtual Cloud Network**.

10. In the left navigation pane, under **Resources**, click **Security Lists**.

11. Select **Default Security List for IAD-FA-LAB03-VCN-01**.

12. Click **Add Ingress Rule.**
    a. For **Source CIDR**, enter `0.0.0.0/0`.
    b. For **Destination Port Range**, enter `80`.
    c. Click **Add Ingress Rules**.

Create Two Compute Instances (Backend Servers)

In this lab, you will create two compute instances and configure them to provide web services. They will serve as the backend servers, and will reside in a private subnet.

Task 1: Build the First Compute Instance

1. In the Console ribbon at the top of the screen, click the **Regions** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

2. From the **Main Menu**, select **Compute**, and then click **Instances**.

3. In the left navigation pane, under **List Scope**, select your assigned <*compartment name*>.

4. Click **Create Instance** and enter the following values:
   - **Name:** `IAD-FA-LAB03-VM-01`
   - **Compartment:** Your assigned <*compartment name*>
   - **Placement**: AD-1
   - **Image:** Oracle Linux 8
   - **Shape:** Click **Change Shape**.
     - **Instance Type:** Virtual Machine
     - **Shape Series**: Ampere
     - **Shape Name:** `VM.Standard.A1.Flex`
     - Leave **Number of OCPU** at one.
     - Leave **Amount of memory (GB)** at six.
     - Click **Select Shape**.
   - **Networking:**
     - **Primary network:** Select existing Virtual Cloud Network.
     - **Virtual Cloud Network in *<assigned compartment>***: `IAD-FA-LAB03-VCN-01`
     - **Subnet in *<assigned compartment>***: Private Subnet-`IAD-FA-LAB03-VCN-01` (regional)
   - **Add SSH Key:** No SSH Keys
   - Click **Show advanced options**.
   - On the **Management** tab, click **Paste cloud-init script** under **Initialization script**.

- Copy and paste the following into the **Cloud-init script** field

  (**Tip:** Copy the below script in a notepad and ensure that the last 2 lines of the script are copied in a single line as a single command):

```
#!/bin/bash -x
iptables -A INPUT -p tcp -m multiport --dports 80,443 -j ACCEPT
yum -y install httpd
systemctl enable httpd.service
systemctl start httpd.service
firewall-offline-cmd --add-service=http
firewall-offline-cmd --add-service=https
systemctl enable firewalld
systemctl restart firewalld
echo Hello World! My name is IAD-FA-LAB03-WS-01>
/var/www/html/index.html
```

  Note: This script configures and enables the compute instance's firewall and httpd processes.

5. Click **Create**.

   Note: The process will take approximately two minutes.

Task 2: Build the Second Compute Instance

1. In the console ribbon at the top of the screen, click the **Regions** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

2. From the **Main Menu**, select **Compute**, and then click **Instances**.

3. In the left navigation pane, under **List Scope**, select your assigned *<compartment name>*.

4. Click **Create Instance** and enter the following values:
   - **Name:** `IAD-FA-LAB03-VM-02`
   - **Compartment:** Your assigned *<compartment name>*
   - **Placement:** AD-1
   - **Image:** Oracle Linux 8
   - **Shape:** Click **Change Shape**
     - **Instance Type:** Virtual Machine
     - **Shape Series:** Ampere

- **Shape Name:** `VM.Standard.A1.Flex`
- Leave **Number of OCPU** at one.
- Leave **Amount of memory (GB)** at six.

- Click **Select Shape**.
- **Networking:**
  - **Primary network:** Select existing Virtual Cloud Network.
  - **Virtual Cloud Network in *<assigned compartment>***: `IAD-FA-LAB03-VCN-01`
  - **Subnet in *<assigned compartment>***: Private Subnet-`IAD-FA-LAB03-VCN-01` (regional)
- **Add SSH Key:** No SSH Keys
- Click **Show advanced options**.
- On the **Management** tab, click **Paste cloud-init script** under **Initialization script**.
- Copy and paste the following into the **Cloud-init script** field

  (**Tip:** Copy the below script in a notepad and ensure that the last 2 lines of the script are copied in a single line as a single command):

```
#!/bin/bash -x
iptables -A INPUT -p tcp -m multiport --dports 80,443 -j ACCEPT
yum -y install httpd
systemctl enable httpd.service
systemctl start httpd.service
firewall-offline-cmd --add-service=http
firewall-offline-cmd --add-service=https
systemctl enable firewalld
systemctl restart firewalld
echo Hello World! My name is IAD-FA-LAB03-WS-02>
/var/www/html/index.html
```

Note: This script configures and enables the compute instance's firewall and httpd processes.

5. Click **Create**.

Note: The process will take approximately two minutes.

Create a Load Balancer

In this lab, you will create a Load Balancer, and configure the listener, the health check, and backend set. You will then add a security rule to the security list of the private subnet.

Tasks

1. From the **Main Menu**, select **Networking**, and then click **Load Balancers**.

2. In the left navigation pane, under **List Scope**, select your assigned *<compartment name>*.

3. Click **Create Load Balancer**.

4. Select **Load Balancer**, click **Create Load Balancer**, and enter the following values:
   - **Load Balancer Name:** `IAD-FA-LAB03-LB-01`
   - **Choose visibility type:** Public
   - **Assign a public IP address:** Ephemeral IP Address
   - In the **Bandwidth** section, under **Shapes**, select **Flexible Shapes**.
   - Under **Choose Networking,** for the **Virtual Cloud Network in *<compartment name>***, select `IAD-FA-LAB03-VCN-01` and for the **Subnet in *<compartment name>***, select Public Subnet-`IAD-FA-LAB03_VCN-01`.
   - Click **Next**.
   - Under **Choose Backends**, select **Weighted Round Robin**.
   - Click **Add Backends**.
   - Select both `IAD-FA-LAB03-VM-01` and `IAD-FA-LAB03-VM-02`.
   - Click **Add Selected Backends**.
   - Leave all values at defaults in the **Specify Health Check Policy** section.

     Note: The default values will add a TCP port 80 rule to the security list for your private subnet.
   - Click **Next**.
   - On the **Configure Listener** page, enter the following values:
     - **Listener Name:** `IAD-FA-LAB03-LISTENER-01`
     - **Specify the type of traffic your listener handles:** HTTP

       Note: The **Specify the port your listener monitors for ingress traffic** value will become 80.
   - Click **Next**.
   - On the **Manage Logging** page, set **Error Logs** to **Not Enabled**.

5. Click **Submit** and wait for the status to become **Active**.

   Note: The process will take approximately three minutes.

6. Verify that the **Backend Set Health** status is **OK**.

7. Locate and copy the Load Balancer's **IP Address**.

8. Paste the copied value into your browser's address bar to visit the site.

9. A webpage stating **Hello World! My name is IAD-FA-LAB03-WS-01** will appear.

10. Reload the page to see the other backend server has provided the message, **Hello World! My name is IAD-FA-LAB03-WS-02**.

Compute: Create a Web Server on an OCI Compute Instance

Lab 4-1 Practices

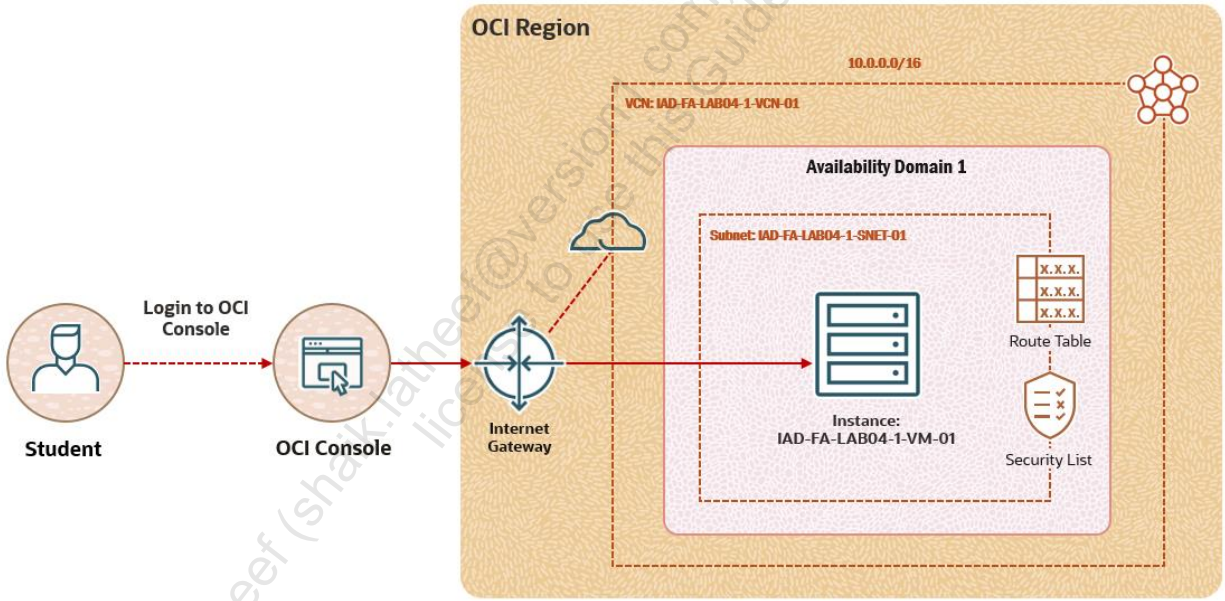
Get Started

Overview

The Oracle Cloud Infrastructure (OCI) Compute lets you provision and manage compute hosts, known as instances. You can launch instances as needed to meet your compute and application requirements. In this lab, you will create a web server on a compute instance.

In this lab, you will:

- Launch Cloud Shell
- Generate SSH keys
- Create a Virtual Cloud Network and its components
- Create a compute instance
- Install an Apache HTTP server on the instance



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Launch Cloud Shell

The OCI Cloud Shell is a web browser–based terminal accessible from the OCI Console. It provides access to a Linux shell, with a pre-authenticated OCI CLI.

In this practice, you will access Cloud Shell via the OCI Console.

Tasks

1. Sign in to your Oracle Cloud Infrastructure Console.

2. In the Console ribbon at the top of the screen, click the Region icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

3. Click the **Cloud Shell** icon next to the Region in the Console ribbon.

   Note: The OCI CLI running in the Cloud Shell will execute commands against the region selected in the Console's region selection menu when the Cloud Shell is started.

   This displays the Cloud Shell in a "drawer" at the bottom of the console.

4. You can use the icons in the top-right corner of the Cloud Shell window to minimize, maximize, and close your Cloud Shell session.

Generate SSH Keys

In this practice, you will generate SSH keys using Cloud Shell.

Tasks

1. From the OCI Console, click the **Cloud Shell** icon next to the region in the Console ribbon.

2. After the Cloud Shell has started, run the following commands:

   ```
   $ mkdir .ssh
   ```

   Important: In case you get an error that says "cannot create director: File exists", you can skip running the first command.

   ```
   $ cd .ssh
   ```

   ```
   $ ssh-keygen -b 2048 -t rsa -f <<sshkeyname>>
   ```

   Replace `<<sshkeyname>>` with **ocifalab4key**. Select the key name you can remember. This will be the key name you will use to connect to the compute instance you create.

   Note: If you receive an error message for the above command, enter the command manually.

   Remember:

   - After entering the third command, press **Enter** twice for no passphrase.
   - Do not include the angle brackets «» and `$` symbol when pasting code into Cloud Shell.

3. Examine the two files that you just created by running the following command:

   ```
   $ ls
   ```

   Note: In the output, there are two files, a private key *<<sshkeyname>>* and a public key *<<sshkeyname>>.pub*. Keep the private key safe and don't share its contents with anyone. The public key will be needed for various activities and can be uploaded to certain systems as well as copied and pasted to facilitate secure communications in the cloud.

4. To list the contents of the public key, run the following command:

   ```
   $ cat <<sshkeyname>>.pub
   ```

   Replace `<<sshkeyname>>` with **ocifalab4key**.

   Note: The angle brackets «» should not appear in your code.

5. Copy the contents of the public key as you will require this in a subsequent step. Make sure that you remove any hard returns that may have been added when copying. The `.pub` key should be one line.

Create a Virtual Cloud Network and Its Components

In this practice, you will create a Virtual Cloud Network (VCN), subnet, and Internet gateway and add route rules in the route table.

Tasks

1. From the **Main Menu**, under **Networking**, click **Virtual Cloud Networks**.
2. Click **Create VCN**.
3. In the **Create a Virtual Cloud Network** dialog box, populate the following information:
   a. **Name:** `IAD-FA-LAB04-1-VCN-01`
   b. **Create in Compartment:** <*your compartment*>
   c. **IPv4 CIDR Blocks:** `10.0.0.0/16` (Press **Enter** to add.)
4. Keep the other options default and click **Create VCN**.

   You can see that the VCN is created successfully.
5. Click **IAD-FA-LAB04-1-VCN-01** VCN to view the details page.
6. Click **Create Subnet**.
7. In the **Create Subnet** dialog box, populate the following information:
   a. **Name:** `IAD-FA-LAB04-1-SNET-01`
   b. **Create in Compartment:** <*your compartment*>
   c. **Subnet Type:** Regional
   d. **IPv4 CIDR Blocks:** `10.0.1.0/24`
   e. **Subnet Access:** Public Subnet
8. Keep the other options default and click **Create Subnet**.

   You can see that the subnet is created successfully, and the state is **Available**.
9. Under **Resources** in the left navigation panel, click **Internet Gateways**.

10. Click **Create Internet Gateway**.

11. In the **Create Internet Gateway** dialog box, populate the following information:

    a. **Name:** `IAD-FA-LAB04-1-IG-01`

    b. **Create In Compartment:** <*your compartment*>

12. Click **Create Internet Gateway**.

    You can see that the Internet gateway is created successfully and the state is **Available**.

13. Under **Resources** in the left navigation panel, click **Route Tables**.

14. Click **Default Route Table** for `IAD-FA-LAB04-1-VCN-01`.

15. Click **Add Route Rules**.

16. In the **Add Route Rules** dialog box, populate the following information:

    a. **Target Type:** Internet Gateway

    b. **Destination CIDR Block:** `0.0.0.0/0`

    c. **Target Internet Gateway:** `IAD-FA-LAB04-1-IG-01`

17. Click **Add Route Rules**.

    You can see that the route rule is successfully added in the default Route Table.

18. Navigate back to the **Virtual Cloud Networks** page from the **Main Menu**.

19. Click **IAD-FA-LAB04-1-VCN-01** VCN to view the details page.

20. Under **Resources** in the left navigation panel, click **Security Lists**.

21. Click **Default Security List** for IAD-FA-LAB04-1-VCN-01.

22. Here, you need to open port 80. Click **Add Ingress Rules**.

23. In the **Add Ingress Rules** dialog box, populate the following information:

    a. **Source Type**: CIDR

    b. **Source CIDR**: 0.0.0.0/0

    c. **IP Protocol**: TCP

    d. **Destination Port Range**: 80

    Note: Do not select the **Stateless** check box. The **Source Port Range** field is set to **All** by default.

24. Click **Add Ingress Rule**.

    You can see that the route rule is successfully added.

Create a Compute Instance

In this practice, you will launch a compute instance and connect to it.

Tasks

1. From the OCI Console **Main Menu**, under **Compute**, click **Instances**.
2. Click **Create instance**.
3. In the **Create compute instance** dialog box, populate the following information:
   a. **Name**: `IAD-FA-LAB04-1-VM-01`
   b. **Create in compartment**: <*your compartment*>
   c. **Placement (Availability domain)**: AD 1

      Click **Show advanced options** and select **On-demand capacity** under Capacity type.
   d. **Image**: Oracle Linux 8
   e. **Shape:** Click **Change Shape** and select the following:
      1) **Instance Type:** Virtual Machine
      2) **Shape Series**: Ampere
      3) **Shape Name:** `VM.Standard.A1.Flex`
      4) Leave **Number of OCPU** at one.
      5) Leave **Amount of memory (GB)** at six.
      6) Click **Select Shape**.
   f. **Primary network:** Select an existing Virtual Cloud Network.
      1) **Virtual cloud network in *<your compartment>*:** `IAD-FA-LAB04-1-VCN-01`
      2) **Subnet:** Select an existing subnet.
      3) **Subnet in *<your compartment>*:** `IAD-FA-LAB04-1-SNET-01` (regional)
      4) **Public IP address:** Assign a public IPv4 address.

g. **Add SSH keys:** Paste public keys.

h. **SSH Keys:** <*public key*> (Paste the public key which you copied in Step 5 of Generate SSH Keys practice.)

Note: Keep the default option for **Boot volume**.

4. Click **Create**.

   You will see that the instance is created successfully, and the state is **Running**.

5. Copy the public IP corresponding to the **IAD-FA-LAB04-1-VM-01** instance and paste it in the Notepad.

6. Click the **Cloud Shell** icon next to Region at the top of the screen.

7. Run the following command using SSH to connect to your instance:

```
$ ssh -i <private_key_file> <username>@<public-ip-address>
```

   a. The `/home/username/.ssh/private_key_file` is the full path and name of the file that contains the private key associated with the instance you want to access.

   b. The `<username>` is the default user `opc`.

   c. The `<public-ip-address>` is the public IP address of the instance.

   Note: Enter **`yes`** in response to "Are you sure you want to continue connecting (yes/no)?"

You are now connected to the instance IAD-FA-LAB04-1-VM-01.

Install an Apache HTTP Server on the Instance

The HTTP server is an open-source web server developed by the Apache Software Foundation. The Apache server hosts web content and responds to requests for this content from web browsers such as Chrome or Firefox.

In this practice, you will install an Apache HTTP web server and connect to it over the public Internet.

Tasks

1. On the OCI Console, click the **Cloud Shell** icon at the top of the screen.

2. While connected to your compute instance via SSH, run the following commands:

   a. Install Apache HTTP:

   ```
   $ sudo yum install httpd -y
   ```

   b. Start the Apache server and configure it to start after system:

   ```
   $ sudo apachectl start

   $ sudo systemctl enable httpd
   ```

   c. Run a quick check on Apache configurations:

   ```
   $ sudo apachectl configtest
   ```

   d. Create firewall rules to allow access to the ports on which the HTTP server listens:

   ```
   $ sudo firewall-cmd --permanent --zone=public --add-
   service=http

   $ sudo firewall-cmd --reload
   ```

   e. Create an index file for your web server.

   ```
   $ sudo bash -c 'echo This is my Web-Server running on Oracle
   Cloud Infrastructure >> /var/www/html/index.html'
   ```

3. Open your browser and enter `http://Public-IPAddress` in the address bar (the IP address of the compute instance).

   You should see the index page of the web server we created in the second step (last point).

```
This is my Web-Server running on Oracle Cloud Infrastructure
```

Object Storage: Create and Manage OCI Object Storage

Lab 5-1 Practices

Get Started

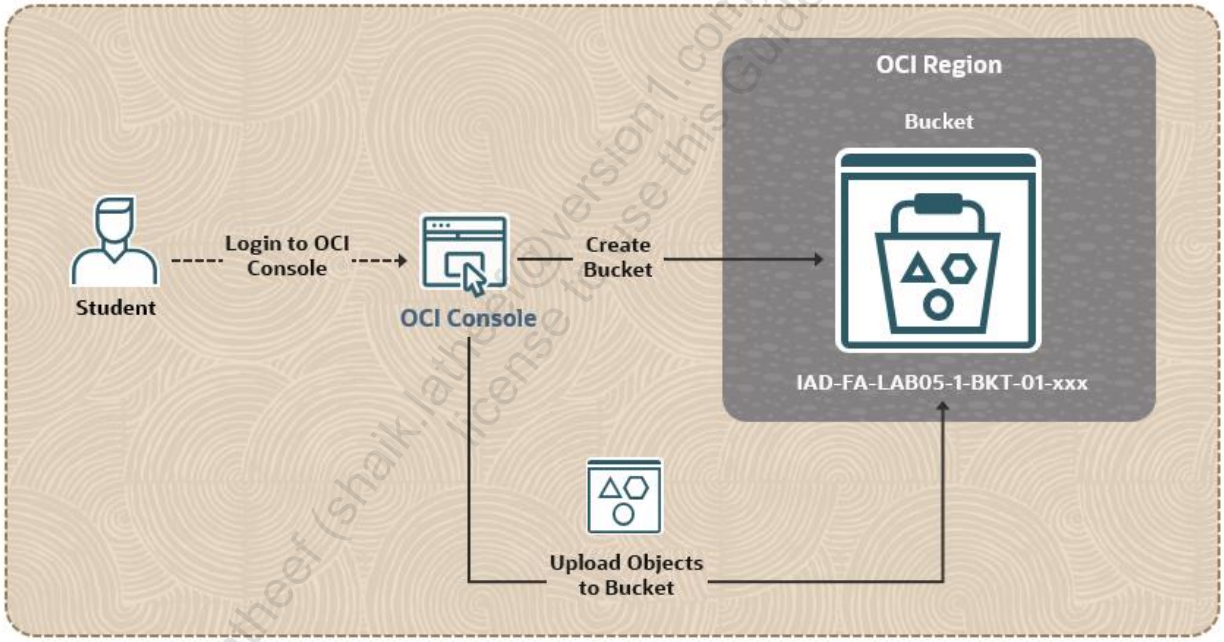
Overview

The Oracle Cloud Infrastructure (OCI) Object Storage provides unlimited capacity with high durability and scalability. It is highly reliable and cost efficient. The object storage resources include namespace, bucket, and object.

Object Storage is characterized by strong consistency and security with encryption. By creating unlimited buckets, you can add as many objects as required with a maximum of 10TiB per object. In this lab, you will work on buckets, object versioning, object life cycle management, replication policy, and retention rule.

In this lab, you will:

- Create an Object Storage bucket
- Upload an object to a bucket



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Create an Object Storage Bucket

In this practice, you will create an Object Storage bucket.

Tasks

1. Sign in to your OCI account.
2. From the **Main Menu**, select **Storage**.
3. Under **Object Storage** and **Archive Storage**, click **Buckets**.
4. From the left navigation panel, select the compartment in which you have permission to work. Then the page updates to display only the resources in that compartment.
5. Click **Create Bucket**.
6. In the **Create Bucket** dialog box, specify the following attributes of the bucket:
   - **Bucket Name:** Enter `IAD-FA-LAB05-1-BKT-01-xxx` as the name for the bucket. Specify a random number in place of xxx to make it unique.
   - **Default Storage Tier:** Select the default tier in which you want to store the data. After it is set, you cannot change the default storage tier of a bucket. When you upload objects, this tier will be selected by default. You can, however, select a different tier. In this case, select **Standard**, which is the primary and default storage tier used for Object Storage.
   - **Enable Auto-Tiering:** Auto-Tiering helps you automatically move objects between Standard and Infrequent Access tiers based on their access patterns. Do not enable this field now.
   - **Enable Object Versioning:** Versioning directs Object Storage to automatically create an object version each time a new object is uploaded, an existing object is overwritten, or when an object is deleted. You can enable it while creating a bucket or later. Do not enable this field now.
   - **Emit Object Events:** Emit Object Events lets the bucket emit events for object state changes. Do not select this field now.
   - **Uncommitted Multipart Uploads Cleanup:** Uncommitted Multipart Uploads Cleanup allows deletion of uncommitted or failed multipart uploads. Do not select this field now.

- **Encryption:** Buckets are encrypted with keys managed by Oracle by default, but you can optionally encrypt the data in this bucket using your own vault encryption key. Select the **Encrypt using Oracle managed keys** option.

- **Tags:** If you have permissions to create a resource, then you also have permissions to apply free-form tags to that resource. Skip this option. You can always apply tags later.

7. Click **Create**.

   The bucket is created immediately, and you can add objects to it.

Upload an Object to a Bucket

In this practice, you will upload an object to your bucket. Object Storage supports uploading individual files up to 10 TiB.

Before you upload an object to a bucket, you must have a bucket. In this case, you will use the bucket that is created from the previous practice.

Tasks

1. In the **Main Menu**, navigate to **Storage**, and then select **Buckets**.
2. Click the bucket **IAD-FA-LAB05-1-BKT-01-*xxx*** to view its details.
3. Under **Objects**, click **Upload**.
4. In the **Object Name Prefix** field, enter the file name prefix `oci/` for the files you plan to upload. This step is optional.
5. The **Storage Tier** field is populated as **Standard**. You can optionally change the storage tier (to Infrequent Access or Archive) to upload objects. In this case, keep it as **Standard**.
6. Select the objects to upload (browse any object from your local machine) by using one of the following options:
   - Drag files from your computer into the **Drop files here...** section.
   - Click the **Select Files** link to display a file selection dialog box.

     As you select files to upload, they are displayed in a scrolling list. If you decide that you do not want to upload a file that you have selected, click **X** to the right of the file name.

     If selected files to upload and files already stored in the bucket have the same name, warning messages to overwrite are displayed.
7. Click **Upload**.

   The selected objects are uploaded. Click **Close** to return to the bucket.

Block Storage: Create, and Attach a Block Volume

Lab 6-1 Practices

Get Started

Overview

The Oracle Cloud Infrastructure (OCI) Block Volume service lets you dynamically provision and manage block storage volumes. You can create, attach, connect, and move volumes, as well as change volume performance, as needed, to meet your storage, performance, and application requirements.

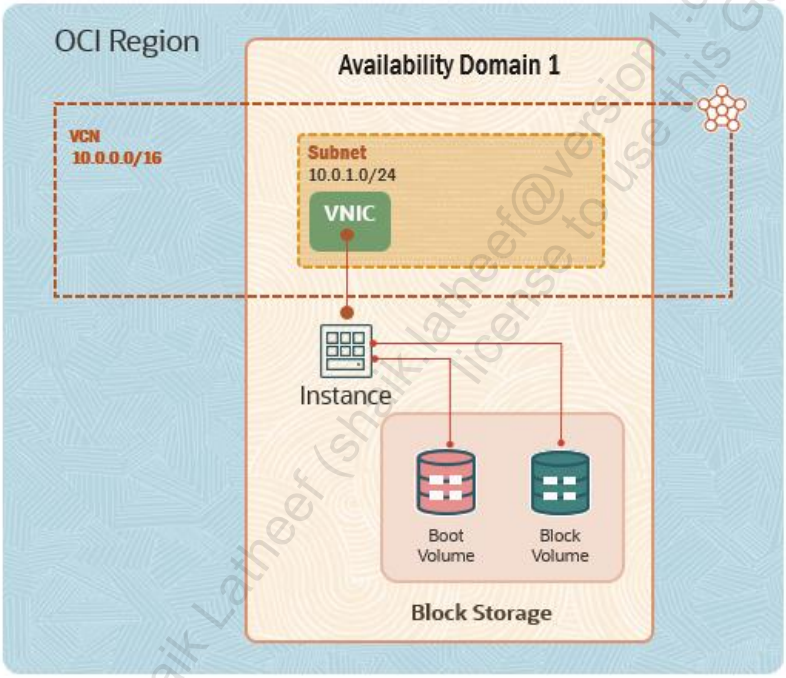
In this lab, you will:

a. Create a Virtual Cloud Network and its components

b. Create a VM instance

c. Create a block volume

d. Attach a block volume to a compute instance



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Create a Virtual Cloud Network and Its Components

In this practice, you will learn how to create a Virtual Cloud Network (VCN), subnet, and Internet gateway, and add route rules in the Route Table.

Tasks

1. Sign in to the OCI Console.
2. In the Console ribbon at the top of the screen, click the **Region** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.
3. From the **Main Menu**, select **Networking**, and then click **Virtual Cloud Networks**.
4. Click **Create VCN.**
5. Enter the following:
   a. **Name:** Enter `IAD-FA-LAB06-1-VCN-01`.
   b. **Create in Compartment:** Select the <*compartment name*> assigned to you.
   c. **IPv4 CIDR Blocks:** Type `10.0.0.0/16` and press **Enter**.

   Note: You can leave all the other options as default.
6. Click **Create VCN**. The VCN is now created successfully.
7. Click **Create Subnet**.
8. In the Create Subnet dialog box, do the following:
   a. **Name:** Enter `IAD-FA-LAB06-1-SNET-01`.
   b. **Create in Compartment:** Select the <*compartment name*> assigned to you.
   c. **Subnet Type:** Select **Regional**.
   d. **IPv4 CIDR Blocks:** Enter `10.0.1.0/24`.
   e. **Subnet Access:** Select **Public Subnet**.

   Note: You can leave all the other options as default.
9. Click **Create Subnet**. The subnet is now created successfully, and the state is **Available**.

10. In the left navigation pane, under **Resources**, click **Internet Gateways**.

11. Click **Create Internet Gateway**.

12. Do the following:

    a. **Name:** Enter `IAD-FA-LAB06-1-IG-01`.

    b. **Create in Compartment:** Select the <*compartment name*> assigned to you.

13. Click **Create Internet Gateway.** The Internet gateway is now created successfully, and the state is **Available**.

14. In the left navigation pane, under **Resources**, click **Route Tables**.

15. Click **Default Route Table for IAD-FA-LAB06-1-VCN-01**.

16. Click **Add Route Rules** and do the following:

    a. **Target Type:** Select **Internet Gateway** from the drop-down list.

    b. **Destination CIDR Block:** Enter `0.0.0.0/0`.

    c. **Target Internet Gateway:** Select **IAD-FA-LAB06-1-IG-01** from the drop-down list.

17. Click **Add Route Rules**. The route rule is now successfully added to the default Route Table.

Create a VM Instance

In this practice, you will learn how to create SSH keys using Cloud Shell and how to launch an instance.

Tasks

1. Sign in to the OCI Console.

2. In the Console ribbon at the top of the screen, click the **Regions** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

3. In the Console ribbon at the top of the screen, click the **Cloud Shell** icon next to the Region selection menu.

4. Once the Cloud Shell is ready, enter the following commands:

```
$ mkdir .ssh
```

   - **Important:** In case you get an error "Cannot create directory: File exists," you can skip running this first command.

```
$ cd .ssh
```

```
$ ssh-keygen -b 2048 -t rsa -f <<sshkeyname>>
```

   - **Remember:** After entering this third command, press **Enter** twice for no passphrase.

   Note: Replace `<<sshkeyname>>` with **ocifalab6key**. Choose a key name you can remember. This will be the key name you will use to connect to the compute instance you create.

   Reminder: The angle brackets «» should not appear in your code.

   Reminder: Do not include the `$` symbol when pasting code into Cloud Shell.

5. Examine the two files that you just created by running the following command:

```
$ ls
```

   Note: In the output, there are two files, a private key *<<sshkeyname>>* and a public key *<<sshkeyname>>.pub*. Keep the private key safe and don't share its contents with anyone. The public key will be needed for various activities and can be uploaded to certain systems, as well as copied and pasted to facilitate secure communications in the cloud.

6. To list the contents of the public key, use the following command:

   ```
   $ cat <<sshkeyname>>.pub
   ```

   Note: Replace `<<sshkeyname>>` with **ocifalab6key**.

   Reminder: The angle brackets «» should not appear in your code.

7. Copy the contents of the public key as you will need this in a subsequent step. Make sure that you remove any hard returns that may have been added when copying. The `.pub` key should be one line.

8. From the **Main Menu**, select **Compute**. Under **Compute**, click **Instances**.

9. Click **Create instance** and do the following:

   a. **Name:** Enter `IAD-FA-LAB06-1-VM-01`.

   b. **Create in compartment:** Select the <*compartment name*> assigned to you.

   c. **Placement:** Select Availability Domain **AD1**. Click **Show advanced options** and select **On-demand capacity** from the **Capacity type** menu.

   d. **Image:** Select Oracle Linux 8.

   e. **Shape:** Click **Change Shape** and select the following:

      1) **Instance Type:** Virtual Machine

      2) **Shape Series**: Ampere

      3) **Shape Name:** `VM.Standard.A1.Flex`

      4) Leave **Number of OCPU** at one.

      5) Leave **Amount of memory (GB)** at six.

      6) Click **Select Shape**.

   f. **Networking:** Select the existing VCN **IAD-FA-LAB06-1-VCN-01** and existing subnet **IAD-FA-LAB06-1-SNET-01 (regional)**. Under **Public IP address**, select **Assign a public IPv4 address**.

   g. **Add SSH keys:** Select **Paste public keys** and paste the contents of the public key, which you copied in Step 6, in the box.

h. **Boot volume:** Keep the default selection.

10. Click **Create**.

    Note: After a couple of minutes, you see that the instance is successfully created, and the state is **Running**.

11. Under **Instance access**, copy the **Public IP address**.

12. Click the **Cloud Shell** icon to open Cloud Shell, and use SSH to connect to your instance by using the following command:

    Note: Enter **`yes`** in response to "Are you sure you want to continue connecting (yes/no)?"

    ```
    $ ssh -i <private_key_file> <username>@<public-ip-address>
    ```

    Reminders:

    - `/home/username/.ssh/private_key_file` is the full path and name of the file that contains the private key associated with the instance you want to access.
    - `<username>` is the default user **opc**.
    - `<public-ip-address>` is the public IP address of the instance.

13. You are now connected to the instance IAD-FA-LAB06-1-VM-01. Run the following command to display information about the block devices:

    ```
    $ lsblk
    ```

    Note: You will only see the boot disk **sda**.

Create a Block Volume

The OCI Block Volume service lets you dynamically provision and manage block storage volumes.

In this practice, you will learn how to create a block volume.

Tasks

1. Sign in to the OCI Console.
2. Open the **Main Menu** and click **Storage**. Under **Block Storage**, click **Block Volumes**.
3. Click **Create Block Volume.**
4. Fill in the required volume information:
    a. **Name:** Enter `IAD-FA-LAB06-1-BV-01`.
    b. **Create in Compartment:** Select the <*compartment name*> assigned to you.
    c. **Availability Domain:** Select the first availability domain.
    d. **Volume Size and Performance:** Select **Custom** and specify the following:
        1) **Volume Size (in GB):** Enter `50`.
        2) **Target Volume Performance**: Drag the VPUs/GB slider to the left to make the performance **Lower Cost** .
    e. **Backup Policies:** Do not specify any policy.
    f. **Cross Region Replication:** Keep the **OFF** default selection.
    g. **Encryption:** Keep the default **Encrypt using Oracle-managed keys** selection.
5. Click **Create Block Volume**. You now see that the Block Volume state becomes **Available**.

Attach a Block Volume to a Compute Instance

You can create, attach, connect, and move volumes. You can also change volume performance, as needed, to meet your storage, performance, and application requirements. After you attach and connect a volume to an instance, you can use the volume like a regular hard drive.

In this practice, you'll learn how to attach a block volume to a compute instance and perform various configuration tasks on the attached volume.

Tasks

1. Open the **Main Menu** and click **Compute**. Under **Compute**, click **Instances**.

2. In the **Instances** list, click the instance **IAD-FA-LAB06-1-VM-01**.

3. In the left navigation pane, under **Resources**, click **Attached block volumes**.

4. Click **Attach block volume**.

5. Specify the volume you want to attach to. For example, to use the volume name, choose **Select volume**, and then select the volume **IAD-FA-LAB06-1-BV-01** from the **Volume** drop-down list.

6. If the instance supports consistent device paths, and the volume you are attaching is not a boot volume, select the path **/dev/oracleoci/oraclevdb** from the **Device path** drop-down list. This enables you to specify a device path for the volume attachment that remains consistent between instance reboots.

7. In the **Attachment type** section, select **Paravirtualized**.

   Note: After you attach a volume using the Paravirtualized attachment type, it is ready to use, and you do not need to run any additional commands.

8. In the **Access** section, select **Read/Write**.

   Note: This is the default option for volume attachments and, with this option, an instance can read and write data to the volume.

9. Click **Attach**. You now see the state as Attached and, since the attachment type is Paravirtualized, you can use the volume without running any additional commands.

10. Ensure that you are connected to the instance **IAD-FA-LAB06-1-VM-01**.

    Note: For help with this, refer to Step 11 in the **Create a VM Instance** practice.

11. Run the following command to display information about the block devices:

    ```
    $ lsblk
    ```

    Note: You now see that the system recognizes a new disk device, and the size is 50 GB.

12. To verify that the volume is attached to the instance, run the following command:

    ```
    $ ll /dev/oracleoci/oraclevd*
    ```

13. To partition the disk using `fdisk`, run the following command:

    ```
    $ sudo fdisk /dev/oracleoci/oraclevdb
    ```

    Note: Enter the following responses as seen in Cloud Shell:

    a. `Command (m for help)`: Enter `n` to create a new partition.

    b. `Select (default p)`: Enter `p`.

    c. `Partition number (1,4, default 1)`: Press **Enter**.

    d. `First sector`: Press **Enter**.

    e. `Last sector`: Press **Enter**.

    f. `Command (m for help)`: Enter `w` to write the new partition.

14. To format the partition, run the following command:

    ```
    $ sudo mkfs -t ext4 /dev/oracleoci/oraclevdb1
    ```

15. To mount the partition, run the following commands:

    ```
    $ sudo mkdir -p /mnt/volume1
    ```

    ```
    $ sudo mount /dev/oracleoci/oraclevdb1 /mnt/volume1
    ```

    Note: On Linux instances, if you want to automatically mount volumes on an instance boot, you need to set some specific options in the `/etc/fstab` file.

16. To display information about the block devices, run the following command:

```
$ lsblk
```

Note: You now see the partition and the mountpoint `/mnt/volume1`.

AppDev: Create a Reusable VCN Configuration with OCI Resource Manager

Lab 7-1 Practices

Get Started

Overview

There are multiple ways to create a VCN and subnet in the OCI Console. Particularly if you want to launch several VCNs with the same configuration, it is beneficial to use Terraform or OCI Resource Manager to streamline and automate that process.

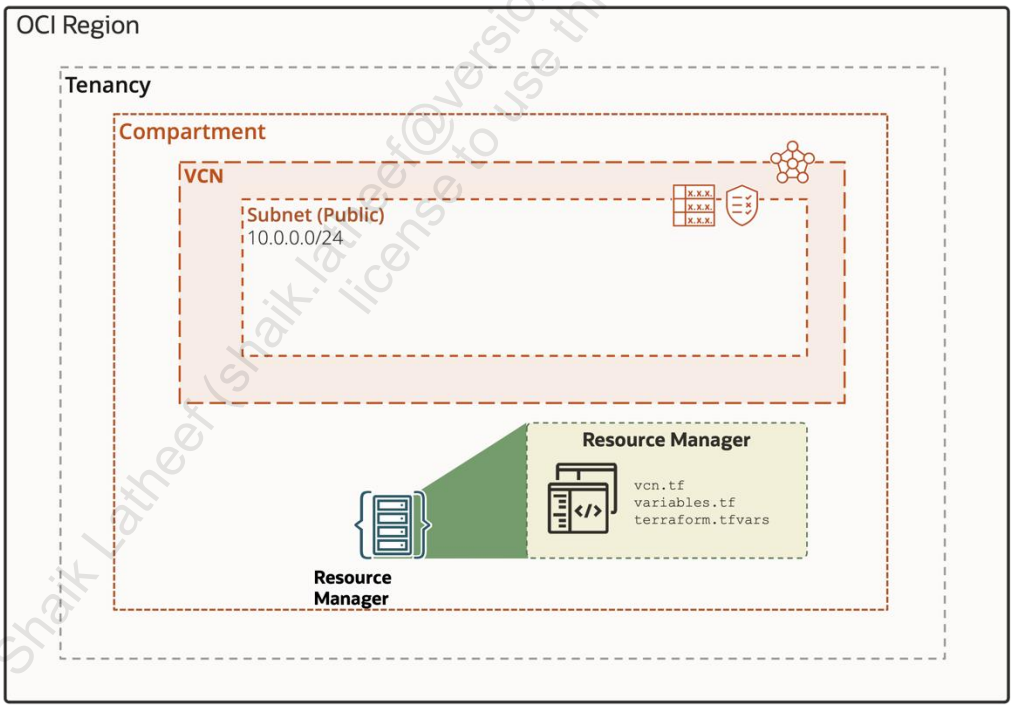
In this lab, you will launch and destroy a VCN and subnet by creating Terraform automation scripts and issuing commands in Code Editor. Next, you will download those Terraform scripts and create a stack by uploading them into the OCI Resource Manager. You will then use that service to launch and destroy the same VCN and subnet.

In this lab, you will:

a. Create a Terraform folder and file in Code Editor

b. Create and destroy a VCN using Terraform

c. Create and destroy a VCN using OCI Resource Manager



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.

- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Create a Terraform Folder and File in Code Editor

In this practice, you will create a folder and file to hold your Terraform scripts.

Task 1: Create a Folder and File

1. Sign in to the OCI Console.

2. In the Console ribbon at the top of the screen, click the **Regions** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

3. Click the **Developer Tools** icon to the right of the **Regions icon**. Click **Code Editor**.

4. Expand the Explorer panel with the top icon on the left panel. It looks like two overlapping documents.

5. Expand the drop-down for your home directory if it is not already expanded. It is okay if it is empty.

6. Create a new folder by clicking **File**, then **New Folder**, and name it `terraform-vcn`. Click **OK**.

7. Create a file in that folder by clicking **File**, then **New File**, and name it `vcn.tf`. To make Code Editor create the file in the correct folder, click the folder name in your home directory to highlight it.

8. First, you will set up Terraform and the OCI Provider in this directory. Add these lines to the file:

```
terraform {
  required_providers {
    oci = {
      source  = "oracle/oci"
      version = ">=4.67.3"
    }
  }
  required_version = ">= 1.0.0"
}
```

9. Save the changes by clicking **File**, then **Save**.

10. Now, run this code. Open a terminal panel in Cloud Editor by clicking **Terminal**, then **New Terminal**.

11. Use `pwd` to check that you are in your home directory.

12. Enter `ls` and you should see your `terraform_vcn` directory.

13. Enter `cd terraform_vcn/` to change to that directory.

14. Enter `terraform init -upgrade` to initialize this directory for Terraform.

15. Use `ls -a` and you should see that Terraform has created a hidden directory and file.

Create and Destroy a VCN Using Terraform

Terraform uses providers to interface between the Terraform engine and the supported cloud platform. The OCI Terraform provider is a component that connects Terraform to the OCI services that you want to manage. In this practice, you will create a Terraform script that will launch a VCN and subnet. You will then alter your script and create two additional files that will apply a compartment OCID variable to your Terraform script.

Task 1: Write the Terraform

1. Open the OCI Provider documentation in the Terraform Registry to familiarize yourself with the OCI Terraform provider. As you go along the lab, it may be helpful to try to find the relevant portions of the documentation.

2. Add the following code block to your Terraform script to declare a VCN, replacing *<your_compartment_ocid>* with the proper OCID. The only *strictly* required parameter is the compartment OCID, but you will add more later.

   If you need to retrieve your compartment OCID, navigate to **Identity & Security**, then **Compartments**. Find your compartment, hover over the OCID, and click **Copy**.

   ```
   resource "oci_core_vcn" "example_vcn" {
     compartment_id = "<your_compartment_ocid>"
   }
   ```

   This snippet declares a resource block of type `oci_core_vcn`. The label that Terraform will use for this resource is `example_vcn`.

3. In the terminal, run `terraform plan`, and you should see that Terraform creates a VCN. Since most of the parameters were unspecified, Terraform will list their values as "(known after apply)." You can ignore the "-out option to save this plan" warning for this lab.

   Note that `terraform plan` parses your Terraform configuration and creates an execution plan for the associated stack, while `terraform apply` applies the execution plan to create (or modify) your resources.

4. Add a display name and CIDR block (the bolded portion) to the code. Note that we want to set the `cidr_blocks` parameter, rather than `cidr_block` (which is deprecated). The region code IAD is used below, for the US East (Ashburn) region. Replace *<your_compartment_ocid>* with the proper OCID.

   ```
   resource "oci_core_vcn" "example_vcn" {
       compartment_id = "<your_compartment_ocid>"
   ```

```
    display_name = "IAD-FA-LAB07-1-VCN-01"
    cidr_blocks = ["10.0.0.0/16"]
}
```

5. Save the changes and run `terraform plan` again. You should see the display name and CIDR block reflected in Terraform's plan.

6. Now add a subnet to this VCN. At the bottom of the file, add the following block:

```
resource "oci_core_subnet" "example_subnet" {
    compartment_id = "<your_compartment_ocid>"
    display_name = "IAD-FA-LAB07-1-SNT-01"
    vcn_id = oci_core_vcn.example_vcn.id
    cidr_block = "10.0.0.0/24"
}
```

7. Note the line where we set the VCN ID. Here we reference the OCID of the previously declared VCN, using the name we gave to Terraform: `example_vcn`. This dependency makes Terraform provision the VCN first, wait for OCI to return the OCID, then provision the subnet. Replace *<your_compartment_ocid>* with the proper OCID.

8. Run `terraform plan` to see that it will now create a VCN and subnet.

Task 2: Provision the VCN

9. Run `terraform apply` and confirm that you want to make the changes by entering **`yes`** at the prompt.

10. Navigate to VCNs in the Console. Ensure that you have the right compartment selected. You should see your VCN. Click its name to see the details. You should see its subnet listed.

Task 3: Terminate the VCN

11. Run `terraform destroy`. Enter **`yes`** to confirm. You should see the VCN terminated. Refresh your browser if needed.

Create and Destroy a VCN Using Resource Manager

You can better manage the infrastructure provisioned through Terraform by migrating to OCI Resource Manager instead of running Terraform locally in Cloud Shell or Code Editor. In this section, we will reuse the Terraform code but replace the CLI with Resource Manager.

Task 1: Create a folder terraform_vcn

1. Create a folder named `terraform_vcn` on your host machine. Download `the vcn.tf` file from Code Editor and move them to the `terraform_vcn` folder to your local machine. To download from Code Editor, right click the file name in the Explorer panel and select **Download**. You could download the whole folder at once, but then you would have to delete Terraform's hidden files.

Task 2: Create a Stack

1. Navigate to Resource Manager in the Console's navigation menu under **Developer Services**. Click **Stacks** under **Resource Manager**.

2. Click **Create stack**.

   a. The first page of the form will be for stack information.

      1) For the origin of the Terraform configuration, keep **My configuration** selected.

      2) Under **Stack configuration**, upload your `terraform_vcn` folder.

      3) Under **Custom providers**, keep **Use custom Terraform providers** unchecked.

      4) Name the stack and give it a description.

      5) Ensure that your compartment is selected

      6) Click **Next**.

   b. The second page will be for variables. Since we have not configured any variables click **Next**.

   c. The third page will be for review.

      1) Keep **Run apply** unchecked.

      2) Click **Create**. This will take you to the stack's details page.

Task 3: Run a Plan Job

1. The stack itself is only a bookkeeping resource; no infrastructure has been provisioned yet. You should be on the stack's page. Click **Plan**. A form will pop up.

    a. Name the job `RM-Plan-01`.

    b. Click **Plan** again at the bottom to submit a job for Resource Manager to run `terraform plan`. This will take you to the job's details page.

2. Wait for the job to complete, and then view the logs. They should match what you saw when you ran Terraform in Code Editor.

Task 4: Run an Apply Job

1. Go back to the stack's details page (use the breadcrumbs). Click **Apply**. A form will pop up.

    a. Name the job `RM-Apply-01`.

    b. Under **Apply job plan resolution**, select the plan job we just ran (instead of "Automatically approve"). This makes it execute based on the previous plan, instead of running a new one.

    c. Click **Apply** to submit a job for Resource Manager to run `terraform apply`. This will take you the job's details page.

2. Wait for the job to finish. View the logs and confirm that it was successful.

Task 5: View the VCN

1. Navigate to VCNs in the Console through the navigation menu under **Networking** and **Virtual Cloud Networks**.

2. You should see the VCN listed in the table. Click on its name to go to its **Details** page.

3. You should see the subnet listed.

Task 6: Run a Destroy Job

1. Go back to the stack's details page in **Resource Manager**.

2. Click the stack created.

3. Click **Destroy**. Click **Destroy** again on the menu that pops up.

4. Wait for the job to finish. View the logs to see that it completed successfully.

5. Navigate back to VCNs in the Console. You should see that it has been terminated.

6. Go back to the stack in Resource Manager. Click the drop-down for **More actions**. Select **Delete stack**. Confirm by selecting **Delete**.

You have now created a Terraform configuration for a VCN; created and destroyed the VCN through Terraform running locally in Cloud Shell/Code Editor; and created and destroyed the VCN through managed Terraform in Resource Manager.

Security: Configure Security Zones Using Maximum Security Zones

Lab 8-1 Practices

Get Started

Overview

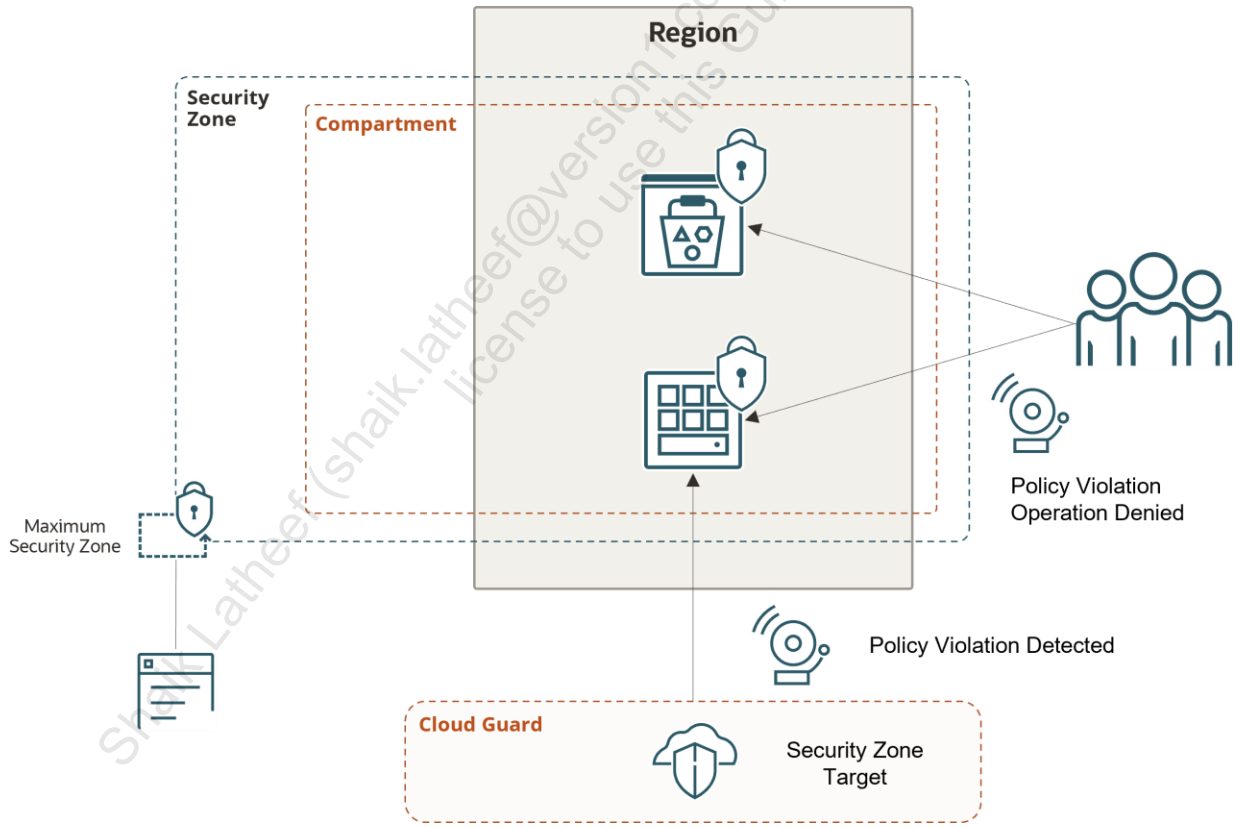
Security zones enforce security posture on OCI cloud compartments and prevent actions that can compromise a customer's security posture. Security zone policies can be applied to various cloud infrastructure types (network, compute, storage, database, and so on) to guarantee cloud resources ensure security and to prevent potential misconfigurations.

In this lab, you will:

a. Set up a security zone with Maximum Security Recipe

b. View the security zone policies attached to a created security zone

c. Test creating a bucket in an assigned compartment using an Oracle-managed key



Prerequisites

- The required IAM policies have been implemented.

- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.
- Your tenancy should have Cloud Guard enabled.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn) (IAD) as your region.
- You must be familiar with navigating the OCI Console.

Set Up Security Zone with Maximum Security Recipe

You will create a security zone for an allocated compartment and check for any security zone policy violations.

Tasks

1. Sign in to the OCI Console.

2. In the Console ribbon at the top of the screen, click the **Region** icon to expand the menu. Ensure that you are in the correct region, **US East (Ashburn)**.

3. From the navigation menu, select **Identity & Security**. Navigate to **Security Zones**, and then click **Overview**.

4. In the left navigation pane, under **Scope**, select *<your assigned compartment>* from the drop-down menu.

   Note: The compartment should not be associated with a security zone. By default, all sub-compartments are also in the same security zone.

5. Click **Create Security Zone**.

6. On the Create Security Zone page, enter the following values:

   a. **Security Zone Recipe:** Select **Oracle-managed** to use Maximum Security Recipe.

   b. **Name**: `IAD-FA-LAB08-1-SZ-01`

   c. **Description**: My Security Zone

   d. **Create for compartment**: *<your assigned compartment>*

7. Click **Create Security Zone**.

   Note: When you create a security zone for a compartment, Cloud Guard does the following:

   - Deletes any existing Cloud Guard target for the compartment and for any child compartments
   - Creates a security zone target for the compartment
   - Adds the default Oracle-managed detector recipes to the security zone target

View the Security Zone Policies Attached with a Created Security Zone

You will identify the recipe associated with the newly formed security zone, and then review its policies.

1. From the navigation menu, select **Identity & Security**. Navigate to **Security Zones**, and then click **Overview**.

2. In the left navigation pane, under Scope, select *<your assigned compartment>* from the drop-down menu.

3. Click the **IAD-FA-LAB08-1-SZ-01** security zone and view the Security Zone details page.

4. On the **Security Zone** information tab, locate the attached recipe and click the **Recipe** for this security zone: Maximum Security Recipe – 20200914.

5. View the Oracle-managed recipe attached to the Security Zone created on the **Recipe details** page.

6. View a few policy statements with associated Resource types:

```
deny public_subnets in VIRTUALNETWORK
deny public_buckets in OBJECTSTORAGE
deny buckets_without_vault_key in in OBJECTSTORAGE
```

Next, you will put a security zone to test by attempting to violate a few of its policies.

Verify Creating a Bucket in an Assigned Compartment Using a Oracle-Managed Key

You will test the security zone. Create a bucket to check if it is restricted in the security zone. As a reference, the security zone recipe has a policy that prohibits bucket creation without a customer-managed vault key.

To create a bucket to observe the security zone violations:

1. Open the navigation menu and click **Storage**. Navigate Object Storage, click **Buckets**.
2. In the left navigation pane, under **List Scope**, select the assigned compartment from the drop-down menu.
3. Click **Create Bucket**.
4. In the **Create Bucket** dialog box, specify the attributes of the bucket:
   a. **Bucket Name:** `IAD-FA-LAB08-1-BKT-01-<user-id>`
      Please specify your user ID in place of `<user-id>` to make it unique.
   b. **Default Storage Tier:** `Standard`
   c. **Encryption:** Encrypt using Oracle-managed keys.

      Note: Leave all the other options in their default setting.
5. Click **Create**.

   You will receive an error indicating a security zone violation: “Encrypt the bucket with a customer-managed encryption key”.
6. Click **Cancel**.

The security zone recipe created earlier has a policy that prohibits bucket creation without a customer-managed key. You will need to create an OCI Vault and a master encryption key, using which you can create a bucket. This way the security zone recipes enforce security posture on OCI cloud compartments and prevent actions that could compromise the security posture of a customer.

Note: Please purge the Security Zone created for this lab.

Purge Security Zone

1. From the navigation menu, select **Identity & Security**. Navigate to **Security Zones** and click **Overview**.

2. Make sure you are in your given compartment.

3. From the list of Security Zones, locate your Security Zone and click its name: **IAD-FA-LAB08-1-SZ-01**.

4. Click **Delete.** Then click **Delete** in the Confirmation window.

Observability and Management: Configure Monitoring and Alarms with Notifications

Lab 9-1 Practices

Get Started

Overview

Oracle Cloud Infrastructure (OCI) Observability and Management provides visibility and actionable insights derived using Machine Learning Algorithms. This platform is open and extensible, and provides cloud-based monitoring and analytics.

Some of the Observability and Management services include Monitoring, Logging, Event Services, Logging Analytics, and Application Performance Monitoring. In this lab, you will create alarms and queries, and trigger alarms.

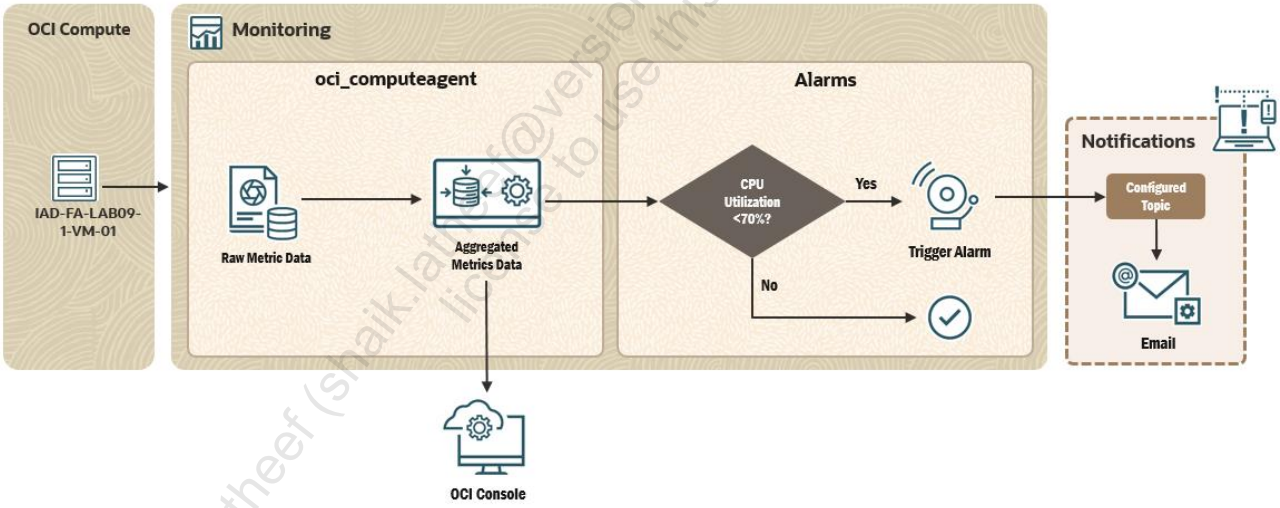
In this lab, you will:

a. Create a Virtual Cloud Network (VCN)

b. Launch a Compute Virtual Machine instance

c. Create alarms and view service metrics

d. Create CPU stress and fire alarms



Prerequisites

- The required IAM policies have been implemented.
- You have access to the OCI Console.
- All the resources required for this lab are available in your assigned compartment.

Assumptions

- Select the region that is available in the tenancy allotted to you. In this lab, we are considering US East (Ashburn, Region Key – IAD) as your region.
- You must be familiar with navigating the OCI Console.

Set Up the Environment

In this practice, you will configure the cloud environment, create a virtual network, and compute instances. The resources created in this practice will help you complete the rest of the lab.

Task 1: Create a VCN

A Virtual Cloud Network (VCN) defines a private network in the cloud environment where you can specify networking parameters such as CIDR block and route tables, along with security controls like access control lists and virtual firewalls. You can also allow connectivity to the public Internet. In this task, you will create a VCN.

Note: For a production VCN environment, it is recommended to further restrict network access controls to meet your security requirements.

1. Sign in to your OCI account.
2. In the console ribbon at the top of the screen, click the **Region** icon to expand the menu and select **US East (Ashburn)**.
3. From the navigation menu, under **Networking**, select **Virtual Cloud Networks**.
4. From the left navigation panel, ensure you are in the compartment allotted to you. Click **Create VCN**.
5. In the **Create a Virtual Cloud Network** dialog box, populate the following information:
   - **Name:** `IAD-FA-LAB09-1-VCN-01`
   - **Create In Compartment:** <*your compartment*>.
   - **IPv4 CIDR Block:** `10.0.0.0/16` (Press **Enter** to add the IP block.)
6. Leave other fields as default. Click **Create VCN**.
7. After the VCN is created, click **IAD-FA-LAB09-1-VCN-01** VCN to view the details page. Under **Resources** in the left navigation panel, click **Internet Gateways**.
8. Click **Create Internet Gateway**.

9. In the **Create Internet Gateway** dialog box, populate the following information:
   - **Name:** `IAD-FA-LAB09-1-IG-01`
   - **Create In Compartment:** <*your compartment*>
10. Click **Create Internet Gateway**.

    Note: If you get a security zone violation error do ensure you have purged the security zone created in Lab 8.
11. Next, make a quick update to the VCN route table to make use of the Internet gateway created in the previous step. Under **Resources** in the left navigation panel, click **Route Tables**.
12. Click **Default Route Table for IAD-FA-LAB09-1-VCN-01** and then click **Add Route Rules**.
13. In the **Add Route Rules** dialog box, populate the following information:
    - **Target Type**: Internet Gateway
    - **Destination CIDR Block:** `0.0.0.0/0`
    - **Target Internet Gateway:** `IAD-FA-LAB09-1-IG-01`
14. Click **Add Route Rules** to complete the process.
15. Finally, create a subnet in the VCN to identify IP space and deploy a VM. Return to the VCN details page by clicking **IAD-FA-LAB09-1-VCN-01** in the breadcrumb list at the top of the page.
16. Under **Resources** in the left navigation panel, click **Subnets**. Then, click **Create Subnet**.
17. In the **Create Subnet** dialog box, populate the following information:
    - **Name:** `IAD-FA-LAB09-1-SNET-01`
    - **Create In Compartment:** <*your compartment*>
    - **Subnet Type:** Regional (Recommended)
    - **IPv4 CIDR Block**: `10.0.0.0/24`
    - **Route Table Compartment in <*your compartment*>:** Default Route Table

- **Subnet Access:** Public Subnet

18. Leave other fields as default. Click **Create Subnet**.

Task 2: Set Up SSH Keys for Virtual Machine Instance

Before launching a Virtual Machine instance, you will create SSH keys to authenticate the instance using Oracle Cloud Shell.

1. In the OCI Console ribbon at the top of the screen, ensure that the correct Region is selected. In this case, the region is **US East (Ashburn)**.

2. Click the **Cloud Shell** icon next to the region.

3. In the Cloud Shell, ensure that you are in the home directory of your account. To check, run the following command:

   ```
   $ pwd
   ```

4. **Reminder:** Do not include the `$` symbol when pasting code into Cloud Shell.

   If you are in your home directory, the value will be `/home/<user_name>`.

5. To change the directory to `.ssh` directory, run the following command:

   ```
   $ cd .ssh/
   ```

6. If the previous step shows an error as "No such file or directory," run the following command:

   ```
   $ mkdir .ssh/
   ```

7. Now, change directory to `.ssh/` by running the following command:

   ```
   $ cd .ssh/
   ```

8. To create SSH keys, run the following command:

   ```
   $ ssh-keygen -b 2048 -t rsa -f <<sshkeyname>>
   ```

   Replace *<<sshkeyname>>* with **ocifalab9key**. Select the key name you can remember. This will be the key name you will use to connect to the compute instance you create.

9. Do not enter a password when prompted, press **Enter**.

   Note: There are two files saved into the `.ssh` directory: **ocifalab9key.pub** (public key) and **ocifalab9key** (private key). **ocifalab9key.pub** will be used while creating compute instances, and **ocifalab9key** will be used to authenticate.

10. Run the following command to view the contents of the **ocifalab9key.pub** public key:

    ```
    $ cat /home/<user_name>/.ssh/<<sshkeyname>>.pub
    ```

    Note: Replace `<user_name>` with your username as noted in Step 3 and `<<sshkeyname>>` with **ocifalab9key**.

11. Copy and paste the content of **ocifalab9key.pub** public key into a Notepad file. You will use this content while creating compute instance.

12. Close the Cloud Shell by clicking **X** at the top-right corner. Then, click **Exit**.

Task 3: Launch Compute Virtual Machine Instance

Now, you will launch a Virtual Machine in your newly created VCN. For this lab, you will create three instances.

1. In the OCI Console ribbon at the top of the screen, ensure that you have selected the same region where you created the VCN.

2. From the navigation menu, under **Compute**, click **Instances**.

3. From the left navigation panel, ensure that you are in the compartment allotted to you. To create the first instance, click **Create instance**.

4. In the **Create compute instance** dialog box, enter **IAD-FA-LAB09-1-VM-01** in the **Name** field.

5. In the **Create in compartment** field, select <*your compartment*>.

6. The **Availability Domain** will be pre-populated to match the subnet you created earlier.

7. Ensure that the **Image** is selected as **Oracle Linux 8**. If not, click **Change Image** and select **Oracle Linux 8**.

8. In the **Shape** field, click **Change Shape**. Then select **VM.Standard.A1.Flex** (under Ampere). Leave the **Number of OCPU** as one and **Amount of Memory (GB)** as six.

9. In the **Primary network** field, select **Select Existing Virtual Cloud Network** and ensure **IAD-FA-LAB09-1-VCN-01** is specified in the **Virtual cloud network** field.

10. In the **Subnet** field, select **Select Existing Subnet**. Ensure the **Subnet** is specified as **IAD-FA-LAB09-1-SNET-01 (regional)**.

    If not, double-check the compartment is set to <*your compartment*>. You may have to switch to a different Availability Domain (see above—the Availability Domain of your subnet and compute instance must match) to allow the selection of your existing subnet, if not already selected.

11. In the **Public IP address** field, select **Assign a public IPv4 address**.

12. In the **Add SSH keys** field, select **Paste public keys**. Then copy the **ocifalab9key.pub** public key from the Notepad (copied earlier in previous task) and paste it in the **SSH keys** field.

13. Keep the other options default and click **Create**.

14. Navigate back to the **Instances** page from the navigation menu. Ensure that the **State** of the instance you just created is **Running**.

15. Copy the Public IP corresponding to the **IAD-FA-LAB09-1-VM-01** instance and paste it in the Notepad.

16. Now, click the **Cloud Shell** icon next to the Region at the top of the screen.

17. Run the following command by using the ocifalab9key - private key:

    ```
    $ ssh -i /home/<user_name>/.ssh/<<sshkeyname>> opc@X.X.X.X
    ```

    - Replace `<user_name>` with your username and `<<sshkeyname>>` with **ocifalab9key**.

    - Replace `X.X.X.X` with the public IP address copied in Step 15.

    Note: The SSH Key is the private key created in the previous task. It is used to authenticate.

18. Enter **Yes** when prompted to connect and ensure you are connected to the instance.

19. Enter **Exit** to close the connection.

Create Alarms and View Service Metrics

In this practice, you will view the service metrics for your instances, confirm that the required monitoring plug-in is enabled, and set up alarm notifications.

Task 1: Confirm that Compute Instance Monitoring Plug-In Is Enabled

To view the service metrics available in the OCI Console, the compute instance monitoring plug-in must be enabled. This plug-in emits metrics about the instance's health, capacity, and performance—such as CPU and memory utilization.

Note: The plug-in will be enabled by default, but it should be confirmed.

1. From the OCI Console navigation menu, under **Compute**, select **Instances**.
2. Click the instance **IAD-FA-LAB09-1-VM-01**.
3. Click **Oracle Cloud Agent** tab.
4. Scroll down to find the **Compute Instance Monitoring** plug-in and ensure that it is running and enabled.

Task 2: Create a Topic and a Subscription inside a Topic

Now that you have confirmed that Monitoring is enabled, you will create an alarm that is triggered when the service metrics reach a designated threshold. You will see this alarm gets triggered later in the practice when you perform a CPU stress test.

To create an alarm, you must first create a notification so that the alarm has a way to notify the relevant parties. For example, an alarm can email an administrator when a CPU usage threshold has been breached.

1. From the OCI Console navigation menu, select **Developer Services**. Under **Application Integration**, select **Notifications**.
2. From the left navigation panel, ensure you are in the compartment assigned to you.
3. Click **Create Topic**.
4. In the **Create Topic** dialog box, enter **IAD-FA-LAB09-1-TOP-01** in the **Name** field and enter **Description**.

5. Click **Create**.

6. Once the topic state changes to **Active**, click the topic to view the details.

7. Click Subscriptions under **Resources**, click **Create Subscription**.

8. In the **Create Subscription** dialog box, select **Email** in the **Protocol** field.

9. In the **Email** field, enter your email address.

10. Click **Create**.

11. Click the subscription that you just created.

12. The Subscription Information will be displayed with the status as **Pending Confirmation**.

13. Check the email account you specified and click the "Confirm subscription" verification link in it. A pop-up browser window will tell you that the subscription has been confirmed.

14. Navigate back to the **Subscriptions** page and verify that the subscription status has changed to **Active**.

    Note: You may need to refresh your browser if the status is not updated.

A topic and a subscription inside a topic are successfully created.

Task 3: Create an Alarm for CPU Utilization

Now that you've created the topic and subscription for a notification, you will create your alarm. This alarm will be activated when the CPU utilization reaches a threshold that you designate.

1. From the OCI Console navigation menu, select **Observability & Management**. Under **Monitoring**, click **Alarm Definitions**.

2. From the left navigation panel, ensure that you are in the compartment assigned to you.

3. Click **Create Alarm**.

4. In the **Create Alarm** dialog box, populate the following information in the **Create alarm** section:

    - **Alarm name:** `IAD-FA-LAB09-1-ALA-01`

- **Alarm severity:** Critical
- **Alarm body:** High Usage of CPU

5. The **Tags** section is optional. Therefore, keep the default selections.
6. Populate the following information in the **Metric description** section:
   - **Compartment:** <*your compartment*>
   - **Metric namespace:** `oci_computeagent`
   - **Metric name**: `CpuUtilization`
   - **Interval**: 1m
   - **Statistic**: Max

   Note: The **Resource Group** field is optional. Therefore, you can skip it for now.
7. Populate the following information in the **Metric dimensions** section:
   - **Dimension name**: `resourceDisplayName`
   - **Dimension value**: `IAD-FA-LAB09-1-VM-01`
8. Populate the following information in the **Trigger rule** section:
   - **Operator:** greater than
   - **Value:** 70
   - **Trigger delay minutes:** 1
9. Populate the following information in the **Define alarm notifications** section:
   - **Destination service:** Notifications
   - **Compartment:** <*your compartment*>
   - **Topic:** `IAD-FA-LAB09-1-TOP-01`

You have created the topic earlier and recall that the topic is the communication channel, such as email. When the alarm is triggered, a notification is sent to the subscribed email addresses.

10. Select the option **Split notifications per metric stream** in the **Message grouping** section.

    With this setting, you are configuring the Alarm to send a message for the specific instance when it reaches the CPU threshold. The UI shows a message, which is just a reference.

11. You can select the message format, which is generally the first option, **Send formatted messages**.

12. You can also choose to have a notification repeated at certain frequencies if an alarm continues. Keep the **Repeat notification** option deselected.

13. You have the option to suppress the notification. Keep the **Suppress notifications** option deselected.

14. Select **Enable this alarm** and click **Save Alarm**.

You should now be able to see the alarm's details.

Create CPU Stress and Fire Alarm

In this practice, you will create a CPU Stress on the instance (IAD-FA-LAB09-1-VM-01), monitor the effect of CPU stress on the instance, and see an event triggered when the CPU utilization is greater than the threshold, which causes the alarm to fire.

Task 1: Create CPU Stress for an Instance

Now that you have created an alarm, Observability and Management monitors the working of instances and sends a notification when the alarm is triggered. For this purpose, the CPU is subjected to stress and forced to run to its maximum capacity. When the CPU Utilization metric is greater than the threshold value, the alarm gets triggered.

This is simulated by means of a CPUStress generator. The following steps are with respect to a Linux OS.

1. From the OCI Console navigation menu, under **Compute**, click **Instances**.
2. Click the instance **IAD-FA-LAB09-1-VM-01**. Copy the Public IP address.
3. Click the **Cloud Shell** icon from the Console ribbon at the top of the page.
4. Connect to the instance by running the following command:

   ```
   $ ssh -i /home/<user_name>/.ssh/<<sshkeyname>> opc@<X.X.X.X>
   ```

   - Replace *<user_name>* with your username and *<<sshkeyname>>* with **ocifalab9key**.
   - Replace *X.X.X.X* with the Public IP address.
5. You should get a message that the FIPS mode is initialized.
6. Run the following command to install the EPEL (Extra Packages for Enterprise Linux) repository on Linux distributions to install additional standard open-source software packages by using YUM and DNF package manager. If you are asked if it is OK, enter **Y**.

   ```
   $ sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
   ```

7. Enter **Y**. You will see **Complete!** when it is complete.

8. Install the stress package. Stress is a generator tool, devised to subject your system to configurables measure of CPU, memory, I/O, disk stress. To install, run the following command:

```
$ sudo yum install stress
```

Note: If you are asked if it is OK, enter `Y` again.

You will get a message when the installation is successful.

Task 2: Induce Stress to the Compute Instance

Now, you need to induce stress to the instance. The stress on the compute instances increases on repeated use of the stress command. Run the following command:

```
$ uptime
$ stress --cpu 8 --timeout 300
```

Task 3: Trigger the Alarm

1. From the OCI Console navigation menu, select **Observability & Management**. Under **Monitoring**, click **Alarm Definitions**.

2. Click the **IAD-FA-LAB09-1-ALA-01** alarm that you created earlier.

3. The icon in IAD-FA-LAB09-1-ALA-01 would have changed to Firing mode due to the stress induced. This happens when the load on the CPU utilization crosses the threshold limits. Please wait for a minute; if the status has not changed to Firing, refresh the page.

4. Scroll down to the **Alarm history** graph, which signifies that the CPU stress has surpassed the set threshold.

5. An email notification is sent to the configured subscription email of the Notifications Topic as the Alarm status changes from OK to Firing.

6. The email provides details about Alarm OCID, Number of Metrics breaching threshold, and Dimensions.

7. Navigate back to the **Alarm Definitions** page and select the check box against the IAD-FA-LAB09-1-ALA-01 alarm.

8. Click **Actions** and select **Add suppressions** from the drop-down list.

9. In the **Suppress Alarms** Wizard, select the default **Start time** and **End time** and click **Apply suppressions** to confirm.

10. Click **Close** and verify that the column **Suppressed** shows the alarm is suppressed for the period.

11. Click the **Cloud Shell** icon to open Cloud Shell where the stress was initiated on the instance. Press Ctrl + C to stop the stress.

12. Navigate back to the **Alarm Definitions** page and click the **IAD-FA-LAB09-1-ALA-01** alarm.

13. The CPU usage alarm icon would have changed to OK mode as the stress is now stopped.

14. Verify an email notification has not been received by the configured subscription email for the status being changed from Firing to OK. This notification is not sent due to Alarm being suppressed for the period.