

GCP Certification Full course in Hindi Associate Cloud Engineer



DEVOPS MADE SIMPLE

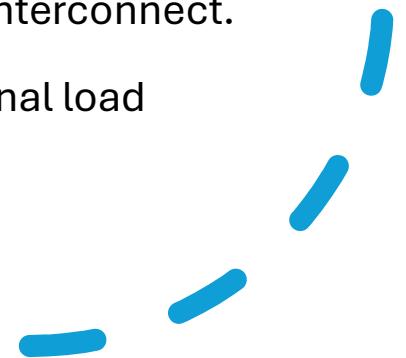
Youtube channel link- [Google Cloud Platform \(GCP\) Full Course 2025 - Hindi | Associate Cloud Engineer Certification - YouTube](#)

Virtual Private Cloud (VPC)

- A Virtual Private Cloud (VPC) network is a virtual version of a physical network that is implemented inside of Google's production network
- Projects can contain multiple VPC networks

A VPC network does the following:

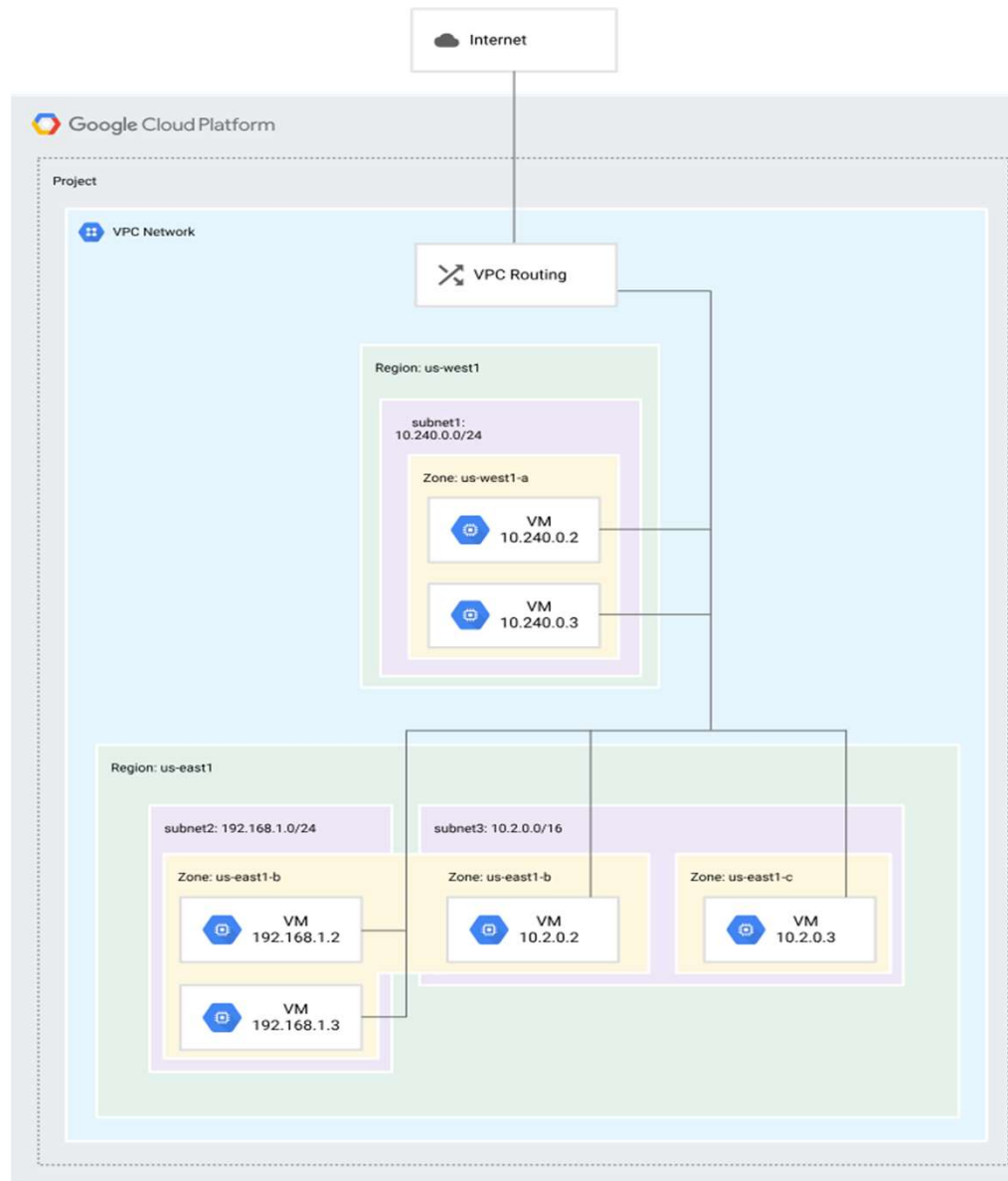
- Provides connectivity for your Compute Engine virtual machine (VM) instances.
- Offers native internal passthrough Network Load Balancers and proxy systems for internal Application Load Balancers.
- Connects to on-premises networks by using Cloud VPN tunnels and VLAN attachments for Cloud Interconnect.
- Distributes traffic from Google Cloud external load balancers to backends.



VPC networks have the following properties:

- VPC networks, including their associated routes and firewall rules, are global resources. They are *not* associated with any particular region or zone.
- Subnets are regional resources.
- Resources within a VPC network can communicate with one another
- VPC networks can be connected to other VPC networks in different projects or organizations by using VPC Network Peering.
- VPC networks can be securely connected in hybrid environments by using Cloud VPN or Cloud Interconnect.





Subnet creation mode

Auto mode VPC network –

- one subnet from each region is automatically created within it.
- As new Google Cloud regions become available, new subnets in those regions are automatically added to auto mode VPC networks
- you can add more subnets manually to auto mode VPC networks in regions that you choose by using IP ranges outside of 10.128.0.0/9.
- Supports single-stack i.e., only IPv4
- Because the subnets of every auto mode VPC network use the same predefined range of IP addresses, you can't connect auto mode VPC networks to one another by using VPC Network Peering or Cloud VPN.

Custom mode VPC network-

- no subnets are automatically created.
- You decide which subnets to create in regions that you choose by using IP ranges that you specify.
- Supports VPC peering and Cloud VPN to connect to other VPCs
- Supports dual-stack i.e., both IPv4 and IPv6

Note:- You can switch a VPC network from auto mode to custom mode, while custom mode VPC networks cannot be changed to auto mode VPC networks.

Calculating Subnet CIDR Range (Classless Inter-domain Routing)

Example 1- 10.4.1.0/24

- IPv4- 32 bits
- here 24 represent number of network bits
- Calculating Available IPs

1.Total no. of bits - network bits

$$32-24 = 8$$

2. No. of available IPs = $2^8 = 256$

- So usable IPs in CIDR range - 10.4.1.1 - 10.4.1.254
- 10.4.1.0 - reserved for subnet
- 10.4.1.255 - reserved for broadcast

Example 2 - 10.4.1.1/32

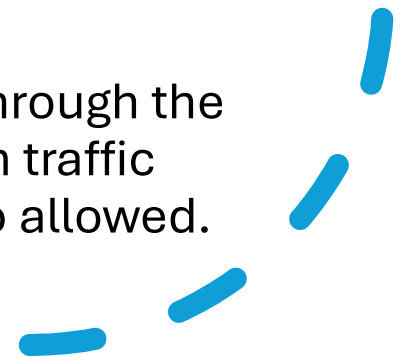
$$32-32 = 0$$

- Now no. of available IPs = 2^0 which is 1
- so only one IP is available which is 10.4.1.1



VPC firewall rules

- Virtual Private Cloud (VPC) firewall rules apply to a given project and network
- VPC firewall rules let you allow or deny connections to or from virtual machine (VM) instances in your VPC network.
- Each firewall rule applies to incoming (ingress) or outgoing (egress) connections, not both.
- Each firewall rule's action is either allow or deny. The rule applies to connections as long as it is enforced
- VPC firewall rules are stateful:
 - When a connection is allowed through the firewall in either direction, return traffic matching this connection is also allowed.



Implied rules

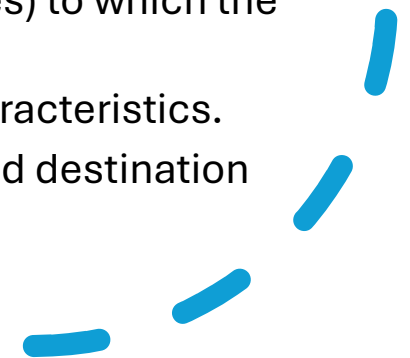
- An ingress rule whose action is deny, source is 0.0.0.0/0, and priority is the lowest possible (65535) protects all instances by blocking incoming connections to them.
 - A higher priority rule might allow incoming access
- An egress rule whose action is allow, destination is 0.0.0.0/0, and priority is the lowest possible (65535) lets any instance send traffic to any destination.
 - A higher priority firewall rule may restrict outbound access

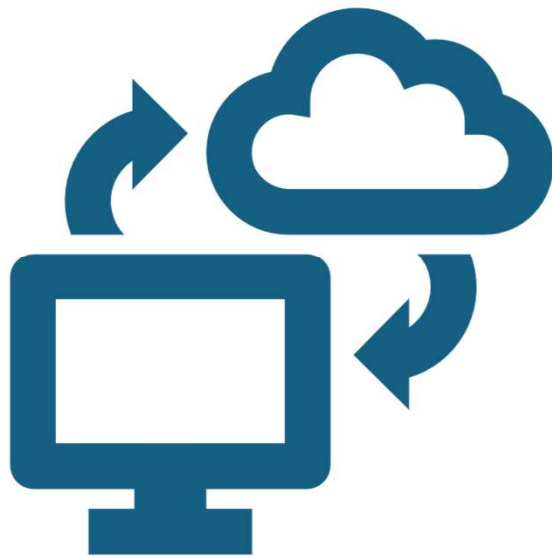


Firewall rule components

Each firewall rule consists of the following configuration components:

- A **direction** from the perspective of the target. Direction can be either ingress or egress.
- A numerical **priority**, which determines whether the rule is applied. Only the highest priority (lowest priority number) rule whose other components match traffic is applied; conflicting rules with lower priorities are ignored.
- An **action on match**, either allow or deny, which determines whether the rule permits or blocks connections.
- The **enforcement status** of the firewall rule: You can enable and disable firewall rules without deleting them.
- A **target**, which defines the instances (including GKE clusters and App Engine flexible environment instances) to which the rule applies.
- A **source or destination** filter for packet characteristics.
- The **protocol** (such as TCP, UDP, or ICMP) and destination port.





IP addresses

IP addresses let Google Cloud resources communicate with other resources in Google Cloud, in on-premises networks, or on the public internet.

External IP address

- External IP addresses are reachable by any host on the internet.
- Resources with external IP addresses can communicate with the public internet.
- External IPv4 addresses for resources can be provided by Google, or you can bring your own IP (BYOIP) addresses to Google Cloud.

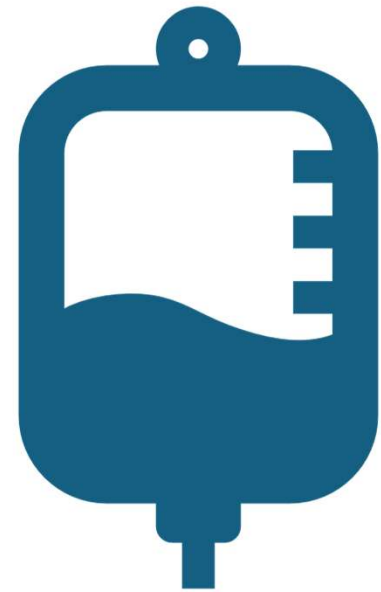
Internal IP address

- Internal IP addresses cannot be reached from the internet and are not publicly routable.
- Resources with internal IP addresses communicate with other resources as if they are all on the same private network.

Ephemeral and Static IP addresses

Internal and external IP addresses can be ephemeral or static.

- An ephemeral IP address is an IP address that doesn't persist beyond the life of the resource.
- In general, the ephemeral IP address is released if you stop or delete the resource.
- Reserving a static IP address assigns the address to your project until you explicitly release it.
- Static addresses are useful if you need to move an IP address from one Google Cloud resource to another.





Reserve a static internal IP address

- Static internal IP addresses provide the ability to reserve internal IP addresses from the IP address range configured in the subnet and then assign those reserved internal IP addresses to resources as needed.
- Only one resource at a time can use a static internal IP address.
- Deleting a resource does not automatically release a static internal IP address. You must manually release static internal IP addresses when you no longer require them.
- You can't unassign or change the internal IPv4 address of an existing resource.
 - For example, you can't assign a new static internal IP address to a running or a stopped VM instance.
- You can promote an ephemeral internal IP address of a resource to a static internal IP address.
- You cannot change the name of a static IP address.
- Static internal IP addresses are regional, meaning they are restricted to the region in which they are reserved.

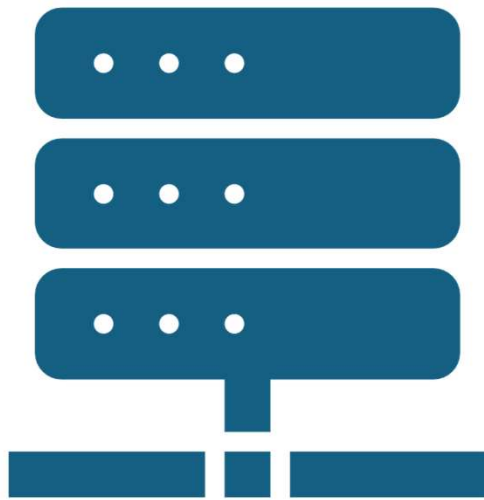
Reserve a static external IP address

- A static external IP address is the IP address that is reserved for your resource until you decide to release it.
- Only one resource at a time can use a static external IP address.
- You can not change the name of a static IP address.

Release a static external IP address

- If you no longer need a static external IPv4 or IPv6 address, you can release the IP address by deleting the IP address resource.
- If you are using the Google Cloud console, you can release a static IP address only if it is not being used by another resource.
- If you're using the gcloud CLI or API, you can release an IP address whether or not it's being used by another resource.
 - If the IP address is not being used by a resource, the IP address is returned to the pool of available external IP addresses.
 - If the IP address is being used by a resource, it remains attached to the resource until the resource is deleted.





Cloud NAT

Cloud NAT provides network address translation (NAT) for outbound traffic to the internet, Virtual Private Cloud (VPC) networks, on-premises networks, and other cloud provider networks.

Cloud NAT provides NAT for the following Google Cloud resources:

- Compute Engine virtual machine (VM) instances
- Google Kubernetes Engine (GKE) clusters
- Cloud Run instances through Serverless VPC Access or Direct VPC egress
- Cloud Run functions instances through Serverless VPC Access
- App Engine standard environment instances through Serverless VPC Access
- Regional internet network endpoint groups (NEGs)



Types of Cloud NAT

By using a Cloud NAT gateway, your Google Cloud resources can connect to resources outside of the source VPC network.

A Cloud NAT gateway supports the following types of NAT:

- Public NAT
- Private NAT

You can use both Public NAT and Private NAT to provide NAT services to the same subnet in a VPC network.



Public NAT

- Public NAT lets Google Cloud resources that don't have external IPv4 addresses communicate with IPv4 destinations on the internet.
- VMs use a set of shared external IP addresses to connect to the internet.
- Cloud NAT gateway allocates a set of external IP addresses and source ports to each VM that uses the gateway to create outbound connections to the internet.



Private NAT

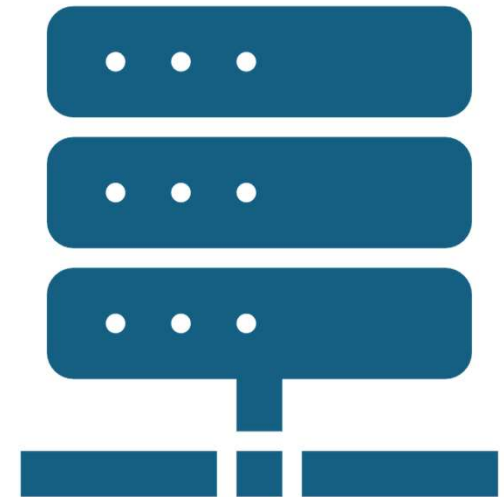
Private NAT enables private-to-private NAT for the following traffic-

- From a VPC network to another VPC network
- From a VPC network to a network outside of Google Cloud

Example- Assume that your Google Cloud resources in a VPC network need to communicate with destinations in a VPC, on-premises, or other cloud provider network that is owned by a different business unit. However, the destination network contains subnets whose IP addresses overlap with the IP addresses of your VPC network. In this scenario, you create a Cloud NAT gateway for Private NAT that translates traffic between the subnets in your VPC network to the non-overlapping subnets of the other network

Benefits of Cloud NAT

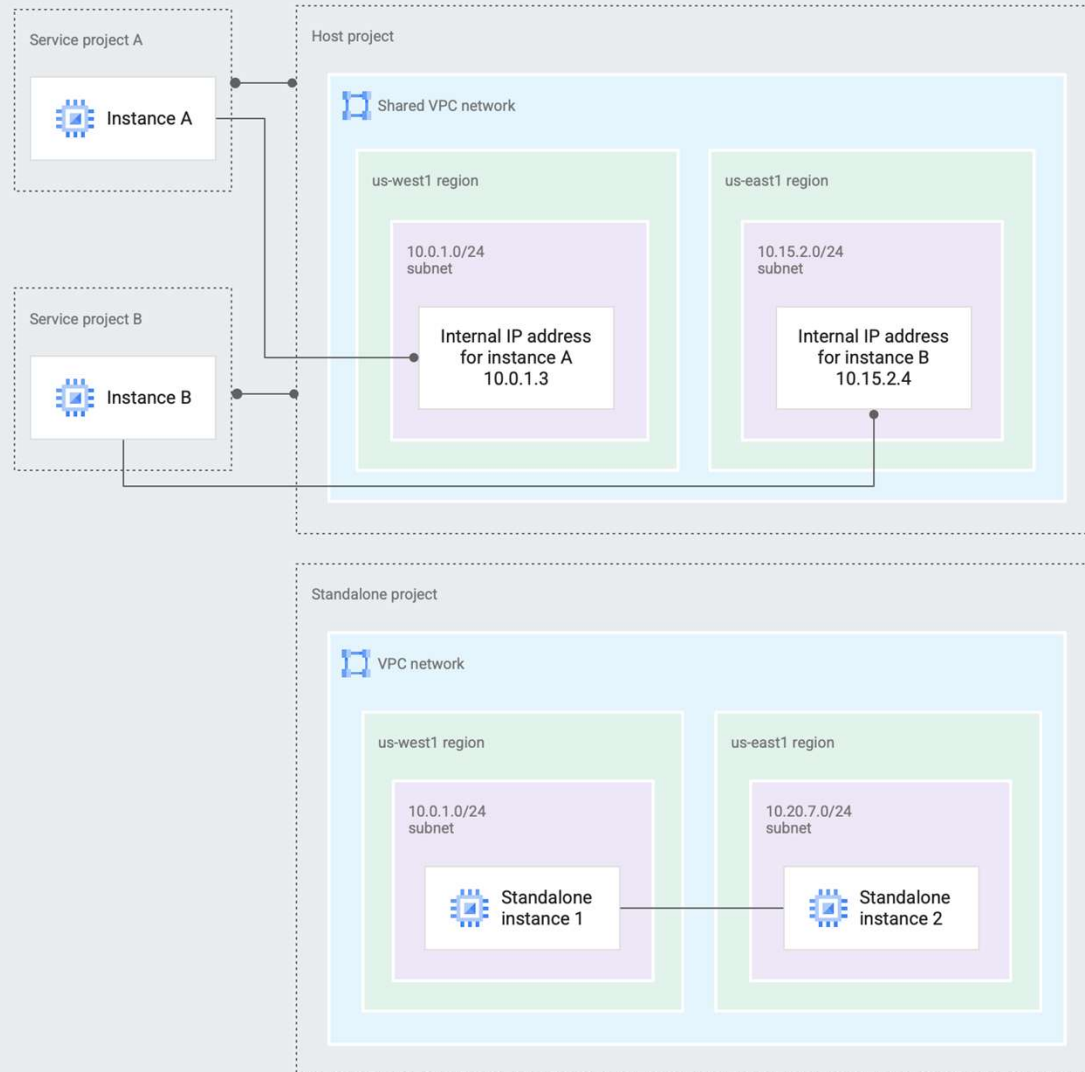
- **Security** - When using a Cloud NAT gateway for Public NAT, you can reduce the need for individual VMs to each have external IP addresses.
- **Availability** - Cloud NAT is a distributed, software-defined managed service. It doesn't depend on any VMs in your project or a single physical gateway device. You configure a NAT gateway on a Cloud Router, which provides the control plane for NAT, holding configuration parameters that you specify.
- **Scalability** - Cloud NAT can be configured to automatically scale the number of NAT IP addresses that it uses, and it supports VMs that belong to managed instance groups, including the groups with autoscaling enabled.
- **Performance** - Cloud NAT does not reduce the network bandwidth per VM. Cloud NAT is implemented by Google's Andromeda software-defined networking.
- **Logging** - For Cloud NAT traffic, you can trace the connections and bandwidth for compliance, debugging, analytics, and accounting purposes.
- **Monitoring** - Cloud NAT exposes key metrics to Cloud Monitoring that give you insight into your fleet's use of NAT gateways. Metrics are sent automatically to Cloud Monitoring. There, you can create custom dashboards, set up alerts, and query metrics.



Shared VPC

- Shared VPC allows an organization to connect resources from multiple projects to a common Virtual Private Cloud (VPC) network so that they can communicate with each other securely and efficiently by using internal IP addresses from that network.
- When you use Shared VPC, you designate a project as a host project and attach one or more other service projects to it. The VPC networks in the host project are called Shared VPC networks. Eligible resources from service projects can use subnets in the Shared VPC network.





- A project that participates in Shared VPC is either a *host project* or a *service project*.
- A host project contains one or more Shared VPC networks. A Shared VPC Admin must first enable a project as a host project. After that, a Shared VPC Admin can attach one or more service projects to it.
- A service project is any project that has been attached to a host project by a Shared VPC Admin.
- A project cannot be both a host project and a service project simultaneously.
- You can create and use multiple host projects; however, each service project can only be attached to a single host project.
- You can create networks, subnets, secondary address ranges, firewall rules, and other network resources in the host project. The host project can then share selected subnets, including secondary ranges, with the service projects.



Required administrative roles

- **Shared VPC Admin** - They perform various tasks necessary to set up Shared VPC, such as enabling host projects, attaching service projects to host projects, and delegating access to some or all the subnets in Shared VPC networks to Service Project Admins
- **Service Project Admin** - Service Project Admins also maintain ownership and control over resources defined in the service projects. They may have additional IAM roles to the service projects, such as project owner.
- **Network Admin** - Network Admins have full control over all network resources except for firewall rules and SSL certificates.
- **Security Admin** - Security Admins manage firewall rules and SSL certificates.



Billing

- Billing for resources that participate in a Shared VPC network is attributed to the service project where the resource is located, even though the resource uses the Shared VPC network in the host project.

Resources

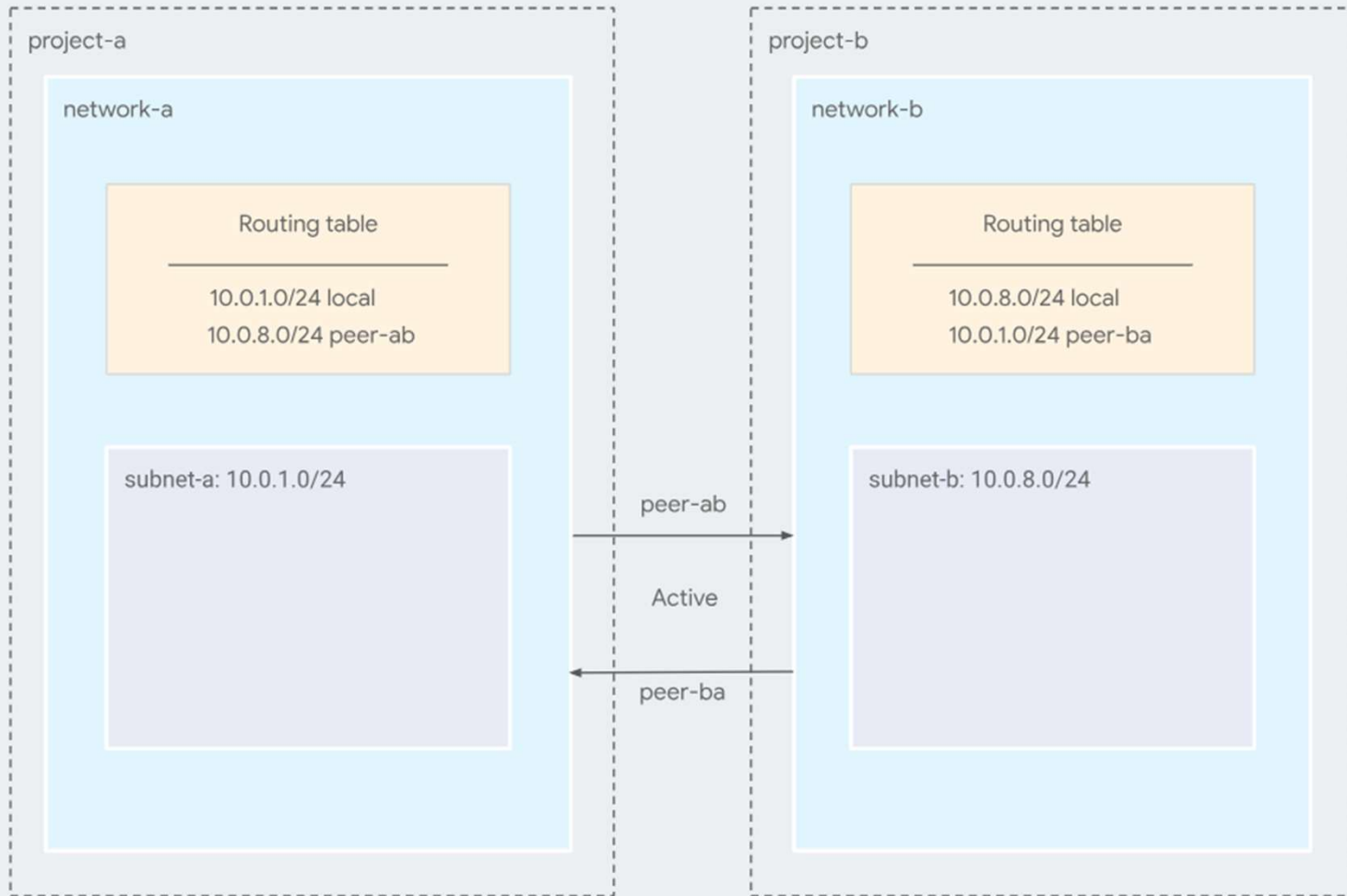
- Use of a Shared VPC network is not mandatory. For example, instance admins can create instances in the service project that use a VPC network in that project.
- Some resources must be re-created in order to use a Shared VPC network. When a Shared VPC Admin attaches an existing project to a host project, that project becomes a service project, but its existing resources do not automatically use shared network resources.



VPC Network Peering

- Google Cloud VPC Network Peering connects two Virtual Private Cloud (VPC) networks so that resources in each network can communicate with each other.
- Peered VPC networks can be in the same project, different projects of the same organization, or different projects of different organizations.
- VPC Network Peering works with Compute Engine, GKE, and App Engine flexible environment.

Google Cloud



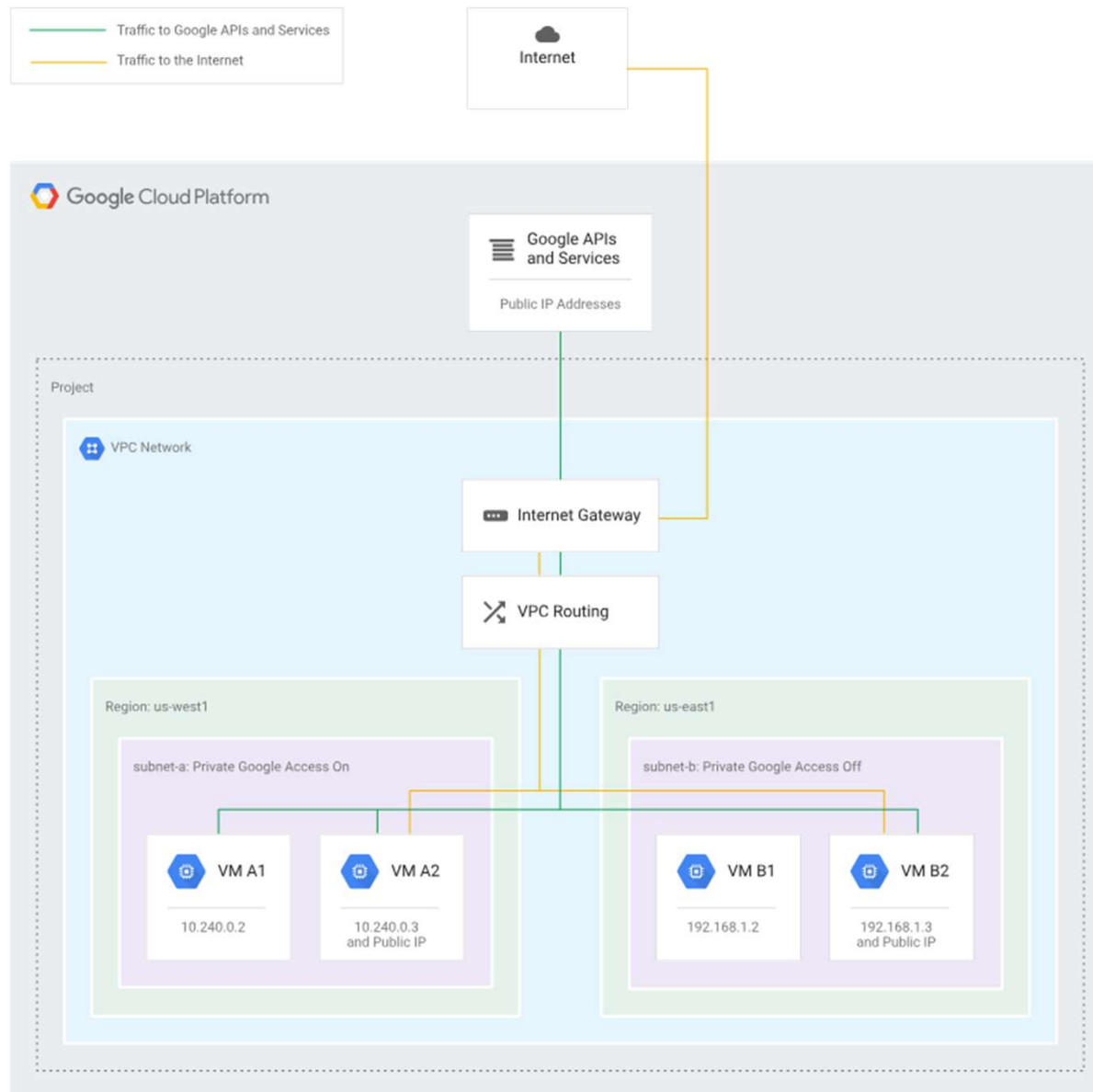
Connectivity

- VPC Network Peering supports connecting VPC networks, not legacy networks.
- VPC Network Peering provides internal IPv4 and IPv6 connectivity between pairs of VPC networks. Traffic flowing between peered networks has the same latency, throughput, and availability as traffic within the same VPC network.
- VPC Network Peering does not provide transitive routing. For example, if VPC networks net-a and net-b are connected using VPC Network Peering, and VPC networks net-a and net-c are also connected using VPC Network Peering, VPC Network Peering does not provide connectivity between net-b and net-c.
- You can't connect two auto mode VPC networks by using VPC Network Peering. This is because each subnet in an auto mode VPC network uses a subnet IP address range that fits within the 10.128.0.0/9 CIDR block.
- You can connect a custom mode VPC network to an auto mode VPC network as long as the subnet IP address ranges doesn't overlap.
- Resources in a peered VPC network can't use Compute Engine internal DNS names created by a local VPC network.



VPC Private Google Access

- VM instances that only have internal IP addresses (no external IP addresses) can use Private Google Access.
 - They can reach the external IP addresses of Google APIs and services.
- If you disable Private Google Access, the VM instances can no longer reach Google APIs and services; they can only send traffic within the VPC network.
- Private Google Access has no effect on instances that have external IP addresses. They don't need any special configuration to send requests to the external IP addresses of Google APIs and services.
- You enable Private Google Access on a subnet-by-subnet basis; it's a setting for subnets in a VPC network.






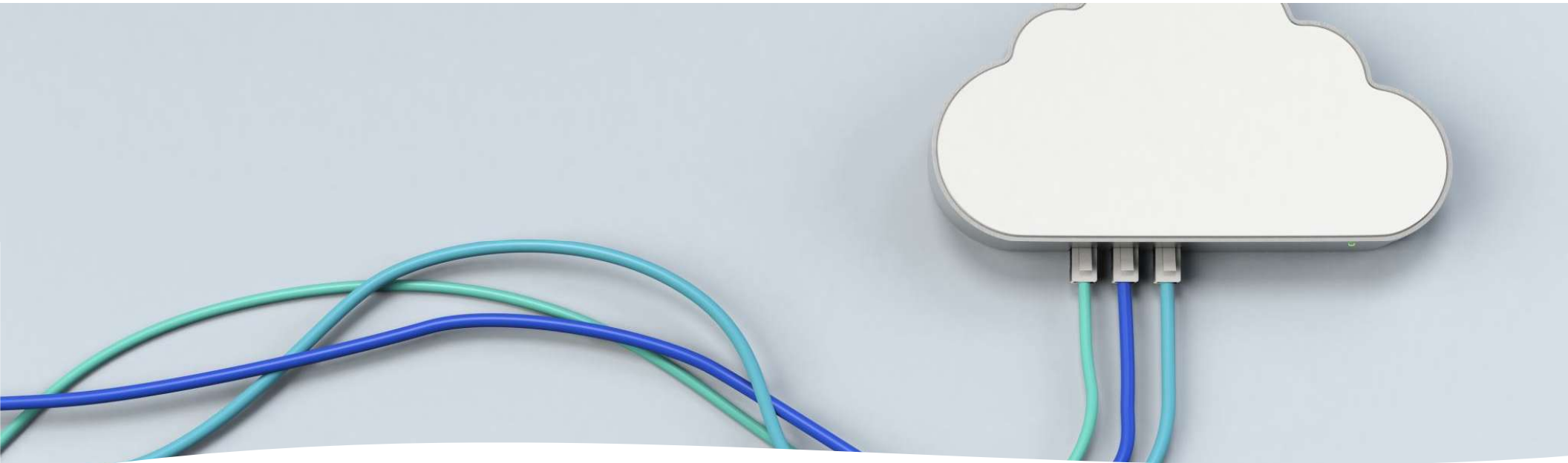
Cloud Domains



- Cloud Domains lets you register and configure a domain in Google Cloud.

Benefits of using Cloud Domains

- Let's you register a domain through Google Cloud and seamlessly attach it to any application
 - Bills your domains through the same Cloud Billing account that you already have.
 - Let's you manage domain registrations per project, not per individual. You can manage one or more of your domains in Cloud Domains as part of a single project.
- **You can use Cloud Domains to do the following:**
 - Search for available domains.
 - Buy a domain name.
 - Manage your registration.
 - Optional: Transfer your domain from Cloud Domains to another registrar.
- 



Cloud DNS

- Cloud DNS is a high-performance, resilient, global Domain Name System (DNS) service
- Cloud DNS lets you publish your zones and records in DNS without the burden of managing your own DNS servers and software.

Zones :

- **Public zone** - A public zone is visible to the internet. You can create DNS records in a public zone to publish your service on the internet.
- **Private zone** - A private zone is any zone that cannot be queried over the public internet.



Records

- A record is a mapping between a DNS resource and a domain name. Each individual DNS record has a type (name and number), an expiration time (time to live), and type-specific data.

Some of the commonly used record types are:

- **A:** Address record, which maps host names to their IPv4 address.
- **AAAA:** IPv6 Address record, which maps host names to their IPv6 address.
- **CNAME:** Canonical name record, which specifies alias names.