Getting Started with VPC Networking

1 hourFree

Rate Lab

**Overview**

Google Cloud Virtual Private Cloud (VPC) provides networking functionality to Compute Engine virtual machine (VM) instances, Kubernetes Engine containers, and App Engine flexible environment. 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 as similar to a physical network, except that it is virtualized within Google Cloud. A VPC network is a global resource that 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

**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 Cloud resources will be made available to you.

This Qwiklabs 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 the Google Cloud Platform for the duration of the lab.

**What you need**

To complete this lab, you need:

* Access to a standard internet browser (Chrome browser recommended).
* Time to complete the lab.

***Note:*** If you already have your own personal GCP account or project, do not use it for this lab.

**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](https://cloud.google.com/compute/docs/regions-zones/#available).

* In the Cloud Console, on the **Navigation menu** (Navigation menu), click **VPC network** > **VPC networks**. Notice the **default** network with its subnets. 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 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.

* In the left pane, click **Routes**. Notice that there is a route for each subnet and one for the **Default internet gateway** (0.0.0.0/0). 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**

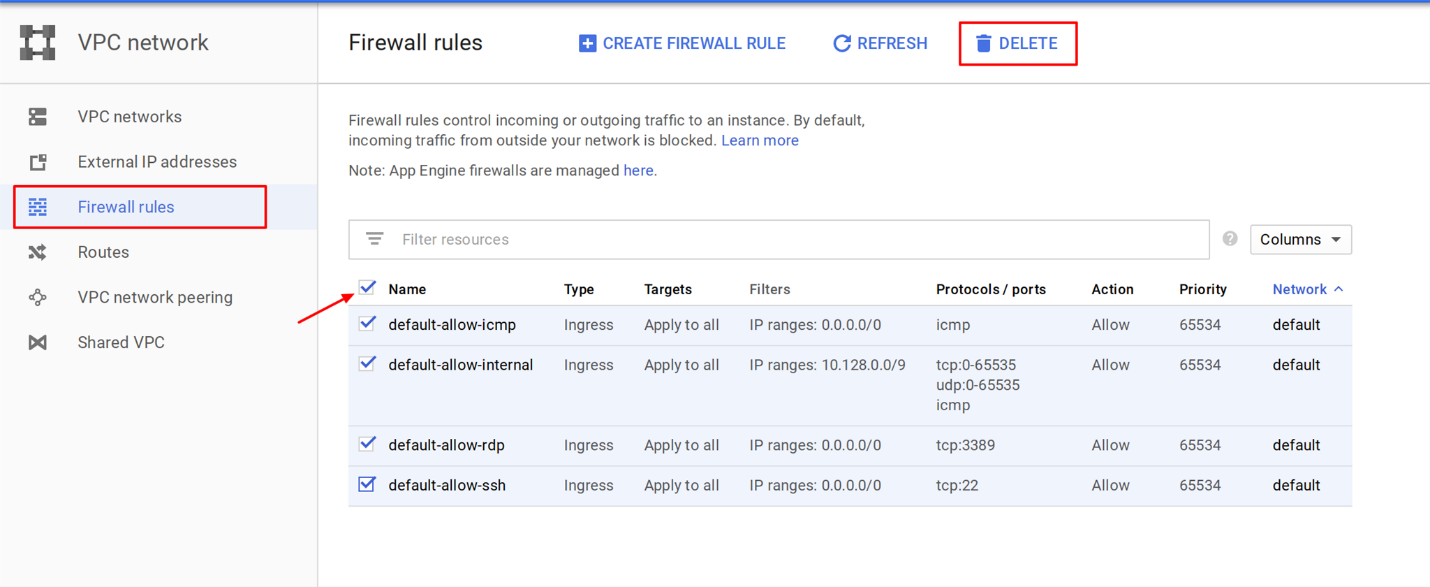
Each VPC network implements a distributed virtual firewall that you can configure. Firewall rules allow you to control which packets are allowed to travel to which destinations. Every VPC network has two implied firewall rules that block all incoming connections and allow all outgoing connections.

* In the left pane, click **Firewall rules**. Notice that there are 4 **Ingress** firewall rules for the **default** network:
  + default-allow-icmp
  + default-allow-rdp
  + default-allow-ssh
  + default-allow-internal

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**, **Filters**, **Protocols/ports**, and **Action** columns explain these rules.

**Delete the Firewall rules**

1. In the left pane, click **Firewall rules**.
2. Select all default network firewall rules.
3. Click **Delete**.
4. Click **Delete** to confirm the deletion of the firewall rules.



**Delete the default network**

1. In the left pane, click **VPC networks**.
2. Select the **default** network.
3. Click **Delete VPC network**.
4. Click **Delete** to confirm the deletion of the **default** network. Wait for the network to be deleted before continuing.
5. In the left pane, click **Routes**. Notice that there are no routes.
6. In the left pane, click **Firewall rules**. Notice that there are no firewall rules.

Without a VPC network, there are no routes and no firewall rules!

**Try to create a VM instance**

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

1. On the **Navigation menu** (Navigation menu), click **Compute Engine** > **VM instances**.
2. Click **Create**.
3. Accept the default values and click **Create**. Notice the error.
4. Click **Management, security, disks, networking, sole tenancy**.
5. Click **Networking**. Notice the **No local network available** error under **Network interfaces**.
6. Click **Cancel**.

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. On the **Navigation menu** (Navigation menu), click **VPC network** > **VPC networks**.
2. Click **Create VPC network**.
3. For **Name**, type **mynetwork**.
4. For **Subnet creation mode**, click **Automatic**. Auto mode networks create subnets in each region automatically.
5. For **Firewall rules**, select all available rules. These are the same standard firewall rules that the default network had. The **deny-all-ingress** and **allow-all-egress** rules are also displayed, but you cannot check or uncheck them because they are implied. These two rules have a lower **Priority** (higher integers indicate lower priorities) so that the allow ICMP, internal, RDP and SSH rules are considered first.
6. Click **Create**. When the new network is ready, notice that a subnet was created for each region.
7. Record the IP address range for the subnets in **us-central1** and **europe-west1**. These will be referred to in the next steps.

If you ever delete the default network, you can quickly re-create it by creating an auto mode network as you just did.

**Create a VM instance in us-central1**

Create a VM instance in the us-central1 region. Selecting a region and zone determines the subnet and assigns the internal IP address from the subnet's IP address range.

1. On the **Navigation menu** (Navigation menu), click **Compute Engine** > **VM instances**.
2. Click **Create**.
3. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value (type value or select option as specified)** |
| Name | mynet-us-vm |
| Region | us-central1 |
| Zone | us-central1-c |
| Machine type | micro (1 shared vCPU) |

1. Click **Create**.
2. Verify that the **Internal IP** for the new instance was assigned from the IP address range for the subnet in **us-central1** (10.128.0.0/20).

The **Internal IP** should be 10.128.0.2 because 10.128.0.1 is reserved for the gateway and you have not configured any other instances in that subnet.

**Create a VM instance in europe-west1**

Create a VM instance in the europe-west1 region.

1. Click **Create instance**.
2. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value (type value or select option as specified)** |
| Name | mynet-eu-vm |
| Region | europe-west1 |
| Zone | europe-west1-c |
| Machine type | micro (1 shared vCPU) |

1. Click **Create**.
2. Verify that the **Internal IP** for the new instance was assigned from the IP address range for the subnet in **europe-west1** (10.132.0.0/20).

The **Internal IP** should be 10.132.0.2 because 10.132.0.1 is reserved for the gateway and you have not configured any other instances in that subnet.

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.

Click *Check my progress* to verify the objective.

Create a VPC network and VM instance

Check my progress

**Task 3. Explore the connectivity for VM instances**

Explore the connectivity for the VM instances. Specifically, try to 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 individually.

**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 that network (external IP).

1. On the **Navigation menu** (Navigation menu), click **Compute Engine** > **VM instances**. Note the external and internal IP addresses for **mynet-eu-vm**.
2. For **mynet-us-vm**, click **SSH** to launch a terminal and connect.

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

1. To test connectivity to **mynet-eu-vm**'s internal IP, run the following command, replacing **mynet-eu-vm**'s internal IP:

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

You can ping **mynet-eu-vm**'s internal IP because of the **allow-internal** firewall rule.

1. Repeat the same test by running the following:

ping -c 3 mynet-eu-vm

You can ping **mynet-eu-vm** by its name because VPC networks have an internal DNS service that allows you to address instances by their DNS names instead of their internal IP addresses. This is very useful because the internal IP address can change when you delete and re-create an instance.

1. To test connectivity to **mynet-eu-vm**'s external IP, run the following command, replacing **mynet-eu-vm**'s external IP:

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

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



mynetwork-allow-internal



mynetwork-allow-rdp



mynetwork-allow-icmp



mynetwork-allow-ssh

Submit

You can SSH to **mynet-us-vm** and ping **mynet-eu-vm**'s internal and external IP address as expected. Alternatively, you can SSH to **mynet-eu-vm** and ping **mynet-us-vm**'s internal and external IP address, which also works.

**Remove the allow-icmp firewall rules**

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

1. On the **Navigation menu** (Navigation menu), click **VPC network** > **Firewall rules**.
2. Select 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-eu-vm**'s internal IP, run the following command, replacing **mynet-eu-vm**'s internal IP:

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

You can ping **mynet-eu-vm**'s internal IP because of the **allow-internal** firewall rule.

1. To test connectivity to **mynet-eu-vm**'s external IP, run the following command, replacing **mynet-eu-vm**'s external IP:

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

The **100% packet loss** indicates that you cannot ping **mynet-eu-vm**'s external IP. This is expected because you deleted the **allow-icmp** firewall rule!

**Remove the allow-internal firewall rules**

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

1. On the **Navigation menu** (Navigation menu), click **VPC network** > **Firewall rules**.
2. Select the **mynetwork-allow-internal** 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-eu-vm**'s internal IP, run the following command, replacing **mynet-eu-vm**'s internal IP:

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

The **100% packet loss** indicates that you cannot ping **mynet-eu-vm**'s internal IP. This is expected because you deleted the **allow-internal** firewall rule!

1. Close the SSH terminal:

exit

**Remove the allow-ssh firewall rules**

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

1. On the **Navigation menu** (Navigation menu), click **VPC network** > **Firewall rules**.
2. Select the **mynetwork-allow-ssh** rule.
3. Click **Delete**.
4. Click **Delete** to confirm the deletion.
5. Wait for the firewall rule to be deleted.
6. On the **Navigation menu**, click **Compute Engine** > **VM instances**.
7. For **mynet-us-vm**, click **SSH** to launch a terminal and connect.

The **Connection failed** message indicates that you cannot SSH to **mynet-us-vm** because you deleted the **allow-ssh** firewall rule!

**Task 4. Review**

In this lab, you explored the default network along with its subnets, routes, and firewall rules. You deleted the default network and determined that you cannot create any VM instances without a VPC network. Thus, you created a new auto mode VPC network with subnets, routes, firewall rules, and two VM instances. Then you tested the connectivity for the VM instances and explored the effects of the firewall rules on connectivity.