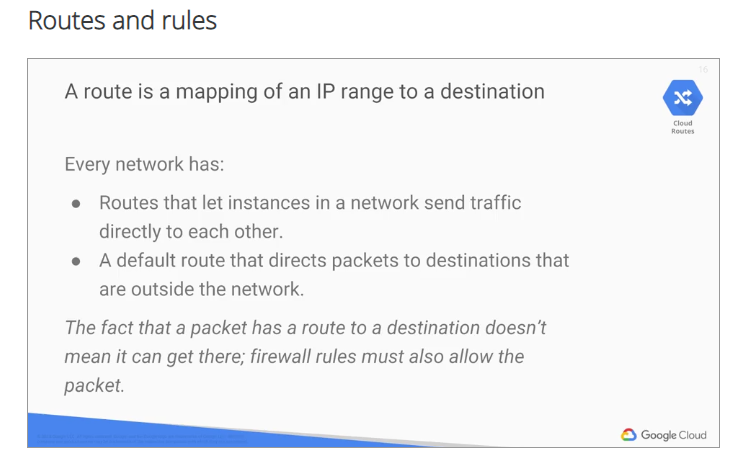
Routes and rules – How GCP routes traffic

# Routes

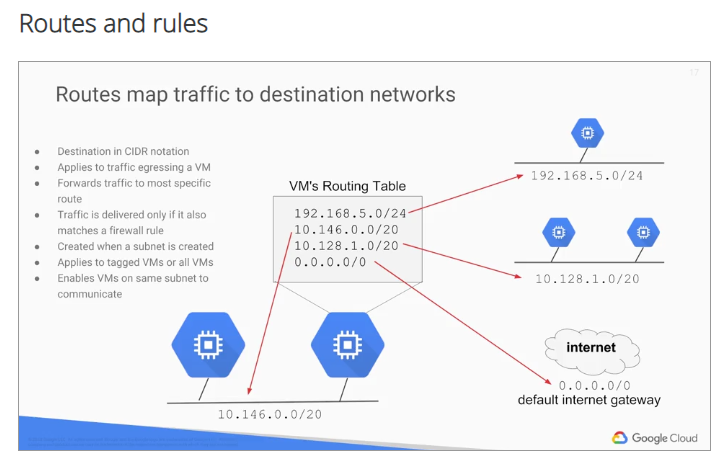
By default, every network has routes that let instances in a network sent traffic directly to each other even across subnets. In addition, every network has **a default route** that directs packets to destinations that are outside the network. Although these routes cover most of your normal routing needs, you can also create special routes that override these routes. Just because a packet has a route to a destination does not mean that it can get there. Firewall rules must also allow the packet.



The **default network** has preconfigured firewall rules that allow all instances in the network to talk with each other.

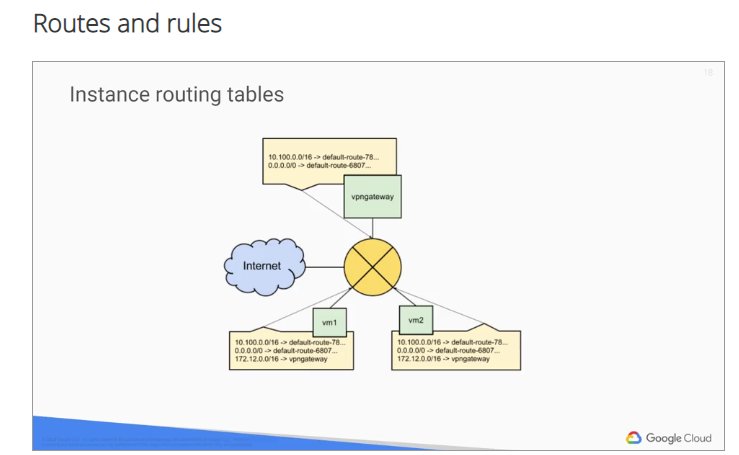
Manually-created networks do not have such rules. So, you must create them as you will experience in the first lab.

Routes match packets by destination IP address. However, no traffic will flow without also matching a firewall rule. A route is created when a network is created, enabling traffic delivery from anywhere. Also, a route is created when the subnet is created. This is what enables VMs on the same subnet to communicate internally.



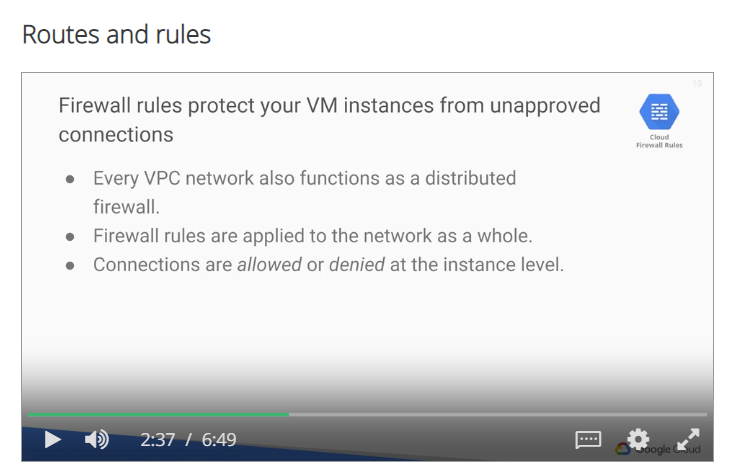
This slide shows a simplified routing table, but let's look at this in more detail.

Each route in the route's collection may apply to one or more instances. A route applies to an instance, if the network and instance tags match. If the network matches and there are no instance tags specified, the route applies to all instances in that network. Compute Engine then uses the route's collection to create individual read-only routing tables for each instance. This diagram shows a massively scalable virtual router at the core of each network. Every virtual machine instance in the network is directly connected to this router, and all packets leaving a virtual machine instance are first handled at this layer before they are forwarded on their next hop. The virtual network router selects the next hop for packet by consulting the writing table for that instance.



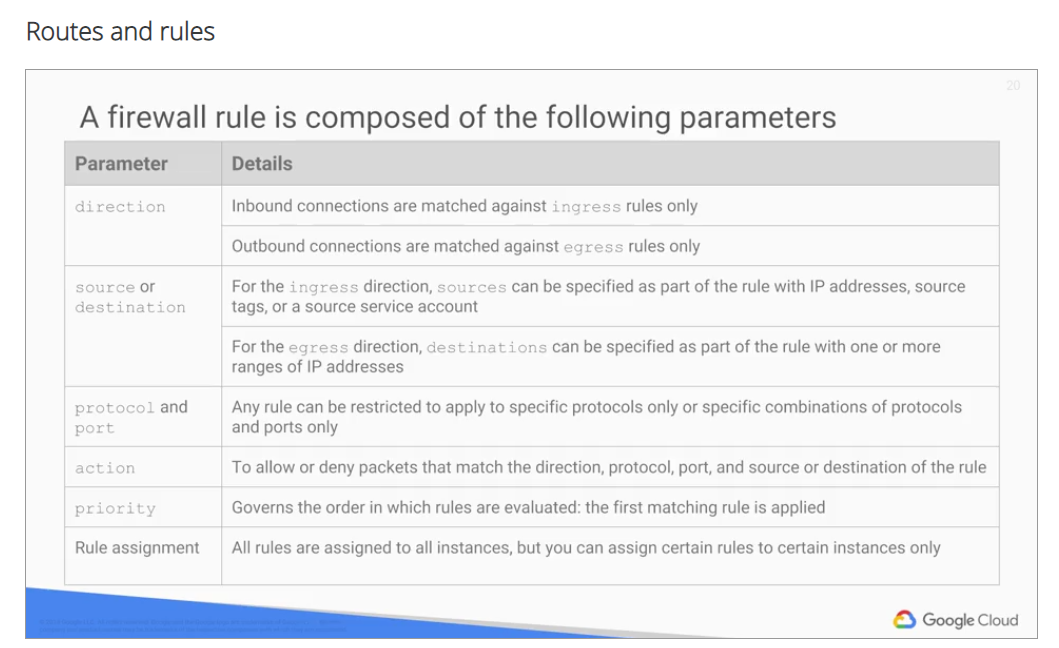
In this diagram, the green boxes are virtual machine instances, the router is the yellow at the center, and the individual routing tables are indicated by tan boxes.

# Firewall



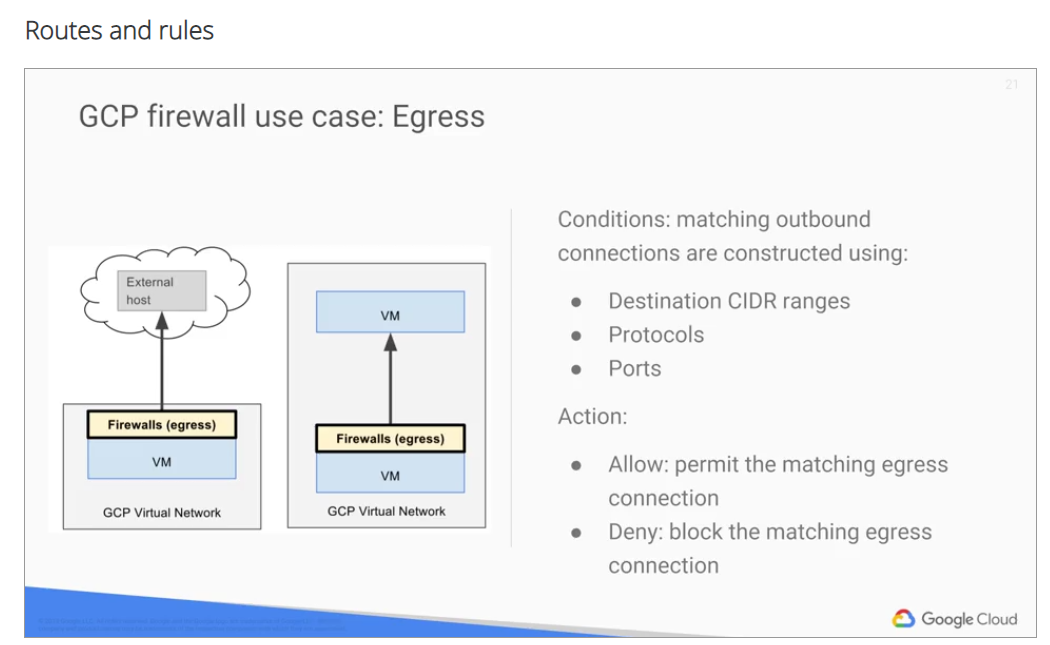
GCP firewall rules protect your virtual machine instances from unapproved connections, *both inbound and outbound known as ingress and egress respectively*.

* Essentially, every VPC network functions as a distributed firewall.
* While firewall rules are applied to the network as a whole,
* connections are allowed or denied at the instance level. You can think of the firewall as existing not only between your instances and other networks, but between individual instances within the same network. Also, if for some reason all firewall rules in a network are deleted, there's still an implied "deny all" ingress rule, and an implied "allow all" egress rule for the network.
* You can express your desired firewall configuration as a set of firewall rules.
  + Conceptually, a firewall rule is composed of the following parameters:
    - the **direction** of the rule - inbound connections are matched against ingress rules only, and outbound connections are matched against egress rules only;
    - The **source** of the connection for ingress packets or the **destination** of the connection for egress packets;
    - There is the **protocol and port** of the connection, where any rule can be restricted to apply to specific protocols only or specific combinations of protocols and ports only;
    - There is the **action** of the rule, which allows or denies packets to match the direction protocol port and source or destination of the rule;
    - the **priority of the rule** which governs the order in which rules are evaluated - the first matching rule is applied;
    - and the rule **assignment**. By default, all rules are assigned to all instances, but you can assign certain rules to certain instances only.

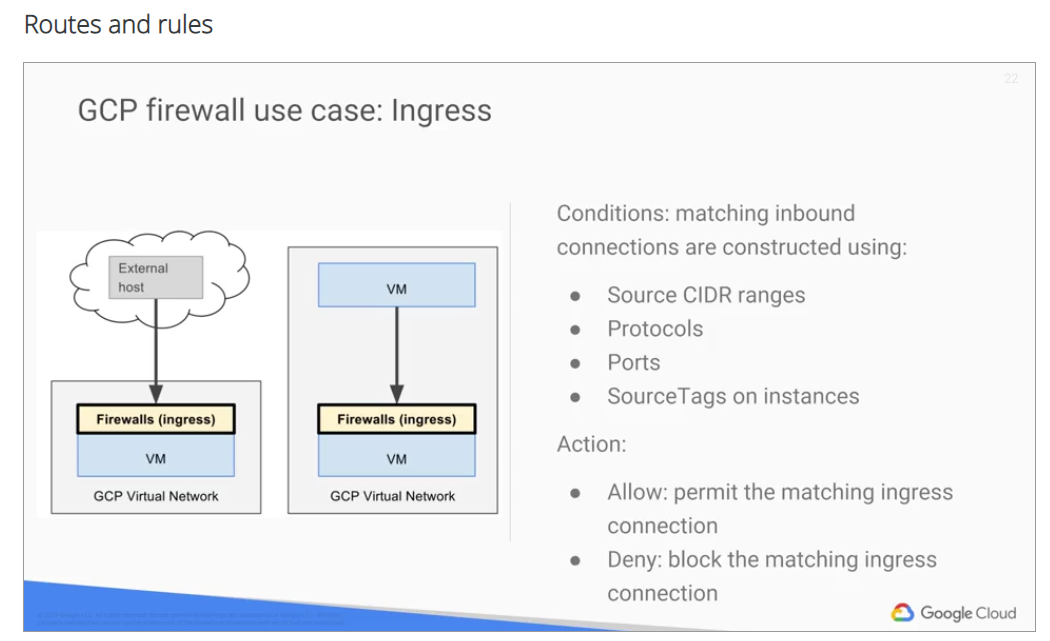


Let's look at some GCP firewall use cases for both egress and ingress.

* Egress firewall rules control outgoing connections originated inside your GCP network.



* + Egress "allow" rules allow open connections that match specific protocols, ports, and IP addresses.
  + Egress "deny" rules, prevent instances from initiating connections that match non-permitted port, protocol, and IP range combinations. For Egress firewall rules, destinations to which a rule applies may be specified using IP CIDR ranges. Specifically, you can use destination ranges to protect from undesired connections, initiated by a VM instance, towards an external destination. For example, as shown on the left, an external host. You can also use destination ranges to protect from undesired connections, initiated by VM instance towards specific GCP CIDR ranges. For example, as shown in the middle, a VM in a specific subnet.
* Ingress firewall rules protect against incoming connections to the instance from any source.



* + Ingress "allow" rules, allow specified protocol, ports, and IP addresses to connect in. The firewall prevents instances from receiving connections, on non-permitted ports and protocols. Rules can be restricted to only affect particular sources. Source CIDR ranges can be used to protect from undesired connections coming to an instance either from external networks or from GCP IP CIDR ranges. In addition, **source text** can be used to protect from undesired connections coming from specific VM instances that are tagged with a **matching tag**. This diagram illustrates a VM receiving a connection from an external address, and another VM receiving a connection from a VM in the same network. You can control ingress connections from a VM instance by constructing inbound connection conditions using source CIDR ranges, protocols, ports, and source tags on instances. However, source tags can only be used for VM to VM connections such as the one shown here in the middle.

# Billing:

First of all, ingress or traffic coming into GCP's network is not charged.

All egress traffic to the same zone, to a different GCP service within the same region, or to other Google products like YouTube, Maps, Drive from a VM in GCP with a public or private IP address is not charged either.

However, there is a charge for egress between zones in the same region and between regions. All these charges are for egress through internal IP addresses. There are different charges for egress through external IP address{es}, regardless of whether the instances are in the same zone. Another thing to consider when designing your network is your throughput and round-trip latency between virtual machines. This is going to vary by location, so check your applications' requirements against current specifics for VPC when choosing where to place your VMs. For example, VM to VM communication within a single zone has much more consistent performance than VM to VM communication between regions in a single continent or even across continents. Keep in mind that as VPC is constantly evolving, there are some features that are marked BETA. These features do not have a service level agreement or SLA and the online documentation states which features are currently in BETA.

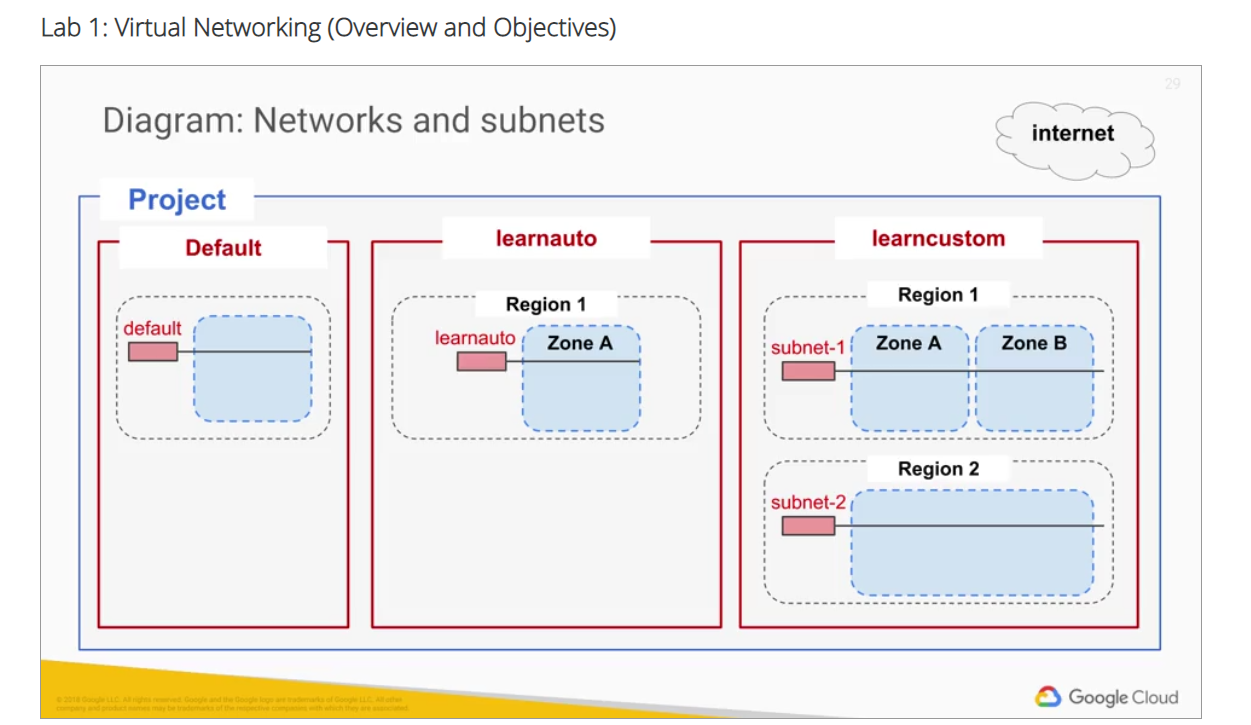
If you make design changes, you might have to delete networks and subnetworks. If that is the case, start by

1. deleting your VMs and any related firewall rules.
2. Then, depending on whether you have an auto type network or a custom type network,
3. you might only be able to delete the entire network or you might be able to delete the subnetworks independently off the entire network.

# Lab

Let's apply some of the network features we just discussed in a lab. In this lab you will build and

explore a complex GCP network structure.



Let me illustrate the different tasks of this lab with some network diagrams. In the first part of the lab you will build this complex multiple network topology. It consists of the default network, an auto mode network, and a custom mode network, along with its associated subnetworks. You will launch VMs in the various regions and subnets. And having these VMs in a variety of locations will enable you to explore the connectivity across and within multiple networks. Finally, you will use ping, trace route and SSH to test connectivity between the instances. You will then modify the firewall rules to meet policy requirements, and test to verify that the changes worked. In most labs you chose regions and zones where objects are located. However, this one is prescriptive about the network layout because it is systematically highlighting the differences between placing instances in a variety of network locations.

## Overview

In this lab you build and explore a complex GCP network structure. In most labs you choose the regions and zones where objects are located; however, this lab is prescriptive about the network layout. The lab systematically highlights the differences between placing instances in a variety of network locations and depending on the instances relative location, how you establish communications between virtual machines.

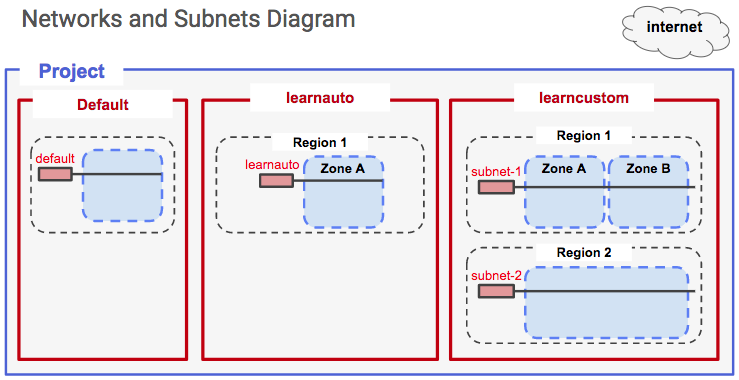
## Objectives

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

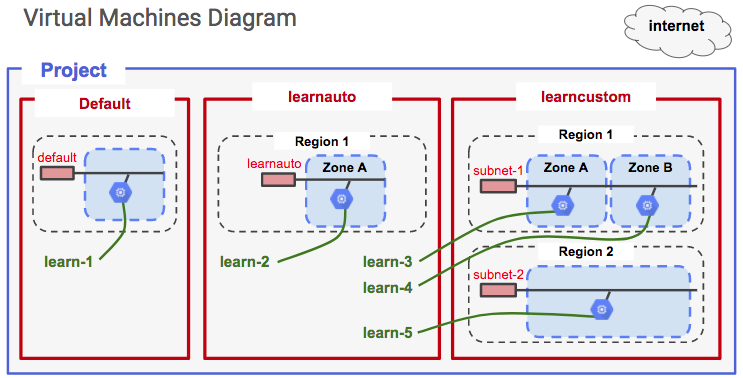
* Create an auto-mode network, a custom-mode network, and associated subnetworks
* Compare connectivity in the various types of networks
* Create routes and firewall rules using IP addresses and tags to enable connectivity
* Convert an auto-mode network to a custom-mode network
* Create, expand, and delete subnetworks

Here is a preview of the lab activities and the networks you will create:

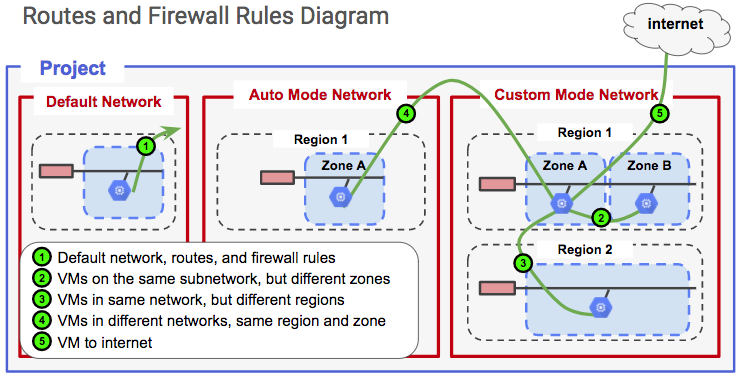
**Task 1: Create the network topology**



**Task 2: Create the VM instances**



**Task 3: Work with routes and firewall rules**



The scoping and connectivity relationships between zones, regions, networks, and subnets are different from networking in other public clouds.

You have been provided with a project in Qwiklabs. The project ID is a unique name across all Google Cloud projects. It is referred to later in this lab as PROJECT\_ID.

## Task 1: Create the network topology

### Explore the default network

The default network is created automatically for you with each new project. The default network layout is not ideal for managing resources. The main benefit is that it is a fast way to get a project set up and running. The default network is great for prototyping solutions and for training purposes.

1. In the Google Cloud Platform (GCP) Console, on the **Navigation menu** (7a91d354499ac9f1.png), click **VPC network** > **VPC networks**.

Notice the default network. It was created automatically for you with a subnetwork in each region.

Example:

asia-east1 | default | 10.140.0.0/20 | 10.140.0.1

For more information, see:

IP Addresses: <https://cloud.google.com/compute/docs/ip-addresses/>

Subnets and CIDR ranges: <https://cloud.google.com/compute/docs/alias-ip/#subnets_and_cidr_ranges>

1. In the left pane, click **Routes**.

Notice that a route was created for each subnetwork, and one global route was created to enable traffic to the internet.

### Create an auto-mode network and subnets

1. In the left pane, click **VPC networks**.
2. Click **Create VPC network**.
3. Specify the following:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **learnauto** |
| **Description** | **Learn about auto-mode networks** |
| **Subnet creation mode** | **Automatic** |

When you click **Automatic**, the list of subnetworks to be created is automatically displayed.

1. For **Firewall rules**, select all listed firewall rules.
2. At the bottom of the page are two links labeled **Equivalent REST** or **command line**. Click **REST** to see POST commands for API programming automation of this process.

|  |
| --- |
| POST https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-18e67002a341bd86/global/networks  {  "routingConfig": {  "routingMode": "REGIONAL"  },  "name": "learnauto",  "description": "learn acout auto-mode networks",  "autoCreateSubnetworks": true  } |
| gcloud compute --project=qwiklabs-gcp-18e67002a341bd86 networks create learnauto --description="learn acout auto-mode networks" --subnet-mode=auto  gcloud compute --project=qwiklabs-gcp-18e67002a341bd86 firewall-rules create learnauto-allow-icmp --description="Allows ICMP connections from any source to any instance on the network." --direction=INGRESS --priority=65534 --network=learnauto --action=ALLOW --rules=icmp --source-ranges=0.0.0.0/0  gcloud compute --project=qwiklabs-gcp-18e67002a341bd86 firewall-rules create learnauto-allow-internal --description="Allows connections from any source in the network IP range to any instance on the network using all protocols." --direction=INGRESS --priority=65534 --network=learnauto --action=ALLOW --rules=all --source-ranges=10.128.0.0/9  gcloud compute --project=qwiklabs-gcp-18e67002a341bd86 firewall-rules create learnauto-allow-rdp --description="Allows RDP connections from any source to any instance on the network using port 3389." --direction=INGRESS --priority=65534 --network=learnauto --action=ALLOW --rules=tcp:3389 --source-ranges=0.0.0.0/0  gcloud compute --project=qwiklabs-gcp-18e67002a341bd86 firewall-rules create learnauto-allow-ssh --description="Allows TCP connections from any source to any instance on the network using port 22." --direction=INGRESS --priority=65534 --network=learnauto --action=ALLOW --rules=tcp:22 --source-ranges=0.0.0.0/0 |

1. Click **Close**.
2. Click **command line** to see commands you could use for automation of this process. You could use these commands to create the network by clicking RUN IN CLOUD SHELL—but don't do it.

Note: These commands tend to include options that are not required. They may not work in a bash script without being altered. Don't rely on them. You should consider these more of a suggestion. If you need to automate with scripts, plan to craft your own commands from examples in the documentation.

1. Click **Close**.
2. Click **Create**.
3. Click **REFRESH** occasionally until the networks are created and appear in the list.

### Explore the auto-mode network

1. In the left pane, click **Routes**.

Notice that a route has been created for each subnetwork, and one route was created to enable traffic from anywhere, including the internet. Traffic is delivered via the most specific matching route: traffic intended for any of the listed subnets gets delivered via virtual network to the host. These routes take precedence over the route that matches all traffic.

1. Click **Destination IP ranges** to sort the list of routes.

Notice that there is an identical subnetwork and route in the learnauto network as there is in the default network. It is possible to have VMs with duplicate Internal IP addresses in the two networks.

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

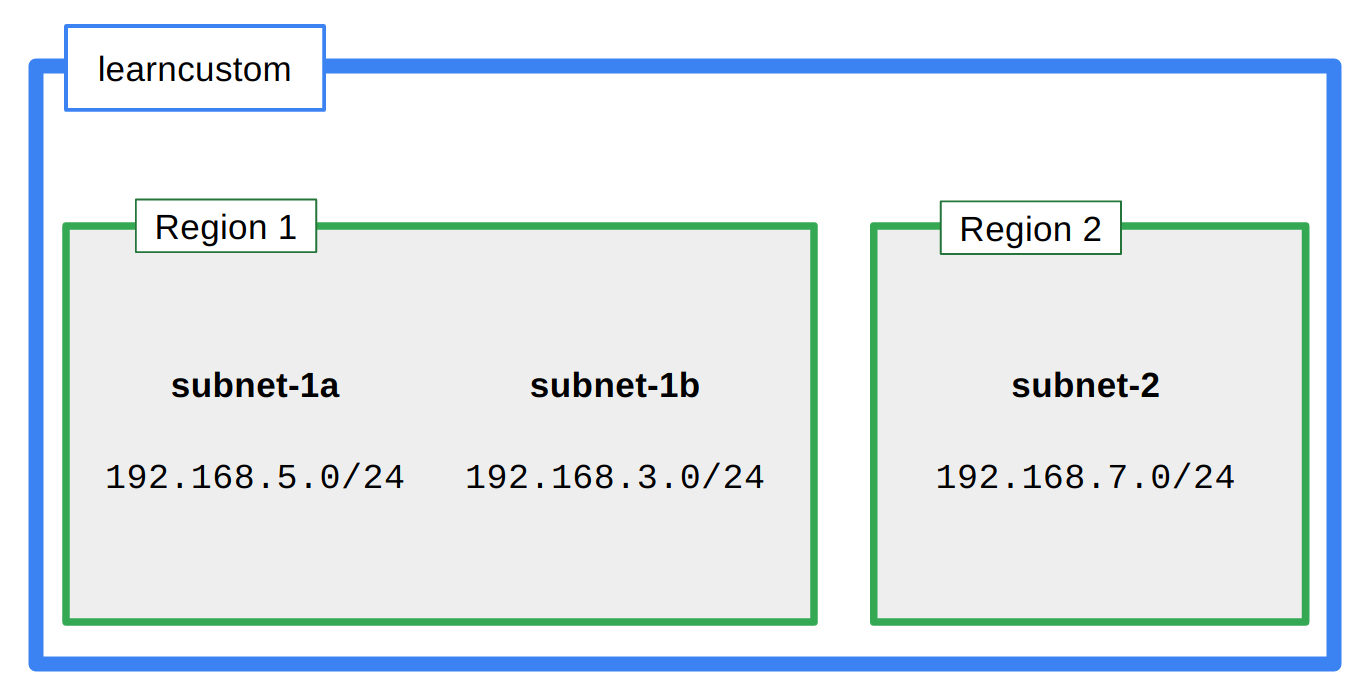
Verify that firewall rules were created for the learnauto network and its subnetworks.

If you delete your default network, you can always recreate it as an auto network using the name "Default."

### Create a custom-mode network

In this subtask, you create a custom-mode network named **learncustom** with three subnetworks:

* (**subnet-1a**) 192.168.5.0/24
* (**subnet-1b**) 192.168.3.0/24, in the same region
* (**subnet-2**) 192.168.7.0/24 in a different region



1. In the left pane, click **VPC networks**.
2. Click **Create VPC network**.
3. Specify the following:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **learncustom** |
| **Description** | **Learn about custom networks** |
| **Subnet creation mode** | **Custom** |

Use the dialog to add three subnets as follows.

1. For the first subnet, specify the following:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **subnet-1a** |
| **Region** | **us-central1** |
| **IP address range** | **192.168.5.0/24** |

1. Click **Add subnet**.
2. For the second subnet, specify the following:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **subnet-1b** |
| **Region** | **us-central1** |
| **IP address range** | **192.168.3.0/24** |

1. Click **Add subnet**.
2. For the third subnet, specify the following:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **subnet-2** |
| **Region** | **us-west1** |
| **IP address range** | **192.168.7.0/24** |

1. Click **Create**.

### Explore the routes and firewall rules

Did creating the custom network automatically create routes?

1. In the left pane, click **Routes**.
2. Click **Network** in the table header to sort by network name. Routes should be displayed for each subnetwork.

Did creating the custom network automatically create firewall rules?

1. In the left pane, click **Firewall rules**.
2. Click **Network** in the table header to sort by network name. No default firewall rules were created for the custom network. You will have to manually add default rules in the next step.

### Create firewall rules for the learncustom network

Notice that for the other networks, the default network and the learnauto network, GCP automatically created default firewall rules allowing SSH traffic (tcp:22), icmp traffic, and rdp (tcp:3389) traffic for Windows VMs. Add a firewall rule to provide the same access for the learncustom network.

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

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **allow-ssh-icmp-rdp-learncustom** |
| **Network** | **learncustom** |
| **Target tags** | **allow-defaults** |
| **Source IP ranges** | **0.0.0.0/0** |
| **Protocols and ports** | **Specified protocols and ports** |

1. For **tcp**, specify ports **22** and **3389**.
2. Specify the **icmp** protocol.

Make sure that the source filter address includes the final "/0". If you specify 0.0.0.0 instead of 0.0.0.0/0, the filter defaults to 0.0.0.0/32, which is an exact host address that doesn't exist.

1. Click **Create**.

### Create an overlapping subnet

In this subtask, you attempt to modify the network by adding a subnet with an overlapping address range but in a different region. What do you predict will happen?

1. In the left pane, click **VPC networks**.
2. Click **learncustom**.
3. Click **Add subnet**.
4. Specify the following, leaving all other values with their defaults:

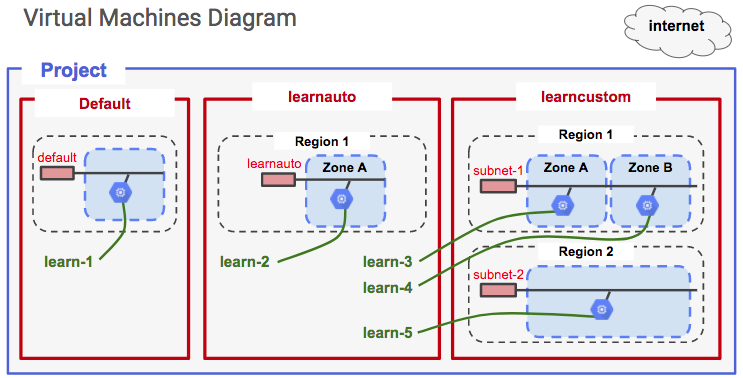
|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Name** | **subnet-3** |
| **Region** | **europe-west1** |
| **IP address range** | **192.168.5.0/24** |

The IP address range label is displayed in red with the following error message: "This IP address range overlaps with a subnet you already added. Enter an address range that doesn't overlap."

1. Click **CANCEL**.

## Task 2: Create the VM instances

To explore the Cloud Virtual Network, you create five micro VMs in different locations in the network. You will not install any additional software on them. They will not run any applications. You will just use them to explore the connectivity across the topologies in the network.



|  |  |  |  |
| --- | --- | --- | --- |
| Name | Network | Region | Zone |
| learn-1 | default | us-west1 | us-west1-a |
| learn-2 | learnauto | us-west1 | us-west1-a |
| learn-3 | learncustom | us-central1 | us-central1-a |
| learn-4 | learncustom | us-central1 | us-central1-b |
| learn-5 | learncustom | us-west1 | us-west1-a |

### Create the learn-1 VM

1. On the **Navigation menu** (7a91d354499ac9f1.png), 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** | **learn-1** |
| **Region** | **us-west1** |
| **Zone** | **us-west1-a** |
| **Machine type** | **micro (1 shared vCPU)** |

1. Click **Management, security, disks, networking, sole tenancy** to access the advanced options.
2. Click **Networking**. The default network interface should already be selected.
3. Click **Create**.

### Create the learn-2 VM

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** | **learn-2** |
| **Region** | **us-west1** |
| **Zone** | **us-west1-a** |
| **Machine type** | **micro (1 shared vCPU)** |

1. Click **Management, security, disks, networking, sole tenancy** to access the advanced options.
2. Click **Networking**.
3. Click the pencil icon to edit **Network interfaces**.
4. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Network** | **learnauto** |
| **Subnetwork** | **learnauto** |

1. Click **Done**.
2. Click **Create**.

### Create the learn-3 VM

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** | **learn-3** |
| **Region** | **us-central1** |
| **Zone** | **us-central1-a** |
| **Machine type** | **micro (1 shared vCPU)** |

1. Click **Management, security, disks, networking, sole tenancy** to access the advanced options.
2. Click **Networking**.
3. Click the pencil icon to edit **Network interfaces**.
4. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Network** | **learncustom** |
| **Subnetwork** | **subnet-1a** |

1. Click **Done**.
2. Click **Create**.

### Create the learn-4 VM

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** | **learn-4** |
| **Region** | **us-central1** |
| **Zone** | **us-central1-b** |
| **Machine type** | **micro (1 shared vCPU)** |

1. Click **Management, security, disks, networking, sole tenancy** to access the advanced options.
2. Click **Networking**.
3. Click the pencil icon to edit **Network interfaces**.
4. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Network** | **learncustom** |
| **Subnetwork** | **subnet-1b** |

1. Click **Done**.
2. Click **Create**.

### Create the learn-5 VM

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** | **learn-5** |
| **Region** | **us-west1** |
| **Zone** | **us-west1-a** |
| **Machine type** | **micro (1 shared vCPU)** |

1. Click **Management, security, disks, networking, sole tenancy** to access the advanced options.
2. Click **Networking**.
3. Click the pencil icon to edit **Network interfaces**.
4. Specify the following, and leave the remaining settings as their defaults:

|  |  |
| --- | --- |
| **Property** | **Value**  (type value or select option as specified) |
| **Network** | **learncustom** |
| **Subnetwork** | **subnet-2** |

1. Click **Done**.
2. Click **Create**.

### Verify that all the test VMs are running

1. On the **VM instances** page, verify that all 5 instances are running.