

[Start Lab](#)

01:00:00

VPC Network Peering

1 hour Free ★★★★☆ Rate Lab

GSP193

Overview

Setup

VPC Network Peering setup

Setting up a VPC Network Peering session

Connectivity Test

Congratulations!

-/100

GSP193



Google Cloud Self-Paced Labs

Overview

[Chat](#)

Google Cloud Virtual Private Cloud (VPC) Network Peering allows private connectivity across two VPC networks regardless of whether or not they belong to the same project or the same organization.

VPC Network Peering allows you to build SaaS (Software-as-a-Service) ecosystems in Google Cloud, making services available privately across different VPC networks within and across organizations, allowing workloads to communicate in private space.

VPC Network Peering is useful for:

- Organizations with several network administrative domains.
- Organizations that want to peer with other organizations.

If you have multiple network administrative domains within your organization, VPC Network Peering allows you to make services available across VPC networks in private space. If you offer services to other organizations, VPC Network Peering allows you to make those services available in private space to those organizations. The ability to offer services across organizations is useful if you want to offer services to other enterprises, and it is useful within your own enterprise if you have several distinct organization nodes due to your own structure or as a result of mergers or acquisitions.

[Chat](#)

VPC Network Peering gives you several advantages over using external IP addresses or VPNs to connect networks, including:

- **Network Latency:** Private networking offers lower latency than public IP networking.
- **Network Security:** Service owners do not need to have their services exposed to the public Internet and deal with its associated risks.
- **Network Cost:** Networks that are peered can use internal IPs to communicate and save Google Cloud egress bandwidth costs. Regular network pricing still applies to all traffic.

Setup

[Chat](#)

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 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 Google Cloud 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.

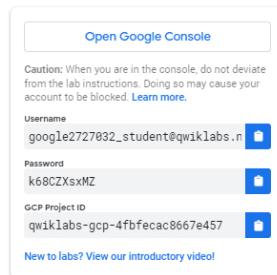
Note: If you already have your own personal Google Cloud account or project, do not use it for this lab.

Note: If you are using a Pixelbook, open an Incognito window to run this lab.

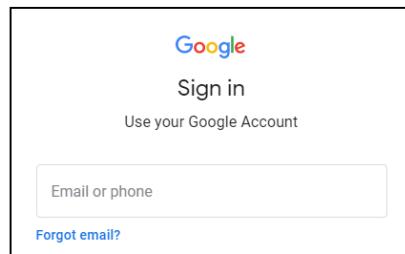
How to start your lab and sign in to the Google Cloud Console

Chat

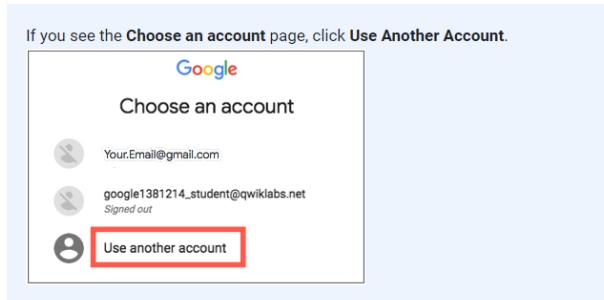
1. Click the **Start Lab** button. If you need to pay for the lab, a pop-up opens for you to select your payment method. On the left is a panel populated with the temporary credentials that you must use for this lab.



2. Copy the username, and then click **Open Google Console**. The lab spins up resources, and then opens another tab that shows the **Sign in** page.



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



3. In the **Sign in** page, paste the username that you copied from the Connection Details panel. Then copy and paste the password.

Chat

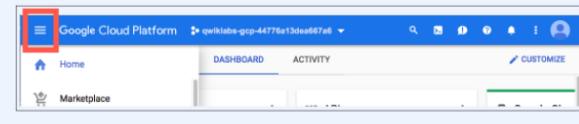
Important: You must use the credentials from the Connection Details panel. Do not use your Qwiklabs credentials. If you have your own Google Cloud account, do not use it for this lab (avoids incurring charges).

4. 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 Cloud Console opens in this tab.

Note: You can view the menu with a list of Google Cloud Products and Services by clicking the **Navigation menu** at the top-left.

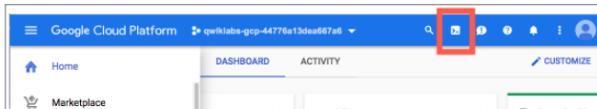


Chat

Activate Cloud Shell

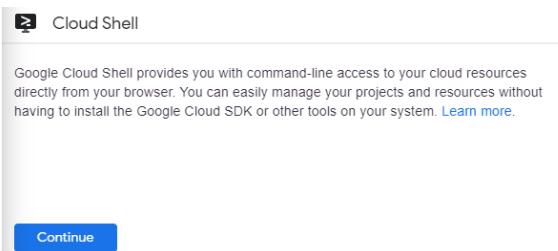
Cloud Shell is a virtual machine that is loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. Cloud Shell provides command-line access to your Google Cloud resources.

In the Cloud Console, in the top right toolbar, click the **Activate Cloud Shell** button.



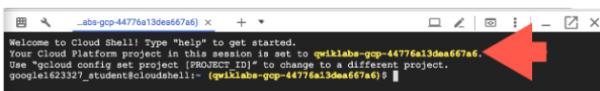
Chat

Click **Continue**.



Chat

It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your *PROJECT_ID*. For example:



Chat

gcloud is the command-line tool for Google Cloud. It comes pre-installed on Cloud Shell and supports tab-completion.

You can list the active account name with this command:

```
gcloud auth list
```

Chat

(Output)

```
Credentialed accounts:  
- <myaccount>@<mydomain>.com (active)
```

Chat

(Example output)

```
Credentialed accounts:  
- google1623327_student@qwiklabs.net
```

You can list the project ID with this command:

```
gcloud config list project
```

Chat

(Output)

```
[core]  
project = <project_ID>
```

Chat

(Example output)

```
[core]  
project = qwiklabs-gcp-44776a13dea667a6
```

For full documentation of `gcloud` see the [gcloud command-line tool overview](#).

VPC Network Peering setup

Within the same organization node, a network could be hosting services that need to be accessible from other VPC networks in the same or different projects.

Alternatively, one organization may want to access services a third-party services offering.

Project names are unique across all of Google Cloud, so you do not need to specify the organization when setting up peering. Google Cloud knows the organization based on the project name.

 Chat

Create a custom network in projects

In this lab you have been provisioned 2 projects, the first project as a Project A and second as Project B.

For managing two projects start a new cloud shell by click + icon.

In the second cloud shell, set project ID by running the following, replacing <PROJECT_ID2> with Project ID for the 2nd project from the Qwiklabs page where you started the lab:

```
gcloud config set project <PROJECT_ID2>
```

Project-A:

Go back to first cloud shell and create a custom network:

 Chat

```
gcloud compute networks create network-a --subnet-mode custom
```

Create a subnet within this VPC and specify a region and IP range by running:

```
gcloud compute networks subnets create network-a-central --network
network-a \
--range 10.0.0.0/16 --region us-central1
```

Create a VM instance:

```
gcloud compute instances create vm-a --zone us-central1-a --network
network-a --subnet network-a-central
```

Run the following to enable SSH and `icmp`, because you'll need a secure shell to communicate with VMs during connectivity testing:

```
gcloud compute firewall-rules create network-a-fw --network network-a --
allow tcp:22,icmp
```

Next you set up project-b in the same way.

 Chat

Click *Check my progress* to verify the objective.

Create a custom network in project A

[Check my progress](#)

Project-B:

Switch to the second cloud shell and create a custom network:

```
gcloud compute networks create network-b --subnet-mode custom
```

Create a subnet within this VPC and specify a region and IP range by running:

```
[REDACTED]
```

```
gcloud compute networks subnets create network-b-central --network
network-b \
--range 10.8.0.0/16 --region us-central1
```

Create a VM instance:

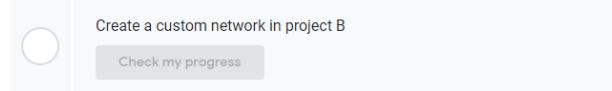
```
gcloud compute instances create vm-b --zone us-central1-a --network
network-b --subnet network-b-central
```

Chat

Run the following to enable SSH and `icmp`, because you'll need a secure shell to communicate with VMs during connectivity testing:

```
gcloud compute firewall-rules create network-b-fw --network network-b --
allow tcp:22,icmp
```

Click *Check my progress* to verify the objective.

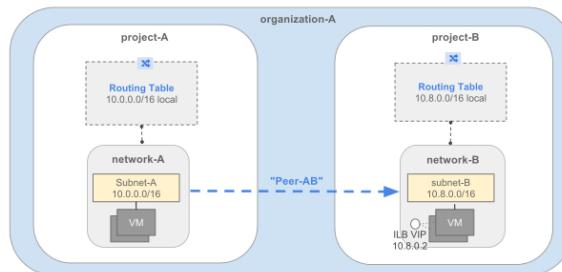


Setting up a VPC Network Peering session

Consider an organization which needs VPC Network Peering to be established between network-A in project-A, and network-B in project-B. In order for VPC Network Peering to be established successfully, administrators of network-A and network-B must separately configure the peering association.

Chat

Peer network-a with network-b:



You will need to select the correct project in the console before you apply the settings. You'll do that by clicking down arrow next to the Project ID at the top of the screen, then selecting which project ID you need.

Chat

Project-A

Chat

Go to the VPC Network Peering in the Cloud Console by navigating to the Networking section and clicking **VPC Network > VPC network peering** in the left menu. Once you're there:

1. Click **Create connection**.
2. Click **Continue**.
3. Type "peer-ab" as the **Name** for this side of the connection.
4. Under **Your VPC network**, select the network you want to peer (network-a).
5. Set the **Peered VPC network** radio buttons to **In another project**.
6. Paste in the **Project ID** of the second project.
7. Type in the **VPC network name** of the other network (network-b).
8. Click **Create**.

At this point, the peering state remains INACTIVE because there is no matching configuration in network-b in project-B.

 Chat

Example Output:

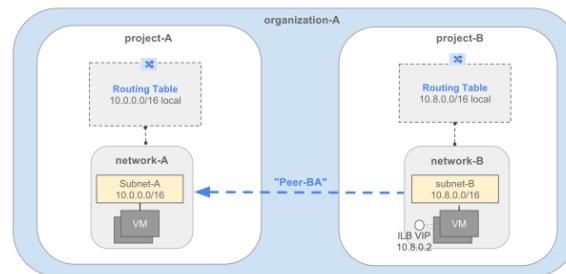
VPC Network Peering				
	 CREATE PEERING CONNECTION	 REFRESH	 DELETE	
Name	Your VPC network	Peered VPC network	Peered project ID	Status
peer-ab	network-a	network-b	qwiklabs-gcp-1ad5e3b27618937e	 Waiting for peer network to connect.

Click *Check my progress* to verify the objective.

Peer network-a with network-b

Check my progress

Peer network-b with network-a



 Chat

Note: Switch to the second project in the console.

Project-B

1. Click **Create connection**.
2. Click **Continue**.
3. Type "peer-ba" as the **Name** for this side of the connection.
4. Under **Your VPC network**, select the network you want to peer (network-b).
5. Set the **Peering VPC network** radio buttons to **In another project**, unless you wish to peer within the same project.
6. Specify the **Project ID** of the first project.
7. Specify **VPC network name** of the other network (network-a).
8. Click **Create**.

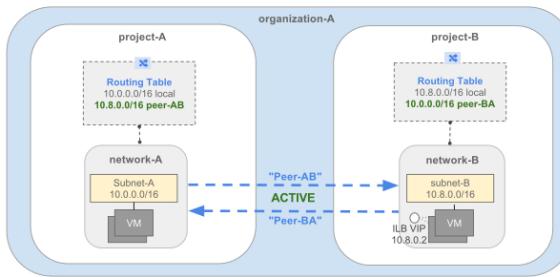
 Chat

Example Output:

VPC Network Peering				
	 CREATE PEERING CONNECTION	 REFRESH	 DELETE	
Name	Your VPC network	Peered VPC network	Peered project ID	Status
peer-ba	network-b	network-a	qwiklabs-gcp-8ad63a49b9d7d4dc	 Connected.

VPC Network Peering becomes ACTIVE and routes are exchanged As soon as the peering moves to an ACTIVE state, traffic flows are set up:

- Between VM instances in the peered networks: Full mesh connectivity.
- From VM instances in one network to Internal Load Balancing endpoints in the peered network.



[Chat](#)

The routes to peered network CIDR prefixes are now visible across the VPC network peers. These routes are implicit routes generated for active peerings. They don't have corresponding route resources. The following command lists routes for all VPC networks for project-a.

```
gcloud compute routes list --project <FIRST_PROJECT_ID>
```

Example Output:

NAME	NETWORK	DEST_RANGE	NEXT_HOP
PRIORITY			
default-route-2a865a00fa31d5df	network-A	0.0.0.0/0	default-internet-gateway 1000
default-route-8af4732e693eae27	network-A	10.0.0.0/16	1000
peering-route-4732ee69e3ecab41	network-A	10.8.0.0/16	peer-ab 1000

[Chat](#)

Click *Check my progress* to verify the objective.

Peer network-b with network-a
[Check my progress](#)

Connectivity Test

Project-A

Navigate to VM instances console: Click **Navigation Menu > Compute Engine > VM instances**.

[Chat](#)

Copy the **INTERNAL_IP** for `vm-a`.

Project-B

Click **Product & services > Compute > Compute Engine > VM instances**.

SSH into `vm-b` instance.

In the SSH shell of `vm-b`, run the following command replacing `<INTERNAL_IP_OF_VM_A>` with the `vm-a` instance INTERNAL_IP:

```
ping -c 5 <INTERNAL_IP_OF_VM_A>
```

Example Output:

```
PING 10.8.0.2 (10.8.0.2) 56(84) bytes of data.
64 bytes from 10.8.0.2: icmp_seq=1 ttl=64 time=0.107 ms
64 bytes from 10.8.0.2: icmp_seq=2 ttl=64 time=0.364 ms
64 bytes from 10.8.0.2: icmp_seq=3 ttl=64 time=0.205 ms
64 bytes from 10.8.0.2: icmp_seq=4 ttl=64 time=0.216 ms
64 bytes from 10.8.0.2: icmp_seq=5 ttl=64 time=0.164 ms
--- 10.8.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4ms
```

[Chat](#)

```
5 packets transmitted, 5 received, 0% packet loss, time 4065ms
rtt min/avg/max/mdev = 0.164/0.404/1.072/0.340 ms
```

You learned how to setup VPC peering across projects in a cloud environment.

Congratulations!



Finish Your Quest

Chat

This self-paced lab is part of the [Security & Identity Fundamentals](#) Quest. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, to recognize your achievement. You can make your badge (or badges) public and link to them in your online resume or social media account. [Enroll in this Quest](#) and get immediate completion credit if you've taken this lab. [See other available Qwiklabs Quests.](#)

Take Your Next Lab

Continue your Quest with [user Authentication: Identity-Aware Proxy](#) or try one of these suggestions:

- [HTTP Load Balancer with Cloud Armor](#)
- [Data Loss Prevention: Qwik Start - Command Line](#)

Next Steps / Learn More

- Read more about VPC networks: <https://cloud.google.com/vpc/docs/vpc>
- Read more about VPC network peering: <https://cloud.google.com/vpc/docs/vpc-peering>

Google Cloud Training & Certification

...helps you make the most of Google Cloud technologies. [Our classes](#) include technical skills and best practices to help you get up to speed quickly and continue your learning journey. We offer fundamental to advanced level training, with on-demand, live, and virtual options to suit your busy schedule. [Certifications](#) help you validate and prove your skill and expertise in Google Cloud technologies.

Manual Last Updated September 15, 2020

Lab Last Tested September 15, 2020

Copyright 2020 Google LLC All rights reserved. Google and the Google logo are trademarks of Google LLC. All other company and product names may be trademarks of the respective companies with which they are associated.

Chat