

VPC Networking Fundamentals

experiment Lab schedule 1 hour universal_currency_alt 1 Credit

show_chart Introductory

GSP210



Google Cloud Self-Paced Labs

Overview

Google Cloud Virtual Private Cloud (VPC) provides networking functionality to Compute Engine virtual machine (VM) instances, Kubernetes Engine containers and App Engine Flex. In other words, without a VPC network you cannot create VM instances, containers or App Engine applications. Therefore, each Google Cloud project has a **default** network to get you started.

You can think of a VPC network the same way you would think of a physical network, except that it is virtualized within Google Cloud. A VPC network is a global resource which consists of a list of regional virtual subnetworks (subnets) in data centers, all connected by a global wide area network (WAN). VPC networks are logically isolated from each other in Google Cloud.

In this lab, you create an auto mode VPC network with firewall rules and two VM instances. Then, you explore the connectivity for the VM instances.

Objectives

In this lab, you learn how to perform the following tasks:

- Explore the default VPC network
- Create an auto mode network with firewall rules
- Create VM instances using Compute Engine
- Explore the connectivity for VM instances

Setup and requirements

Before you click the Start Lab button

Read these instructions. Labs are timed and you cannot pause them. The timer, which starts when you click **Start Lab**, shows how long Google Cloud resources will be made available to you.

This hands-on lab lets you do the lab activities yourself in a real cloud environment, not in a simulation or demo environment. It does so by giving you new, temporary credentials that you use to sign in and access Google Cloud for the duration of the lab.

To complete this lab, you need:

- Access to a standard internet browser (Chrome browser recommended).

Note: Use an Incognito or private browser window to run this lab. This prevents any conflicts between your personal account and the Student account, which may cause extra charges incurred to your personal account.

- Time to complete the lab---remember, once you start, you cannot pause a lab.

Note: If you already have your own personal Google Cloud account or project, do not use it for this lab to avoid extra charges to your account.

How to start your lab and sign in to the Google Cloud console

1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is the **Lab Details** panel with the following:

- The **Open Google Cloud console** button
- Time remaining
- The temporary credentials that you must use for this lab
- Other information, if needed, to step through this lab

2. Click **Open Google Cloud console** (or right-click and select **Open Link in Incognito Window** if you are running the Chrome browser).

The lab spins up resources, and then opens another tab that shows the **Sign in** page.

Tip: Arrange the tabs in separate windows, side-by-side.

Note: If you see the **Choose an account** dialog, click **Use Another Account**.

3. If necessary, copy the **Username** below and paste it into the **Sign in** dialog.

"Username"

content_co

You can also find the **Username** in the **Lab Details** panel.

4. Click **Next**.

5. Copy the **Password** below and paste it into the **Welcome** dialog.

"Password"

content_co

You can also find the **Password** in the **Lab Details** panel.

6. Click **Next**.

Important: You must use the credentials the lab provides you. Do not use your Google Cloud account credentials.

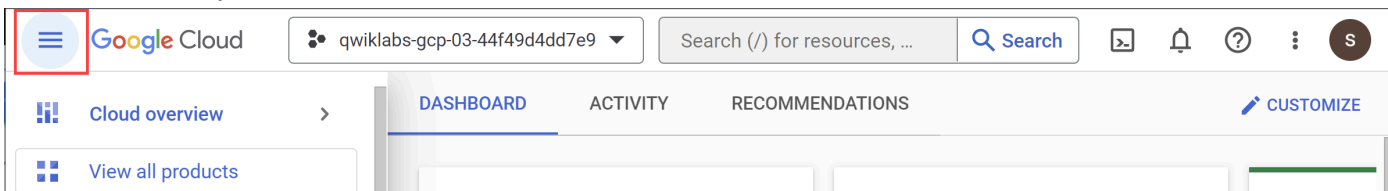
Note: Using your own Google Cloud account for this lab may incur extra charges.

7. Click through the subsequent pages:

- Accept the terms and conditions.
- Do not add recovery options or two-factor authentication (because this is a temporary account).
- Do not sign up for free trials.

After a few moments, the Google Cloud console opens in this tab.

Note: To view a menu with a list of Google Cloud products and services, click the **Navigation menu** at the top-left.



Task 1. Explore the default network

Each Google Cloud project has a **default** network with subnets, routes, and firewall rules.

View the subnets

The **default** network has a subnet in each Google Cloud region.

1. In the Cloud console, navigate to **Navigation menu** (≡) > **VPC network** > **VPC networks**.
2. Click on the **default** network. Notice the **default** network details and the subnets.

Note: Each subnet is associated with a Google Cloud region and a private RFC 1918 CIDR block for its internal **IP addresses range** and a **gateway**.

View the routes

Routes tell VM instances and the VPC network how to send traffic from an instance to a destination, either inside the network or outside of Google Cloud.

Each VPC network comes with some default routes to route traffic among its subnets and send traffic from eligible instances to the Internet.

1. In the left pane, click on **Routes**.
2. In the **Effective Routes** tab, select the **default** network and the **us-central1** region.

Notice that there is a route for each subnet and one for the **Default internet gateway** (0.0.0.0/0).

Note: These routes are managed for you but you can create custom static routes to direct some packets to specific destinations. For example, you can create a route that sends all outbound traffic to an instance configured as a NAT gateway.

View the firewall rules

Each VPC network implements a distributed virtual firewall that you can configure. Firewall rules allow you to control which packets are allowed to travel to which destinations.

Every VPC network has two implied firewall rules that block all incoming connections and allow all outgoing connections.

1. In the left pane, click on **Firewall**.

Notice that there are 4 **Ingress** firewall rules for the **default** network:

- default-allow-icmp
- default-allow-internal
- default-allow-rdp
- default-allow-ssh

Note: These firewall rules allow **ICMP**, **RDP** and **SSH** ingress traffic from anywhere (0.0.0.0/0) and all **TCP**, **UDP** and **ICMP** traffic within the network (10.128.0.0/9). The **Targets**, **Source filters**, **Protocols/ports** and **Action** columns explain these rules.

Delete the default network

1. Select all firewall rules and click **DELETE**.
2. In the left pane, click on **VPC networks**.
3. Click on the **default** network.
4. Click **Delete VPC network** at the top of the page,
5. Then click **DELETE** to confirm the deletion of the **default** network.

Note: Wait for the network to be deleted before moving on.

6. In the left pane, click on **Routes**.

Notice that there are no routes. You may need to click the **Refresh** button at the top of the page.

Note: Without a VPC network, there are no routes!

Try to create a VM instance

Verify that you cannot create a VM instance without a VPC network.

1. In the Cloud console, navigate to **Navigation menu** (≡) > **Compute Engine** > **VM instances**.
2. Click **+CREATE INSTANCE** to create a VM instance.
3. Leave all the values at their default and click **Create**.

Note: Notice the error.

4. Expand the **Advanced options** section, and then scroll down to **Network interfaces**.

Note: Notice the error *No more networks available in this project* under the **Network** box.

5. Click **Cancel**.


Note: As expected, you cannot create a VM instance without a VPC network!

Task 2. Create a VPC network and VM instances

Create a VPC network so that you can create VM instances.

Create an auto mode VPC network with Firewall rules

Replicate the **default** network by creating an auto mode network.

1. In the Console, navigate to **Navigation menu** () > **VPC network** > **VPC networks**, and then click **+CREATE VPC NETWORK**.

2. Set the **Name** to `mynetwork`.

3. For **Subnet creation mode**, click **Automatic**.

Auto mode networks create subnets in each region automatically.

4. For **Firewall rules**, check all available rules.

These are the same standard firewall rules that the default network had.

Note: The **deny-all-ingress** and **allow-all-egress** rules are also displayed, but you cannot check or uncheck them as they are implied. These two rules have a lower **Priority** (higher integers indicate lower priorities) so that the allow ICMP, custom, RDP and SSH rules are considered first.

5. Click **CREATE**, then wait for **mynetwork** to be created.

Notice that a subnet was created for each region.

6. Click on the **mynetwork** name and record the IP address range for the subnets in **REGION** and **REGION**. Refer to these in the next steps.

Note: If you ever delete the default network, you can quickly re-create it by creating an auto

mode network as you just did.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.



Create a VPC network.

Check my progress

Create a VM instance in REGION

Create a VM instance in the **REGION** region. Selecting a region and zone determines the subnet and assigns the internal IP address from the subnet's IP address range.

1. In the Console, navigate to **Navigation menu (≡) > Compute Engine > VM instances**,
2. Click **+CREATE INSTANCE**.
3. Set the following values, leaving all others at their defaults:

Property	Value (type value or select option as specified)
Name	mynet-us-vm
Region	REGION
Zone	ZONE
Series	E2
Machine type	e2-micro

4. Click **Create**, then wait for the instance to be created.

5. Verify that the **Internal IP** was assigned from the IP address range for the subnet in **REGION** **REGION** IP .

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.



Create a VM instance in **REGION** .

Check my progress

Create a VM instance in **REGION**

Create a VM instance in the **REGION** region.

1. Click **+CREATE INSTANCE**.
2. Set the following values, leaving all others at their defaults:

Property	Value (type value or select option as specified)
Name	mynet-second-vm
Region	REGION
Zone	ZONE
Series	E2
Machine type	e2-micro

3. Click **Create**, then wait for the instance to be created.

Note: If you receive an error stating that this zone does not have enough resources to fulfill the

request, try re-running steps 1-3 with a different zone.

4. Verify that the **Internal IP** was assigned from the IP address range for the subnet in **REGION** **REGION** **IP**.

The **Internal IP** should be **REGION** **IP** as x.x.x.1 is reserved for the gateway and you have not configured any other instances in that subnet.

Note: The **External IP addresses** for both VM instances are ephemeral. If an instance is stopped, any ephemeral external IP addresses assigned to the instance are released back into the general Compute Engine pool and become available for use by other projects.

When a stopped instance is started again, a new ephemeral external IP address is assigned to the instance. Alternatively, you can reserve a static external IP address, which assigns the address to your project indefinitely until you explicitly release it.

Test completed task

Click **Check my progress** to verify your performed task. If you have completed the task successfully, you are granted an assessment score.



Create a VM instance in **REGION**.

Check my progress

Task 3. Explore the connectivity for VM instances

Explore the connectivity for the VM instances. Specifically, SSH to your VM instances using `tcp:22` and ping both the internal and external IP addresses of your VM instances using ICMP. Then, explore the effects of the firewall rules on connectivity by removing the firewall rules one-by-one.

Verify connectivity for the VM instances

The firewall rules that you created with **mynetwork** allow ingress SSH and ICMP traffic from within **mynetwork** (internal IP) and outside of that network (external IP).

1. In the Console, navigate to **Navigation menu** () > **Compute Engine** > **VM instances**.

Note the external and internal IP addresses for **mynet-second-vm**.

2. For **mynet-us-vm**, click **SSH** to launch a terminal and connect. You may have to click **SSH** twice.

You are able to SSH because of the **allow-ssh** firewall rule, which allows incoming traffic from anywhere (0.0.0.0/0) for **tcp:22**.

Note: The SSH connection works seamlessly because Compute Engine generates an SSH key for you and stores it in one of the following locations:

- By default, Compute Engine adds the generated key to project or instance metadata.
- If your account is configured to use OS Login, Compute Engine stores the generated key with your user account.

Alternatively, you can control access to Linux instances by creating SSH keys and editing public SSH key metadata.

3. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

```
ping -c 3 <Enter mynet-second-vm's internal IP
```

content_co

here>

You are able to ping **mynet-second-vm**'s internal IP because of the **allow-custom** firewall rule.

4. To test connectivity to **mynet-second-vm**'s external IP, run the following command using **mynet-second-vm**'s external IP:

```
ping -c 3 <Enter mynet-second-vm's external IP  
here>
```

content_co

Task 4. Test your understanding

Below are a multiple choice questions to reinforce your understanding of this lab's concepts. Answer them to the best of your abilities.



Which firewall rule allows the ping to mynet-second-vm's external IP address?

- ☐ mynetwork-allow-icmp
- ☐ mynetwork-allow-rdp
- ☐ mynetwork-allow-ssh
- ☐ mynetwork-allow-custom

Submit



Google Cloud firewall rules let you allow or deny traffic to and from your virtual machine (VM) instances based on a configuration.

- ☐ True
- ☐ False



Firewall rules can be shared among networks.

- ☐ True
- ☐ False

Note: You were able to SSH to **mynet-us-vm** and ping **mynet-second-vm**'s internal and external IP address as expected. Alternatively, you could SSH to **mynet-second-vm** and ping **mynet-us-vm**'s internal and external IP address, which also works.

Task 5. Remove the allow-icmp firewall rules

Remove the **allow-icmp** firewall rule and try to ping the internal and external IP address of **mynet-second-vm**.

1. In the Console, navigate to **Navigation menu** (≡) > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-icmp** rule.
3. Click **DELETE**.
4. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

5. Return to the **mynet-us-vm** SSH terminal.
6. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

```
ping -c 3 <Enter mynet-second-vm's internal IP here>
```

content_copy

You are able to ping **mynet-second-vm**'s internal IP because of the **allow-custom** firewall rule.

7. To test connectivity to **mynet-second-vm**'s external IP, run the following command using **mynet-second-vm**'s external IP:


```
ping -c 3 <Enter mynet-second-vm's external IP here>
```

content_copy

Note: The **100% packet loss** indicates that you are unable to ping **mynet-second-vm**'s external IP. This is expected because you deleted the **allow-icmp** firewall rule!

Task 6. Remove the allow-custom firewall rules

Remove the **allow-custom** firewall rule and try to ping the internal IP address of **mynet-second-vm**.

1. In the Console, navigate to **Navigation menu** () > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-custom** rule and then click **DELETE**.
3. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

4. Return to the **mynet-us-vm** SSH terminal.
5. To test connectivity to **mynet-second-vm**'s internal IP, run the following command using **mynet-second-vm**'s internal IP:

```
ping -c 3 <Enter mynet-second-vm's internal IP  
here>
```

content_co

Note: The **100% packet loss** indicates that you are unable to ping **mynet-second-vm**'s internal IP. This is expected because you deleted the **allow-custom** firewall rule!

6. Close the SSH terminal:

```
exit
```

content_co

Task 7. Remove the allow-ssh firewall rules

Remove the **allow-ssh** firewall rule and try to SSH to **mynet-us-vm**.

1. In the Console, navigate to **Navigation menu** (≡) > **VPC network** > **Firewall**.
2. Check the **mynetwork-allow-ssh** rule and then click **DELETE**.
3. Click **DELETE** to confirm the deletion.

Wait for the firewall rule to be deleted.

4. In the Console, navigate to **Navigation menu** (≡) > **Compute Engine** > **VM instances**.
5. For **mynet-us-vm**, click **SSH** to launch a terminal and connect.

Note: The **Connection failed** message indicates that you are unable to SSH to **mynet-us-vm** because you deleted the **allow-ssh** firewall rule!