Batch: A2 Roll No.: 1911031

Experiment No. 09

Grade: AA / AB / BB / BC / CC / CD /DD

Title: Configuring Networks via gcloud

Objective: To configure a network via google cloud

Expected Outcome of Experiment:

CO	Outcome
CO5	Configure and experiment with advanced cloud technologies

Books/ Journals/ Websites referred:

1. https://www.vmware.com/topics/glossary/content/network-configuration.html#:~:text=Network%20configuration%20is%20the%20process,need%20for%20extensive%20manual%20configuration.
2. https://www.techtarget.com/searchnetworking/definition/network-configuration-management
3. https://www.geeksforgeeks.org/advantages-and-disadvantages-of-computer-networking/

Abstract:-

What is network configuration?

Network configuration is the process of assigning network settings, policies, flows, and controls. In a virtual network, it's easier to make network configuration changes because physical network devices appliances are replaced by software, removing the need for extensive manual configuration.

Network configuration can also be automated and managed via a centralized configuration manager network configuration manager, further reducing manual IT workload and making it easier to:

- **Maintain** a network
- **Make** configuration changes
- **Relaunch** devices
- **Track** and report data

Some network configuration basics include switch/router configuration, host configuration, software and firewall configuration, and network topology which can be controlled through rest APIs.

Why is network configuration important?

The right network configuration is essential to supporting the flow of traffic through a network, and it can also support and enhance network security and improve network stability. In addition, the use of network configuration management manager and or configuration tools can provide a number of benefits, including:

- Automated data tracking and reporting, allowing administrators to spot any configuration changes and potential threats or issues
- An easy way to make bulk changes, such as a blanket password change in a situation where passwords are compromised
- The means to swiftly roll back network settings to a previous configuration
- Reduced downtime, thanks to increased visibility and the ability to quickly identify changes
- Streamlined maintenance and repair of network devices (physical or virtual) and connections
- The ability to relaunch a device when it fails, thanks to centralized storage management of device configurations

What is zero-configuration networking?

Zero-configuration networking refers to a set of technologies that allow network administrators to set up a network and connect devices without having to manually configure each device's network settings.

This is particularly useful for allowing end users to easily connect to the network. However, for an administrator of an enterprise network, there are advantages to actively configure and monitor the network rather than relying on default settings.
What are network topologies?

Different types of network configuration in computer networks are commonly referred to as network topologies. A network topology describes how the nodes or devices (physical or virtual) in a network are arranged and how they communicate with each other.

Network topology can be physical (referring to where physical devices are placed in relation to each other) or logical (referring to how data is transmitted through the network, including any virtual or cloud resources). When choosing a network topology, an organization must consider the size of its network, its performance requirements and the flow of its traffic, among other factors.

Common network topologies include:

Bus: Every node in the network is connected along a linear path. This simple topology is used most often for small networks.

Ring: Nodes are connected in a loop, and traffic may flow in one direction or in both directions. Ring networks tend to be cost-effective, but not as scalable or stable as other network topologies.

Star: A central node connects to all other nodes in the network. This is a common and stable topology that's often used for local area networks (LANs).

Mesh: Nodes are linked in such a way that multiple paths between nodes are possible. This type of network topology increases the resiliency of the network, but also increases cost. A network may be fully meshed (all nodes connecting to all other nodes) or partially meshed (only some nodes having multiple connections to other nodes).

Spine-Leaf (Tree): Multiple star topologies are connected together in a larger star configuration.

Hybrid: A combination of other topologies are used together within one network.

How can you check your network configuration?

In a command-line environment, the commands **ipconfig** (for Windows network configuration) and **ifconfig** (for Linux network configuration, as well as Mac OSX and other Linux-like environments) allow you to view information about your network configuration and to configure your networsk interface.

With a network configuration manager or with APIs, you can check and set up the network configuration in a centralized software interface, allowing you to more easily configure, monitor and administer your network. A network configuration manager also enables the use of automation to make policy changes and updates.
How to configure a network switch and router?

When setting up a network switch and router, it's important to customize settings and apply all necessary configurations to ensure that your network will work properly. Some of the configurable settings on a network switch and router include:

- **IP address**—for identification
- **Password**—for added security
- **Channel and band selection**—to improve performance
- **Default gateway**—to make the device visible to network management tools
- **Neighbor discovery**—for added visibility
- **Correct time**—for proper troubleshooting and detailed error logs

A network configuration manager is the easiest way to perform network switch configuration and apply these settings consistently to every device on your enterprise network.
What is network monitoring?

Network monitoring is a function of network management that monitors a network and alerts network administrators to potential issues. The thresholds or conditions for alerting the administrator can be configured based on network traffic flow and business needs. When issues do occur, networking configuration management allows the administrator to quickly correct the problem by modifying the configuration or adding more network resources.

Related Theory: -

The computer network is defined as a set of interconnected autonomous systems that facilitate distributed processing of information. It results in better performance with a high speed of processing.

Advantages of Network:
These are the main advantages of Computer Networks:

1. **Central Storage of Data –**
   Files can be stored on a central node (the file server) that can be shared and made available to each and every user in an organization.

2. **Anyone can connect to a computer network –**
   There is a negligible range of abilities required to connect to a modern computer network. The effortlessness of joining makes it workable for even youthful kids to start exploiting the data.

3. **Faster Problem-solving –**
   Since an extensive procedure is disintegrated into a few littler procedures and each is taken care of by all the associated gadgets, an explicit issue can be settled in lesser time.

4. **Reliability –**
   Reliability implies backing up information. Due to some reason equipment crashes, and so on, the information gets undermined or inaccessible on one PC, another duplicate of similar information is accessible on another workstation for future use, which prompts smooth working and further handling without interruption.

5. **It is highly flexible –**
   This innovation is known to be truly adaptable, as it offers clients the chance to investigate everything about fundamental things, for example, programming without influencing their usefulness.

6. **Security through Authorization –**
   Security and protection of information are additionally settled through the system. As just the system clients are approved to get to specific records or applications, no other individual can crack the protection or security of information.

7. **It boosts storage capacity –**
   Since you will share data, records, and assets with other individuals, you need to guarantee all information and substance are legitimately put away in the framework. With this systems administration innovation, you can do the majority of this with no issue, while having all the space you require for capacity.

Disadvantages of Network:
These are the main disadvantages of Computer Networks:

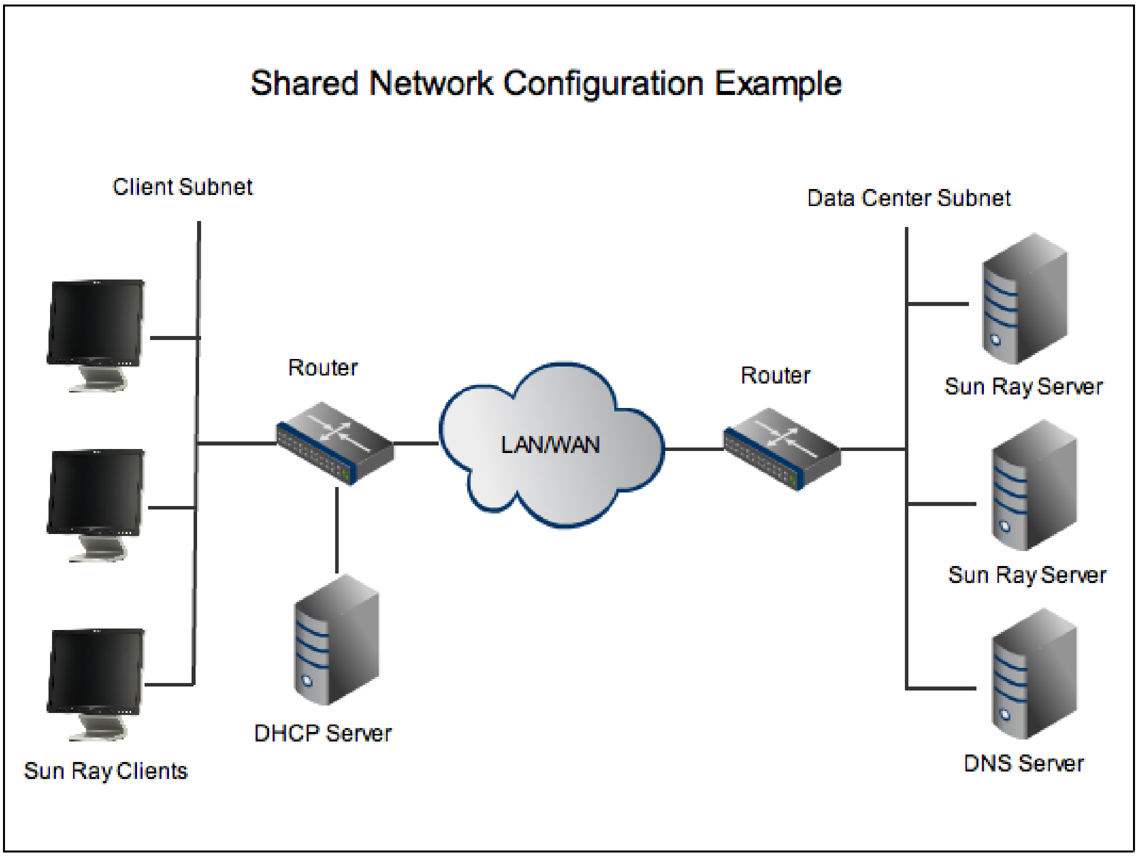
1. **It lacks robustness –**
   If a PC system's principal server separates, the whole framework would end up futile. Also, if it has a bridging device or a central linking server that fails, the entire network would also come to a standstill. To manage these issues, gigantic systems ought to have a ground-breaking PC to fill in as a document server to

influence setting up and keeping up the system less demanding.

2. **It lacks independence –**
PC organizing includes a procedure that is worked utilizing PCs, so individuals will depend on a greater amount of PC work, rather than applying an exertion for their jobs that needs to be done. Besides this, they will be subject to the primary document server, which implies that, in the event that it separates, the framework would end up futile, making clients inactive.

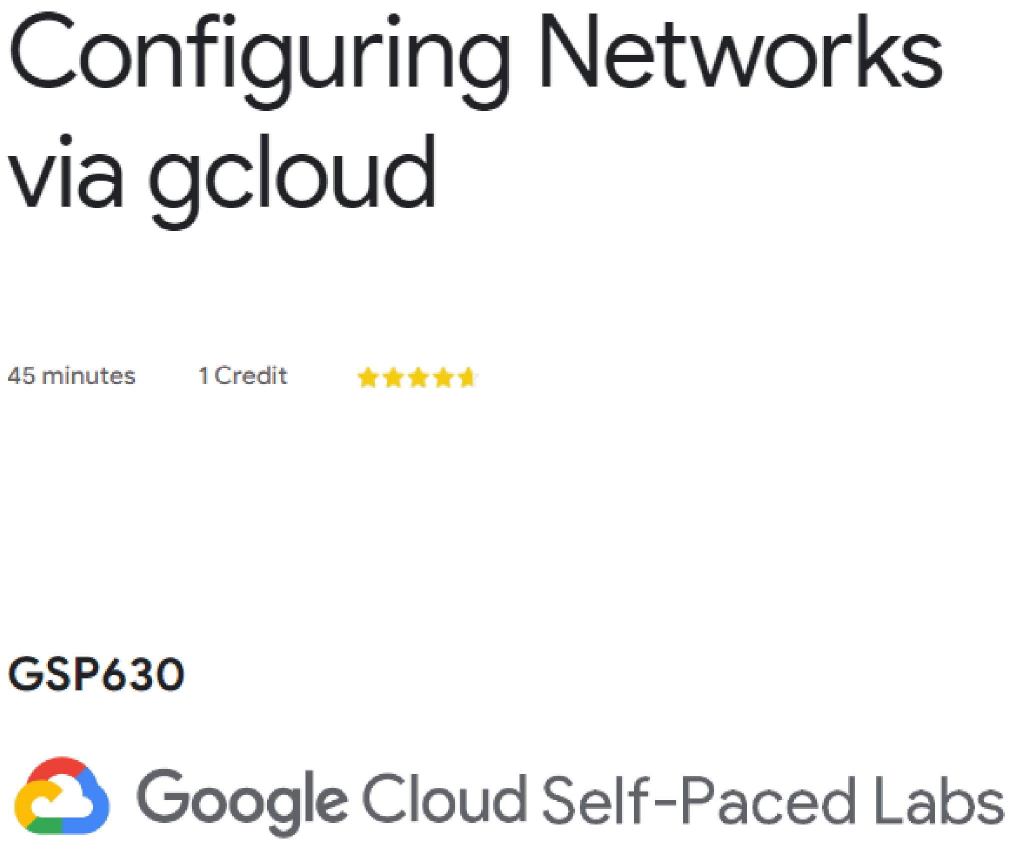
3. **Virus and Malware –**
On the off chance that even one PC on a system gets contaminated with an infection, there is a possibility for alternate frameworks to get tainted as well. Infections can spread on a system effectively, in view of the availability of different gadgets.

4. **Cost of the network –**
The expense of executing the system including cabling and equipment can be expensive.



Implementation Details:

1. Enlist all the Steps followed and various options explored

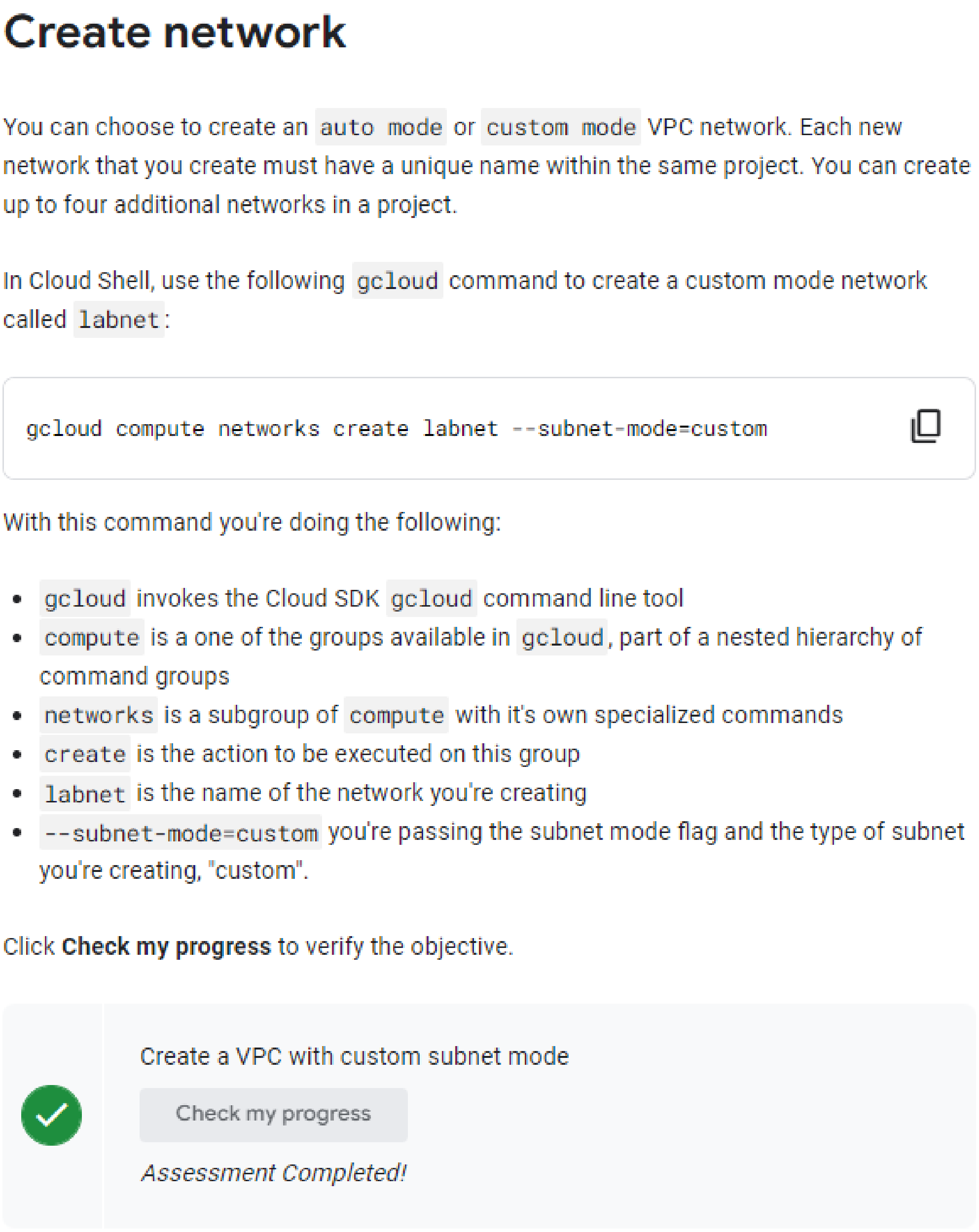


Configuring Networks via gcloud

45 minutes 1 Credit ★★★★½

GSP630

Google Cloud Self-Paced Labs



Create network

You can choose to create an `auto mode` or `custom mode` VPC network. Each new network that you create must have a unique name within the same project. You can create up to four additional networks in a project.

In Cloud Shell, use the following `gcloud` command to create a custom mode network called `labnet`:

```
gcloud compute networks create labnet --subnet-mode=custom
```

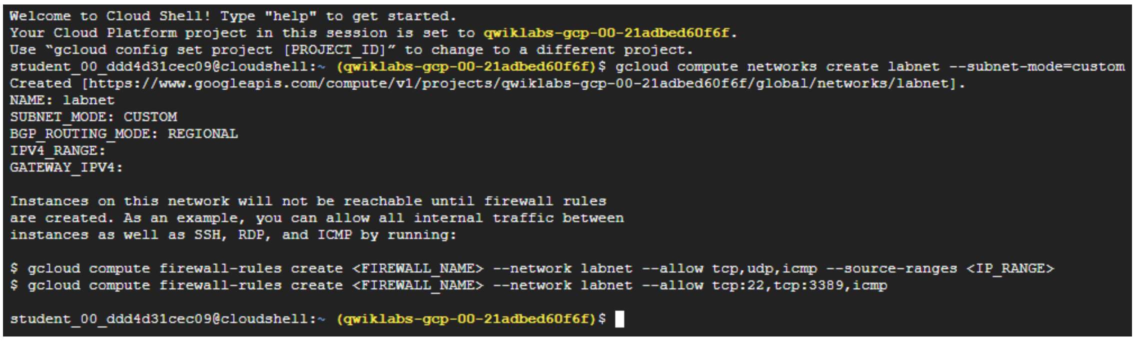
With this command you're doing the following:

- `gcloud` invokes the Cloud SDK `gcloud` command line tool
- `compute` is a one of the groups available in `gcloud`, part of a nested hierarchy of command groups
- `networks` is a subgroup of `compute` with it's own specialized commands
- `create` is the action to be executed on this group
- `labnet` is the name of the network you're creating
- `--subnet-mode=custom` you're passing the subnet mode flag and the type of subnet you're creating, "custom".

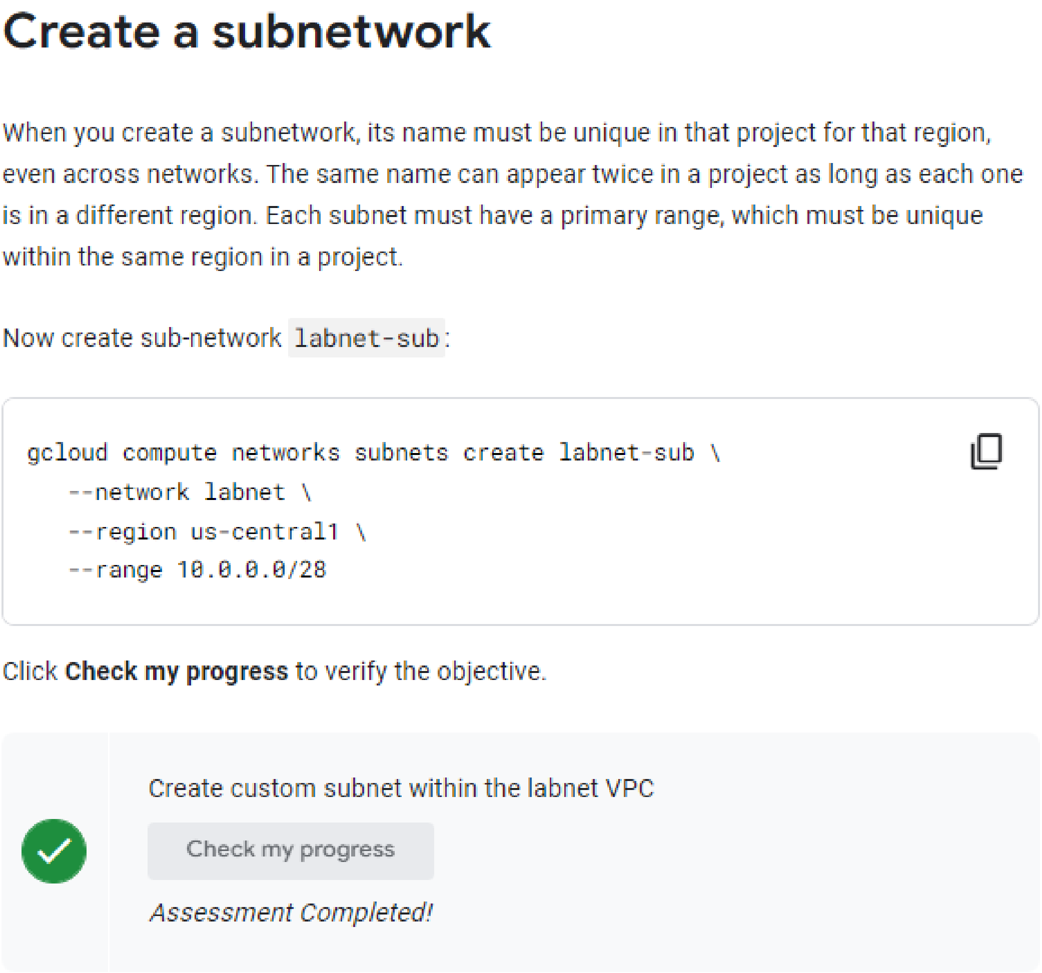
Click **Check my progress** to verify the objective.

✓ Create a VPC with custom subnet mode

Assessment Completed!



```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to qwiklabs-gcp-00-21adbed60f6f.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks create labnet --subnet-mode=custom
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/networks/labnet].
NAME: labnet
SUBNET_MODE: CUSTOM
BGP_ROUTING_MODE: REGIONAL
IPV4_RANGE:
GATEWAY_IPV4:

Instances on this network will not be reachable until firewall rules
are created. As an example, you can allow all internal traffic between
instances as well as SSH, RDP, and ICMP by running:

$ gcloud compute firewall-rules create <FIREWALL_NAME> --network labnet --allow tcp,udp,icmp --source-ranges <IP_RANGE>
$ gcloud compute firewall-rules create <FIREWALL_NAME> --network labnet --allow tcp:22,tcp:3389,icmp

student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Create a subnetwork

When you create a subnetwork, its name must be unique in that project for that region, even across networks. The same name can appear twice in a project as long as each one is in a different region. Each subnet must have a primary range, which must be unique within the same region in a project.

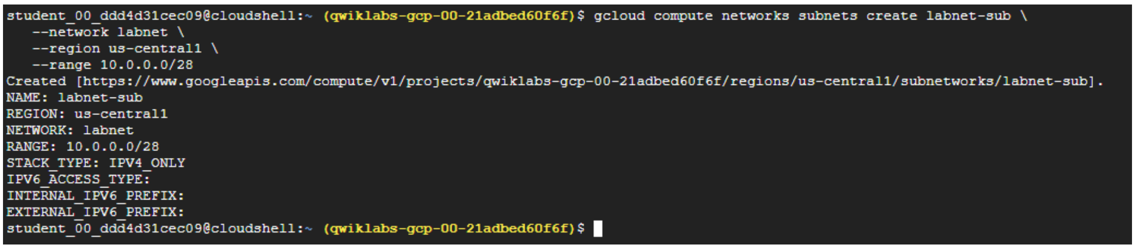
Now create sub-network `labnet-sub`:

```
gcloud compute networks subnets create labnet-sub \
   --network labnet \
   --region us-central1 \
   --range 10.0.0.0/28
```

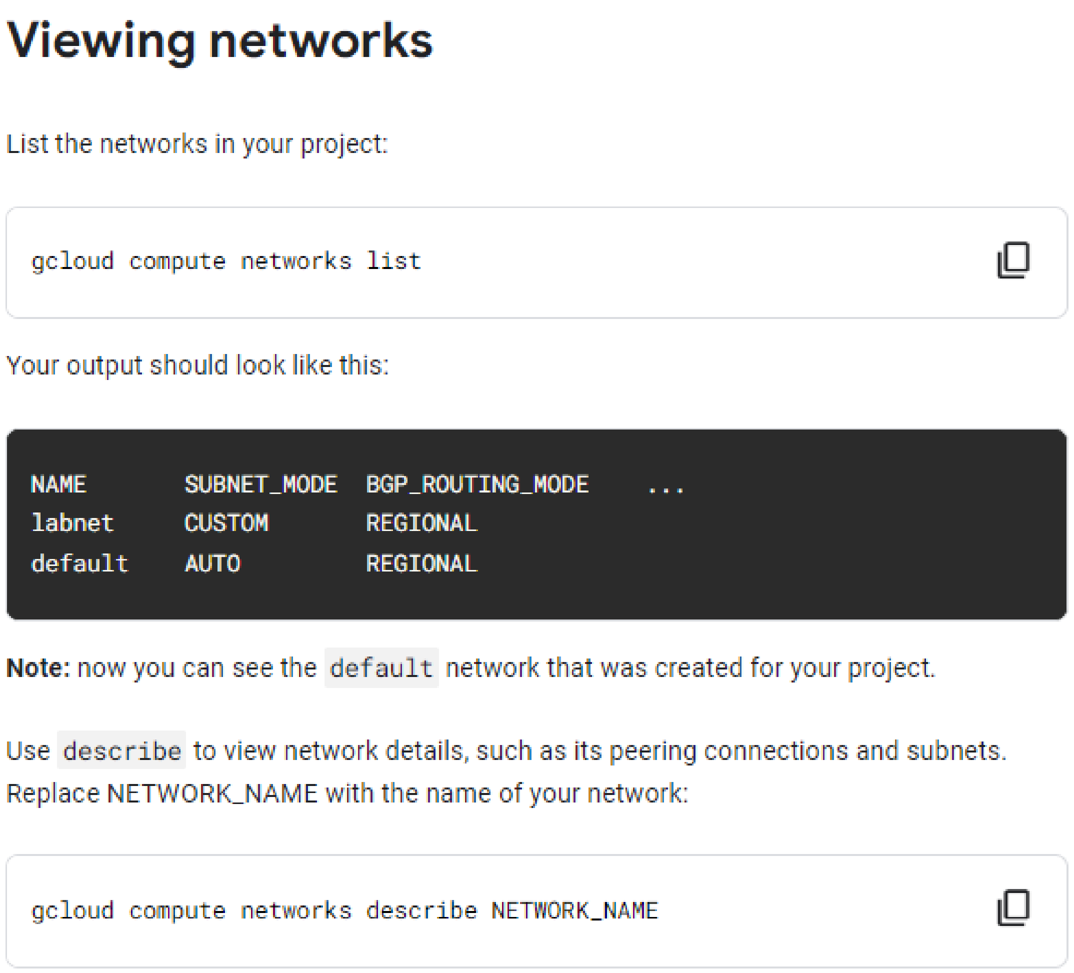
Click **Check my progress** to verify the objective.

Create custom subnet within the labnet VPC

Assessment Completed!



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks subnets create labnet-sub \
   --network labnet \
   --region us-central1 \
   --range 10.0.0.0/28
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/regions/us-central1/subnetworks/labnet-sub].
NAME: labnet-sub
REGION: us-central1
NETWORK: labnet
RANGE: 10.0.0.0/28
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Viewing networks

List the networks in your project:

```
gcloud compute networks list
```

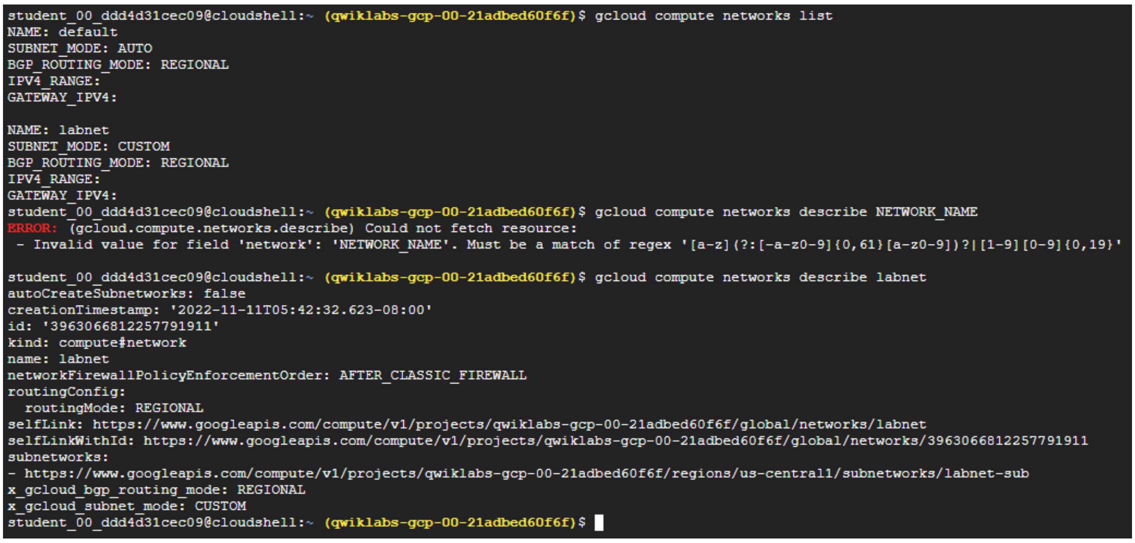
Your output should look like this:

```
NAME       SUBNET_MODE  BGP_ROUTING_MODE   ...
labnet     CUSTOM       REGIONAL
default    AUTO         REGIONAL
```

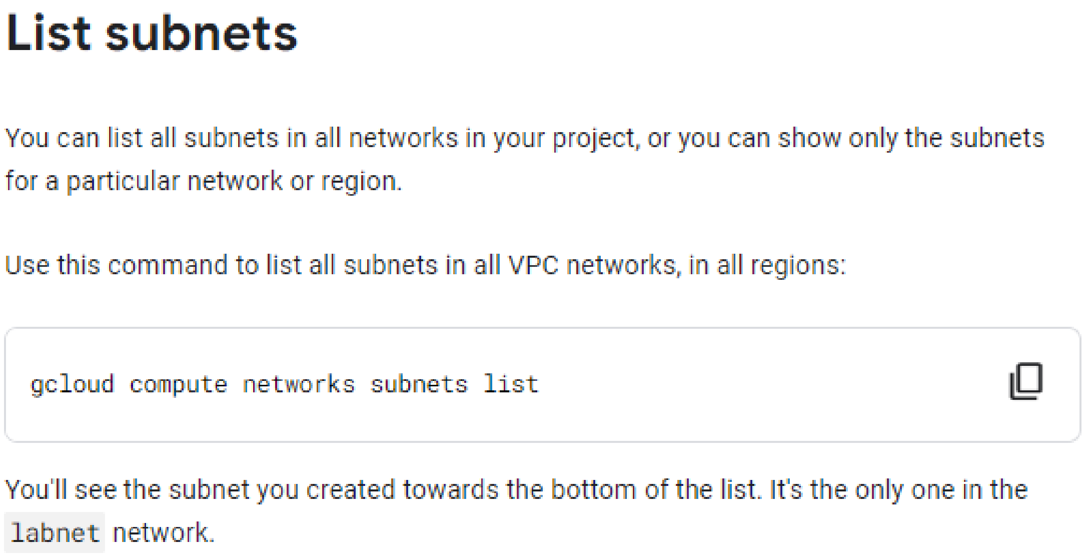
Note: now you can see the `default` network that was created for your project.

Use `describe` to view network details, such as its peering connections and subnets. Replace NETWORK_NAME with the name of your network:

```
gcloud compute networks describe NETWORK_NAME
```



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks list
NAME: default
SUBNET_MODE: AUTO
BGP_ROUTING_MODE: REGIONAL
IPV4_RANGE:
GATEWAY_IPV4:

NAME: labnet
SUBNET_MODE: CUSTOM
BGP_ROUTING_MODE: REGIONAL
IPV4_RANGE:
GATEWAY_IPV4:
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks describe NETWORK_NAME
ERROR: (gcloud.compute.networks.describe) Could not fetch resource:
 - Invalid value for field 'network': 'NETWORK_NAME'. Must be a match of regex '[a-z](?:[-a-z0-9]{0,61}[a-z0-9])?|[1-9][0-9]{0,19}'

student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks describe labnet
autoCreateSubnetworks: false
creationTimestamp: '2022-11-11T05:42:32.623-08:00'
id: '3963066812257791911'
kind: compute#network
name: labnet
networkFirewallPolicyEnforcementOrder: AFTER_CLASSIC_FIREWALL
routingConfig:
  routingMode: REGIONAL
selfLink: https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/networks/labnet
selfLinkWithId: https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/networks/3963066812257791911
subnetworks:
- https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/regions/us-central1/subnetworks/labnet-sub
x_gcloud_bgp_routing_mode: REGIONAL
x_gcloud_subnet_mode: CUSTOM
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



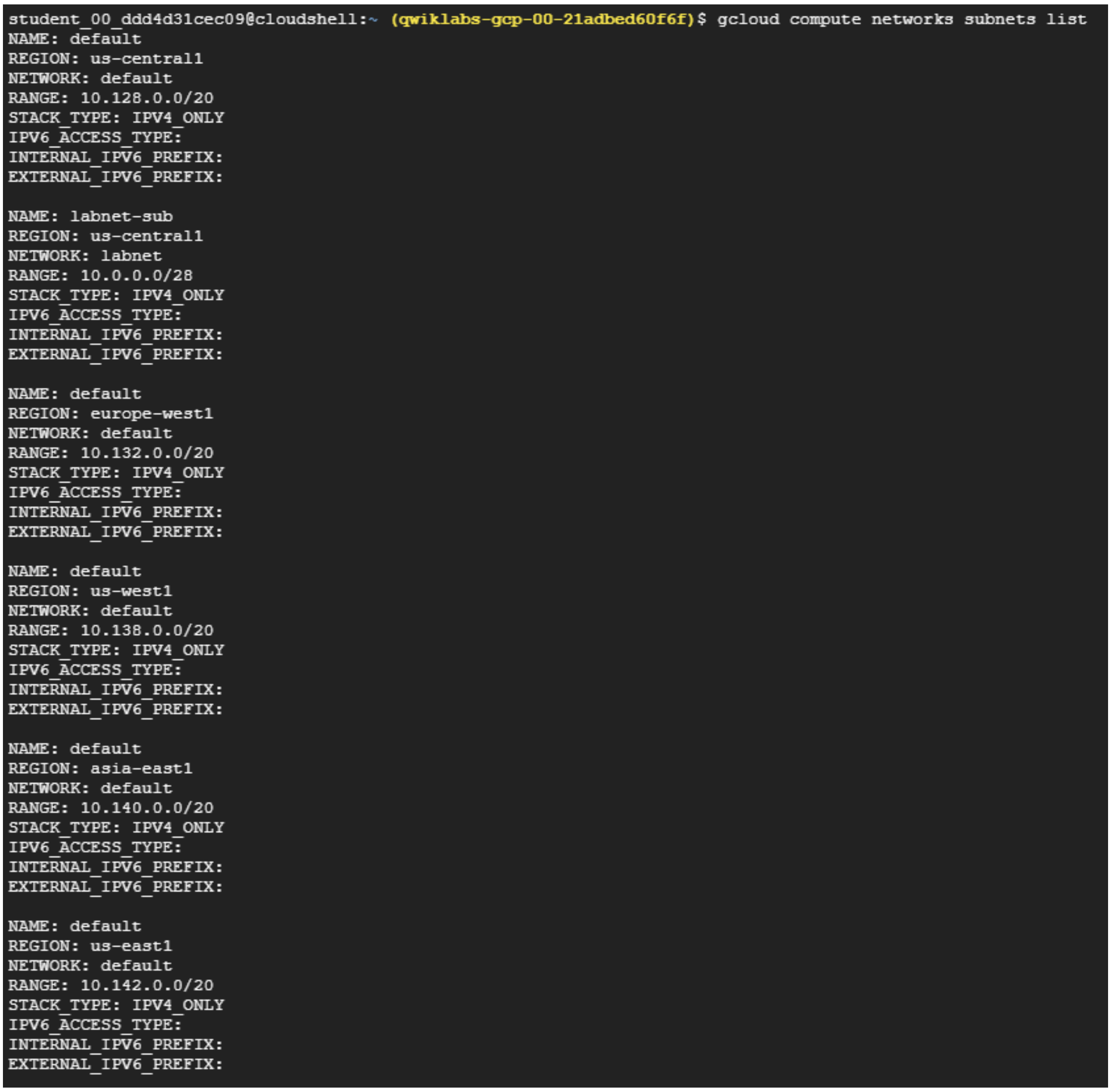
List subnets

You can list all subnets in all networks in your project, or you can show only the subnets for a particular network or region.

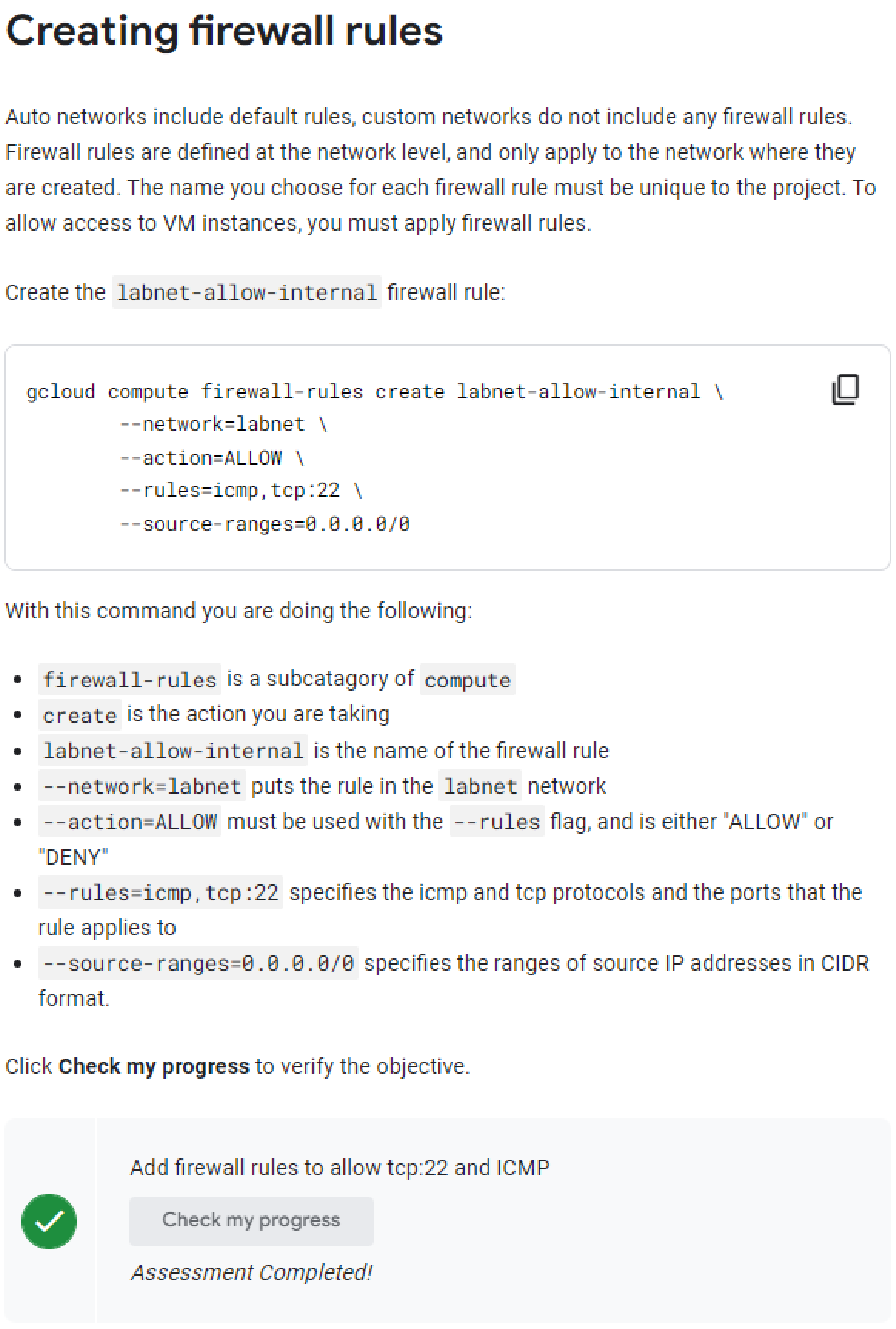
Use this command to list all subnets in all VPC networks, in all regions:

```
gcloud compute networks subnets list
```

You'll see the subnet you created towards the bottom of the list. It's the only one in the `labnet` network.



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks subnets list
NAME: default
REGION: us-central1
NETWORK: default
RANGE: 10.128.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: labnet-sub
REGION: us-central1
NETWORK: labnet
RANGE: 10.0.0.0/28
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: default
REGION: europe-west1
NETWORK: default
RANGE: 10.132.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: default
REGION: us-west1
NETWORK: default
RANGE: 10.138.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: default
REGION: asia-east1
NETWORK: default
RANGE: 10.140.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: default
REGION: us-east1
NETWORK: default
RANGE: 10.142.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:
```



Creating firewall rules

Auto networks include default rules, custom networks do not include any firewall rules. Firewall rules are defined at the network level, and only apply to the network where they are created. The name you choose for each firewall rule must be unique to the project. To allow access to VM instances, you must apply firewall rules.

Create the `labnet-allow-internal` firewall rule:

```
gcloud compute firewall-rules create labnet-allow-internal \
        --network=labnet \
        --action=ALLOW \
        --rules=icmp,tcp:22 \
        --source-ranges=0.0.0.0/0
```

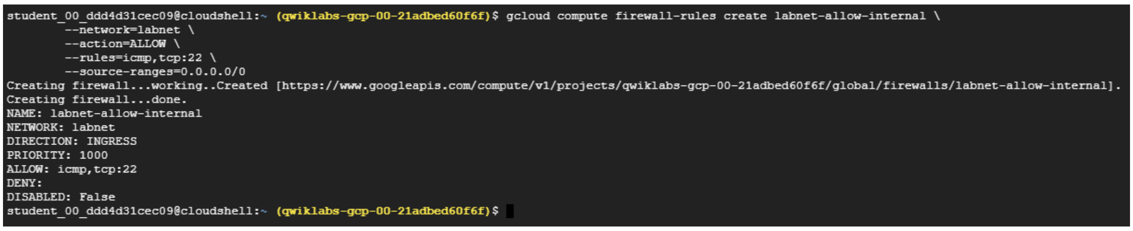
With this command you are doing the following:

- `firewall-rules` is a subcatagory of `compute`
- `create` is the action you are taking
- `labnet-allow-internal` is the name of the firewall rule
- `--network=labnet` puts the rule in the `labnet` network
- `--action=ALLOW` must be used with the `--rules` flag, and is either "ALLOW" or "DENY"
- `--rules=icmp,tcp:22` specifies the icmp and tcp protocols and the ports that the rule applies to
- `--source-ranges=0.0.0.0/0` specifies the ranges of source IP addresses in CIDR format.

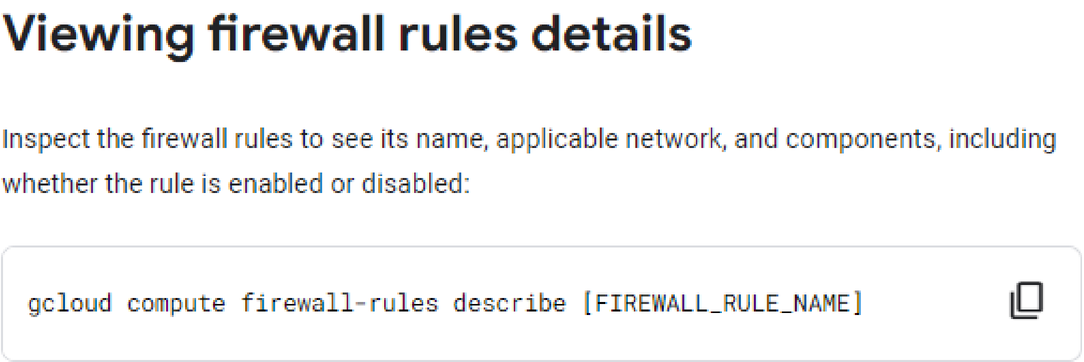
Click **Check my progress** to verify the objective.

✓ Add firewall rules to allow tcp:22 and ICMP

Assessment Completed!



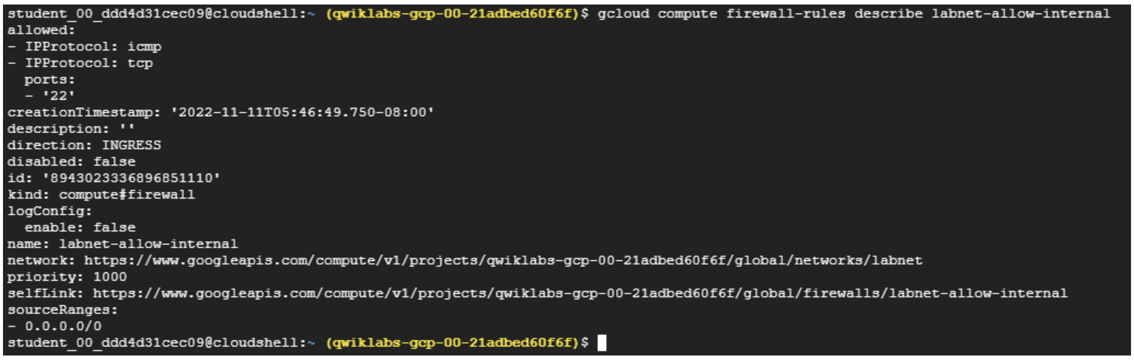
```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute firewall-rules create labnet-allow-internal \
        --network=labnet \
        --action=ALLOW \
        --rules=icmp,tcp:22 \
        --source-ranges=0.0.0.0/0
Creating firewall...working..Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/firewalls/labnet-allow-internal].
Creating firewall...done.
NAME: labnet-allow-internal
NETWORK: labnet
DIRECTION: INGRESS
PRIORITY: 1000
ALLOW: icmp,tcp:22
DENY:
DISABLED: False
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



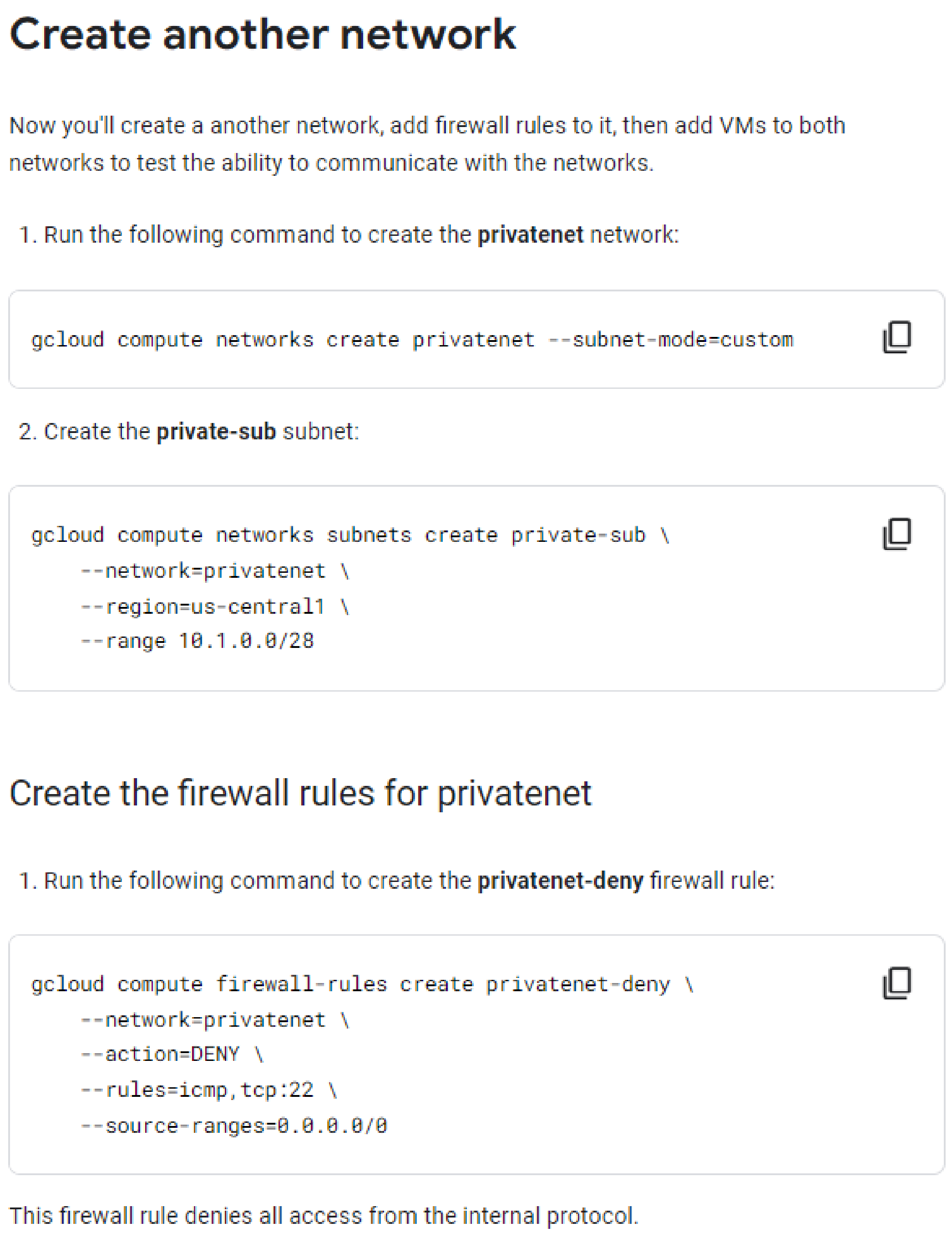
Viewing firewall rules details

Inspect the firewall rules to see its name, applicable network, and components, including whether the rule is enabled or disabled:

```
gcloud compute firewall-rules describe [FIREWALL_RULE_NAME]
```



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute firewall-rules describe labnet-allow-internal
allowed:
- IPProtocol: icmp
- IPProtocol: tcp
  ports:
  - '22'
creationTimestamp: '2022-11-11T05:46:49.750-08:00'
description: ''
direction: INGRESS
disabled: false
id: '8943023336896851110'
kind: compute#firewall
logConfig:
  enable: false
name: labnet-allow-internal
network: https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/networks/labnet
priority: 1000
selfLink: https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/firewalls/labnet-allow-internal
sourceRanges:
- 0.0.0.0/0
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Create another network

Now you'll create a another network, add firewall rules to it, then add VMs to both networks to test the ability to communicate with the networks.

1. Run the following command to create the **privatenet** network:

```
gcloud compute networks create privatenet --subnet-mode=custom
```

2. Create the **private-sub** subnet:

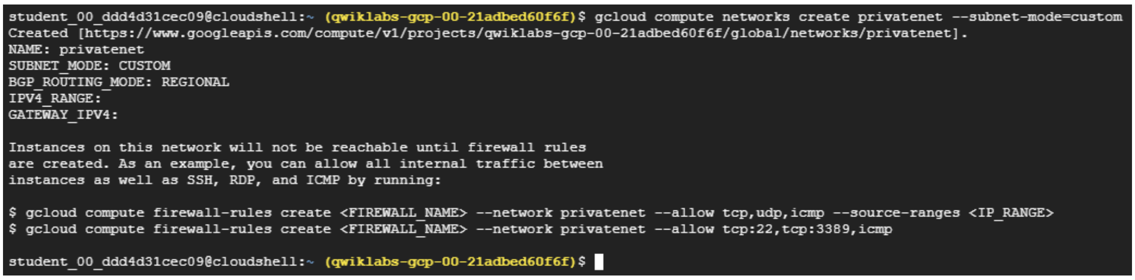
```
gcloud compute networks subnets create private-sub \
    --network=privatenet \
    --region=us-central1 \
    --range 10.1.0.0/28
```

Create the firewall rules for privatenet

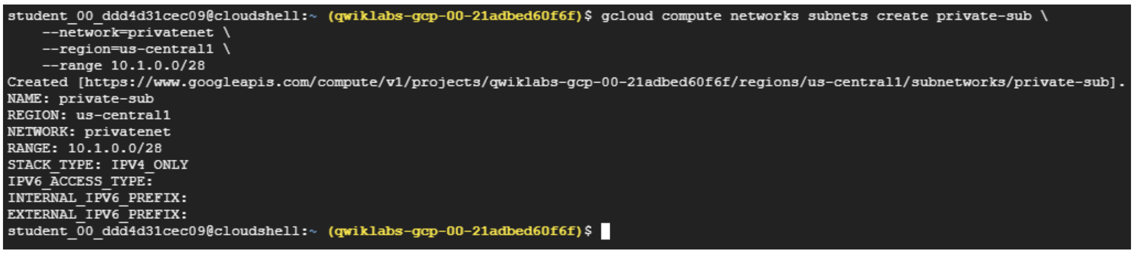
1. Run the following command to create the **privatenet-deny** firewall rule:

```
gcloud compute firewall-rules create privatenet-deny \
    --network=privatenet \
    --action=DENY \
    --rules=icmp,tcp:22 \
    --source-ranges=0.0.0.0/0
```

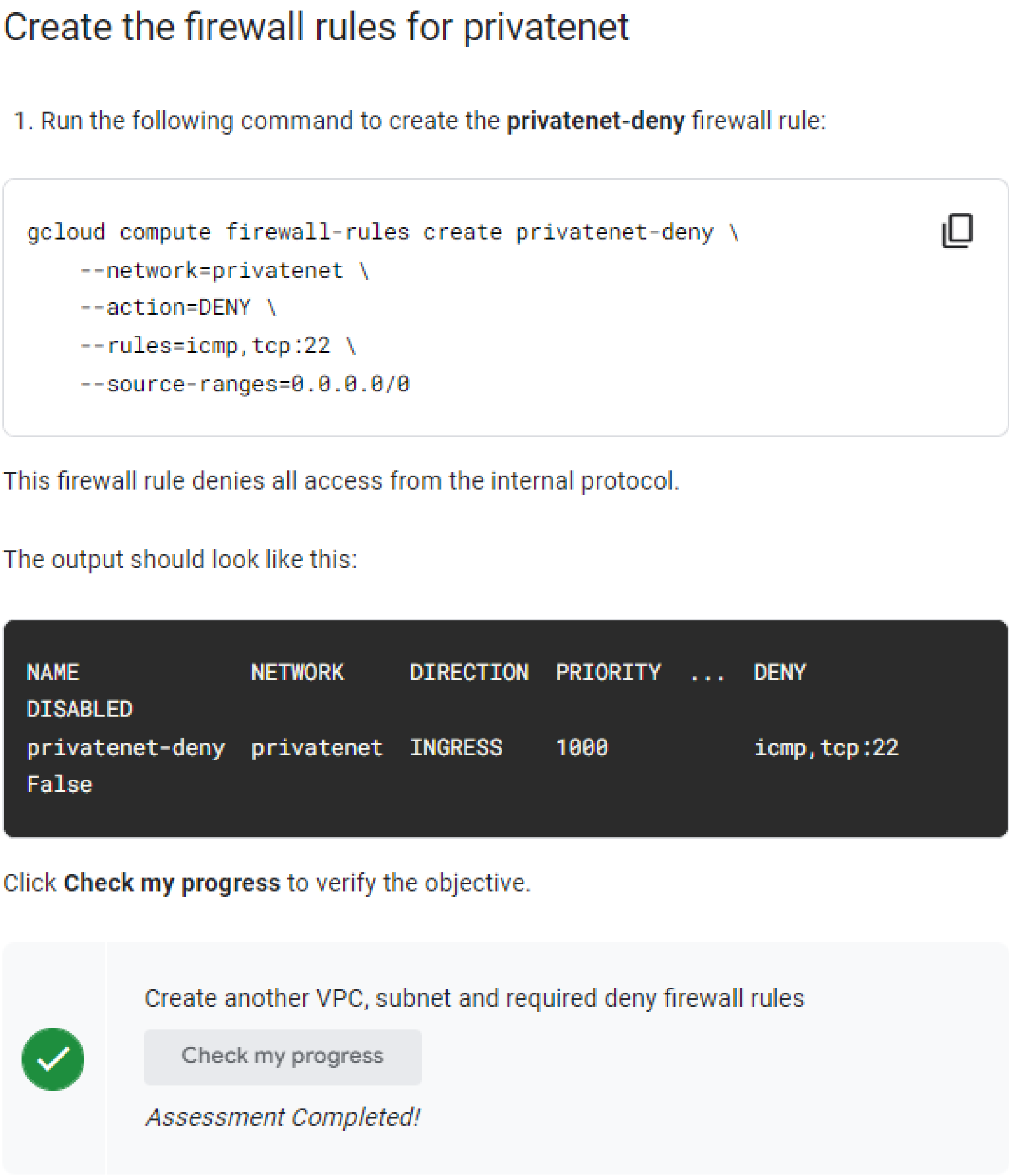
This firewall rule denies all access from the internal protocol.



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks create privatenet --subnet-mode=custom
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/networks/privatenet].
NAME: privatenet
SUBNET_MODE: CUSTOM
BGP_ROUTING_MODE: REGIONAL
IPV4_RANGE:
GATEWAY_IPV4:

Instances on this network will not be reachable until firewall rules
are created. As an example, you can allow all internal traffic between
instances as well as SSH, RDP, and ICMP by running:

$ gcloud compute firewall-rules create <FIREWALL_NAME> --network privatenet --allow tcp,udp,icmp --source-ranges <IP_RANGE>
$ gcloud compute firewall-rules create <FIREWALL_NAME> --network privatenet --allow tcp:22,tcp:3389,icmp

student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute networks subnets create private-sub \
    --network=privatenet \
    --region=us-central1 \
    --range 10.1.0.0/28
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/regions/us-central1/subnetworks/private-sub].
NAME: private-sub
REGION: us-central1
NETWORK: privatenet
RANGE: 10.1.0.0/28
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Create the firewall rules for privatenet

1. Run the following command to create the **privatenet-deny** firewall rule:

```
gcloud compute firewall-rules create privatenet-deny \
    --network=privatenet \
    --action=DENY \
    --rules=icmp,tcp:22 \
    --source-ranges=0.0.0.0/0
```

This firewall rule denies all access from the internal protocol.

The output should look like this:

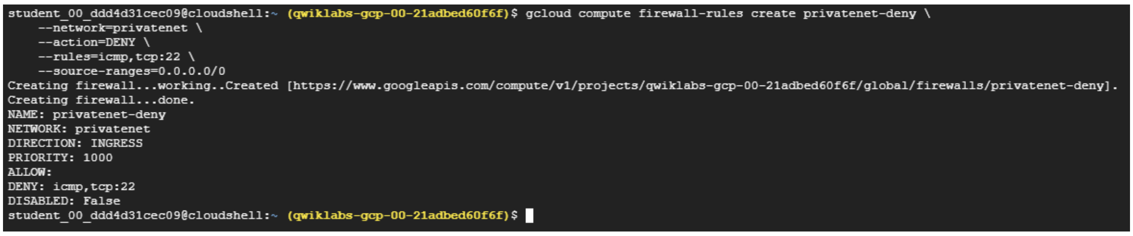
```
NAME             NETWORK     DIRECTION  PRIORITY  ...  DENY
DISABLED
privatenet-deny  privatenet  INGRESS    1000           icmp,tcp:22
False
```

Click **Check my progress** to verify the objective.

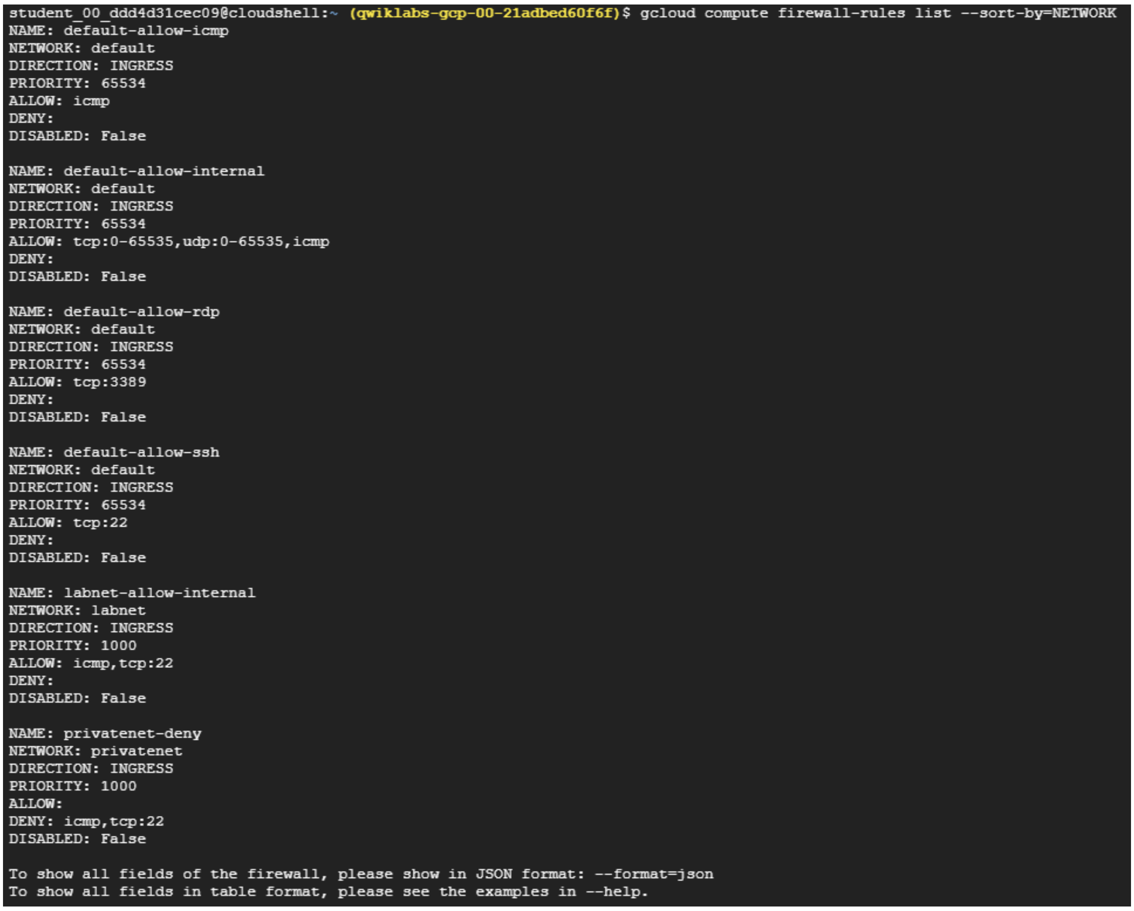
✓ Create another VPC, subnet and required deny firewall rules

Check my progress

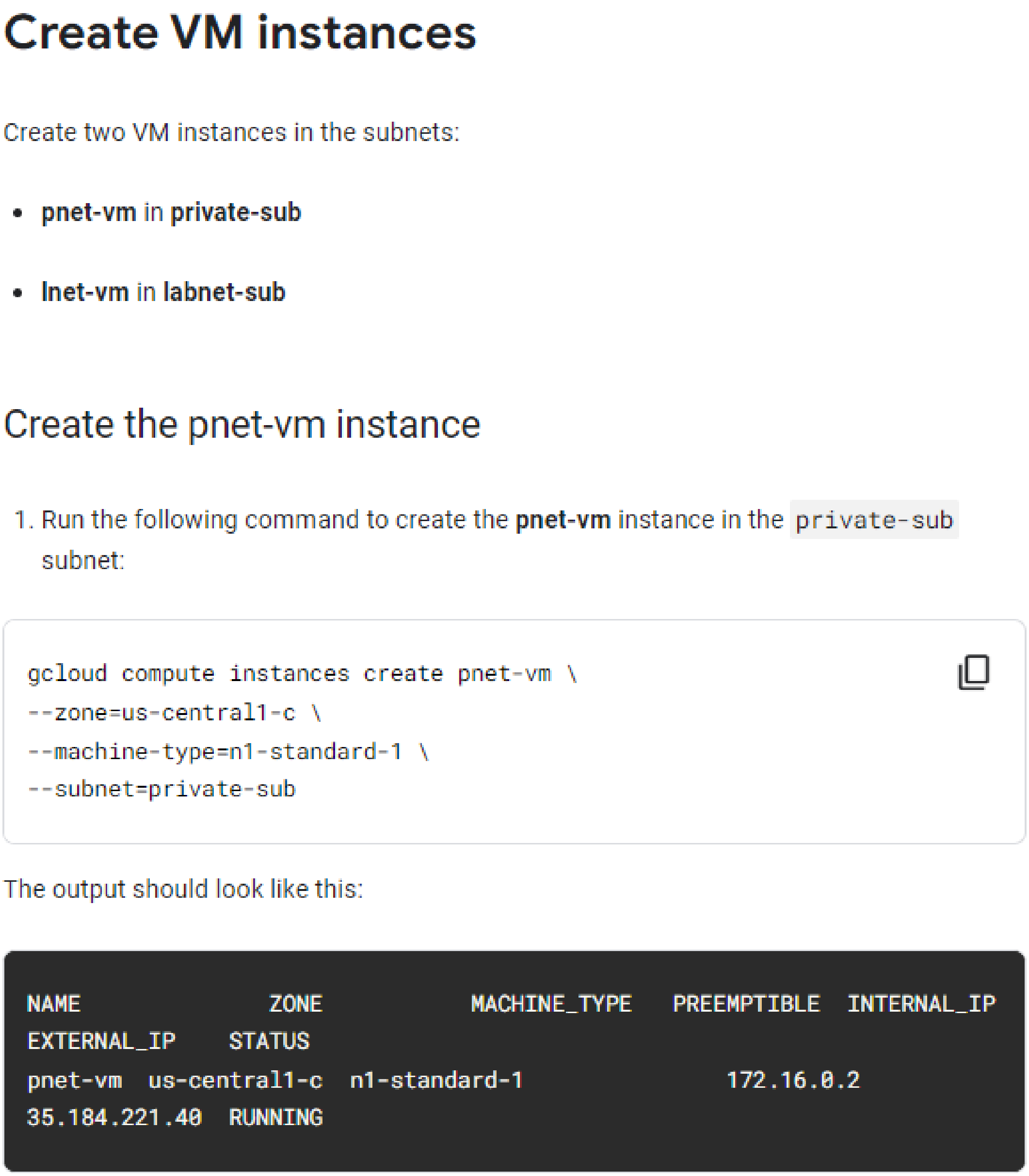
Assessment Completed!



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute firewall-rules create privatenet-deny \
    --network=privatenet \
    --action=DENY \
    --rules=icmp,tcp:22 \
    --source-ranges=0.0.0.0/0
Creating firewall...working..Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/global/firewalls/privatenet-deny].
Creating firewall...done.
NAME: privatenet-deny
NETWORK: privatenet
DIRECTION: INGRESS
PRIORITY: 1000
ALLOW:
DENY: icmp,tcp:22
DISABLED: False
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute firewall-rules list --sort-by=NETWORK
NAME: default-allow-icmp
NETWORK: default
DIRECTION: INGRESS
PRIORITY: 65534
ALLOW: icmp
DENY:
DISABLED: False

NAME: default-allow-internal
NETWORK: default
DIRECTION: INGRESS
PRIORITY: 65534
ALLOW: tcp:0-65535,udp:0-65535,icmp
DENY:
DISABLED: False

NAME: default-allow-rdp
NETWORK: default
DIRECTION: INGRESS
PRIORITY: 65534
ALLOW: tcp:3389
DENY:
DISABLED: False

NAME: default-allow-ssh
NETWORK: default
DIRECTION: INGRESS
PRIORITY: 65534
ALLOW: tcp:22
DENY:
DISABLED: False

NAME: labnet-allow-internal
NETWORK: labnet
DIRECTION: INGRESS
PRIORITY: 1000
ALLOW: icmp,tcp:22
DENY:
DISABLED: False

NAME: privatenet-deny
NETWORK: privatenet
DIRECTION: INGRESS
PRIORITY: 1000
ALLOW:
DENY: icmp,tcp:22
DISABLED: False

To show all fields of the firewall, please show in JSON format: --format=json
To show all fields in table format, please see the examples in --help.
```



Create VM instances

Create two VM instances in the subnets:

- **pnet-vm** in **private-sub**
- **lnet-vm** in **labnet-sub**

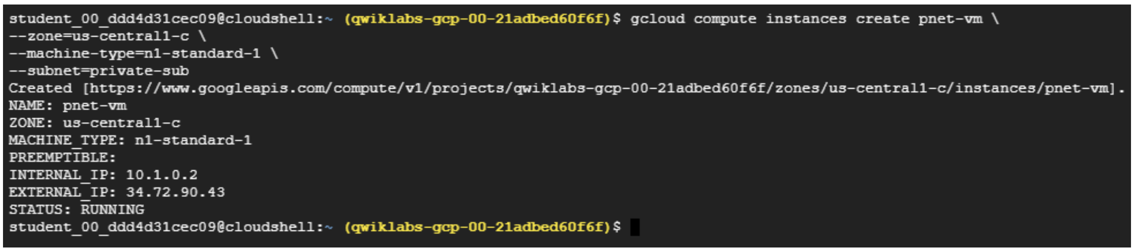
Create the pnet-vm instance

1. Run the following command to create the **pnet-vm** instance in the `private-sub` subnet:

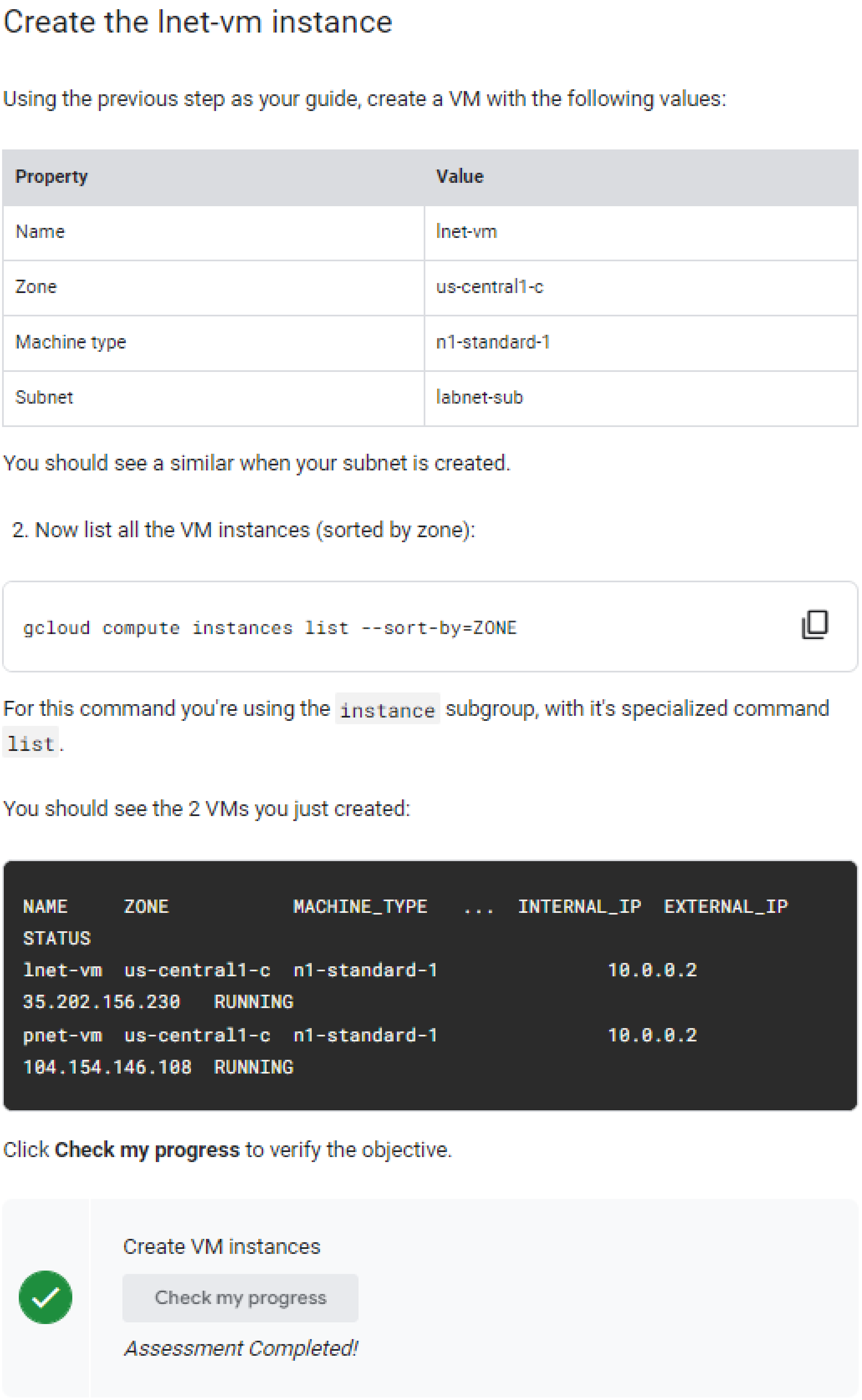
```
gcloud compute instances create pnet-vm \
--zone=us-central1-c \
--machine-type=n1-standard-1 \
--subnet=private-sub
```

The output should look like this:

```
NAME                ZONE          MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP
EXTERNAL_IP    STATUS
pnet-vm   us-central1-c  n1-standard-1             172.16.0.2
35.184.221.40  RUNNING
```



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute instances create pnet-vm \
--zone=us-central1-c \
--machine-type=n1-standard-1 \
--subnet=private-sub
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/zones/us-central1-c/instances/pnet-vm].
NAME: pnet-vm
ZONE: us-central1-c
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.1.0.2
EXTERNAL_IP: 34.72.90.43
STATUS: RUNNING
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Create the lnet-vm instance

Using the previous step as your guide, create a VM with the following values:

Property	Value
Name	lnet-vm
Zone	us-central1-c
Machine type	n1-standard-1
Subnet	labnet-sub

You should see a similar when your subnet is created.

2. Now list all the VM instances (sorted by zone):

```
gcloud compute instances list --sort-by=ZONE
```

For this command you're using the `instance` subgroup, with it's specialized command `list`.

You should see the 2 VMs you just created:

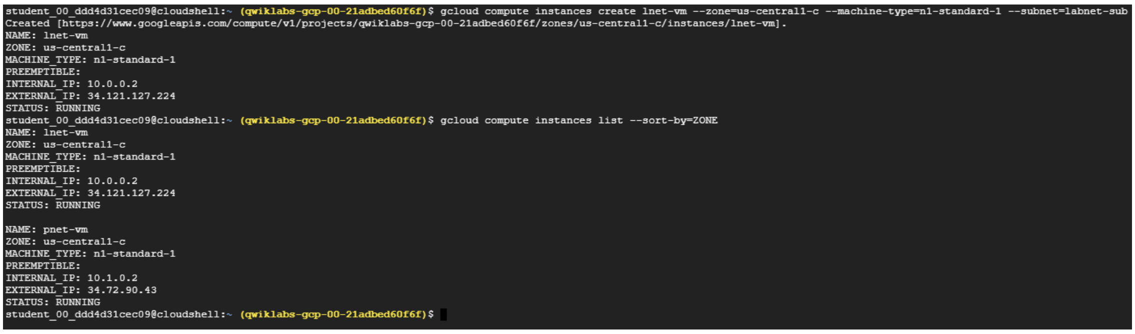
```
NAME     ZONE           MACHINE_TYPE   ...  INTERNAL_IP  EXTERNAL_IP
STATUS
lnet-vm  us-central1-c  n1-standard-1            10.0.0.2
35.202.156.230   RUNNING
pnet-vm  us-central1-c  n1-standard-1            10.0.0.2
104.154.146.108  RUNNING
```

Click **Check my progress** to verify the objective.

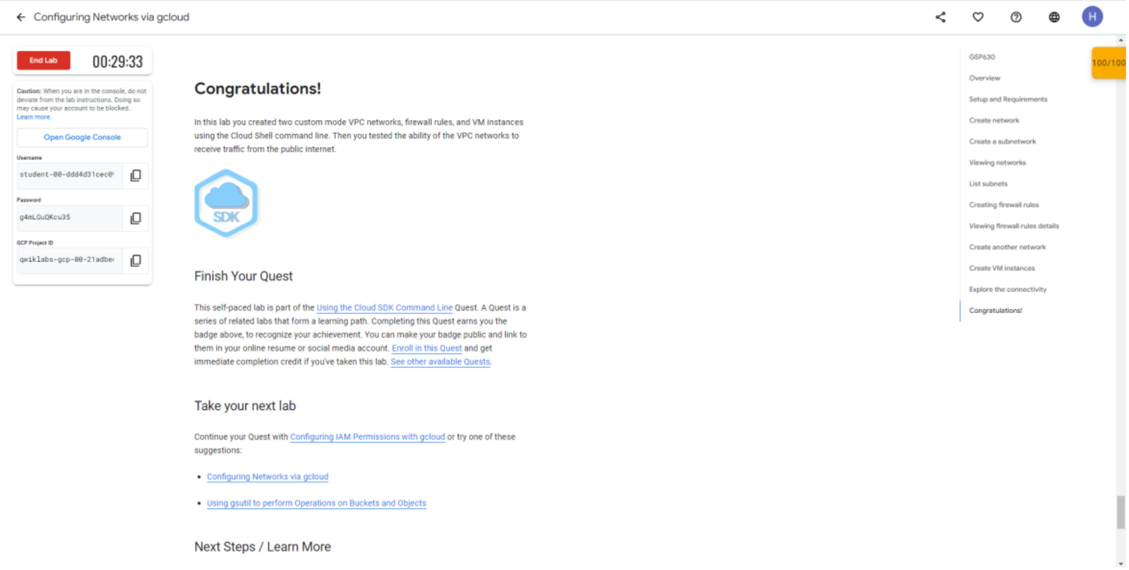
✓ Create VM instances

Check my progress

Assessment Completed!



```
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute instances create lnet-vm --zone=us-central1-c --machine-type=n1-standard-1 --subnet=labnet-sub
Created [https://www.googleapis.com/compute/v1/projects/qwiklabs-gcp-00-21adbed60f6f/zones/us-central1-c/instances/lnet-vm].
NAME: lnet-vm
ZONE: us-central1-c
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.0.0.2
EXTERNAL_IP: 34.121.127.224
STATUS: RUNNING
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$ gcloud compute instances list --sort-by=ZONE
NAME: lnet-vm
ZONE: us-central1-c
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.0.0.2
EXTERNAL_IP: 34.121.127.224
STATUS: RUNNING

NAME: pnet-vm
ZONE: us-central1-c
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.1.0.2
EXTERNAL_IP: 34.72.90.43
STATUS: RUNNING
student_00_ddd4d31cec09@cloudshell:~ (qwiklabs-gcp-00-21adbed60f6f)$
```



Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. Learn more.

Username
student-00-ddd4d31cec0

Password
g4mLGuQKcu35

GCP Project ID
qwiklabs-gcp-00-21adbe

Congratulations!

In this lab you created two custom mode VPC networks, firewall rules, and VM instances using the Cloud Shell command line. Then you tested the ability of the VPC networks to receive traffic from the public internet.

Finish Your Quest

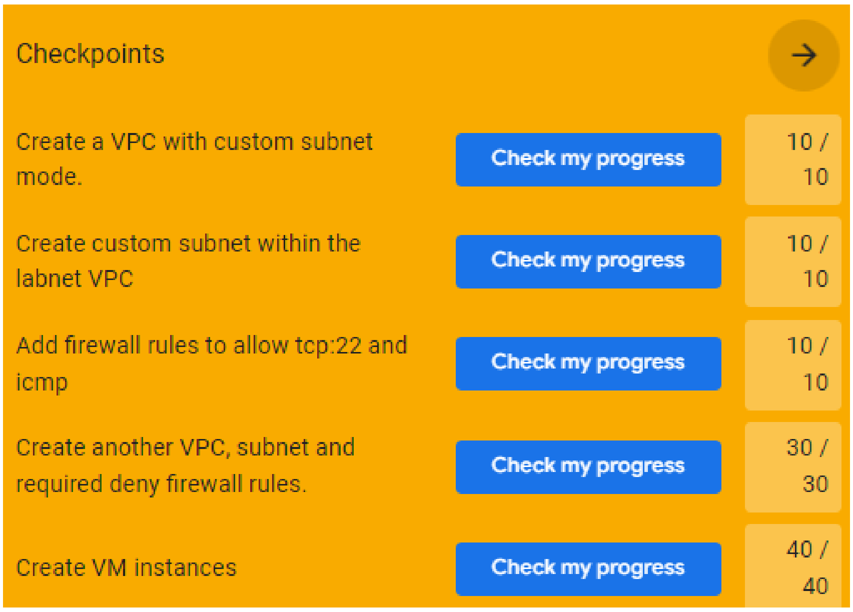
This self-paced lab is part of the Using the Cloud SDK Command Line Quest. A Quest is a series of related labs that form a learning path. Completing this Quest earns you the badge above, to recognize your achievement. You can make your badge public and link to them in your online resume or social media account. Enroll in this Quest and get immediate completion credit if you've taken this lab. See other available Quests.

Take your next lab

Continue your Quest with Configuring IAM Permissions with gcloud or try one of these suggestions:

- Configuring Networks via gcloud
- Using gsutil to perform Operations on Buckets and Objects

Next Steps / Learn More



Checkpoints

Create a VPC with custom subnet mode.	Check my progress	10 / 10
Create custom subnet within the labnet VPC	Check my progress	10 / 10
Add firewall rules to allow tcp:22 and icmp	Check my progress	10 / 10
Create another VPC, subnet and required deny firewall rules.	Check my progress	30 / 30
Create VM instances	Check my progress	40 / 40

Various options explored:

- **Cloud Interconnect:** Network Connectivity provides two options for extending your on-premises network to your VPC networks in Google Cloud. You can create a dedicated connection (Dedicated Interconnect) or use a service provider (Partner Interconnect) to connect to VPC networks. When choosing one of the following connection types, consider your connection requirements, such as the connection location and capacity.
- **Router appliance:** Router appliance is an alternative way of enabling connectivity between sites outside of Google Cloud through a Network Connectivity Center hub. You peer your Router appliance with Cloud Router to provide this connectivity. appliance spoke, which you attach to a Network Connectivity Center hub.
- **Direct Peering:** Direct Peering enables you to establish a direct peering connection between your business network and Google's edge network and exchange high-throughput cloud traffic. This capability is available at any of more than 100 locations in 33 countries around the world. For more information about Google's edge locations, see Google's peering site.
- **Cloud Router:** Cloud Router is a fully distributed and managed Google Cloud service that uses the Border Gateway Protocol (BGP) to advertise IP address ranges. It programs custom dynamic routes based on the BGP advertisements that it receives from a peer. Instead of a physical device or appliance, each Cloud Router consists of software tasks that act as BGP speakers and responders.

2. Explain your program logic, classes and methods used, as applicable.
Methods used:

- **Creation of a network:** Networking, also known as computer networking, is the practice of transporting and exchanging data between nodes over a shared medium in an information system. Networking comprises not only the design, construction and use of a network, but also the management, maintenance and operation of the network infrastructure, software and policies.
- **Creation of subnetworks:** A subnet, or subnetwork, is a segmented piece of a larger network. More specifically, subnets are a logical partition of an IP network into multiple, smaller network segments. The Internet Protocol (IP) is the method for sending data from one computer to another over the internet. Each computer, or host, on the internet has at least one IP address as a unique identifier.
- **Creation of virtual machine instances:** A VM is a virtualized instance of a computer that can perform almost all of the same functions as a computer,

including running applications and operating systems. Virtual machines run on a physical machine and access computing resources from software called a hypervisor. The hypervisor abstracts the physical machine's resources into a pool that can be provisioned and distributed as needed, enabling multiple VMs to run on a single physical machine.

- **Creation of firewall rules:** The firewall rules are the access control mechanism used by firewalls to safeguard your network from harmful applications and unauthorized access. Firewall rules determine which types of traffic your firewall accepts and which are denied. A collection of firewall rules make up the firewall access policy.

3. Explain the Importance of the approach followed by you

Network configuration can reduce downtime by allowing system administrators to rapidly identify changes being made in the network. It also helps ensure that software versions and hardware components are up to date and comply with licensing agreements. Visibility and accountability is also improved, as system personnel have an easy way to determine the identity of components and software operating on the network.

In addition, network configuration can:

- Streamline the processes of maintenance, repair, expansion and upgrading.
- Minimize configuration errors as part of change management.
- Optimize network security.
- Ensure that changes made to a device or system do not adversely affect other devices or systems.
- Roll back changes to a previous configuration if system updating or replacement efforts are unsatisfactory.
- Archive the details of all network configuration changes.

Conclusion: - Successfully understood how to configure a network in google cloud and what are the advantages of doing so. How network configuration works and what benefits does it provide.