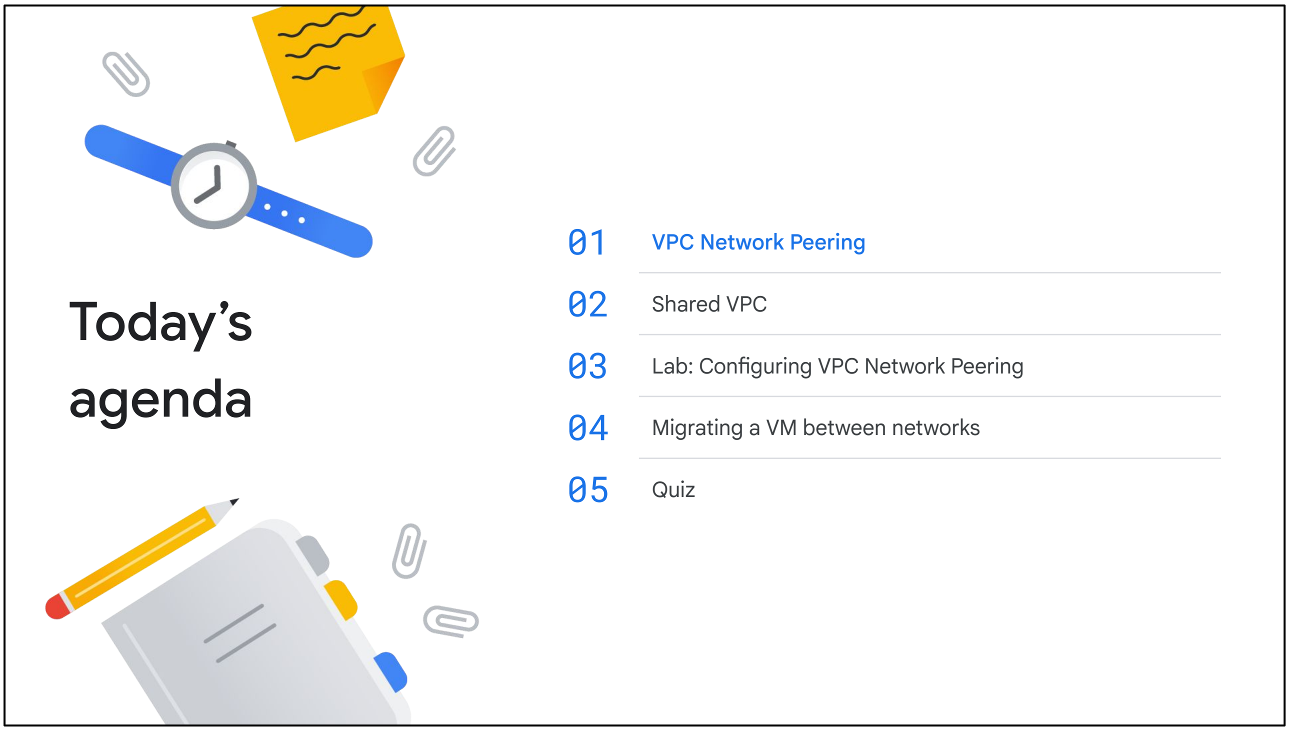
Google Cloud

# Networking in Google Cloud

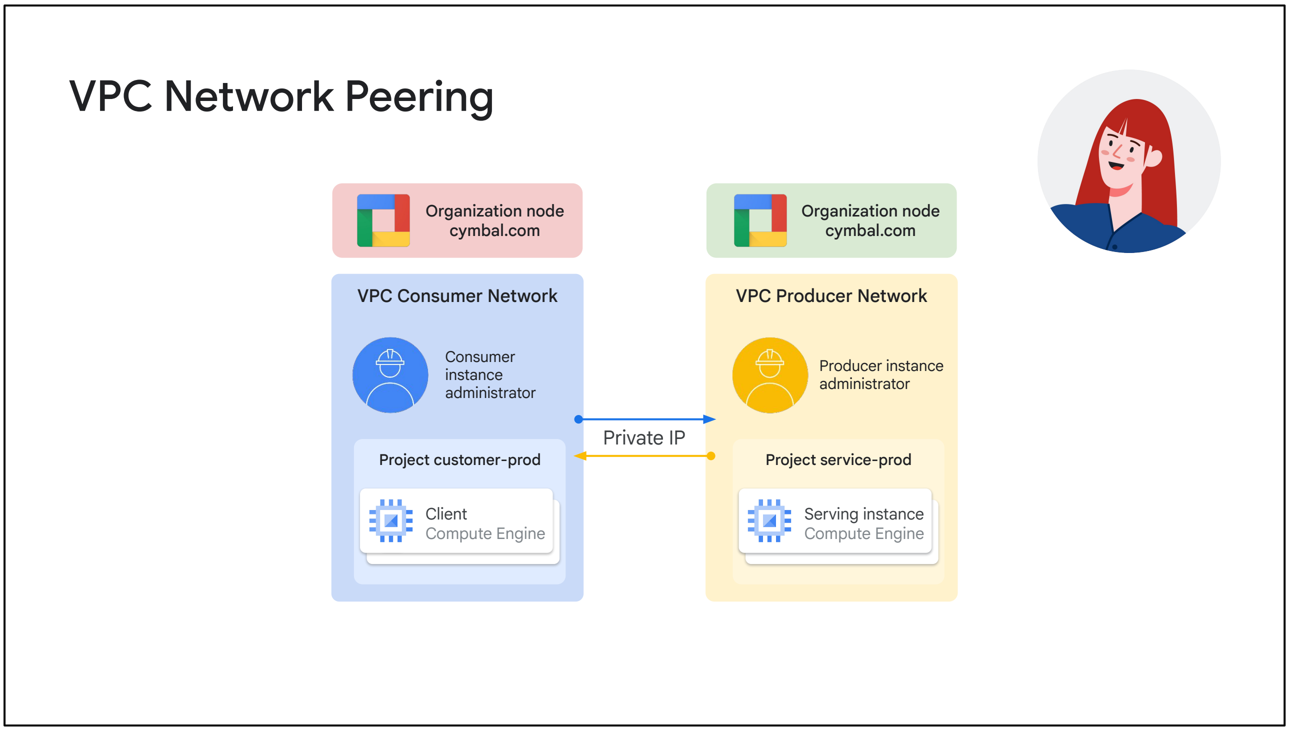
Sharing VPC Networks

Welcome to the Sharing VPC Networks module.

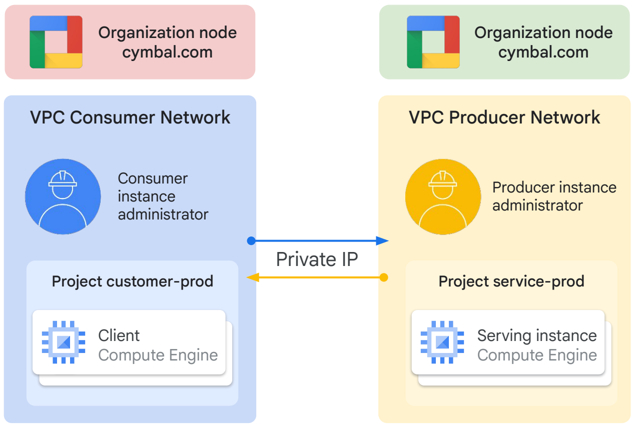


## Today's agenda

This covers more details on sharing a VPC network. You will learn how to connect isolated Virtual Private Clouds (VPCs) using VPC Network Peering, enable centralized network management with Shared VPC, and migrate virtual machines (VMs) between networks. Through hands-on labs and practical examples, you'll gain a understanding of these essential networking concepts, ensuring optimal performance, security, and resource utilization.

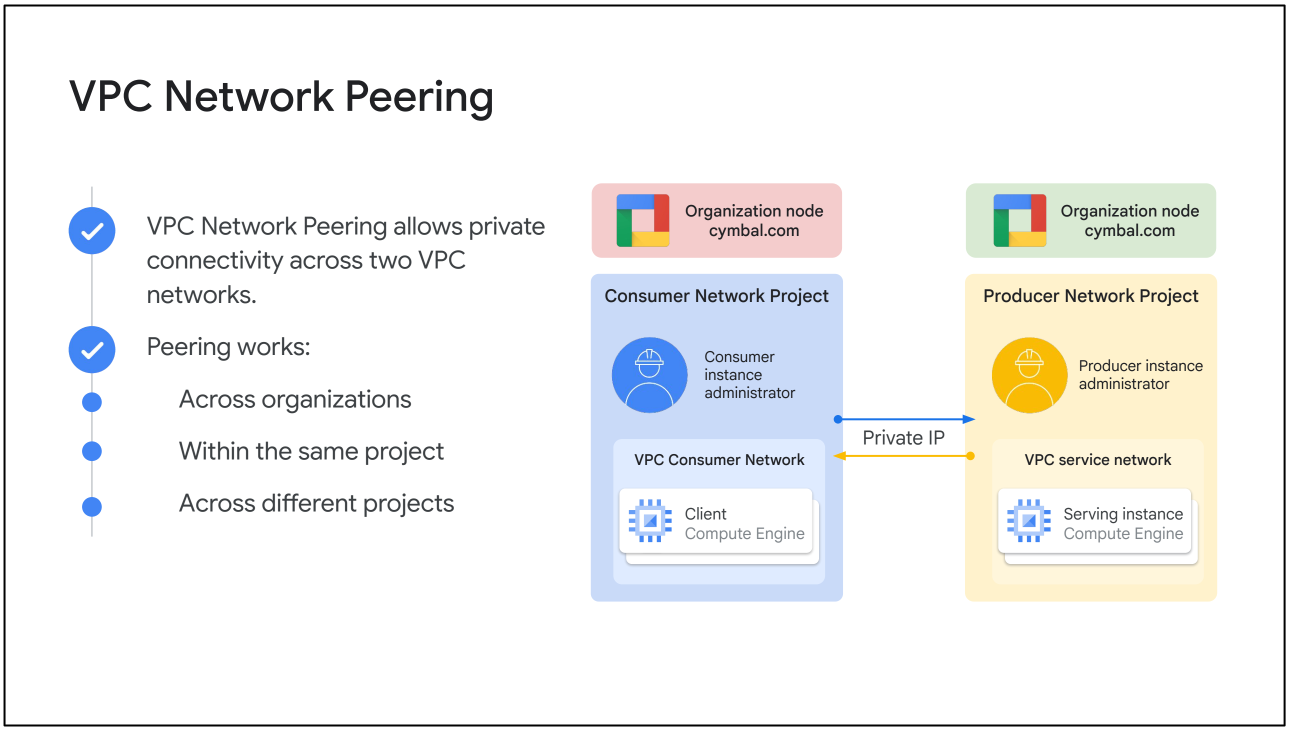


## VPC Network Peering



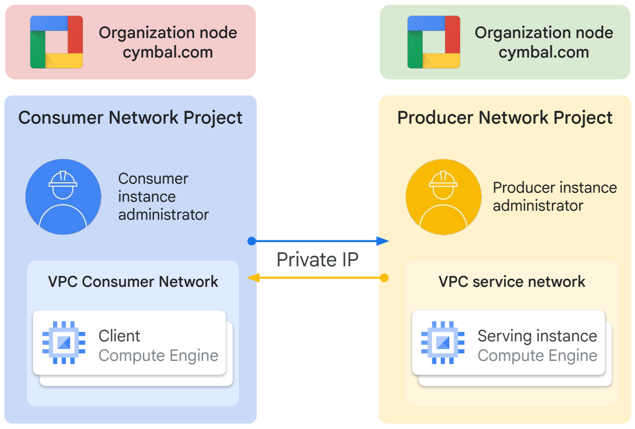
Let us start with a use case. Lucian, a network engineer at Cymbal Corporation, faces a new challenge: seamlessly connecting and establishing a direct connection between their two organizations by consumer and producer. Public internet peering introduces unacceptable latency and security risks, while dedicated VPNs are proving cost-prohibitive. Lucian needs a solution that's private, secure, and scales with their evolving needs.

Lucian opts to use VPC Network Peering.



## VPC Network Peering

- ✓ VPC Network Peering allows private connectivity across two VPC networks.
- ✓ Peering works:
  - Across organizations
  - Within the same project
  - Across different projects


Organization node cymbal.com
Organization node cymbal.com
Consumer Network Project
Producer Network Project
Consumer instance administrator
Producer instance administrator
Private IP
VPC Consumer Network
VPC service network
Client Compute Engine
Serving instance Compute Engine

VPC Network Peering allows private connectivity across two VPC networks. Even if both VPC networks belong to the same organization or the same project, you can still peer them.

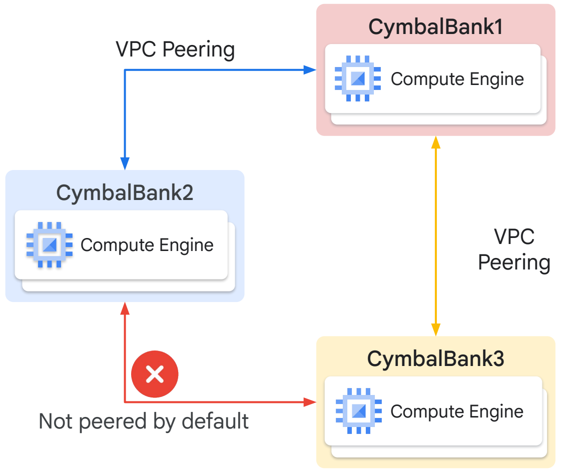
Remember that each VPC network will have firewall rules that define what traffic is allowed or denied between the peered networks.

For example, in this diagram, two organizations represent a consumer and a producer, respectively. Each organization has its own organization node, VPC network, VM instances, Network Admin, and Instance Admin. To successfully establish VPC Network Peering, two peering relationships must be created. The producer network administrator must peer the producer network with the consumer network. The consumer network administrator must peer the consumer network with the producer network. When both peering connections are created, the VPC Network Peering session becomes active and routes are exchanged. The peering relationships let the VM instance use their internal IP addresses to communicate privately.

VPC Network Peering is a decentralized or distributed approach to multi-project networking. Each VPC network may remain in the control of separate administrator groups and maintains its own global firewall and routing tables. Historically, such projects would consider external IP addresses or VPNs to facilitate private communication between VPC networks. However, VPC Network Peering does not incur the network latency, security, and cost drawbacks that are present when you use external IP addresses or VPNs.

## Using VPC Network Peering

- ✓ You can peer Compute Engine, Kubernetes Engine, and App Engine flexible environments.
- ✓ Peered VPC networks remain administratively separate.
- ✓ Each side of a peering association is set up independently.
- ✓ No subnet IP ranges overlap across peered VPC networks.
- ✓ Transitive peering is not supported.



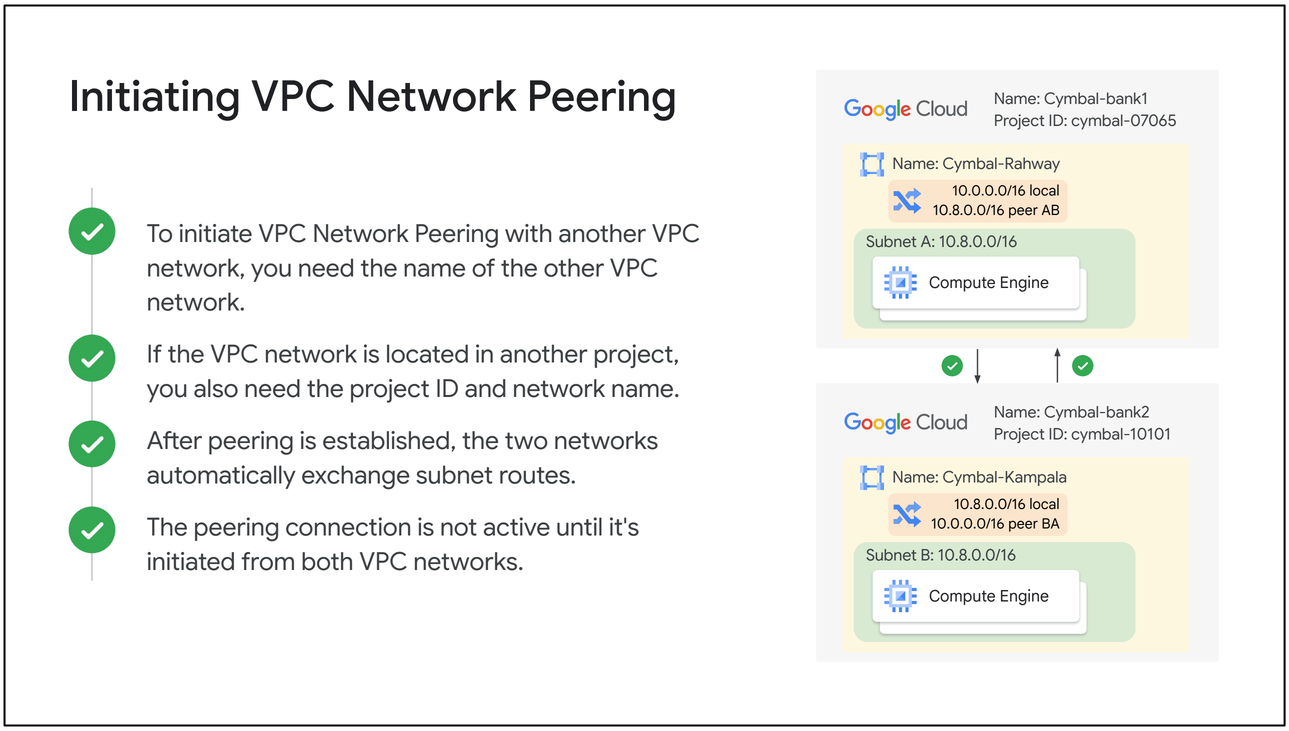
Let's talk about a few points to remember when using VPC Network Peering. First of all, VPC Network Peering works with Compute Engine, Google Kubernetes Engine, and App Engine flexible environments. You need to have the network administrator role to perform peering.

Peered VPC networks remain administratively separate. In other words, routes, firewalls, VPNs, and other traffic management tools are administered and applied separately in each of the VPC networks. These tools are not managed centrally for all network peers.

Each side of a VPC Network Peering association is set up independently. Peering will be active only when the configuration from both sides matches. This arrangement allows either side to delete the peering association at any time.

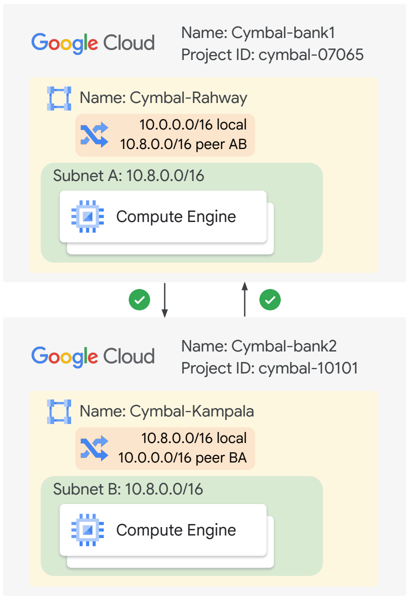
A subnet CIDR prefix in one peered VPC network cannot overlap with a subnet CIDR prefix in another peered network. For example, two auto mode VPC networks that only have the default subnets cannot peer. Google Cloud returns an error if any of the following operations result in an overlap.

Only directly peered networks can communicate, which means that transitive peering is not supported. Let me explain this with an example. Suppose VPC network CymbalBank1 is peered with CymbalBank2 and CymbalBank3. In that scenario, CymbalBank2 and CymbalBank3 are not directly connected. VPC network CymbalBank2 cannot communicate with VPC network CymbalBank3 over a peered connection. If CymbalBank1 offered SaaS services to CymbalBank2 and CymbalBank3, this situation could be critical.



## Initiating VPC Network Peering

- To initiate VPC Network Peering with another VPC network, you need the name of the other VPC network.
- If the VPC network is located in another project, you also need the project ID and network name.
- After peering is established, the two networks automatically exchange subnet routes.
- The peering connection is not active until it's initiated from both VPC networks.



Next, let's explore the steps involved in initiating a VPC peering connection. Before you begin, you must have the name of the VPC network to which you will peer with. If that VPC network is located in another project, you must also have the project ID and network name.

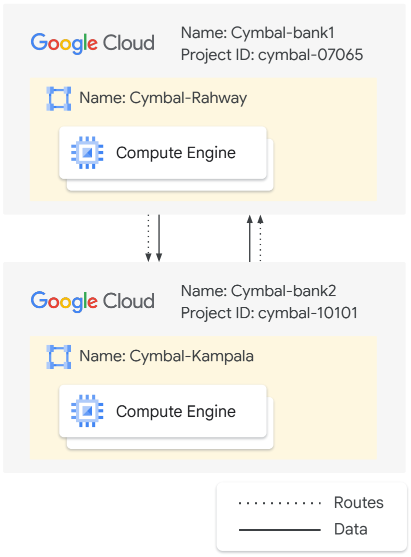
A peering configuration establishes the intent to connect to another VPC network. The VPC networks are not connected until each one has a peering configuration for the other. After the other network has a corresponding configuration to peer with your network, the peering state changes to active in both networks. At that point, the VPC networks are connected through VPC Network Peering. After peering is established, the two networks automatically exchange subnet routes.

Until both VPC networks create peering configurations for each other, no peering connection exists between them. The peering state remains inactive. An inactive peering state indicates that there is not yet a full VPC Network Peering connection.

Suppose you want to peer the two VPC networks shown on the slide. Let’s begin with the Cymbal-Rahway VPC network. You must know the name of the other VPC network, which is Cymbal-Kampala. Because Cymbal-Kampala is not in the same project as Cymbal-Rahway, you must also have the project ID. Cymbal-Kampala is in the Cymbal-bank2 project, whose project ID is cymbal-10101.

## Sharing custom routes

- Custom routes help achieve a desired granularity that cannot be achieved with a default route.
- Sharing custom routes and dynamic with peered VPC networks lets networks learn routes directly from their peered networks.
- If a custom route in a peered network is updated, your VPC network automatically learns and uses the updated custom route.
- You can import and export routes.



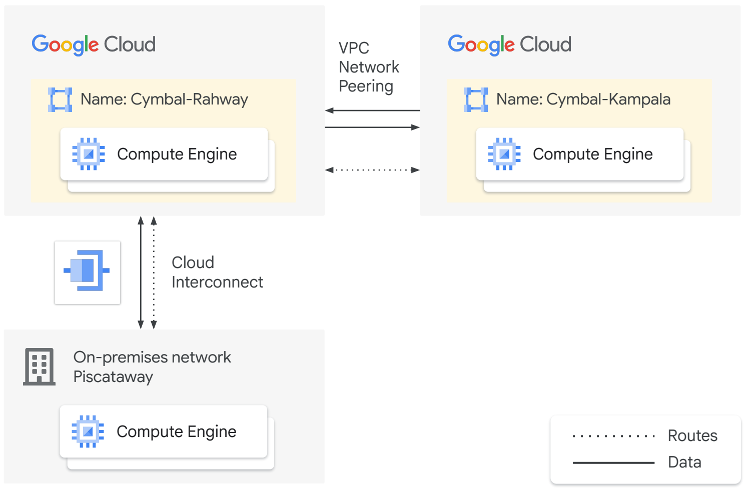
Custom routes help achieve a desired granularity that cannot be achieved with a default route. Sharing custom routes and dynamic routes with peered VPC networks let networks learn routes directly from their peered networks. For example, custom routes can be used to route traffic between subnets through a network virtual appliance.

For example, if a custom route in a peered network is updated, your VPC network automatically learns and uses the updated custom route. You are not required to configure any additional settings in your VPC network.

When you create or modify a peering configuration, you can choose to import routes, export routes, or both. The peer network administrator must similarly configure their peering configuration before routes are exchanged. This process ensures that both network administrators explicitly agree to exchange custom routes before they are exchanged.

## A sample scenario

Share dynamic routes so that peered networks can reach your on-premises network.

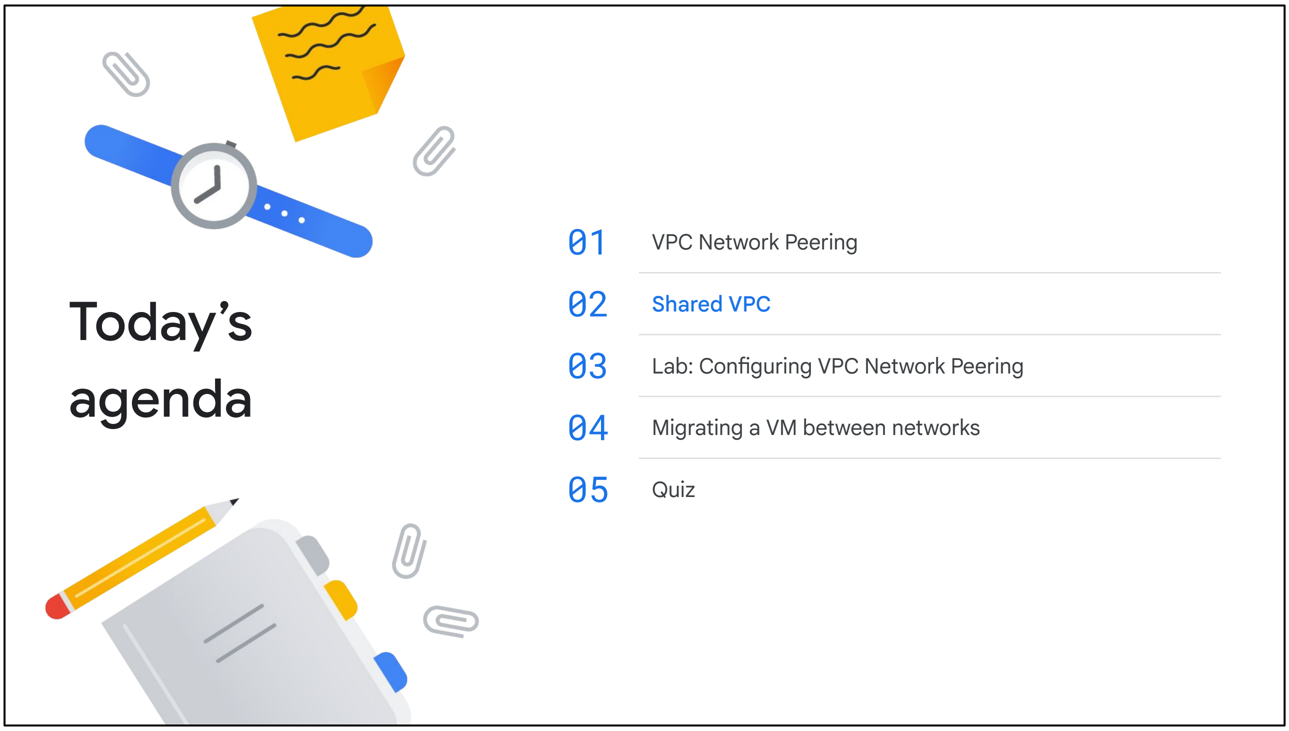


Let’s consider a sample scenario.

Consider you are connecting your on-premises environment through a VPN tunnel or a Cloud Interconnect connection. Cloud Interconnect establishes direct, private connections between your on-premises network and your Google Cloud Virtual Private Cloud (VPC) network. VPN on the other hand provides a secure remote access that establishes an encrypted connection over the public internet.

When using either of these connections, you can share dynamic routes.Sharing these routes enables peered networks to reach your on-premises network, as shown in the graphic. For dynamic routes, you must add Cloud Router custom route advertisements in your VPC network. These advertisements announce peered network subnets to your on-premises network.

Local routes are always preferred over dynamic routes that are learned by using VPC Network Peering.



## Today's agenda

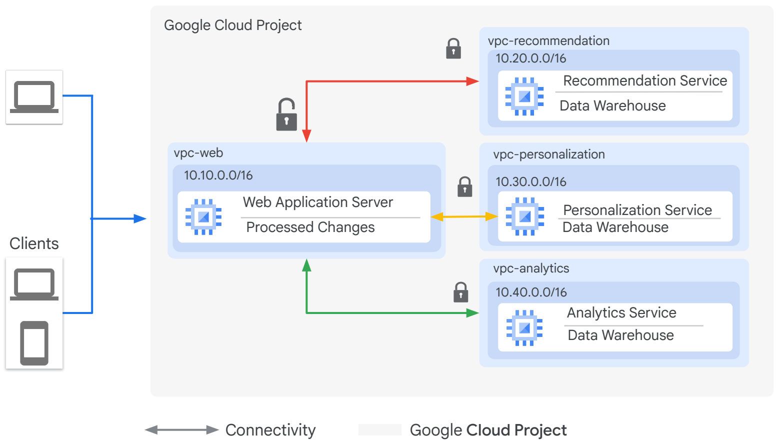
In this module, we are going to cover two configurations for sharing VPC networks across Google Cloud projects.

First, we will go over shared VPC, which allows you to share a network across several projects in your Google Cloud organization.

Then, we will go over VPC Network Peering, which allows you to configure private communication across projects in the same or different organizations.

Let's start by talking about shared VPC.

## Use case: Complexity managing isolated networks



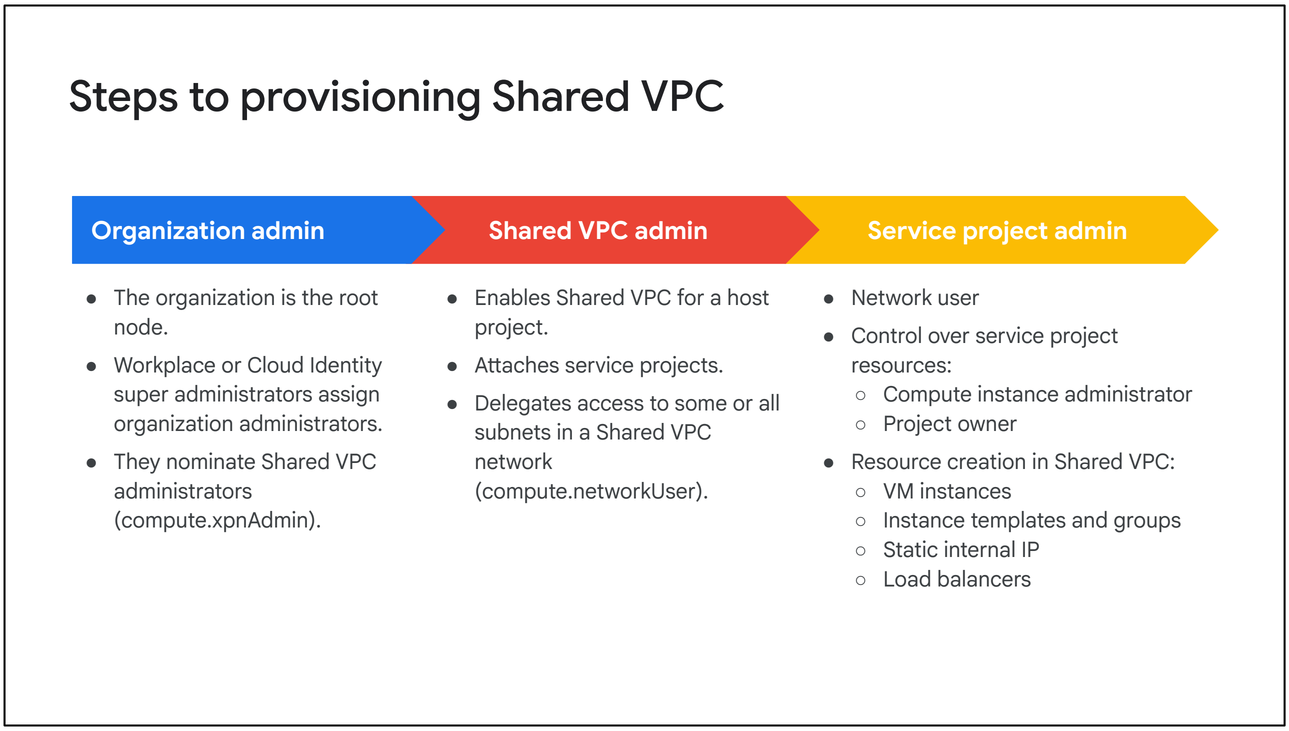
Kim is a network administrator at Cymbal Corporation. Cymbal has four networks that belongs to the Web Application Server’s project, namely the Recommendation, personalization, analytics and web. The recommendation Service, the Personalization Service, and the Analytics Service are each hosted on a different VPC.

In this model, Kim is facing challenges such as time-consuming management of network, difficulty scaling, and limitation communicating between networks. In addition to this, it is a constant churn to set up access policy, new projects, billing, etc,.Kim would like to centrally manage the VPC networks while still segregating workloads that use these networks. What should Kim do?

Kim can benefit by implementing Shared VPC among these networks.

When you use shared VPC, you designate a project as a host project and attach one or more other service projects to it. In this case, the Web Application Server’s project is the host project, and the three other projects are the service projects.

The overall VPC network is called the shared VPC network.



## Steps to provisioning Shared VPC

**Organization admin**

- The organization is the root node.
- Workplace or Cloud Identity super administrators assign organization administrators.
- They nominate Shared VPC administrators (compute.xpnAdmin).

**Shared VPC admin**

- Enables Shared VPC for a host project.
- Attaches service projects.
- Delegates access to some or all subnets in a Shared VPC network (compute.networkUser).

**Service project admin**

- Network user
- Control over service project resources:
  - Compute instance administrator
  - Project owner
- Resource creation in Shared VPC:
  - VM instances
  - Instance templates and groups
  - Static internal IP
  - Load balancers

Shared VPC makes use of Cloud IAM roles for delegated administration.

Let me walk through how to provision Shared VPC by focusing on the required administrative roles.

The first required role is the organization administrator. The Organization resource represents an organization, for example, a company, and is the root node in the Google Cloud resource hierarchy.

The Google Workplace or Cloud Identity super administrators are the first users who can access the organization, and they assign the organization admin role to users.

The organization admin's role in provisioning shared VPC is to nominate Shared VPC Admins by granting them appropriate project creation and deletion roles, and the compute.xpnAdmin role for the organization.

Note that shared VPC is also referred to as "XPN" in the API and command-line interface.

Next, the Shared VPC administrator performs various tasks necessary to set up Shared VPC, such as enabling Shared VPC on the host project, attaching service projects to the host project, and delegating access to some or all of the subnets in Shared VPC networks to Service Project Admins by granting the compute.networkUser role.

Typically, a Shared VPC administrator is also the project owner for a given host project.

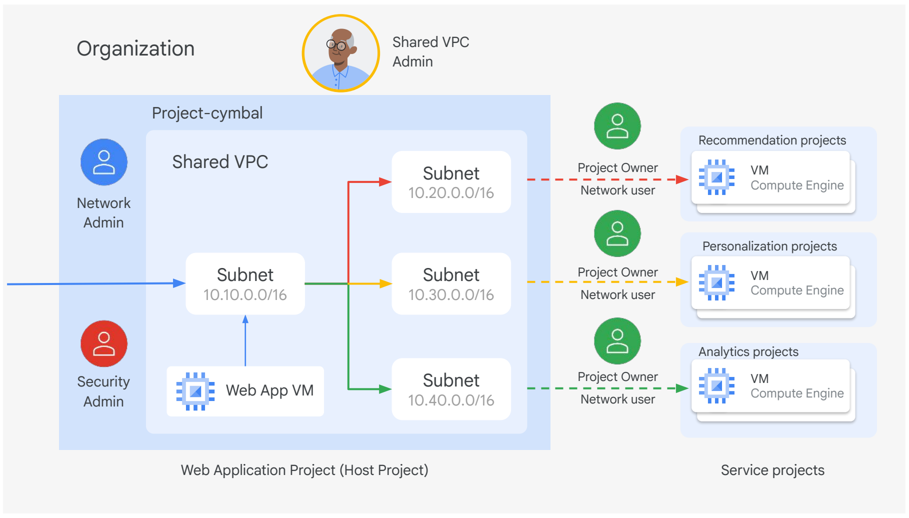
In addition to being a network user, service project administrators also maintain ownership and control over resources defined in their service projects.

The service project administrators must at least have the compute.instanceAdmin role to the corresponding service project. However, typically, the service project administrators are project owners of their service projects. This allows them to create and manage resources in the shared VPC.

These resources could be:

- VM instances
- Instance templates and groups
- Static internal IP addresses
- And load balancers

## Use case: Centralize Control using Shared VPC

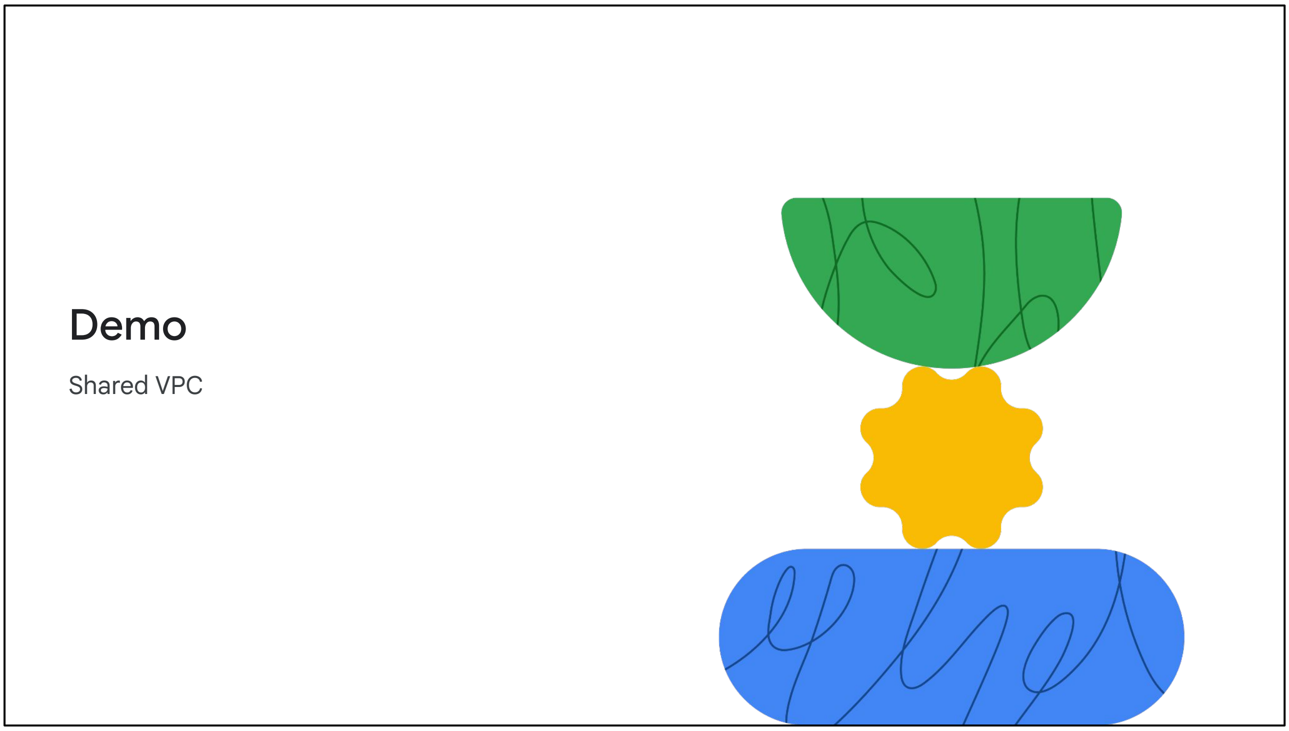


Let's come back to our original example that had one host project and 3 service projects:

In this diagram, the Shared VPC administrator, which was nominated by an organization administrator, configured the web application project to be a host project with subnet-level permissions. Doing so allowed the Shared VPC administrator to selectively share subnets from the VPC network.

Next, the Shared VPC administrator attached the three service projects to the host project and gave each project owner the Network User role for the corresponding subnets. Each project owner then created VM instances from their service projects in the shared subnets. By the way, billing for those VM instances is attributed to the project where the resources are created, which are the service projects.

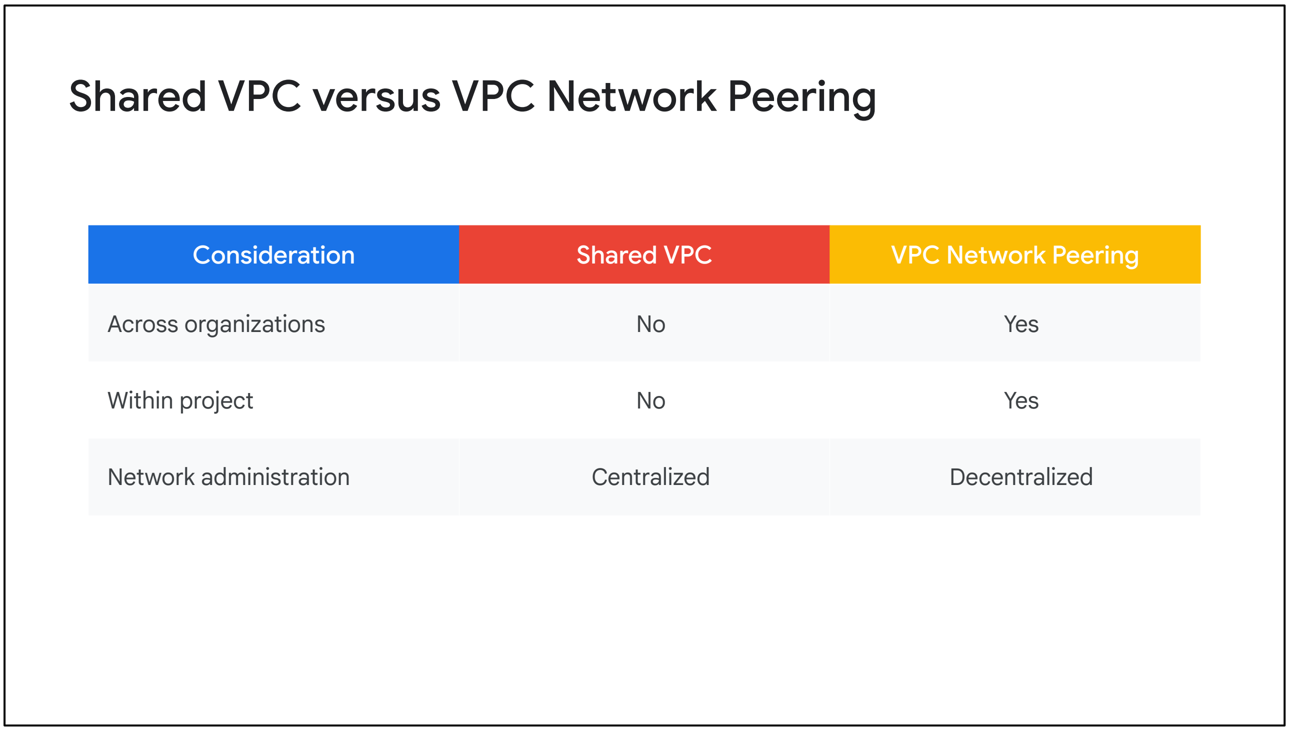
Shared VPC administrators have full control over the resources in the host project, including administration of the shared VPC network. They can optionally delegate the network administrator and security administrator roles for the host project. Overall, Shared VPC is a centralized approach to multi-project networking because security and network policy occurs in a single designated VPC network.



## Demo

Shared VPC

Next, let’s watch a demo that describes how to set up Shared VPC. We’ll see what must done to configure the host project and service projects.



## Shared VPC versus VPC Network Peering

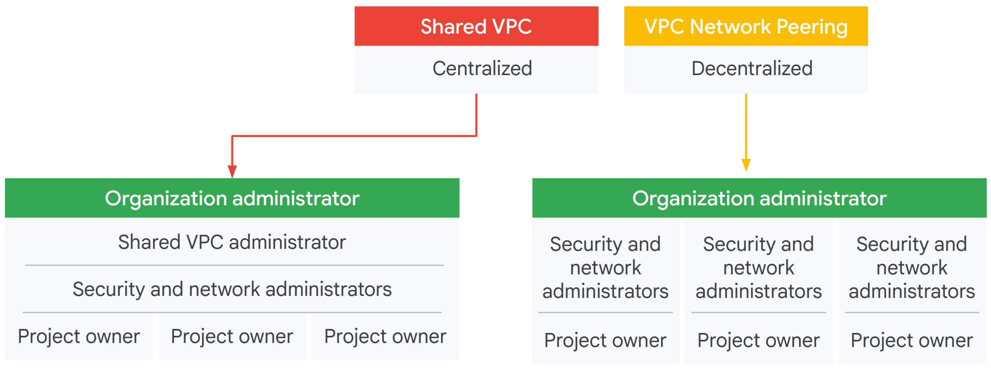
| Consideration | Shared VPC | VPC Network Peering |
|---|---|---|
| Across organizations | No | Yes |
| Within project | No | Yes |
| Network administration | Centralized | Decentralized |

Let’s compare both of these configurations to help you decide which one is appropriate for a given situation.

If you want to configure private communication between VPC networks in different organizations, you must use VPC Network Peering. Shared VPC only works within the same organization.

If you want to configure private communication between VPC networks in the same project, you must use VPC Network Peering. The VPC networks can be in the same project, but it’s not required. Shared VPC only works across projects.

## Shared VPC versus VPC Network Peering


Shared VPC
Centralized
VPC Network Peering
Decentralized
Organization administrator
Shared VPC administrator
Security and network administrators
Project owner
Project owner
Project owner
Organization administrator
Security and network administrators
Security and network administrators
Security and network administrators
Project owner
Project owner
Project owner

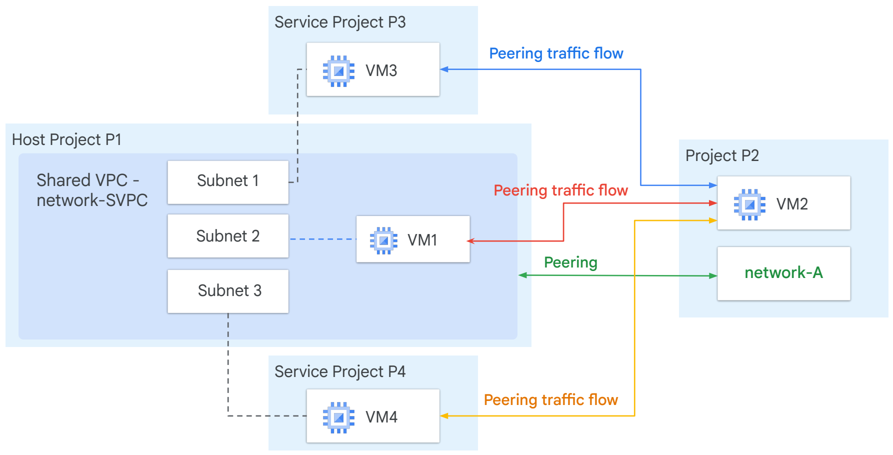
In our opinion, the biggest difference between the two configurations is the network administration models. Shared VPC is a centralized approach to multi-project networking, because security and network policy occurs in a single designated VPC network.

In contrast, VPC Network Peering is a decentralized approach. Each VPC network is controlled by administrator groups in that VPC network's organization. Each VPC network can maintain its own global firewall and routing tables.

For more information about the limits of VM instances per VPC network, see Per Network on the Quotas and Limits page of the Google Cloud documentation.

Now that we've compared both of these configurations to share VPC networks across Google Cloud projects, let's look at one last use case.

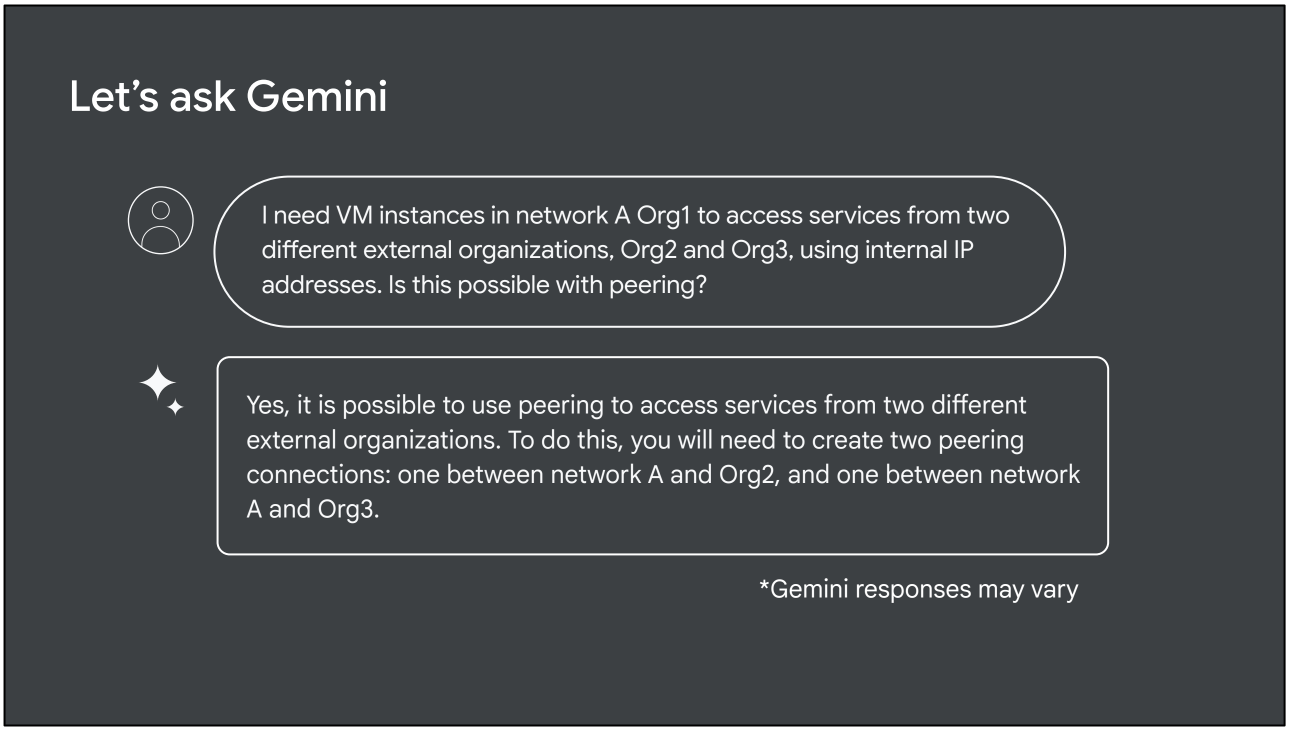
## Peering with a Shared VPC network



In Google Cloud, you can peer with a Shared VPC network. Here you can see an example of a Shared VPC network called network-SVPC. network-SVPC is in host project P1. Service projects P3 and P4 can attach VM instances to network-SVPC, which enables private communication between VMs 1, 2, and 4.

If we establish a peering session between network-A and network-SVPC, all VM instances will have private, internal IP connectivity. Each VPC network has firewall rules that define which traffic is allowed or denied between the networks.

You can also set up VPC Network Peering between two Shared VPC networks.



## Let’s ask Gemini

I need VM instances in network A Org1 to access services from two different external organizations, Org2 and Org3, using internal IP addresses. Is this possible with peering?

Yes, it is possible to use peering to access services from two different external organizations. To do this, you will need to create two peering connections: one between network A and Org2, and one between network A and Org3.

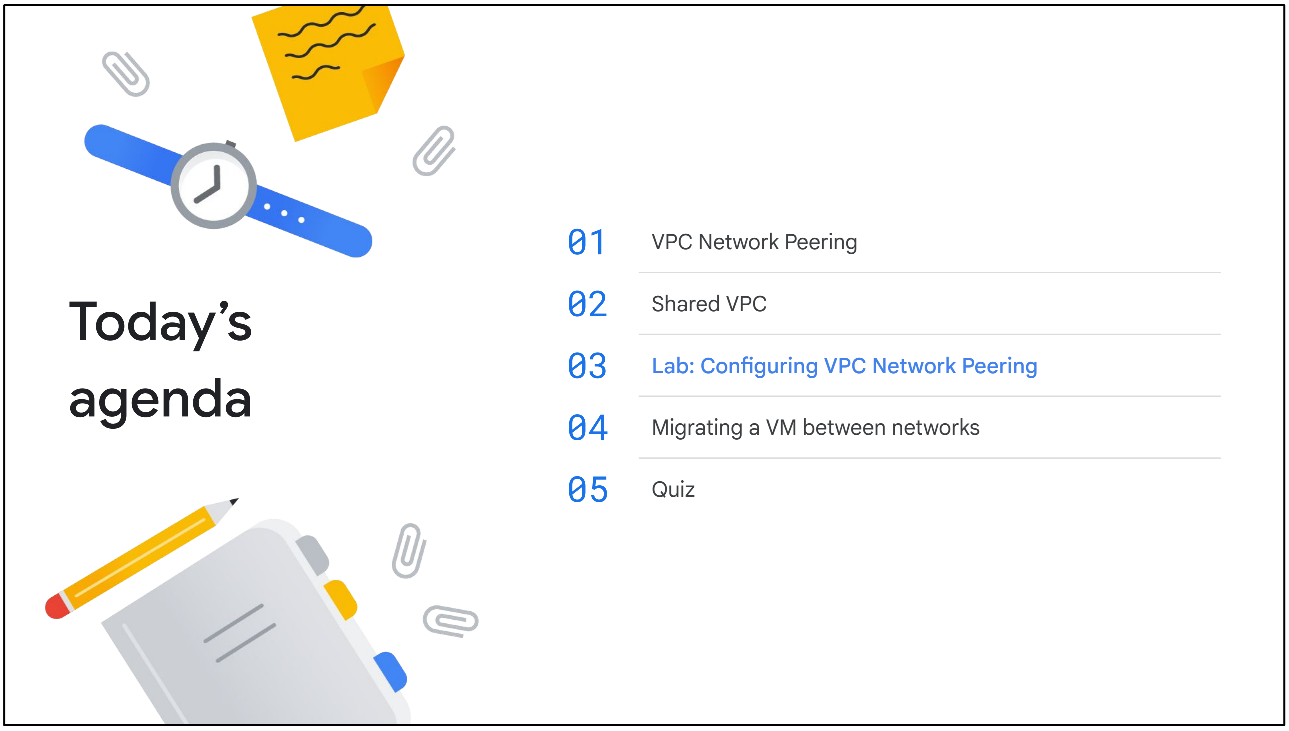
*Gemini responses may vary

If you still are not sure where to start and how, you can always lean on Gemini an AI-powered collaborator in Google Cloud, for some help. Wondering how?

Ensure that Gemini is set up for your Google Cloud user account and project. Then, you are all set to simply chat with Gemini to get help. Using the Gemini pane, you enter prompts, which are questions or statements that describe the help you want, and Gemini returns responses.

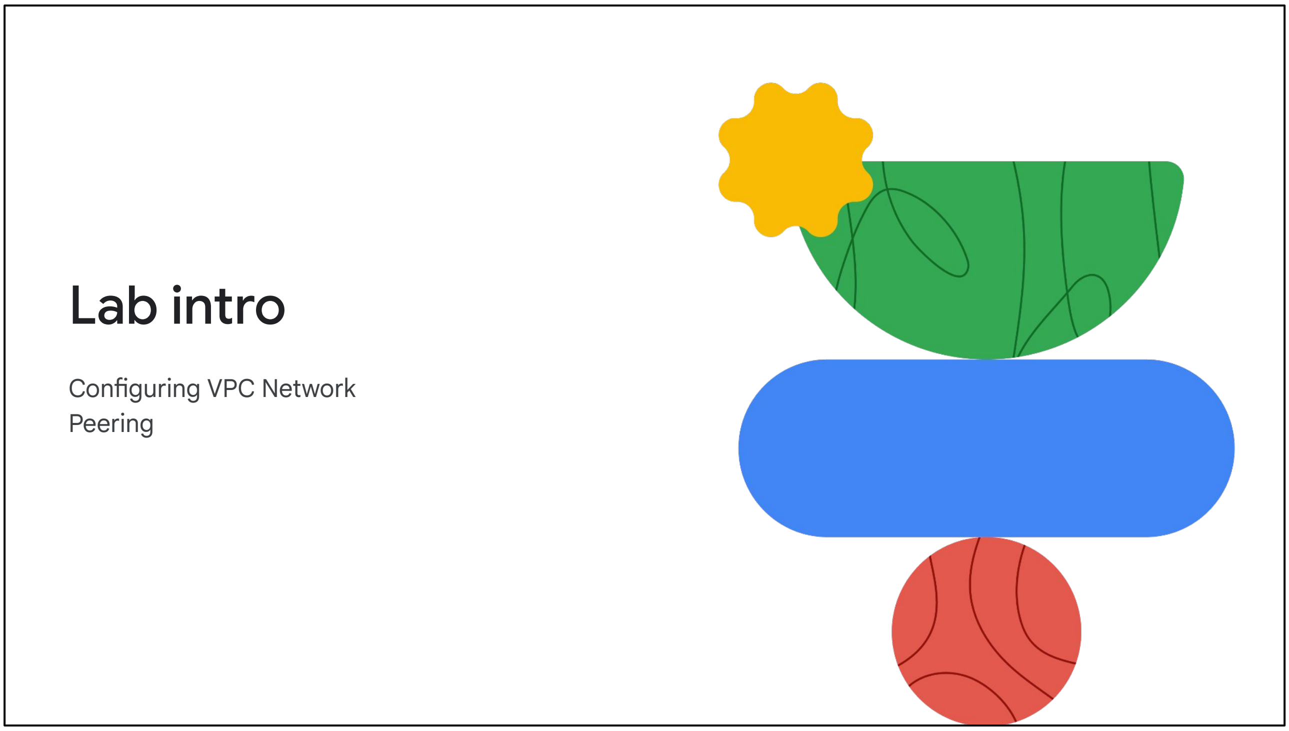
In the Google Cloud console toolbar, click chat_spark Open Gemini and type your prompt. In this case, let’s ask a peering question. I need VM instances in network A Org1 to access services from two different external organizations, Org2 and Org3, using internal IP addresses. Is this possible with peering?

Gemini will guide you through the process and also provide steps. How cool is that?



## Today's agenda

Next, you will complete a lab exercise.

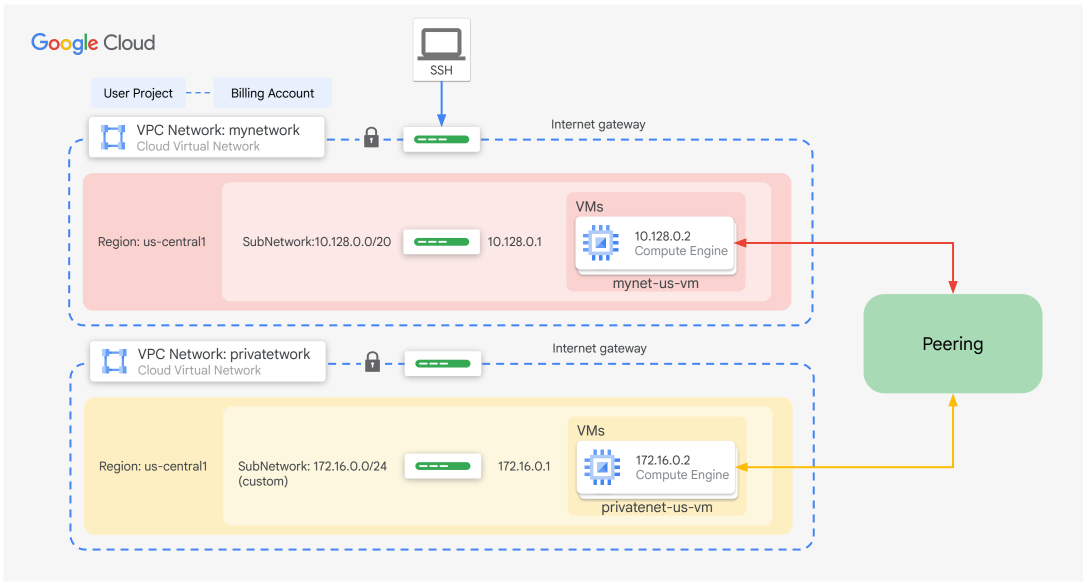


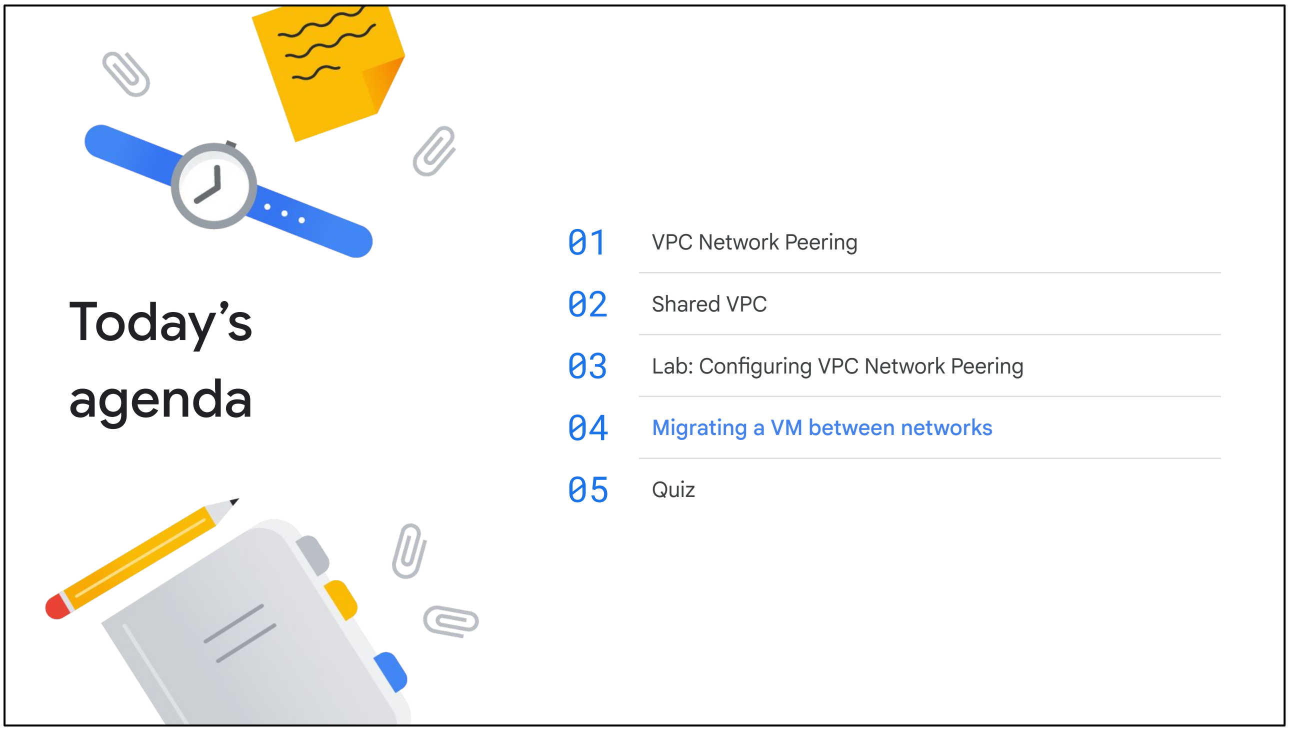
## Lab intro

Configuring VPC Network Peering

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

- Explore connectivity between non-peered VPC networks.
- Configure VPC Network Peering.
- Verify private communication between peered VPC networks.
- Delete VPC Network Peering.


Google Cloud
SSH
User Project
Billing Account
VPC Network: mynetwork
Cloud Virtual Network
Internet gateway
Region: us-central1
SubNetwork:10.128.0.0/20
10.128.0.1
VMs
10.128.0.2
Compute Engine
mynet-us-vm
Peering
VPC Network: privatetwork
Cloud Virtual Network
Internet gateway
Region: us-central1
SubNetwork: 172.16.0.0/24
(custom)
172.16.0.1
VMs
172.16.0.2
Compute Engine
privatenet-us-vm

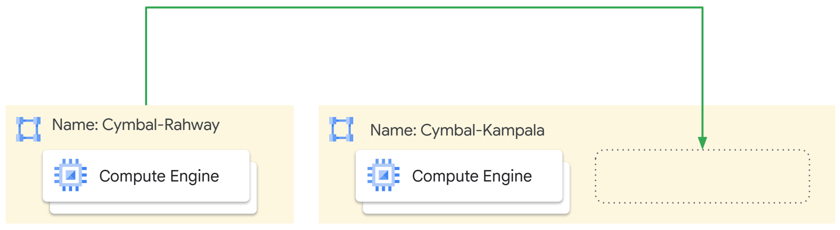


## Today's agenda

Finally, we discuss how to migrate a VM instance from one network to another.

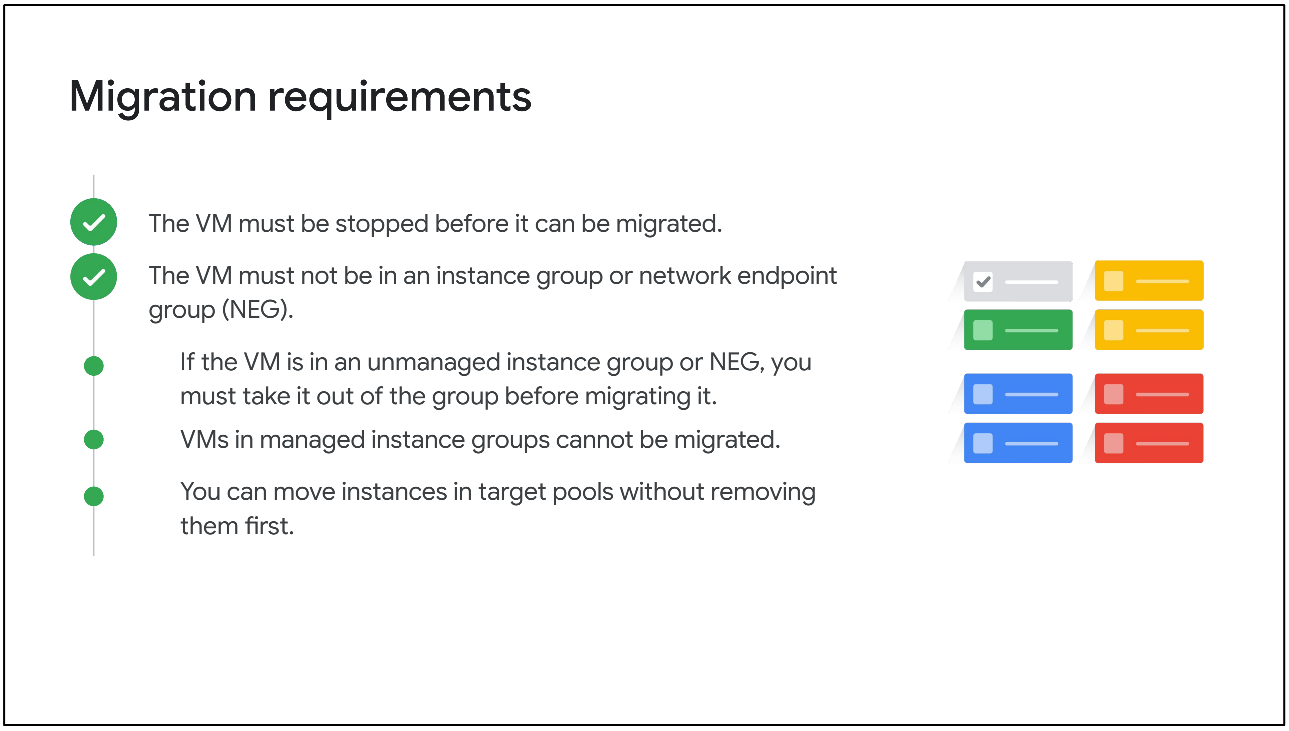
When a VM is connected to more than one network using multiple interfaces, the migration process updates one of the interfaces and leaves the rest in place.

## Supported migrations



Luka needs to migrate a VM from one VPC network to another. What can she do?

Before she migrates it, it is important that she is aware of the supported migration requirements and limitations. So let’s go over them.

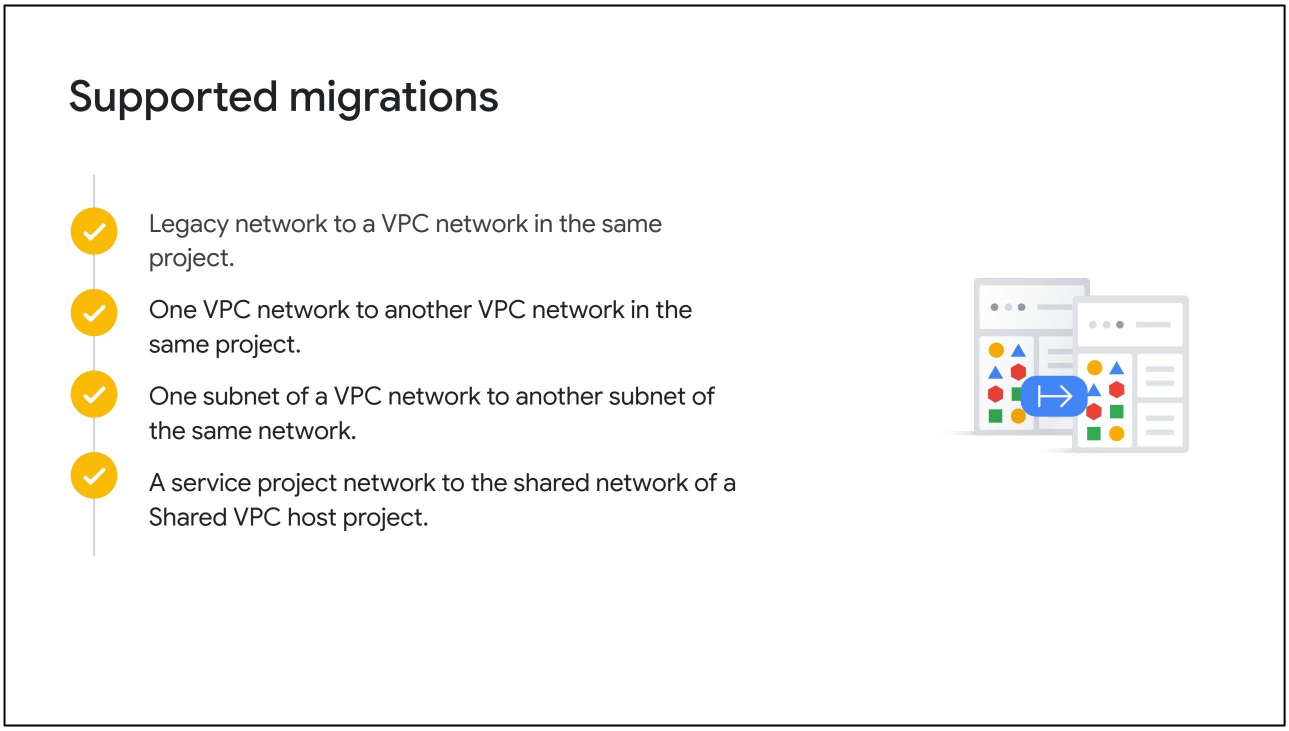


## Migration requirements

- The VM must be stopped before it can be migrated.
- The VM must not be in an instance group or network endpoint group (NEG).
  - If the VM is in an unmanaged instance group or NEG, you must take it out of the group before migrating it.
  - VMs in managed instance groups cannot be migrated.
  - You can move instances in target pools without removing them first.

Before you migrate a VM, you must meet the following requirements:

- The migration is a "cold" migration. The VM must be stopped before it can be migrated.
- The VM must not be in an instance group or network endpoint group (NEG).
  - If the VM is in an unmanaged instance group or NEG, you must take it out of the group before migrating it.
  - VMs in managed instance groups cannot be migrated. Instead, you must copy your instance template to the new network and use it to rebuild the managed instance group.
  - A target pool is a group of Google Compute Engine instances that receive incoming traffic from a load balancer. You can move instances in target pools without removing them first. The target pool expands to cover both networks.



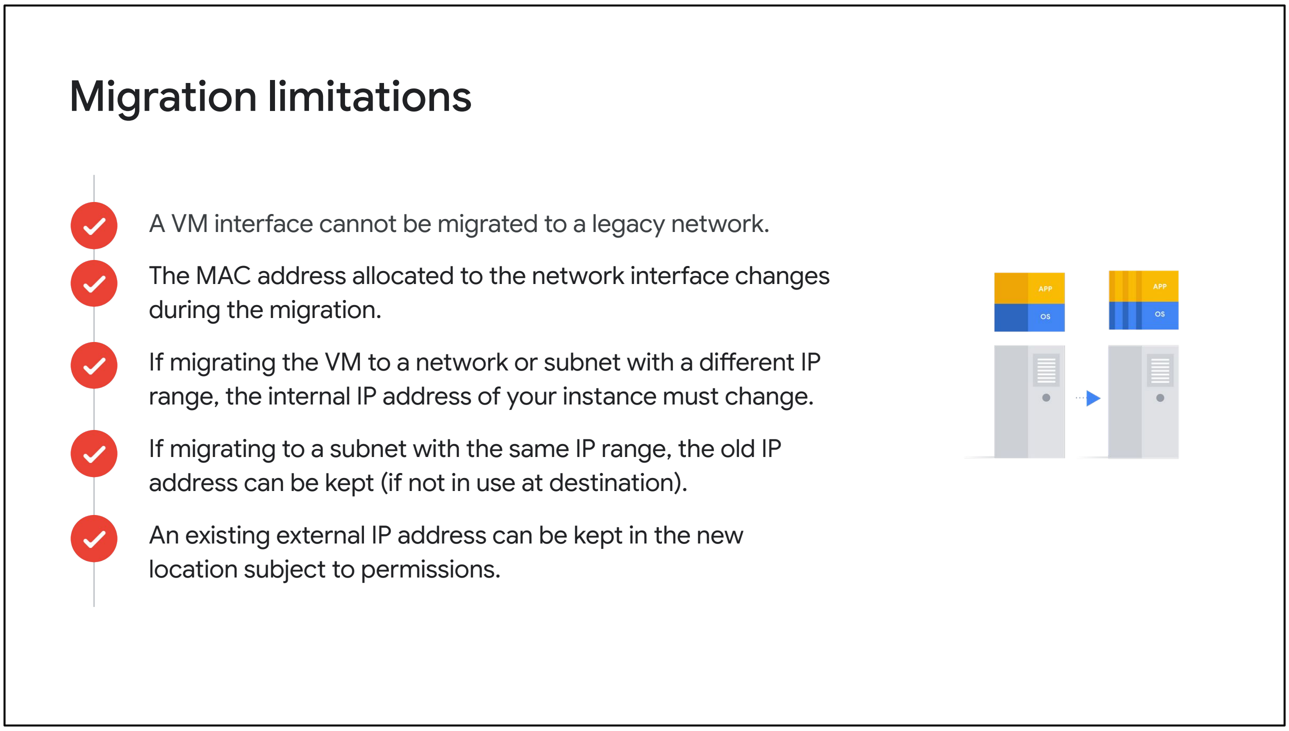
## Supported migrations

- ✓ Legacy network to a VPC network in the same project.
- ✓ One VPC network to another VPC network in the same project.
- ✓ One subnet of a VPC network to another subnet of the same network.
- ✓ A service project network to the shared network of a Shared VPC host project.

The migrations supported are:

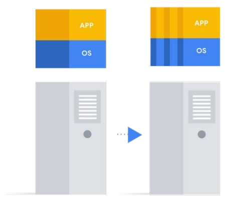
- From legacy network to a VPC network in the same project.
- From one VPC network to another VPC network in the same project.
- From one subnet of a VPC network to another subnet of the same network.
- From a service project network to the shared network of a Shared VPC host project.

In all cases, the VM stays in the region and zone where it was before. Only the attached network changes.



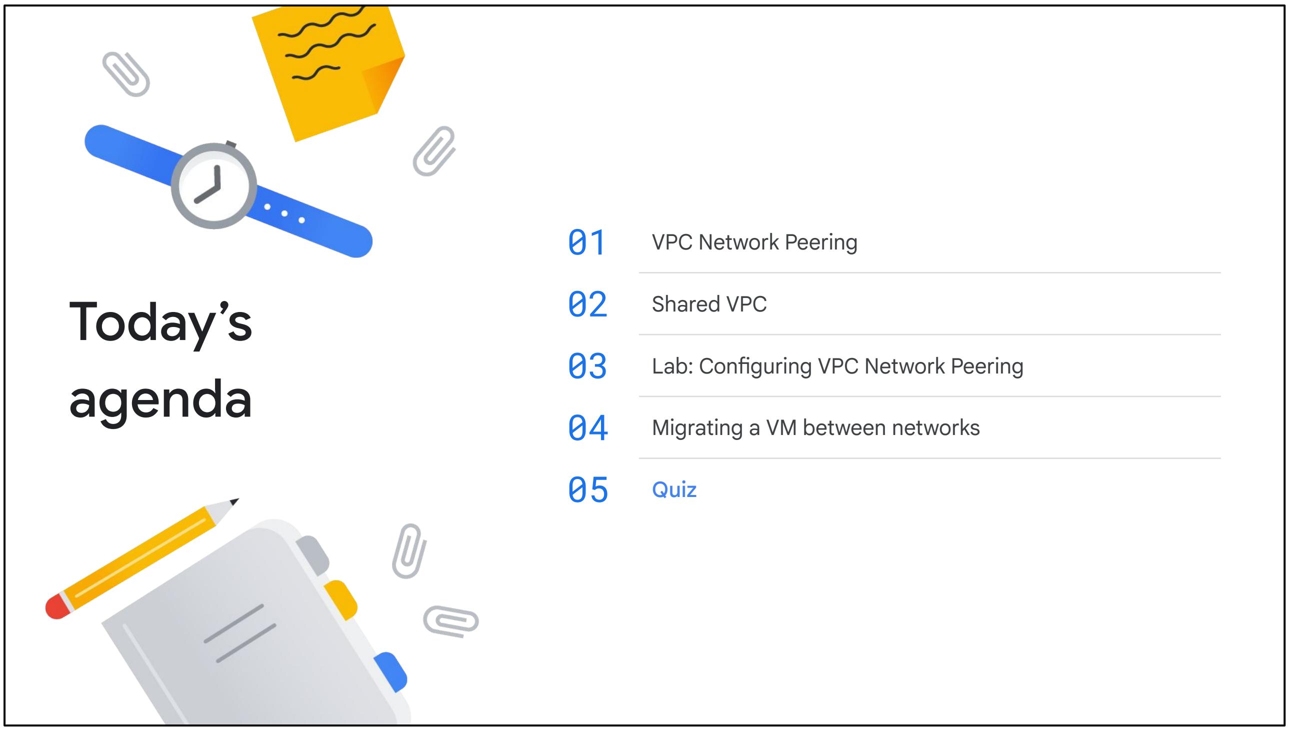
## Migration limitations

- ✓ A VM interface cannot be migrated to a legacy network.
- ✓ The MAC address allocated to the network interface changes during the migration.
- ✓ If migrating the VM to a network or subnet with a different IP range, the internal IP address of your instance must change.
- ✓ If migrating to a subnet with the same IP range, the old IP address can be kept (if not in use at destination).
- ✓ An existing external IP address can be kept in the new location subject to permissions.



There are limitations to migrating that you should consider.

- You cannot migrate a VM interface to a legacy network.
- The MAC address allocated to the network interface will change during the migration. This could have an impact on services tightly coupled with MAC addresses such as third-party license agreements.
- If you are migrating the VM to a network or subnet with a different IP range, the internal IP address of your instance must change.
- If you are migrating to a subnet with the same IP range, you can keep the old IP address, as long as it is not already in use at the destination, by specifying it during the migration.
- You can keep the VM's existing external IP address in the new location. However, to do this you must have the `compute.subnetworks.useExternalIp` permission on the target network, and the target network cannot have external IP addresses disabled by the constraints/compute.vmExternalIpAccess constraint.



## Today's agenda

Let's test your knowledge.

## Quiz | Question 1

### Question

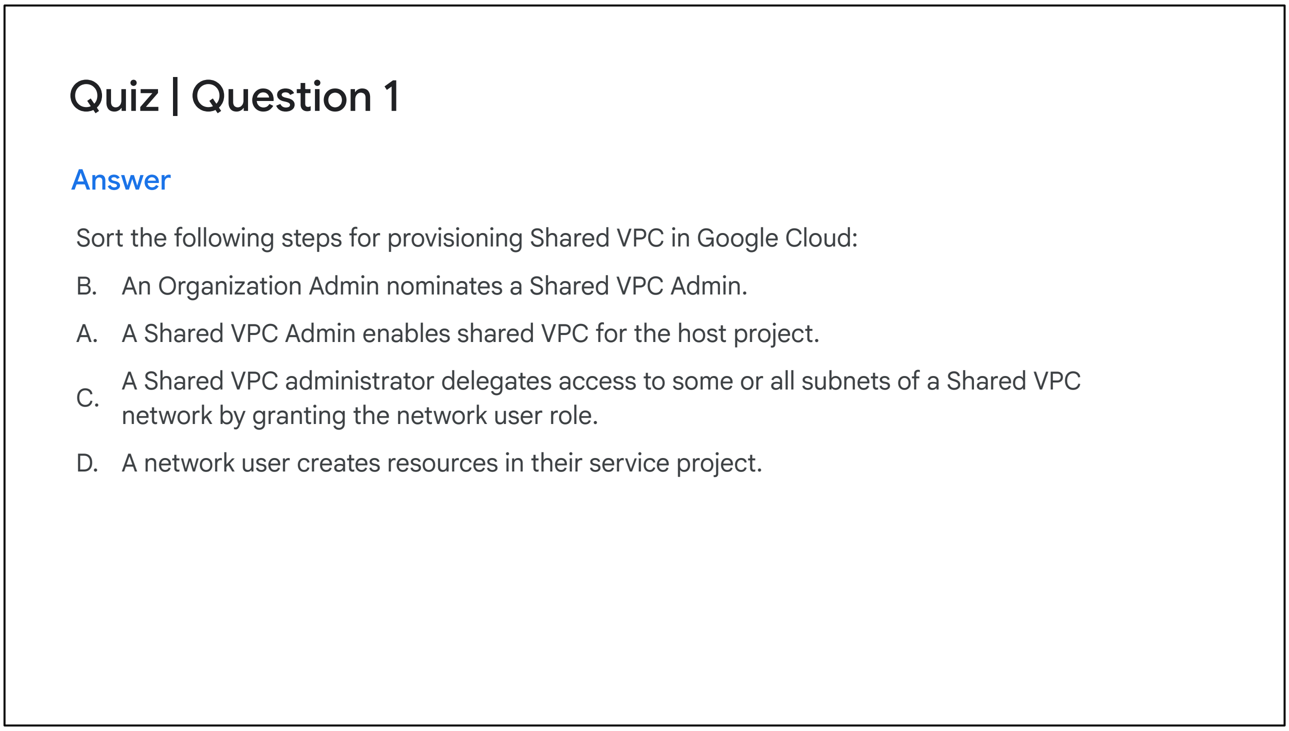
Sort the following steps for provisioning Shared VPC in Google Cloud:

A. A Shared VPC administrator enables Shared VPC for the host project.

B. An organization administrator nominates a Shared VPC administrator.

C. A Shared VPC Admin delegates access to some or all subnets of a shared VPC network by granting the Network User role.

D. A Network User creates resources in their Service Project.



## Quiz | Question 1

Answer

Sort the following steps for provisioning Shared VPC in Google Cloud:

B. An Organization Admin nominates a Shared VPC Admin.

A. A Shared VPC Admin enables shared VPC for the host project.

C. A Shared VPC administrator delegates access to some or all subnets of a Shared VPC network by granting the network user role.

D. A network user creates resources in their service project.

**Explanation:**
First, an organization administrator nominates a Shared VPC administrator.
The Shared VPC administrator then enables Shared VPC for the host project.
Next, the Shared VPC administrator delegates access to some or all subnets of a shared VPC network by granting the network user role.
Finally, the network user creates resources in their service project.

## Quiz | Question 2

### Question

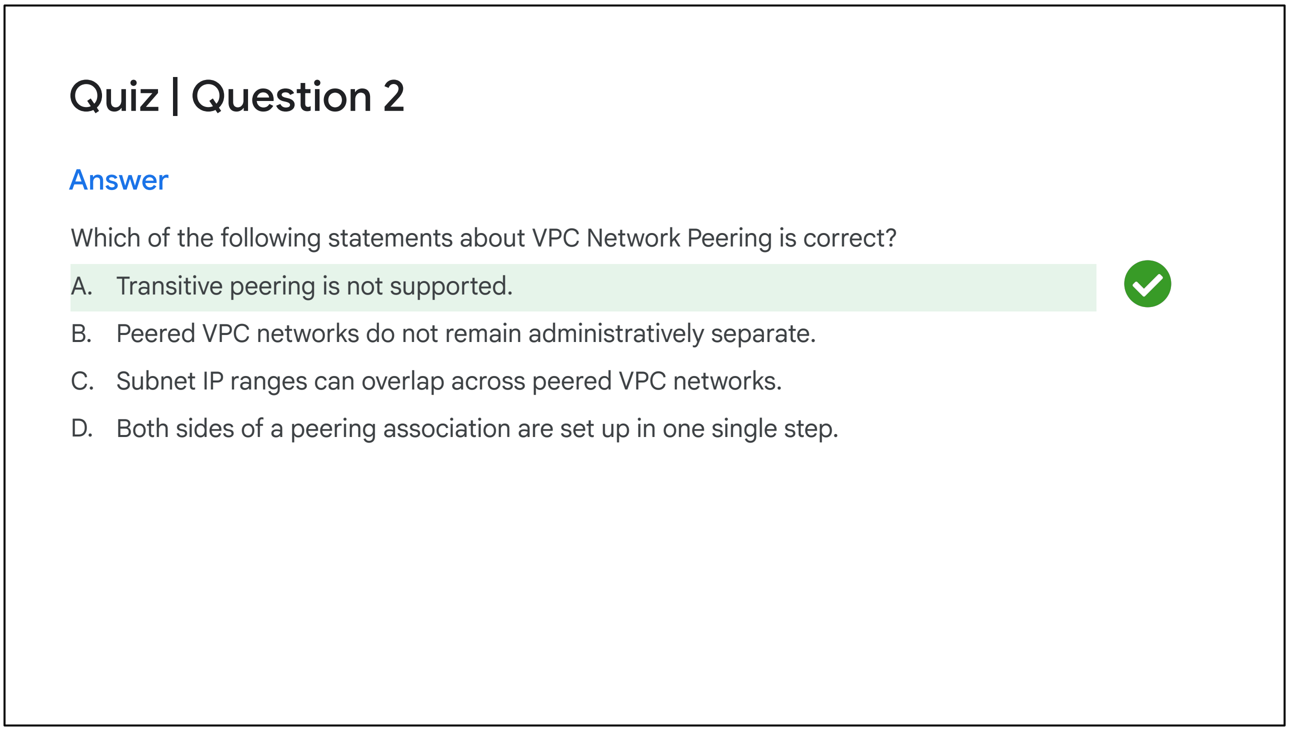
Which of the following statements about VPC Network Peering is correct?

A. Transitive peering is not supported.

B. Peered VPC networks do not remain administratively separate.

C. Subnet IP ranges can overlap across peered VPC networks.

D. Both sides of a peering association are set up in one single step.



## Quiz | Question 2

### Answer

Which of the following statements about VPC Network Peering is correct?

A. Transitive peering is not supported. ✓

B. Peered VPC networks do not remain administratively separate.

C. Subnet IP ranges can overlap across peered VPC networks.

D. Both sides of a peering association are set up in one single step.

**Explanation**:

A. Correct! Transitive peering is not supported in VPC Network Peering.

B. That's incorrect. Peered VPC networks remain administratively separate. This means that routes, firewalls, VPNs, and other traffic management tools are administered and applied separately in each of the VPC networks.

C. That's incorrect. Subnet IP ranges cannot overlap across peered VPC networks.

D. That's incorrect. Each side of a peering association is set up independently.

## Quiz | Question 3

### Question

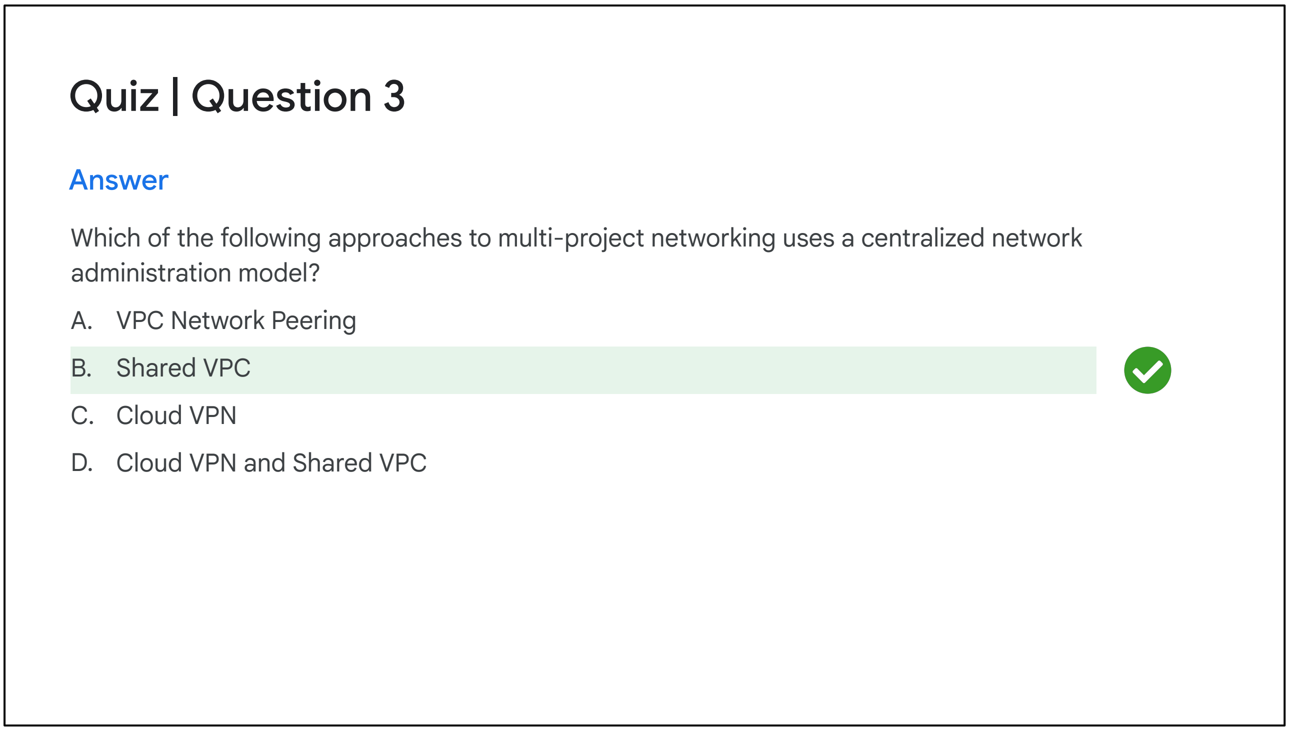
Which of the following approaches to multi-project networking uses a centralized network administration model?

A. VPC Network Peering

B. Shared VPC

C. Cloud VPN

D. Cloud VPN and Shared VPC



## Quiz | Question 3

Answer

Which of the following approaches to multi-project networking uses a centralized network administration model?

A. VPC Network Peering

B. [✓ Shared VPC]

C. Cloud VPN

D. Cloud VPN and Shared VPC

**Explanations:**

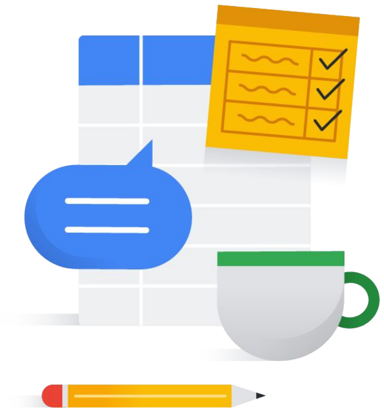
A. That’s incorrect. VPC Network Peering is a decentralized approach, because each VPC network can remain under the control of separate administrator groups and maintains its own global firewall and routing tables.

B. Correct! Shared VPC is a centralized approach to multi-project networking, because security and network policy occurs in a single designated VPC network.

C. That’s incorrect. Please review the course material.

D. That’s incorrect. Please review the course material.

## Debrief



In this module, we looked at Shared VPC and VPC Network Peering, which are two configurations for sharing VPC networks across Google Cloud projects.

You got to explore VPC Network Peering in a lab, and we compared both configurations and their network administration models to help you decide when to choose which.

Google Cloud's flexibility to support multiple approaches to network administration allows organizations like yours to more carefully map resource policies, administrative controls, and related accounting to existing structures.

In addition, administrators can carefully control the manner in which environments interact with each other, on-premises networks, and the public internet.

Thank you.

THANK YOU