VPCs and Interconnecting Networks

VPC

Overview

Resources in GCP projects are split across VPCs (Virtual Private Clouds)

Routes and forwarding rules must be configured to allow traffic within a VPC and with the outside world

Traffic flows only after firewall rules are configured specifying what traffic is allowed or not

VPN, peering, shared VPCs are some of the ways to connect VPCs or a VPC with an on premise network

Virtual Private Cloud

A global private isolated virtual network partition that provides managed networking functionality

(www.docker.com)

The GCP Virtual Private Cloud

Provides global, scalable, flexible networking for your cloud based services

https://cloud.google.com/vpc/docs/vpc

VPC

- Global

  Resources from across zones, regions

- Multi-tenancy

  VPCs can be shared across GCP projects

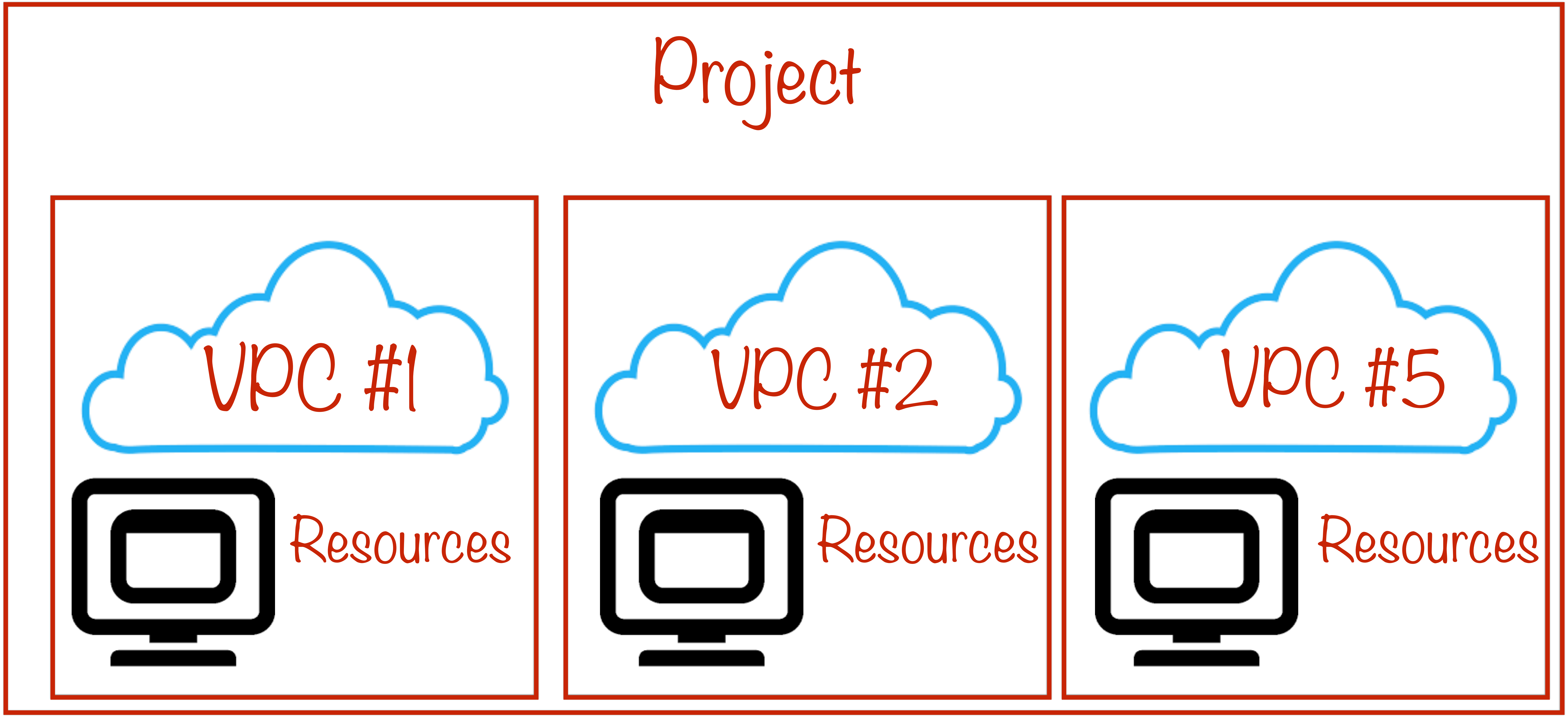
- Private and secure

  IAM, firewall rules

- Scalable

  Add new VMs, containers to the network

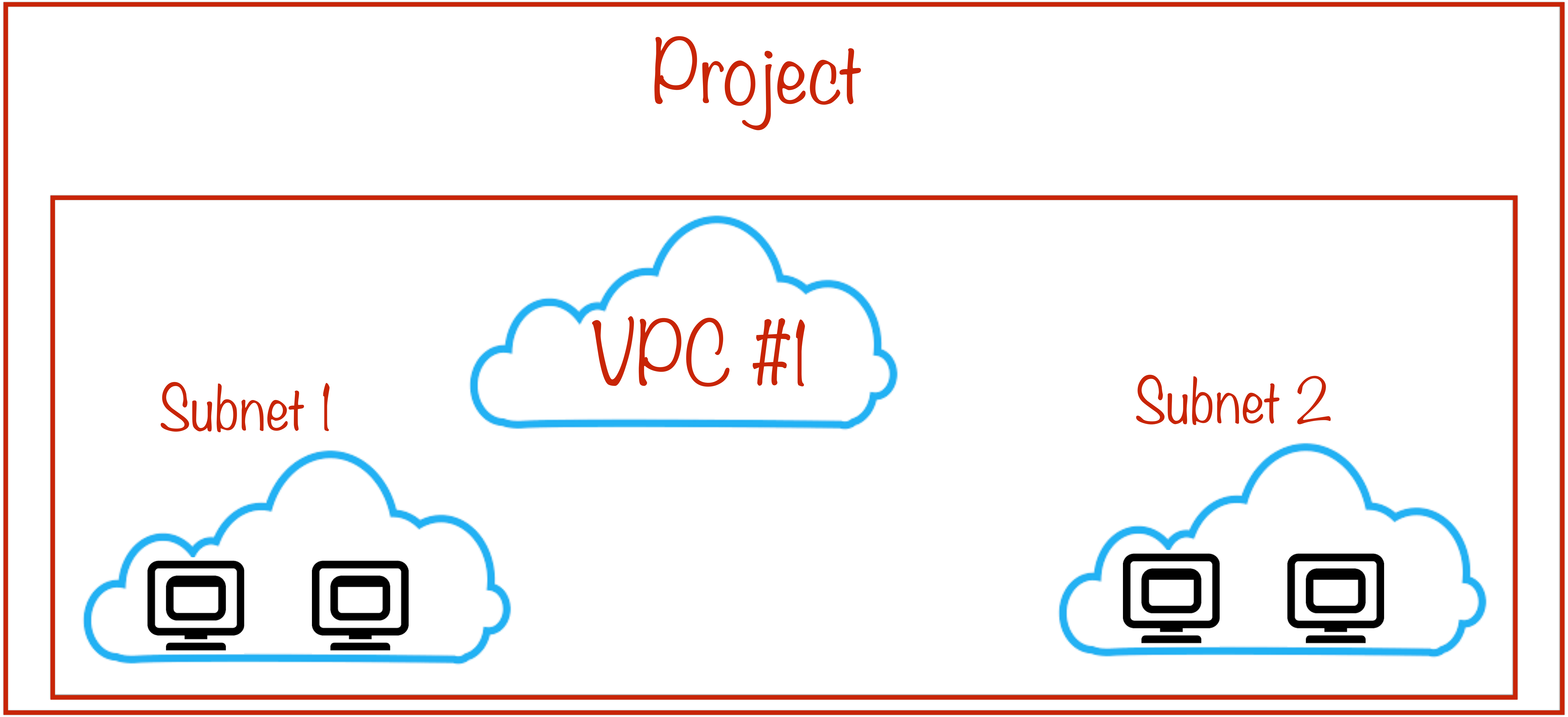
Projects and VPCs



A single project has a quota of 5 networks

A single network has a limit of 7000 instances

Projects and VPCs



Subnets

Subnet 1

Logical partitioning of the network

- Defined by a IP address prefix range
- Specified in CIDR notation
- IP ranges cannot overlap between subnets
- Subnets in the GCP can contain resources only from a single region

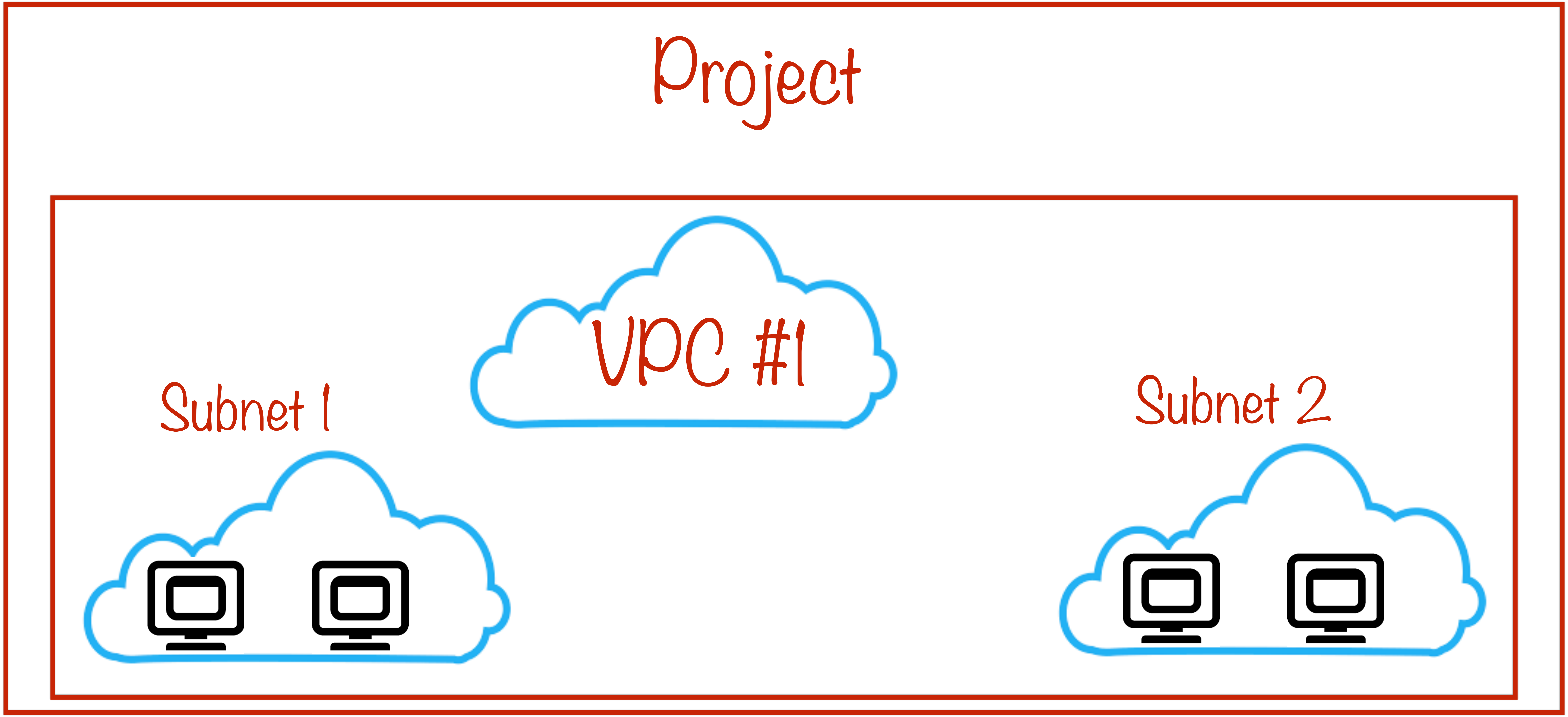
Subnets

Subnet 1

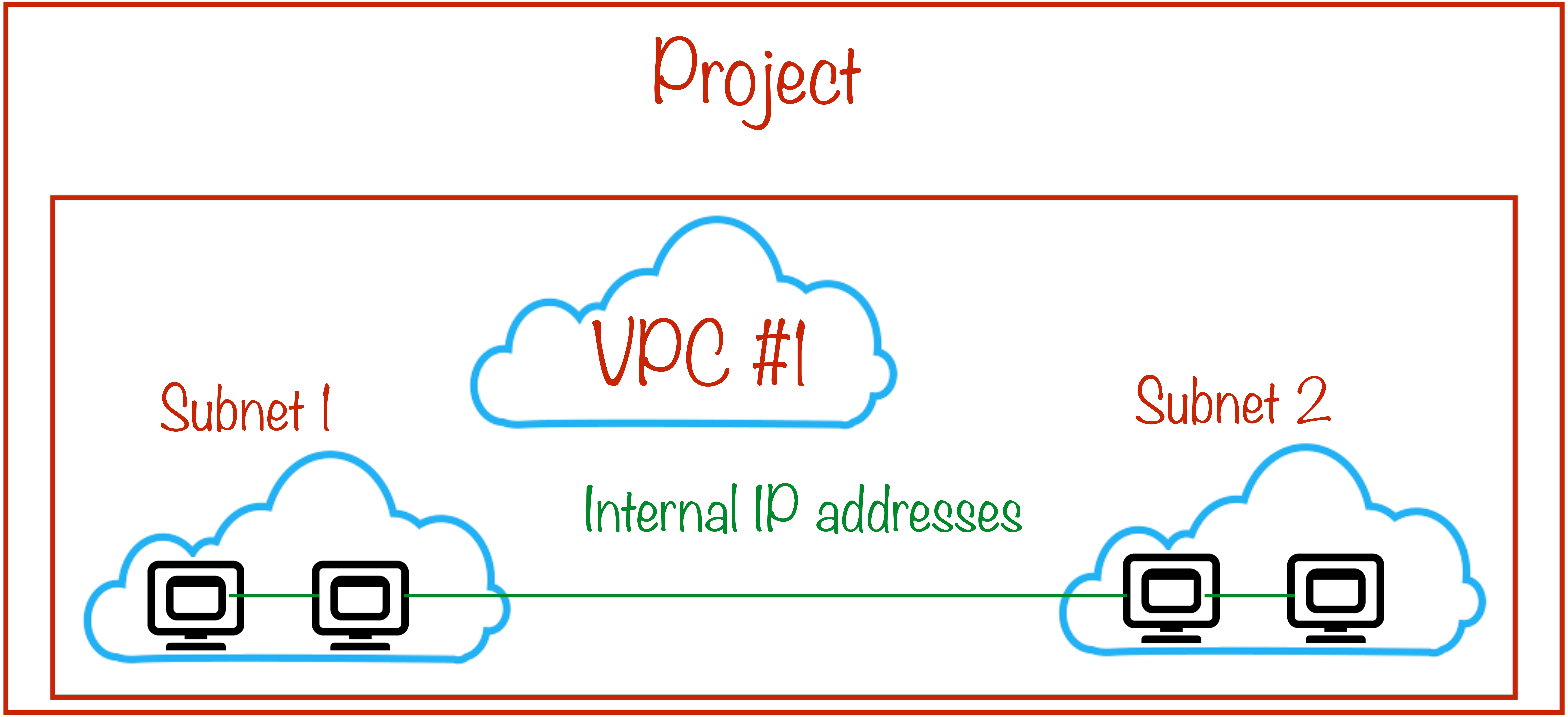
CIDR notation

- 10.123.9.0/24
- Contains all IP addresses in the range 10.123.9.**0** to 10.123.9.**255**
- the /24 represents the number of bits which is the network prefix
- Each subnet has a contiguous private RFC1918 IP space

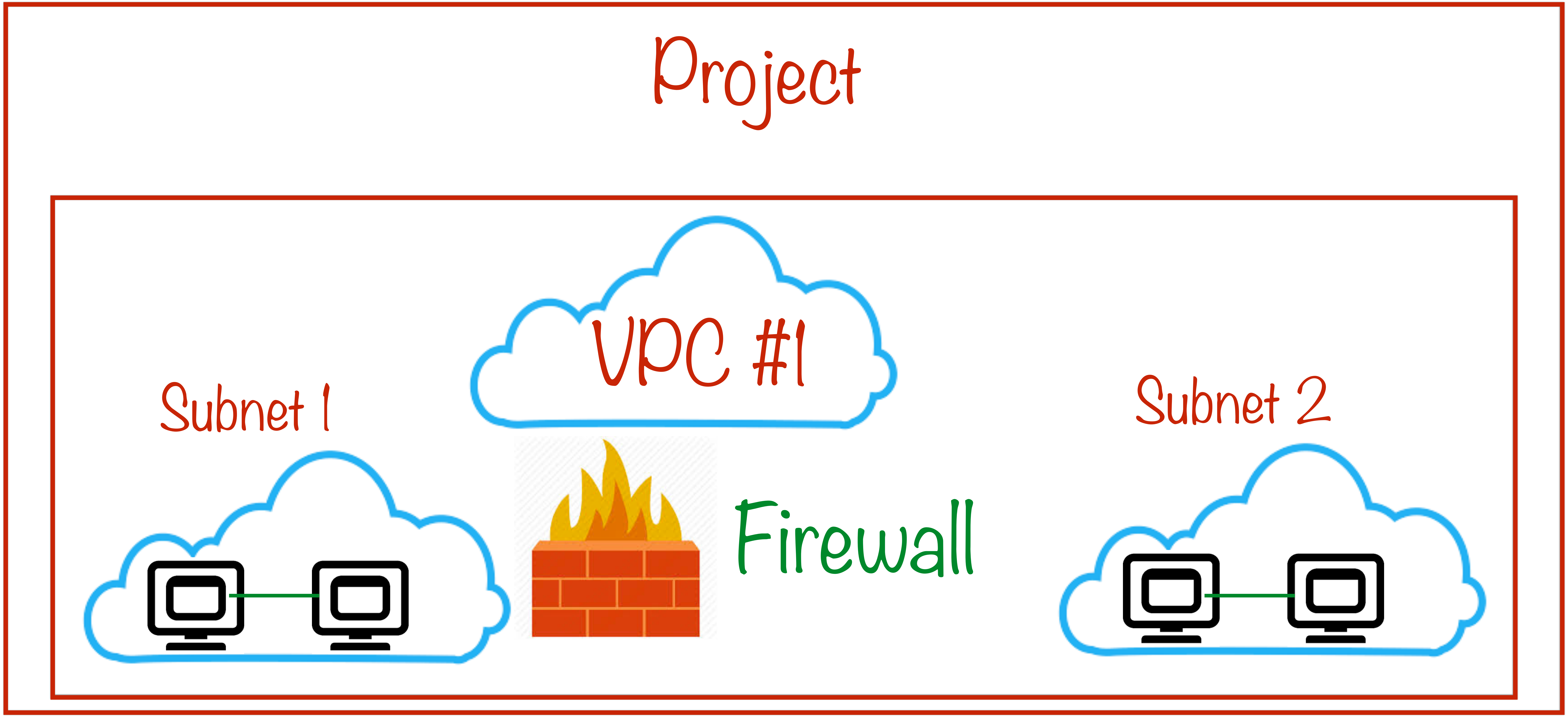
Projects and VPCs



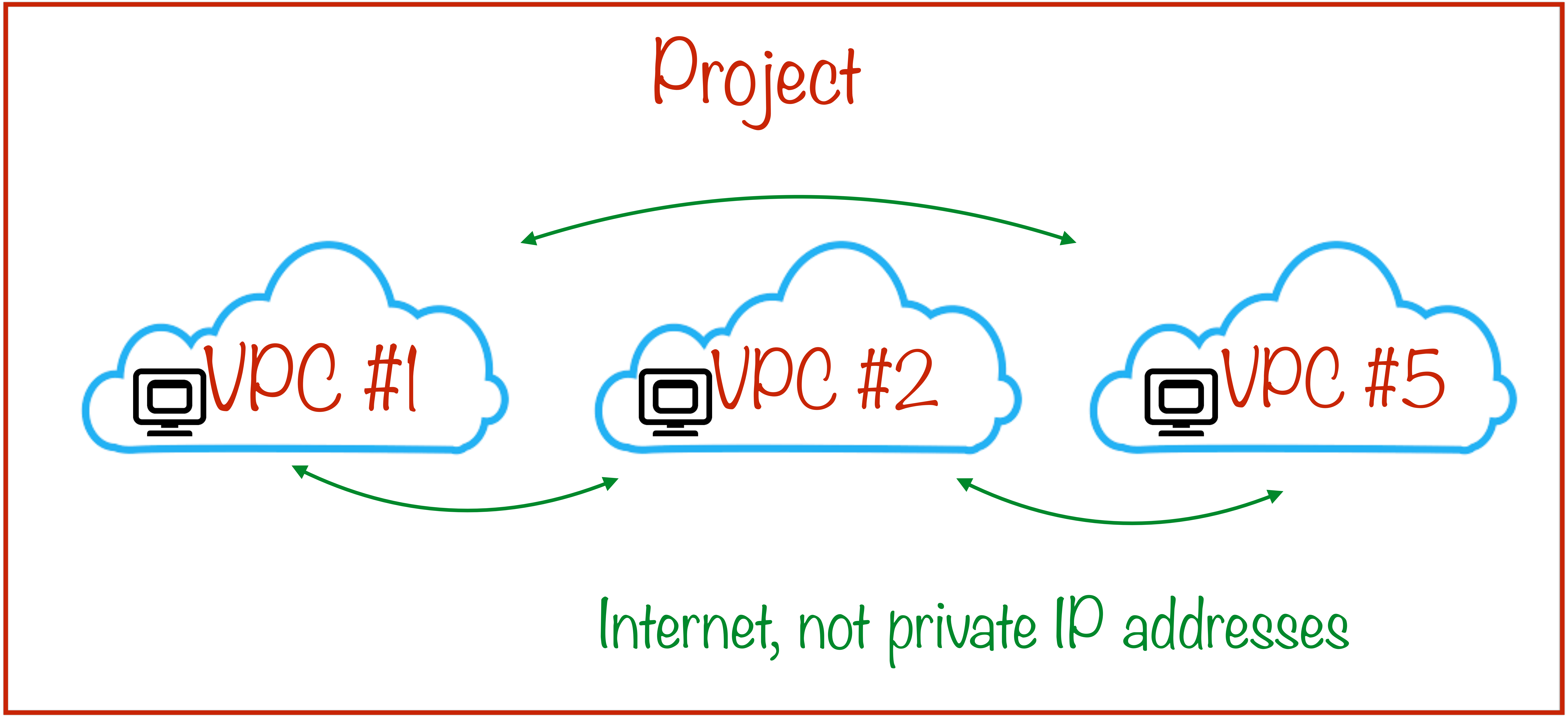
Projects and VPCs



Projects and VPCs



Projects and VPCs



Project

VPC #1

VPC #2

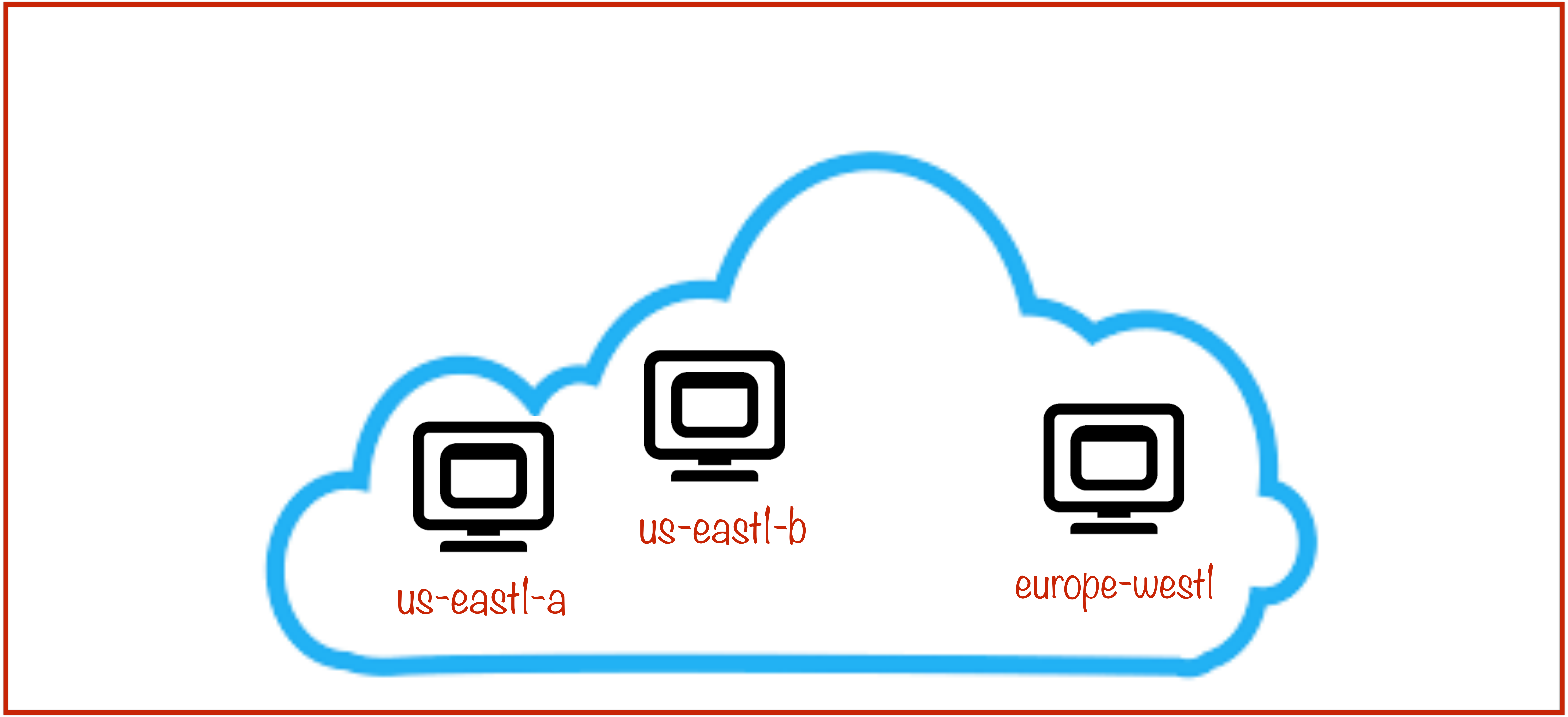
VPC #5

Internet, not private IP addresses

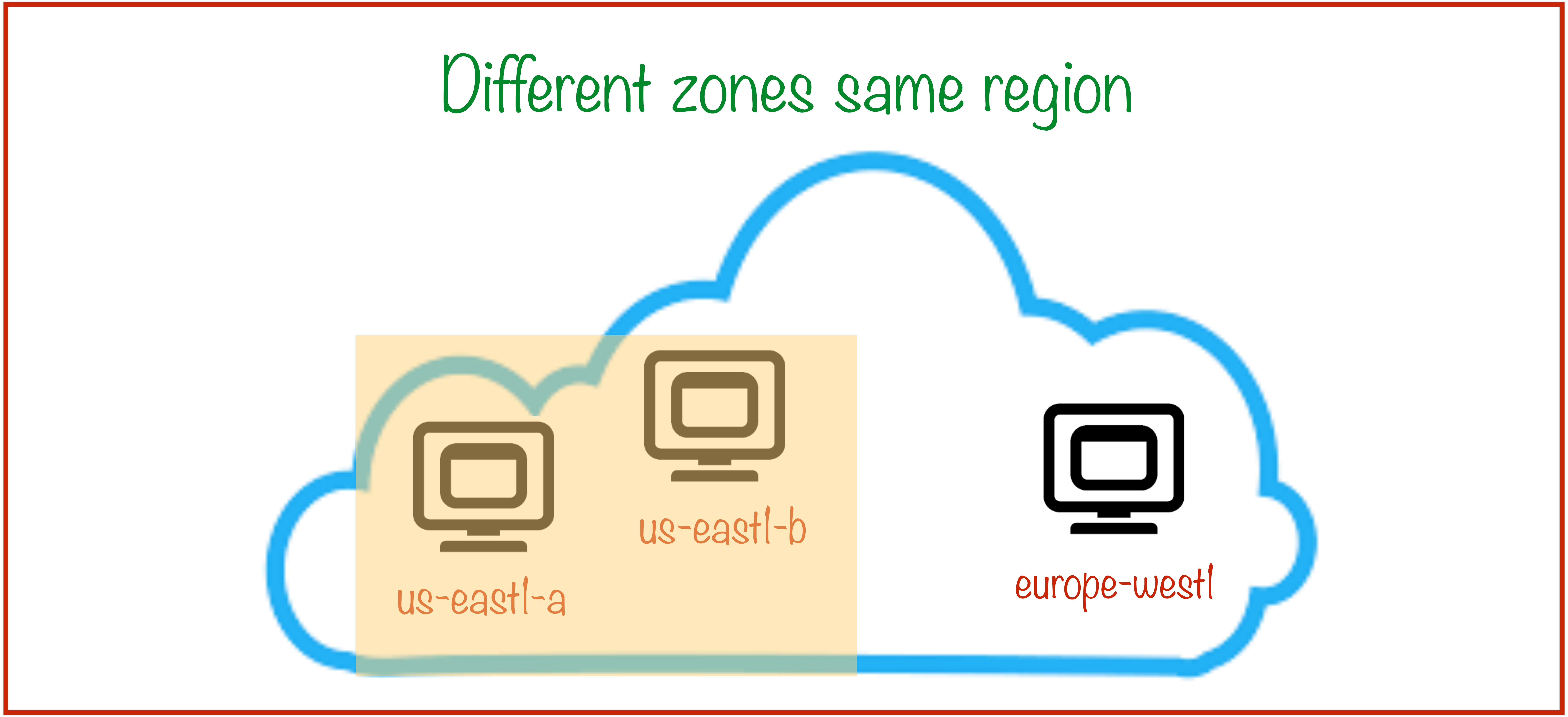
Within a network the resources communicate with each other often and are trusted

Resources in other networks are treated just like any other external resource (even if they are in the same project)

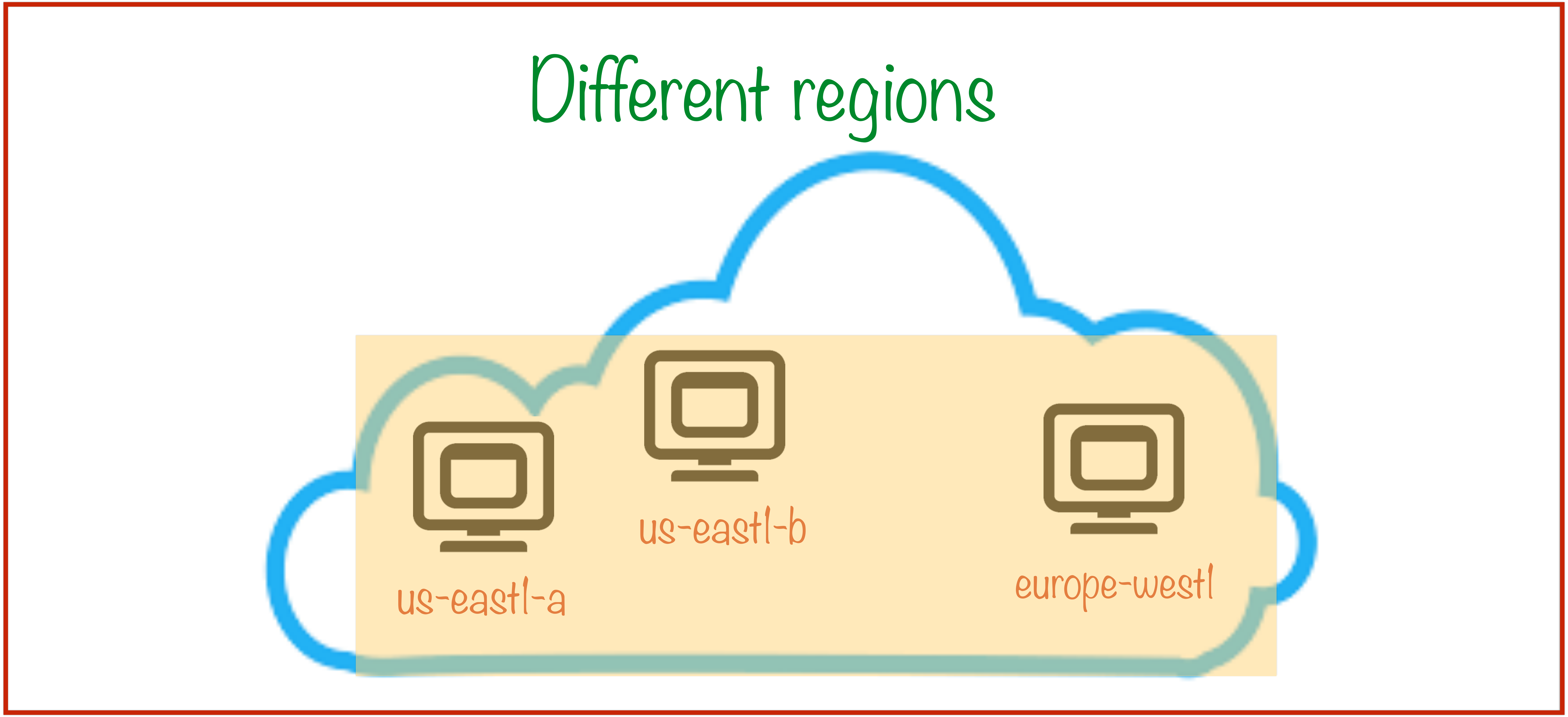
VPCs are Global



VPCs are Global

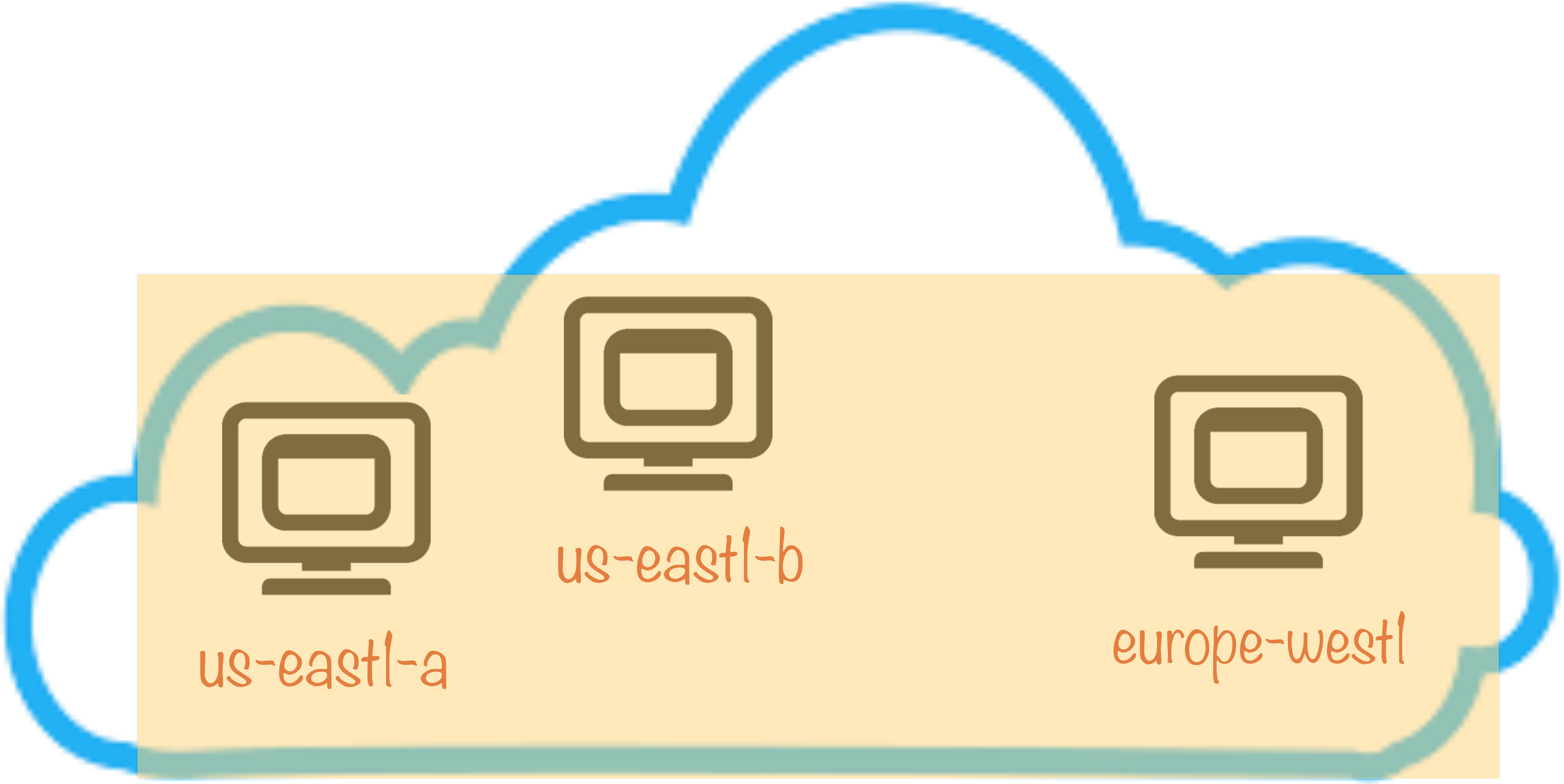


VPCs are Global



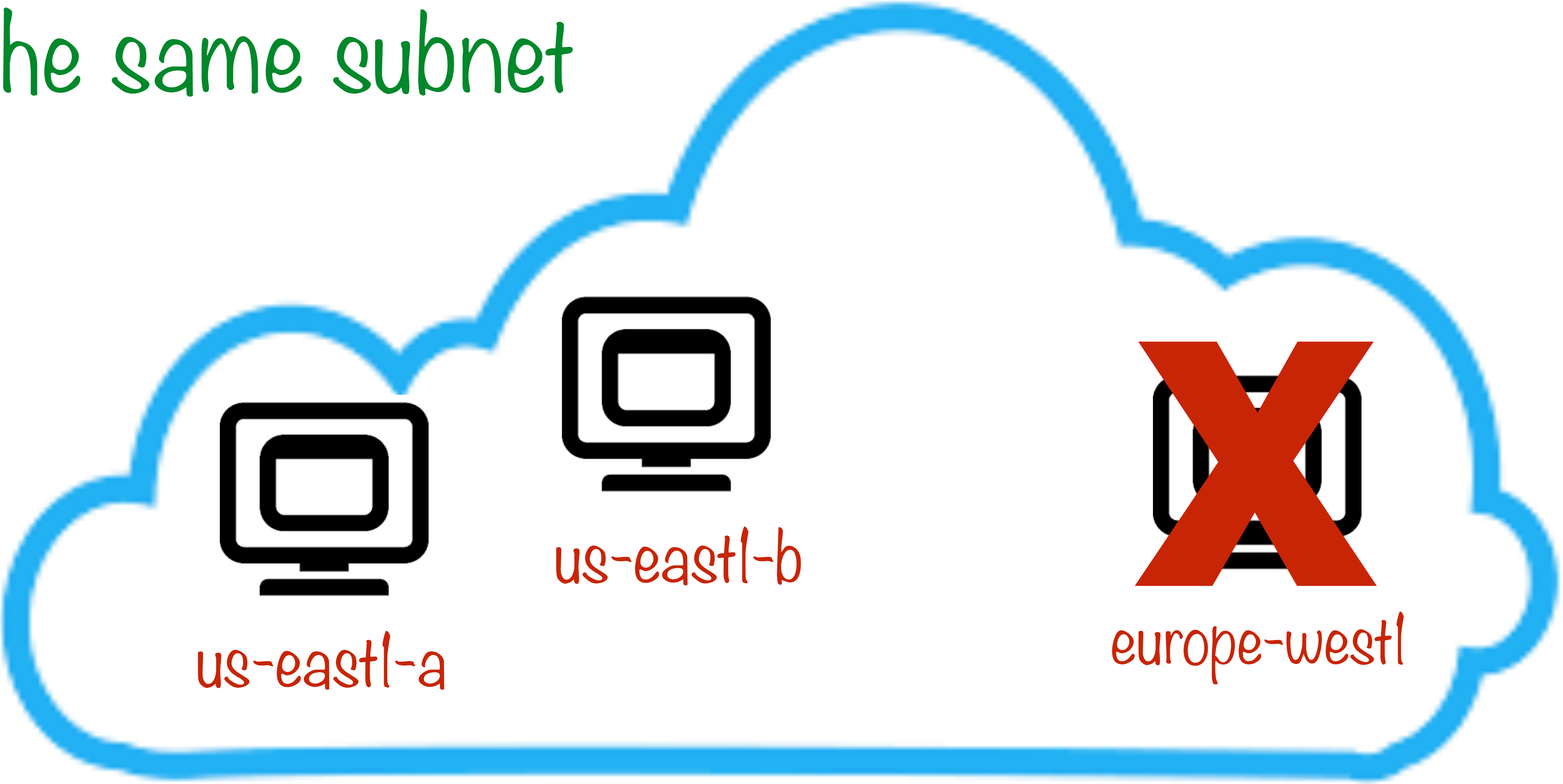
VPCs are Global

All machines communicate using internal IP addresses



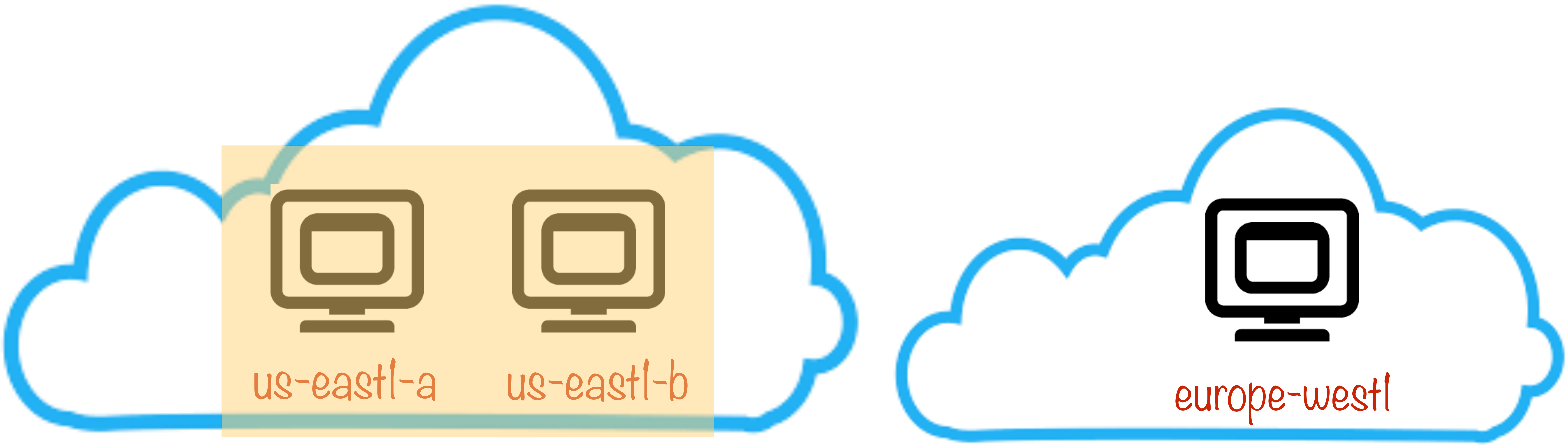
Subnets are Regional

Instances from different regions cannot be on the same subnet



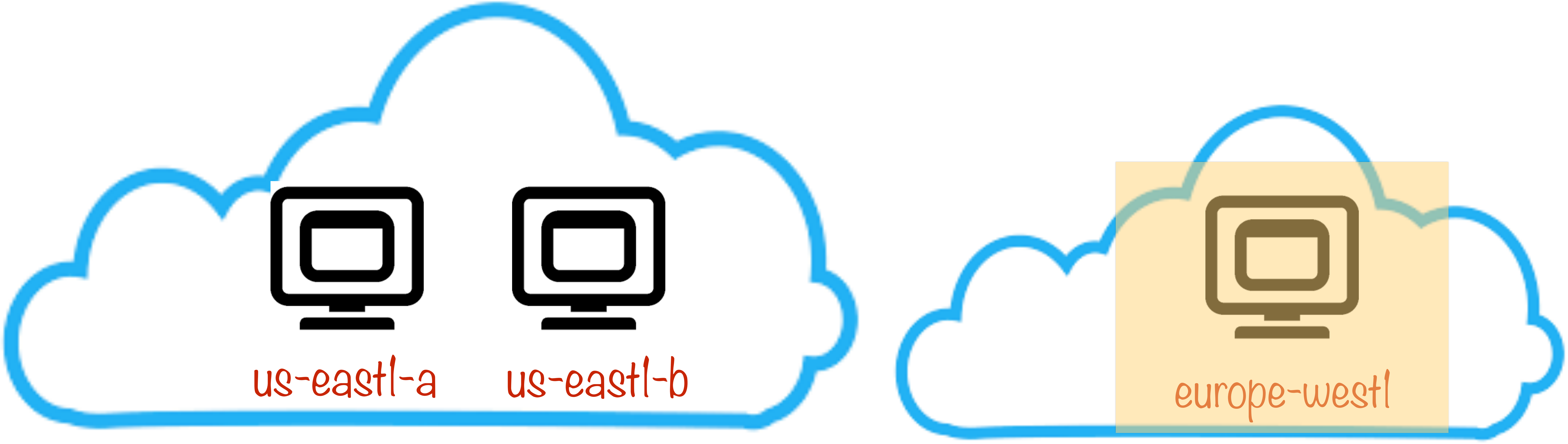
Subnets are Regional

Subnets can have resources from multiple zones



Subnets are Regional

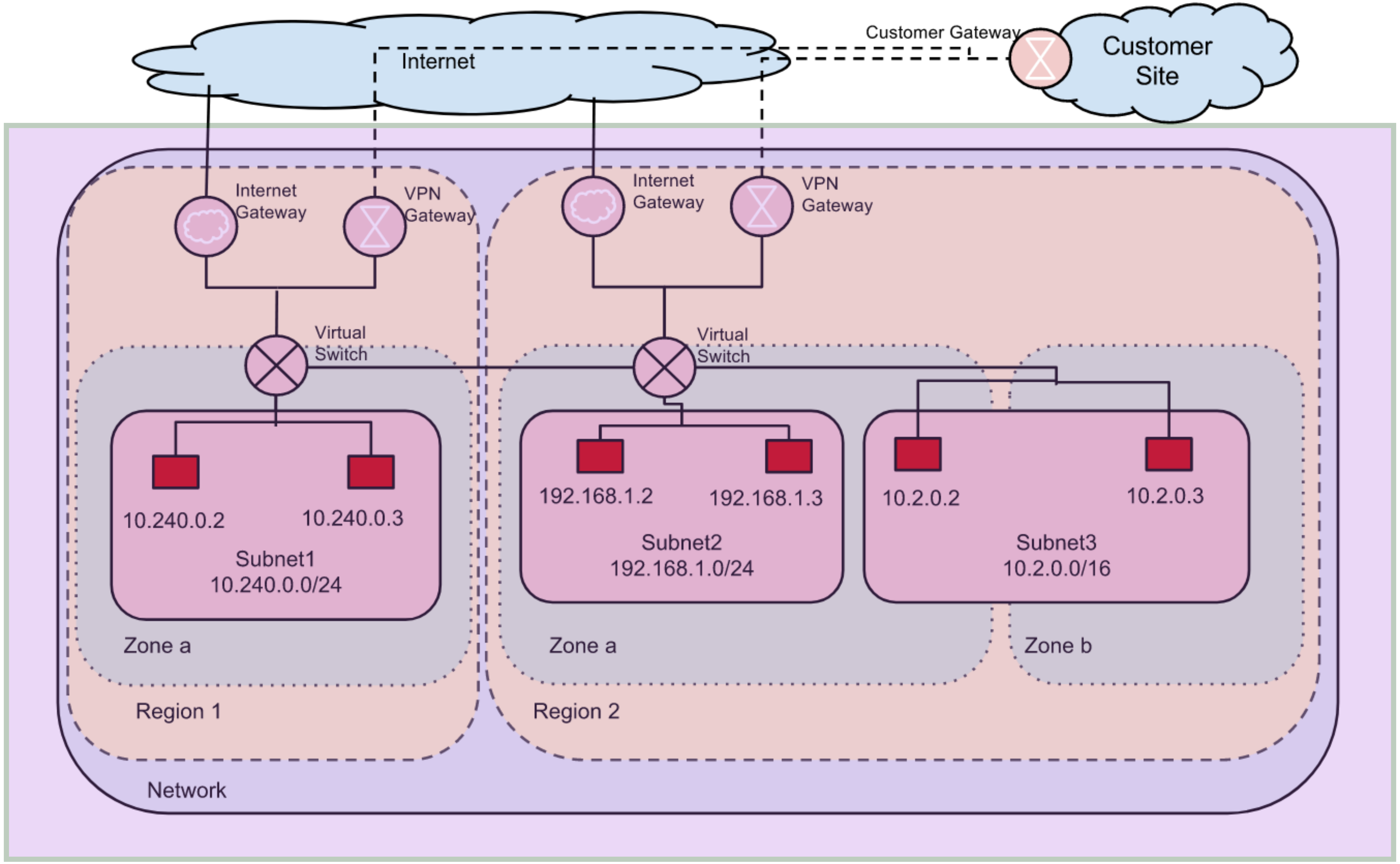
Or from a single zone



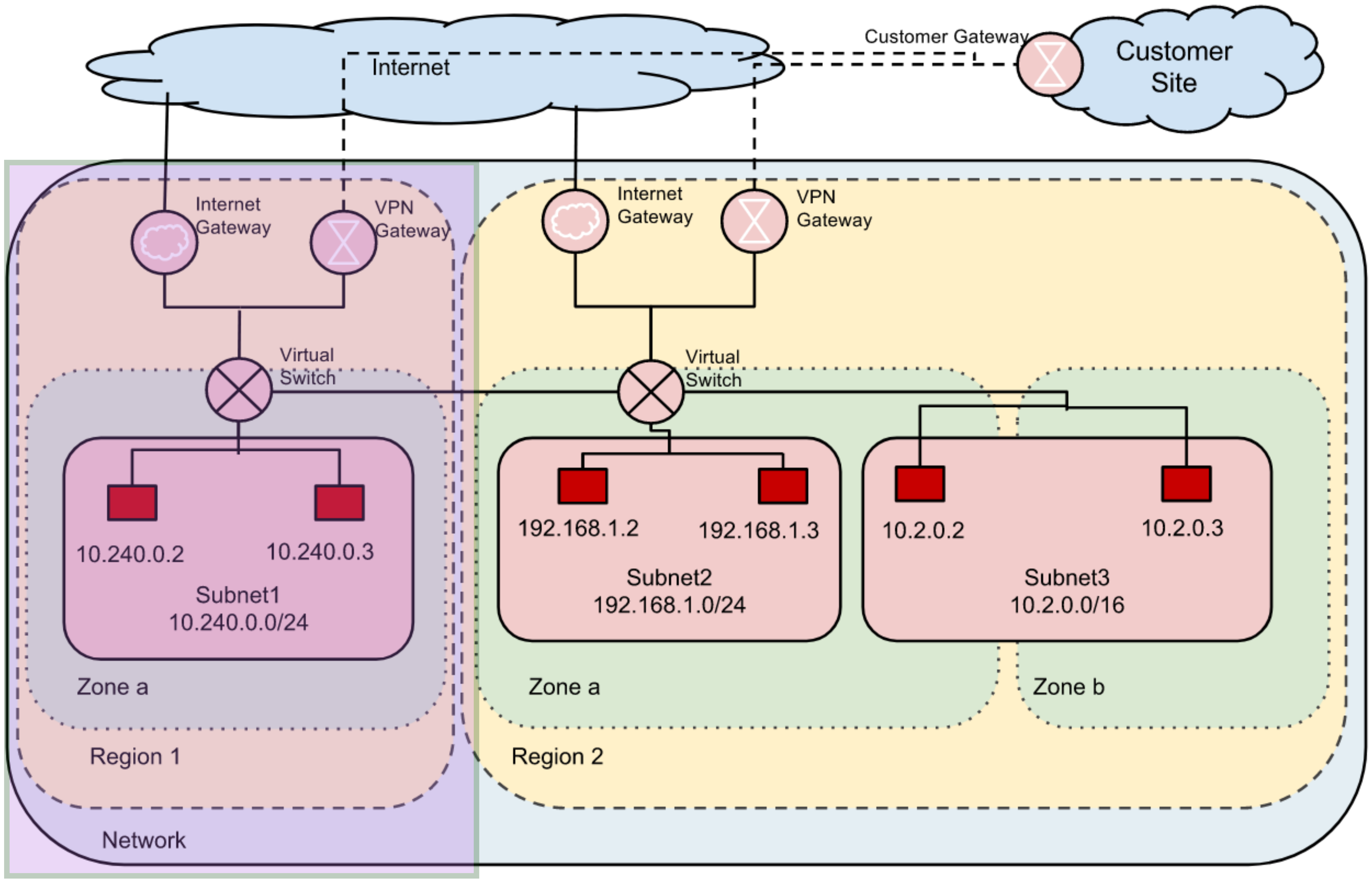
Networks are global - instances can be in different regions/zones

Subnets are regional - instances can be different zones

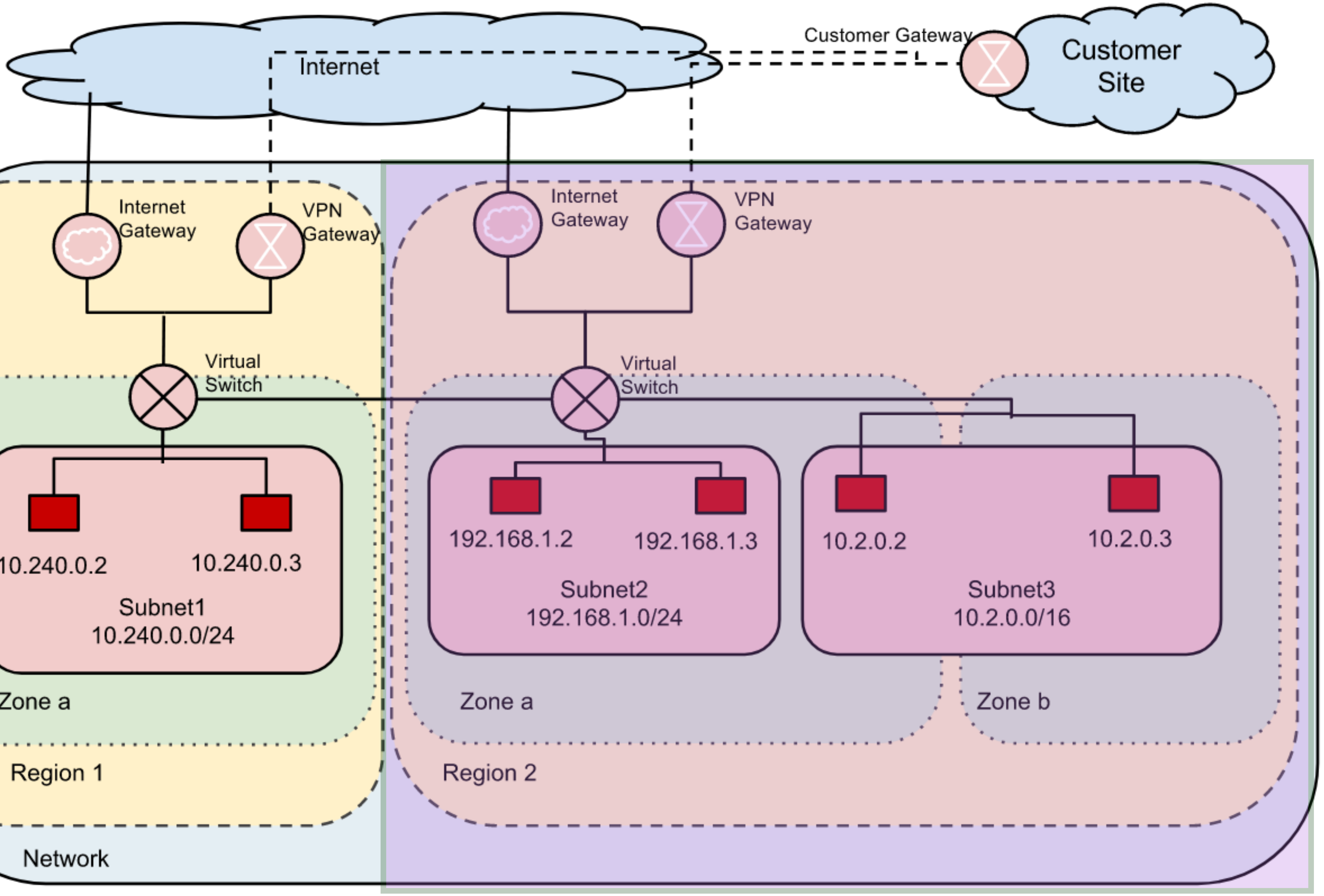
Example of a Network



Example of a Network

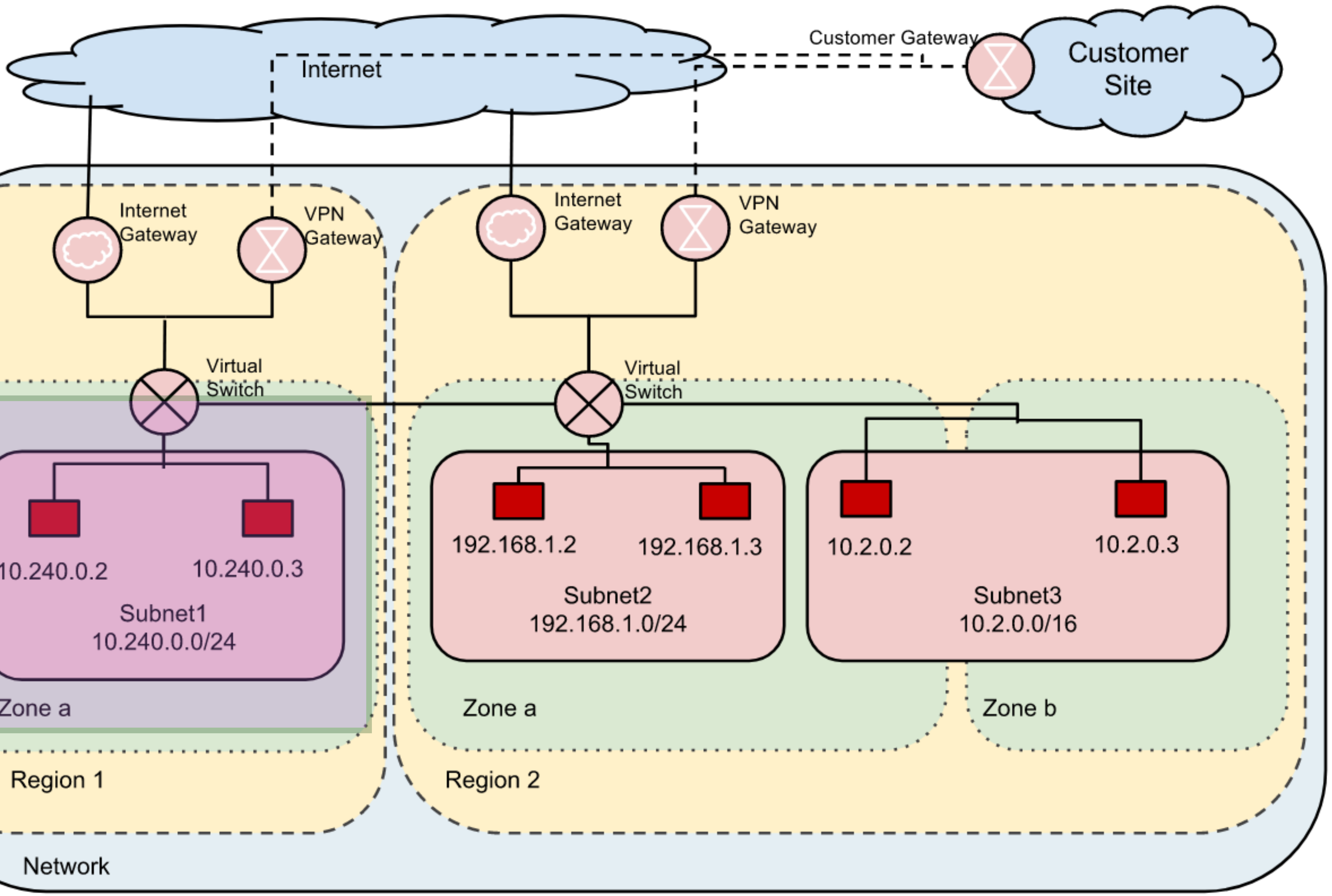


Example of a Network

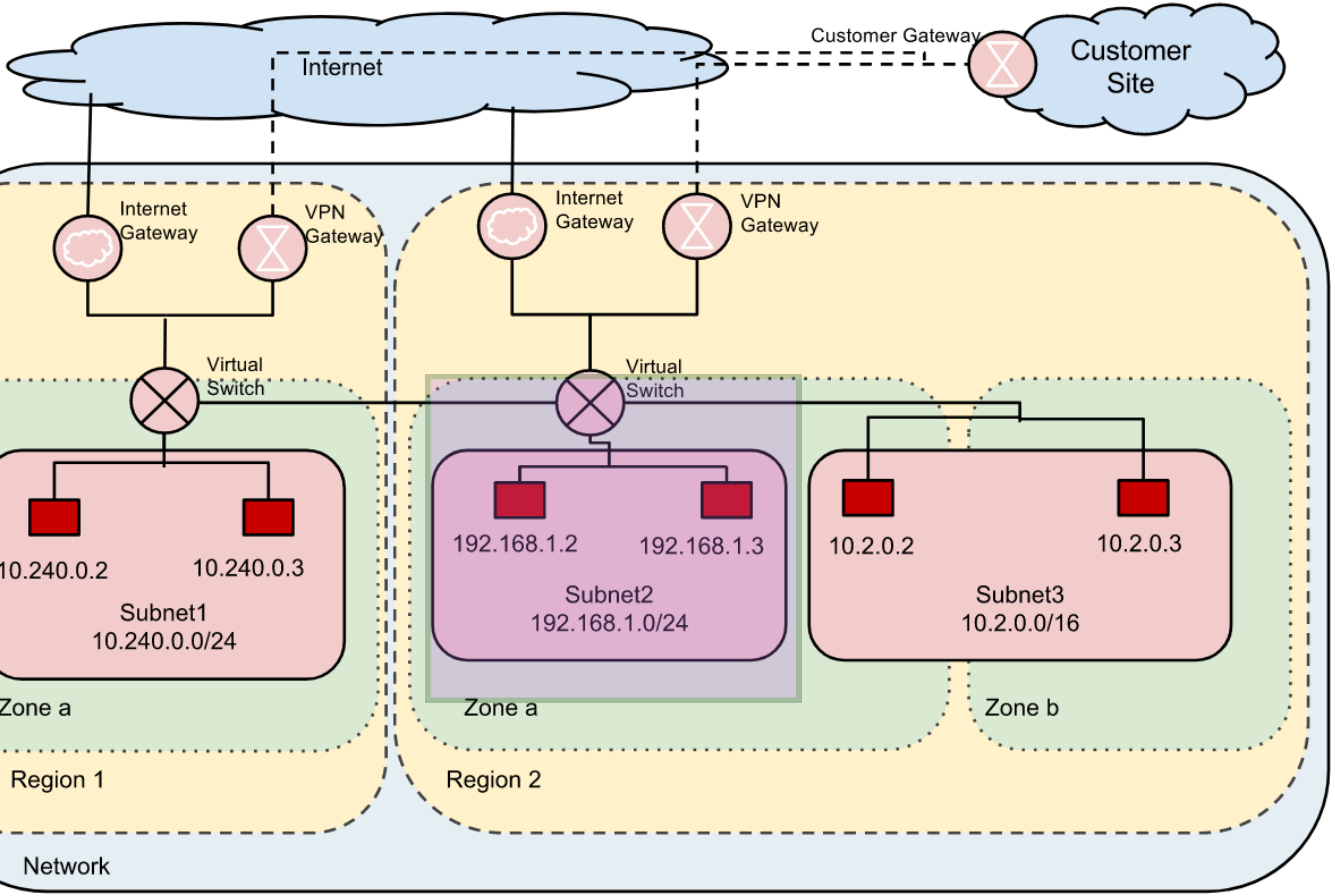


Example of a Network

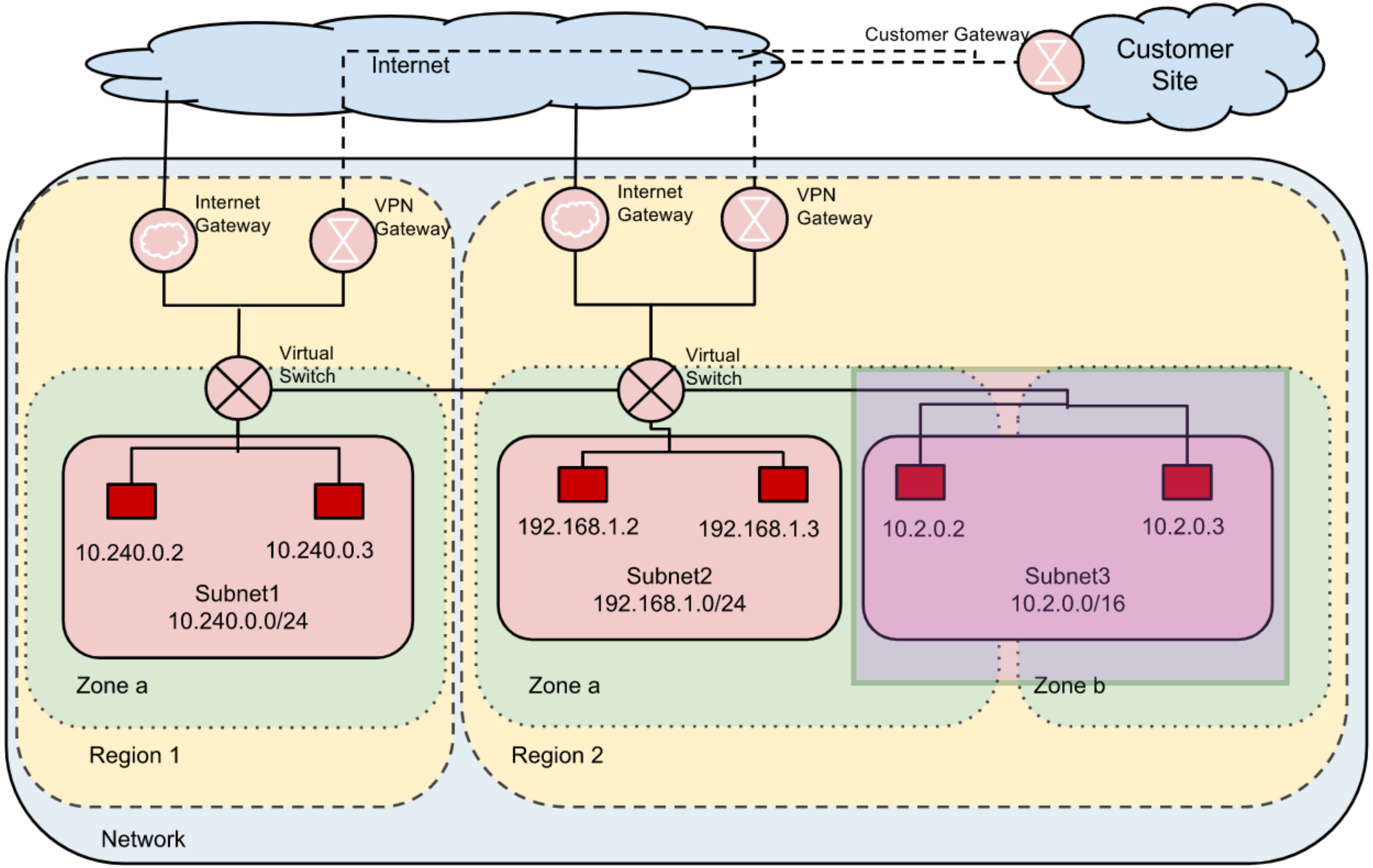
Notice the IP address range of the subnets



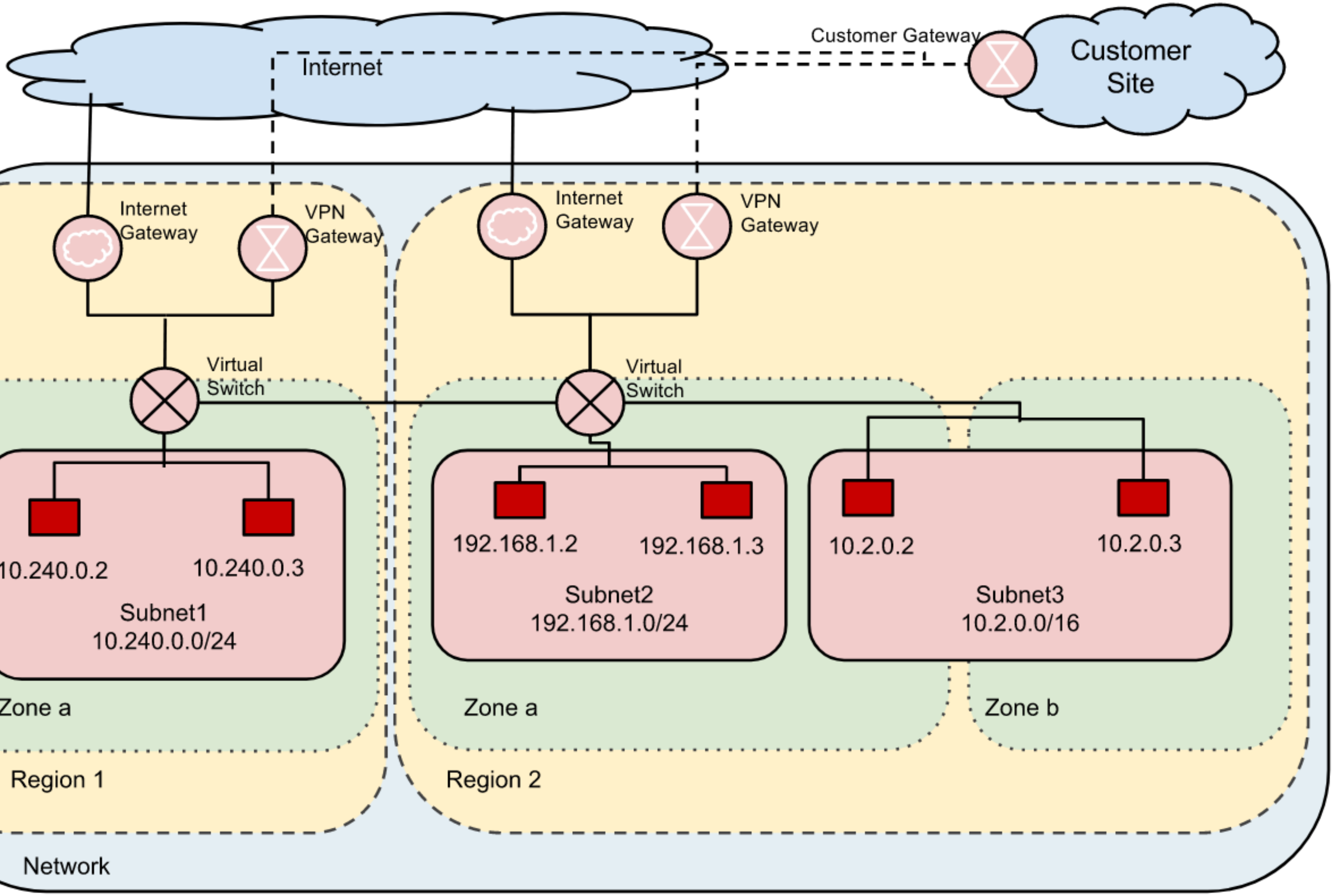
Example of a Network



Example of a Network



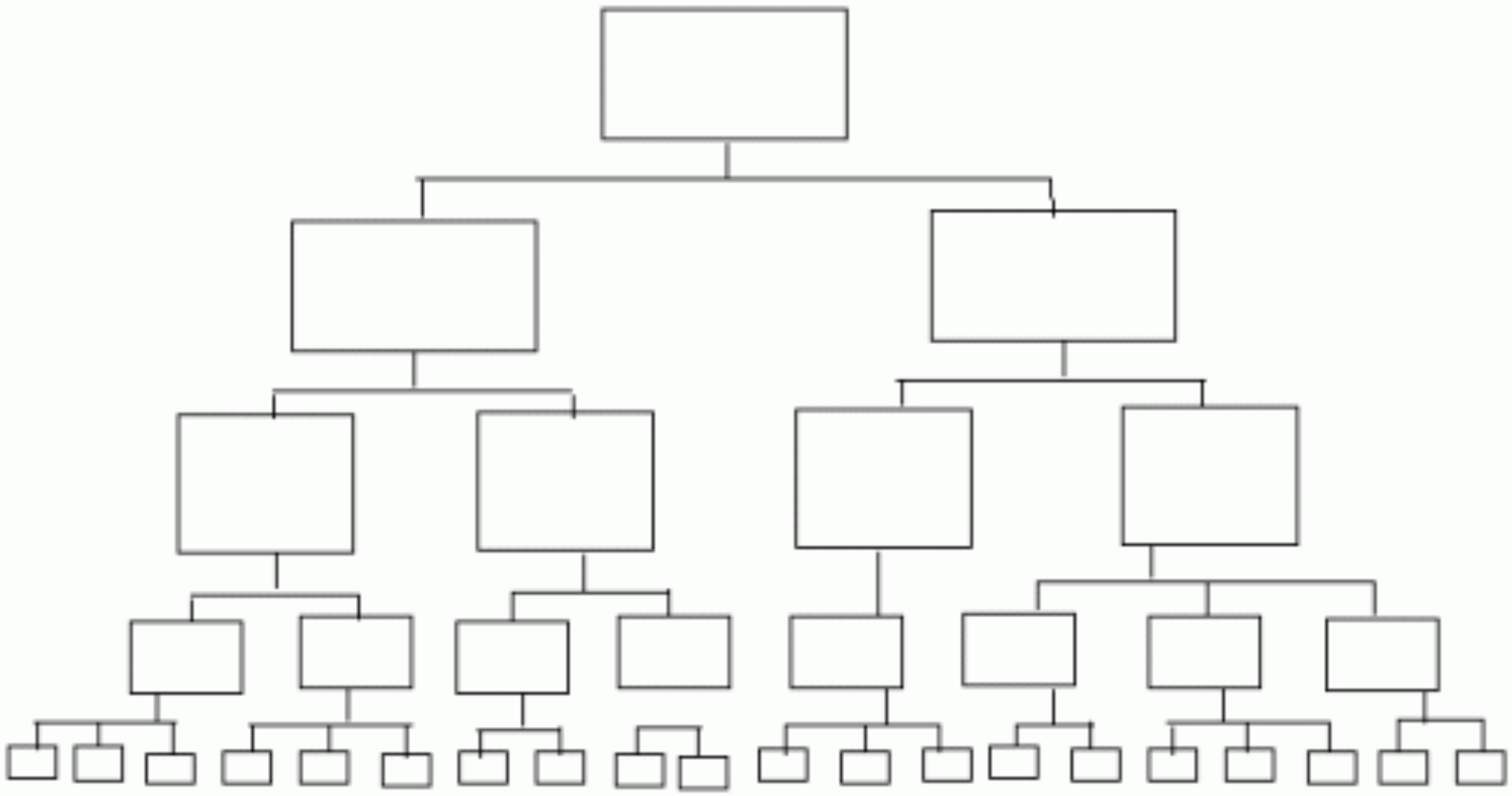
Example of a Network



Traditional networks had a range of IP addresses assigned to it

Each subnet comprised of a smaller range if IP addresses from the network range

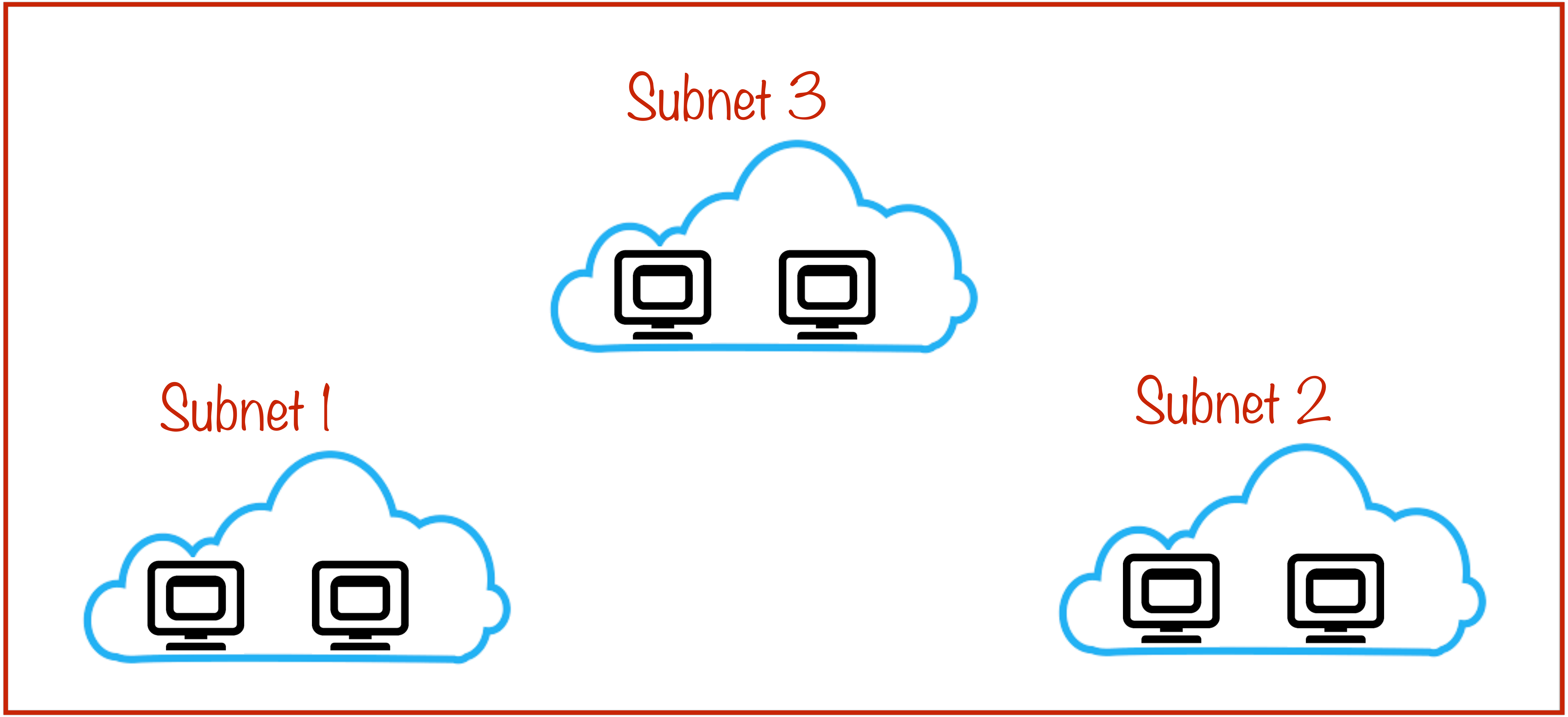
Traditional Networks



GCP networks are a collection of subnets which have their own IP ranges

Subnet IP ranges do not have to fit into the network's larger IP range

Subnetworks in GCP



Types of VPC Networks

Auto Mode

Automatically sets up a single subnet in each region - can manually create more subnets

Custom Mode

No subnets are set up by default, we have to manually configure all subnets

The "default" Network

Auto Mode

Automatically sets up a single subnet in each region - can manually create more subnets

- Every GCP project has an auto-mode network set up by default
- It comes with a number of routes and firewall rules preconfigured
- Gets us up and running without thinking about networks

IP Addresses

IP Addresses

- Can be assigned to resources e.g. VMs
- Each VM has an internal IP address
- One or more secondary IP addresses
- Can also have an external IP address

Internal IP Addresses

- Use within a VPC
- Cannot be used across VPCs unless we have special configuration (like shared VPCs or VPNs)
- Can be ephemeral or static, typically ephemeral
- VMs know their internal IP address (VM name and IP is available to the network DNS)

External IP Addresses

- Use to communicate across VPCs
- Traffic using external IP addresses can cause additional billing charges
- Can be ephemeral or static
- VMs are not aware of their external IP address

Internal vs External

Internal	External
Ephemeral, changes every 24 hours or on VM restarts	Can be ephemeral or static
Allocated from the range of IP addresses available to a subnet to which the resource belongs	Ephemeral: Allocated from a pool of external IP addresses.
	Static: Reserved - charged when not assigned to VM
VMs know their internal IP	VMs unaware of external IP

Internal vs External

Internal

Hostname is mapped to internal IP

"instance-1.c.test-project123.internal"

VPC networks automatically resolve internal IP addresses to host names

External

Hosts with external IPs allow connections from outside the VPC

Need to publish public DNS records to point to the instance with the external IP

Can use Cloud DNS

Ephemeral vs Static

Ephemeral

Available only till the VM is stopped, restarted or terminated

No distinction between regional and global IP addresses

Static

Permanently assigned to a project and available till explicitly detached

Regional or global resources

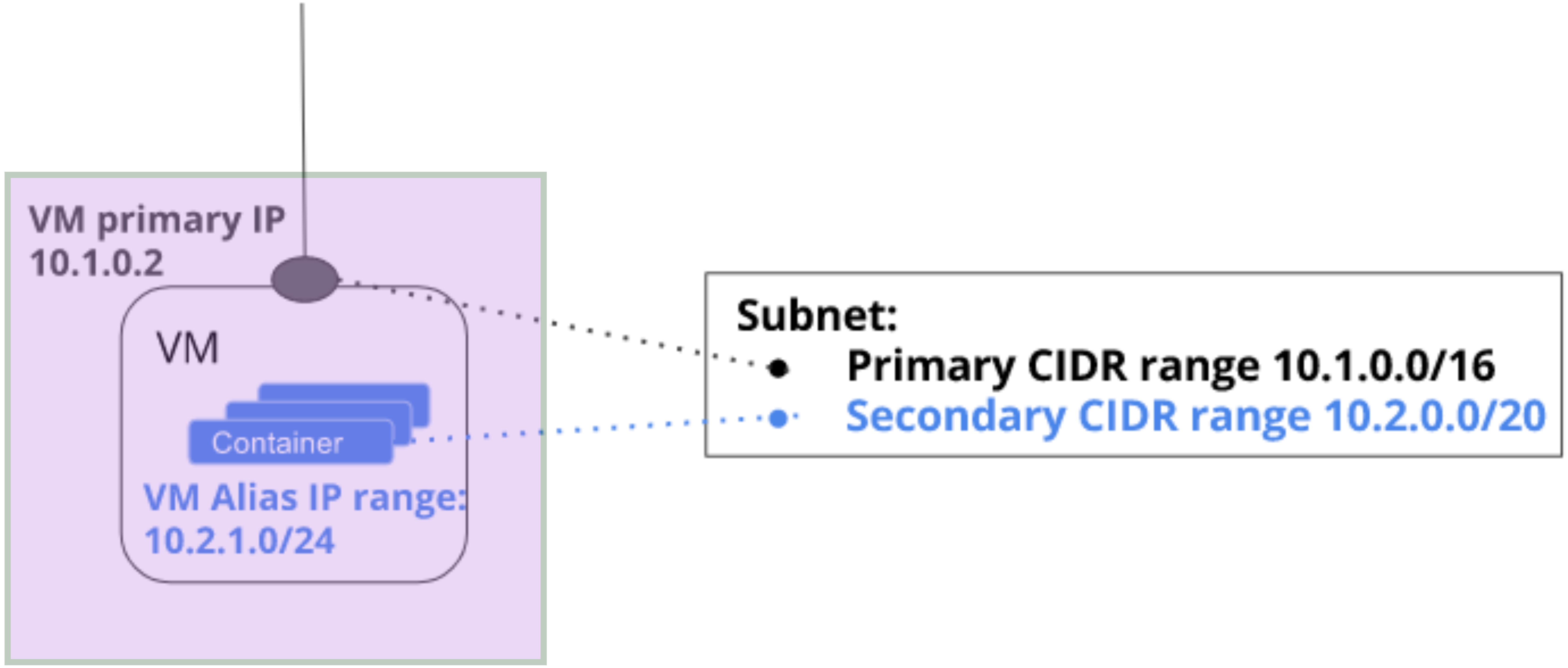
- Regional: Allows resource of the region to use the address
- Global: Used only for global forwarding rules in global load balancing

Unassigned static IPs incur a cost

Alias IP Ranges

- A single service on a VM requires just one IP address
- Multiple services on the same VM may need different IP addresses
- Subnets have a primary and secondary CIDR range
- Using IP aliasing can set up multiple IP addresses drawn from the primary or secondary CIDR ranges

Alias IP Ranges

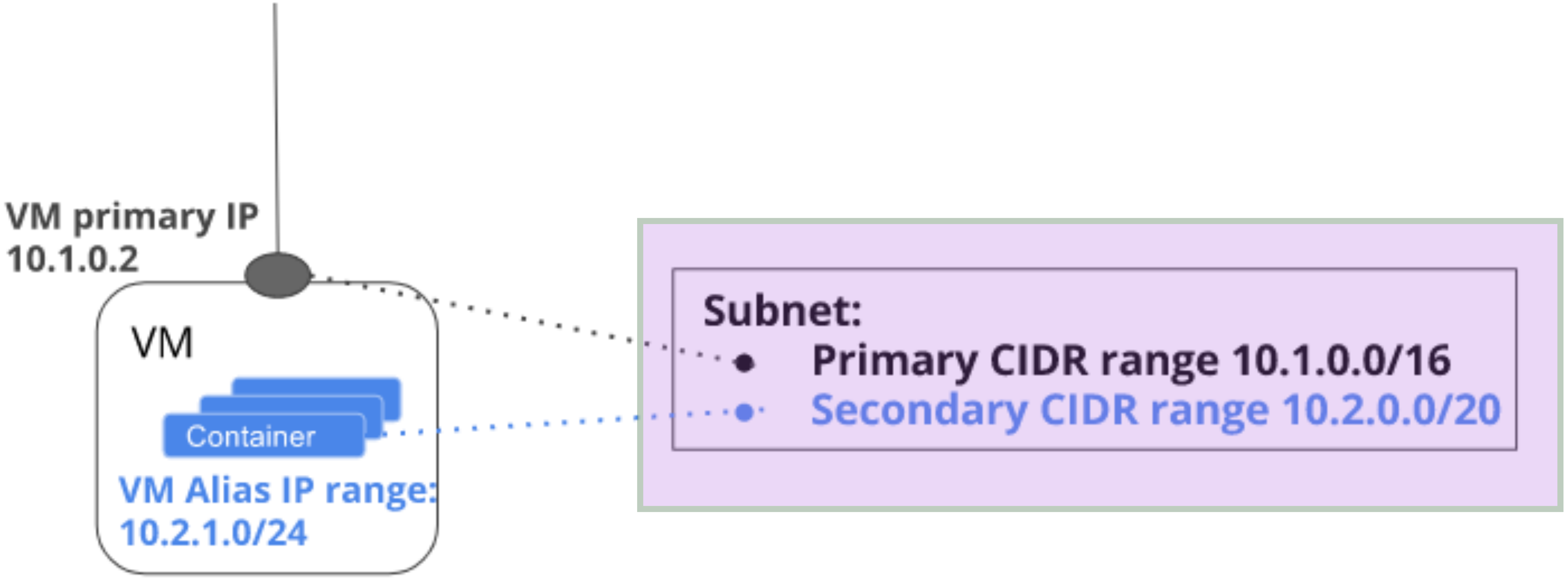


VM primary IP
10.1.0.2

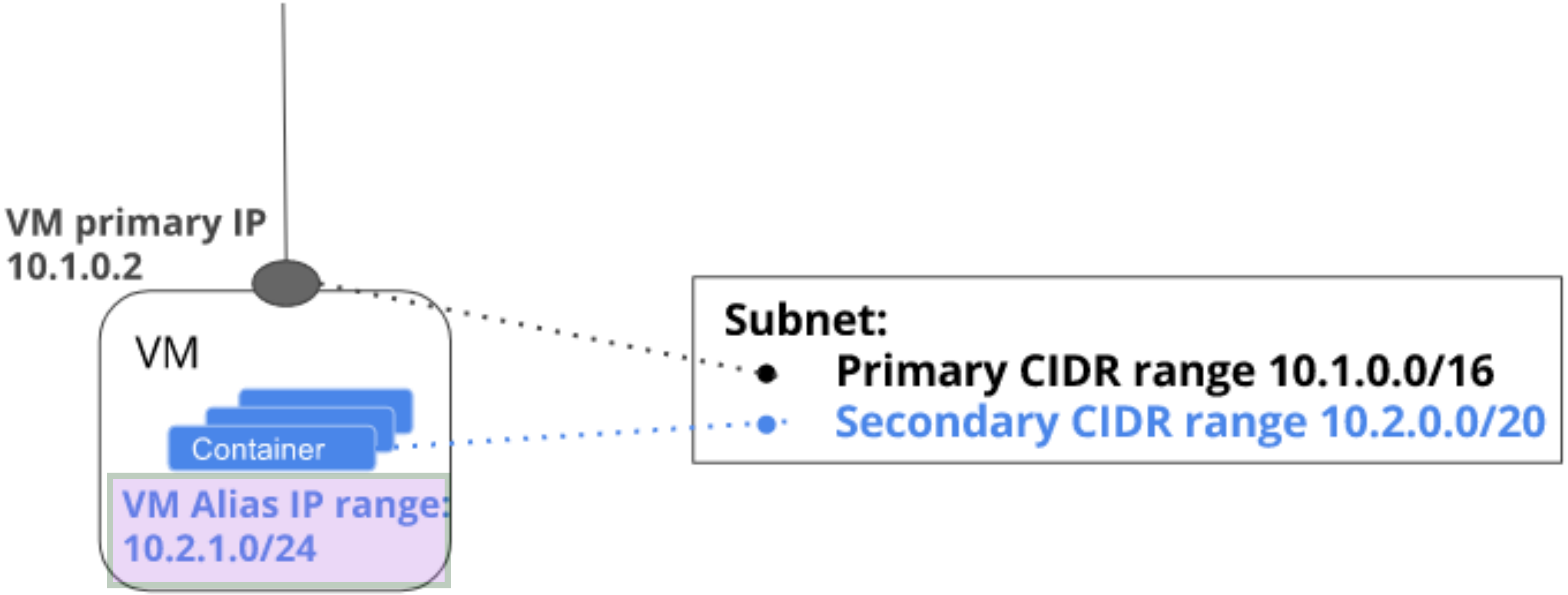
Subnet:

- Primary CIDR range 10.1.0.0/16
- Secondary CIDR range 10.2.0.0/20

Alias IP Ranges



Alias IP Ranges



VM primary IP
10.1.0.2

VM

Container

VM Alias IP range:
10.2.1.0/24

Subnet:

- ● **Primary CIDR range 10.1.0.0/16**
- ● **Secondary CIDR range 10.2.0.0/20**

Alias IP Ranges

- Multiple containers or services on a VM can have their own IP
- VPCs automatically set up routes for the IPs
- Containers don't need to do their own routing, simplifies traffic management
- Can separate infrastructure from containers (infra will draw from the primary range, containers from the secondary range)

Routes

Routes

A route is a mapping of an IP range to a destination. Routes tell the VPC network where to send packets destined for a particular IP address.

https://cloud.google.com/vpc/docs/routes

Routes

Where to send packets destined for an IP?

The answer lies in a route

Eg all internet-bound packets to proxy server first

2 Default Routes for a Network

Direct packets to destinations to specific destinations which carry it to the outside world (uses external IP addresses)

Allow instances on a VPC to send packets directly to each other (uses internal IP addresses)

The existence of a route does not mean that a packet will get to the destination

Firewall rules have to be configured to allow the packet through

Creating a Network

Default route for internet traffic

One route for every subnet that is created

What is a route made of?

- **name:** User-friendly name
- **network:** The name of the network to which this route applies
- **destRange:** The destination IP range that this route applies to
- **instanceTags:** Instance tags that this route applies to, applies to all instances if empty
- **priority:** Used to break ties in case of multiple matches

What is a route made of?

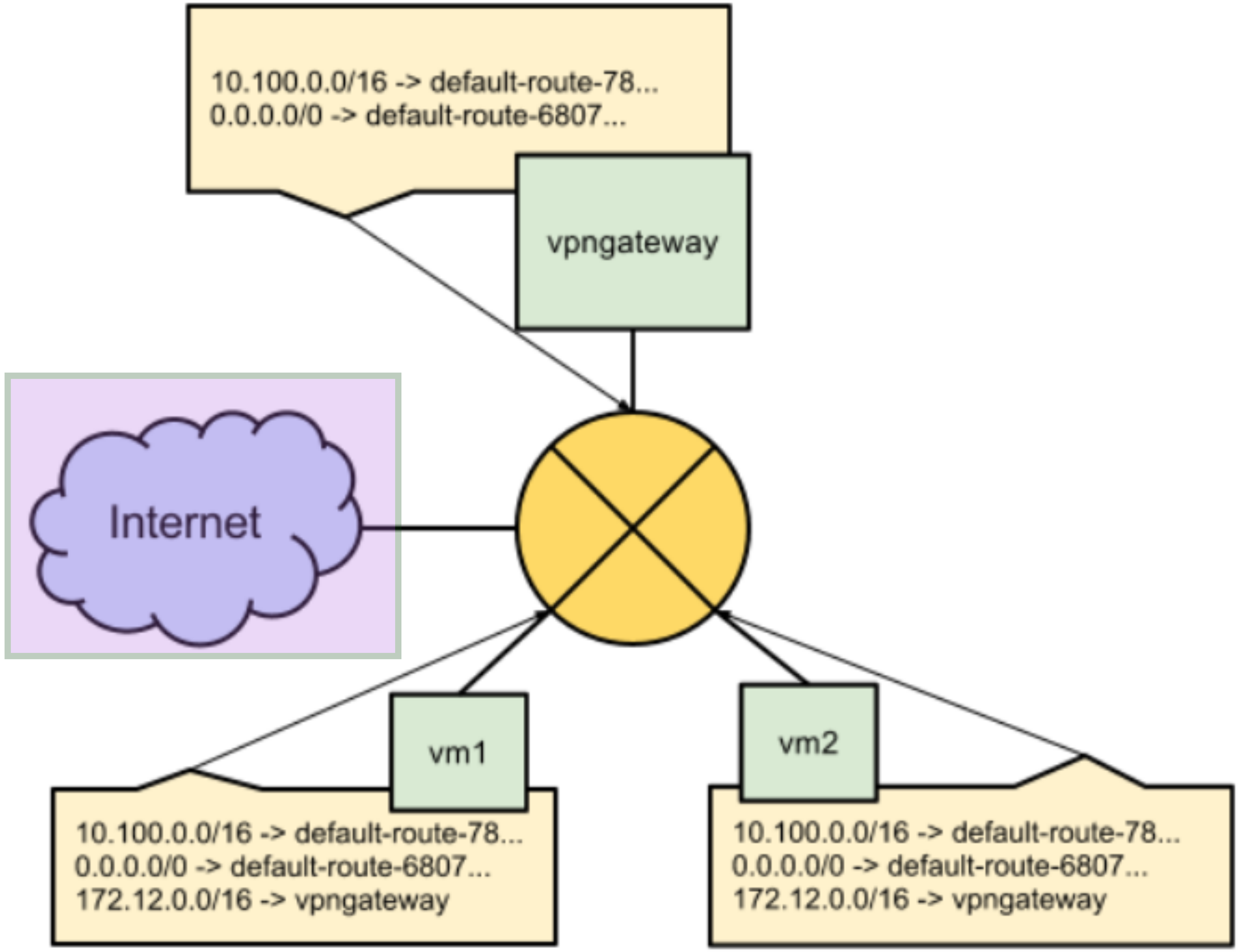
And one of

- **nextHopInstance:** Fully qualified URL. Instance must already exist
- **nextHopIp:** The IP address
- **nextHopNetwork:** URL of network
- **nextHopGateway:** URL of gateway
- **nextHopVpnTunnel:** URL of VPN tunnel

Instance Routing Tables

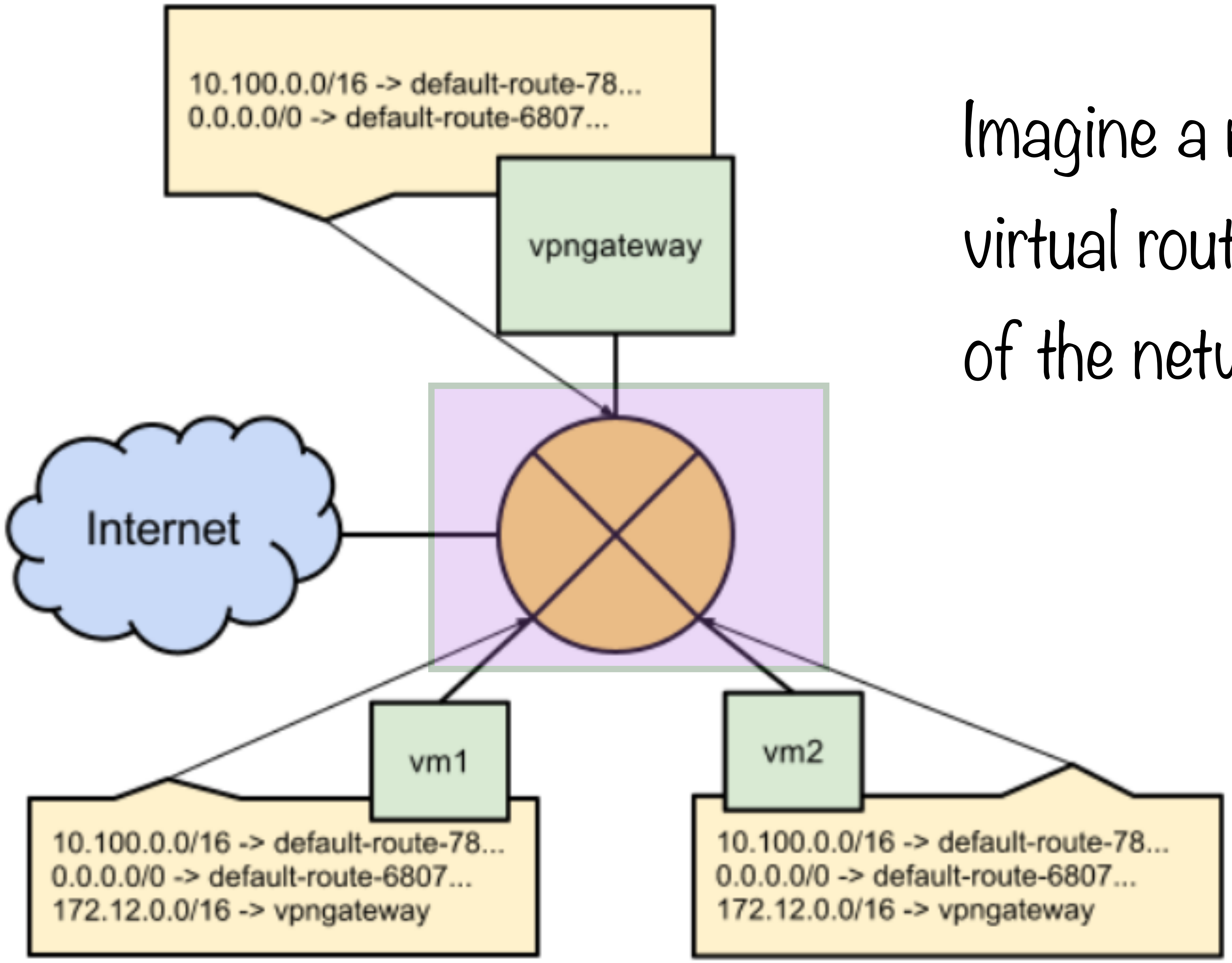
- Every route in a VPC might map to 0 or more instances
- Routes apply to an instance if the tag of the route and instance match
- If no tag, then route applies to all instances in a network
- All routes together form a routes collection

Routes and VMs



10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
vpngateway
Internet
vm1
vm2
10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway
10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

Routes and VMs



10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...

vpngateway

Internet

vm1

10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

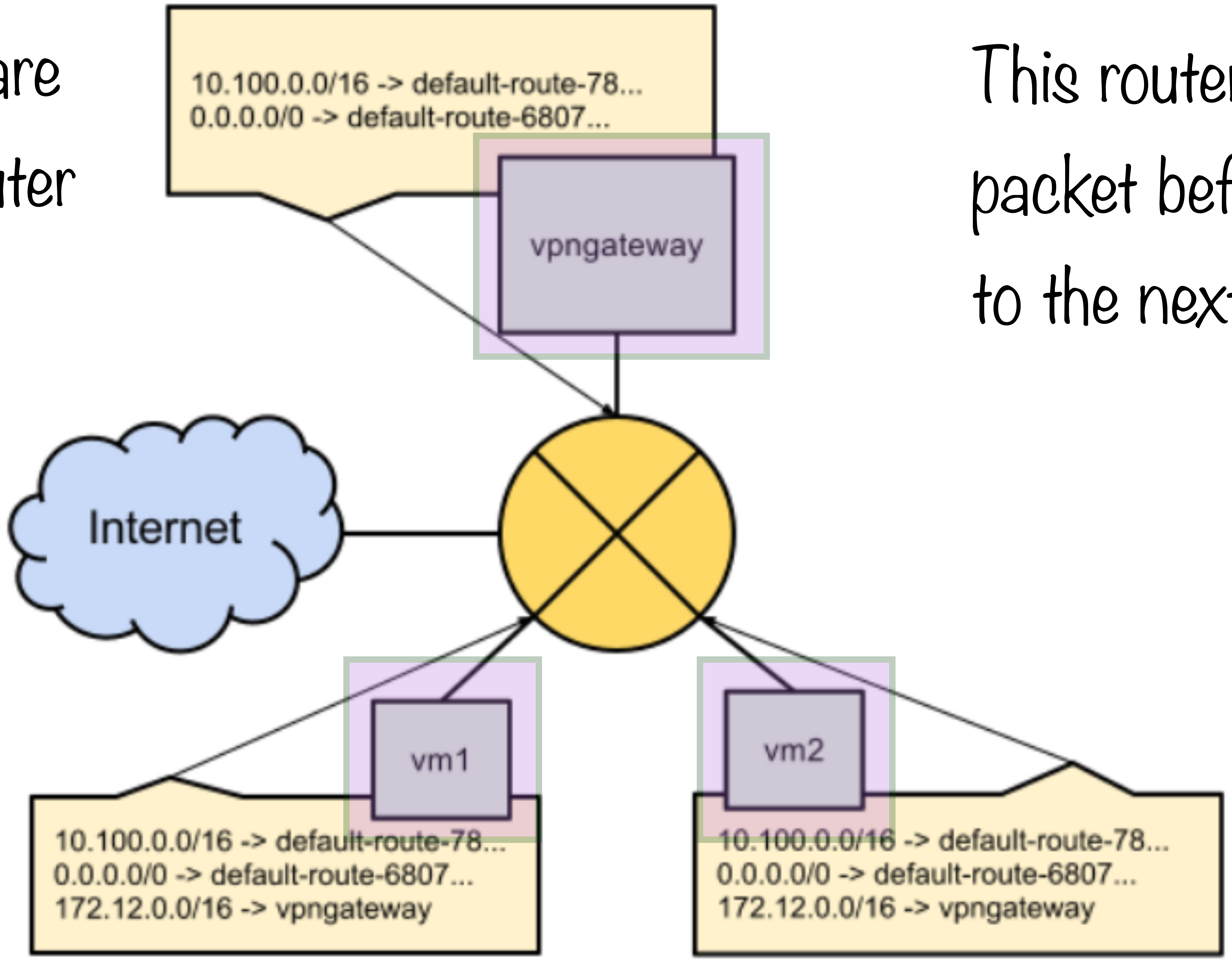
vm2

10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

Imagine a massively scalable virtual router at the center of the network

Routes and VMs

All virtual machines are connected to this router



10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...

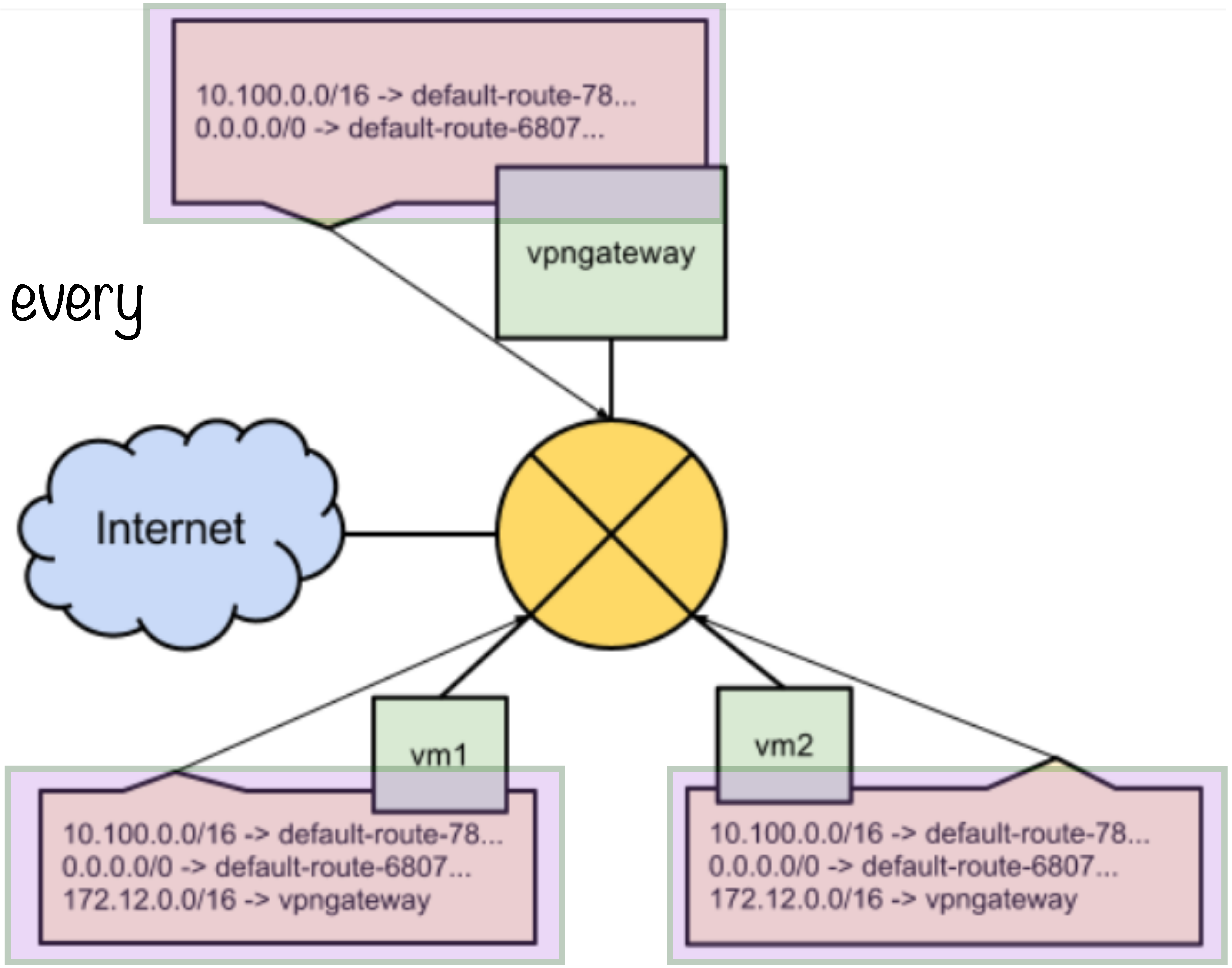
10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

This router handles every packet before it is passed on to the next hop

Routes and VMs

The routing table for every instance



10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...

10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

10.100.0.0/16 -> default-route-78...
0.0.0.0/0 -> default-route-6807...
172.12.0.0/16 -> vpngateway

Using Routes

- Many-to-one NATs
  - Multiple hosts mapped to one public IP
- Transparent proxies
  - Direct all external traffic to one machine

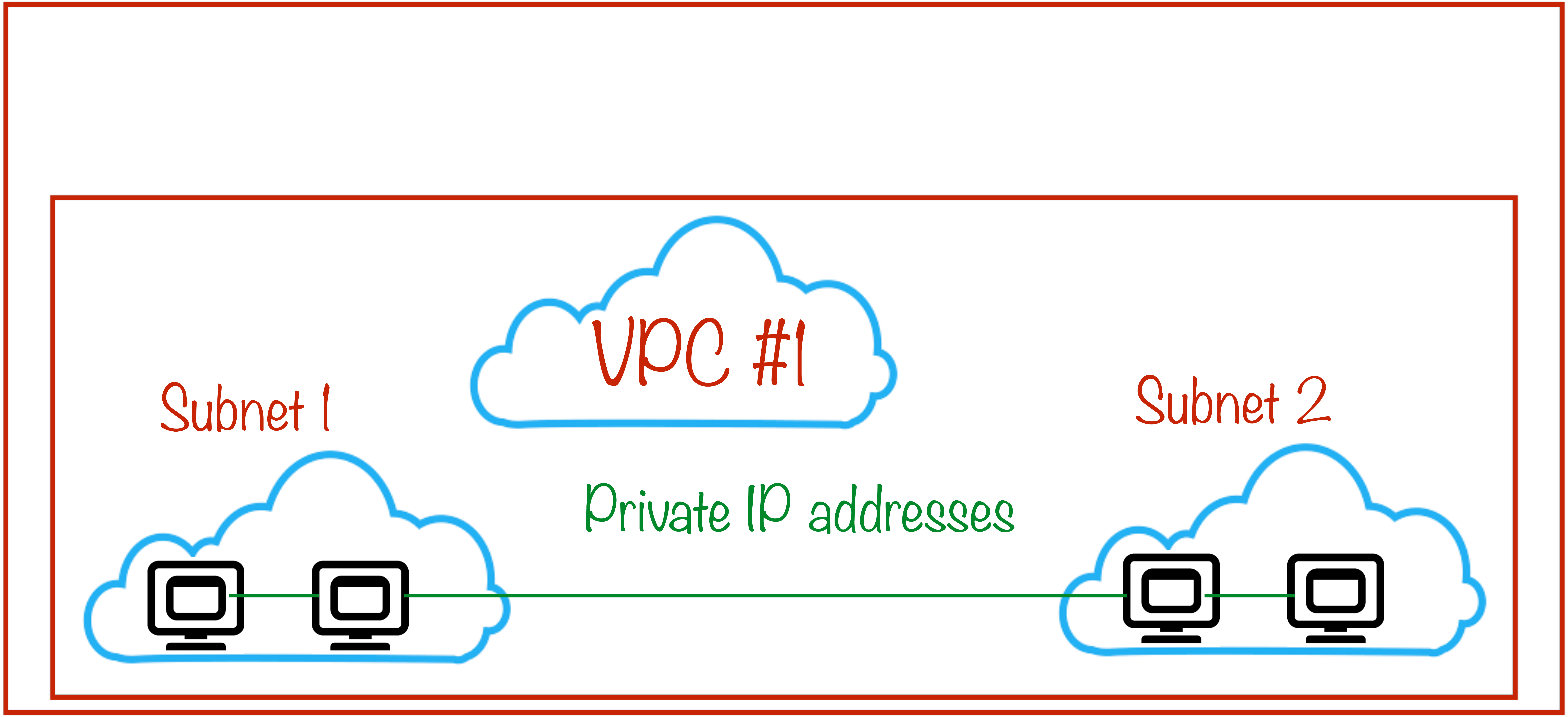
Firewall Rules

Firewall Rules

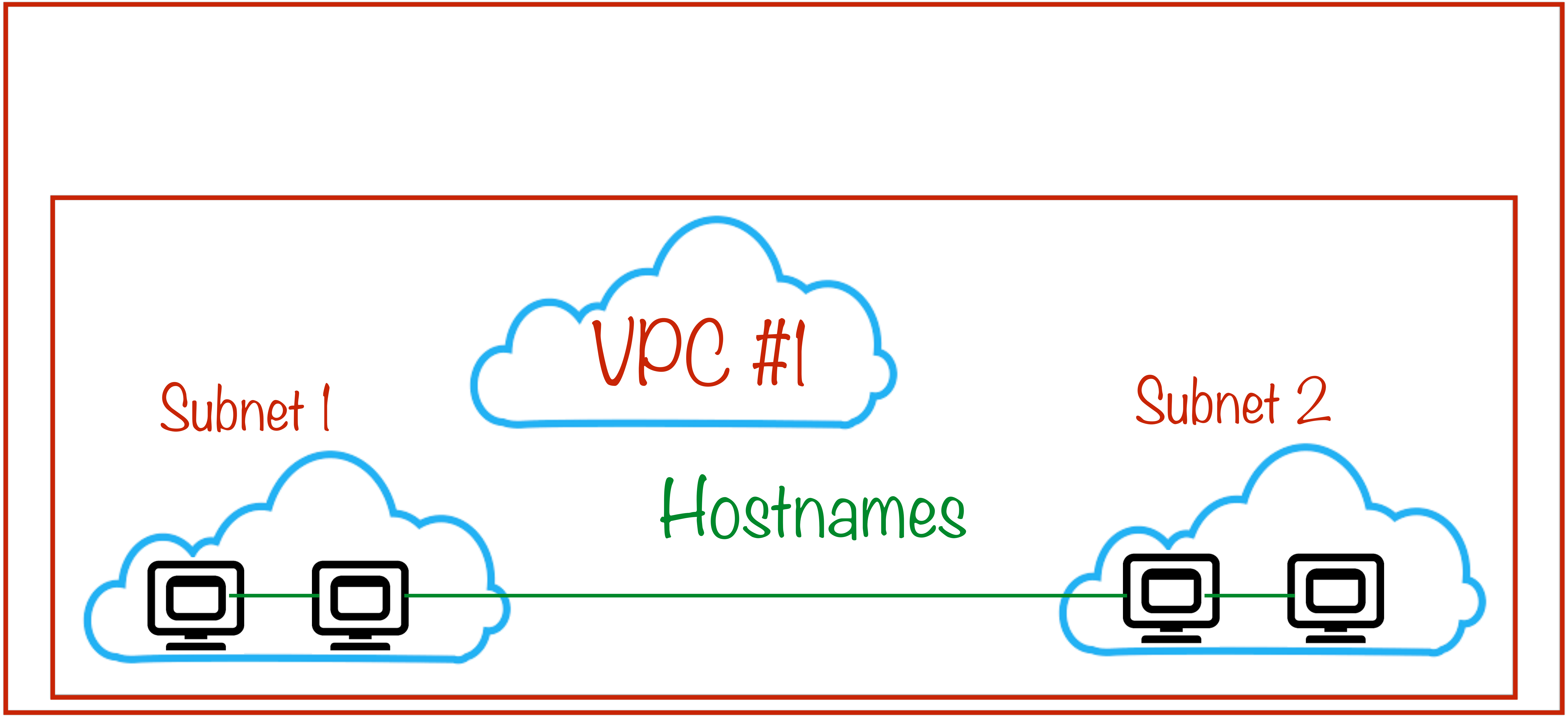
Protects your virtual machine (VM) instances from unapproved connections, both inbound (ingress) and outbound (egress). You can create firewall rules to **allow** or **deny** specific connections based on a combination of IP addresses, ports, and protocol.

https://cloud.google.com/vpc/docs/firewalls

Routes

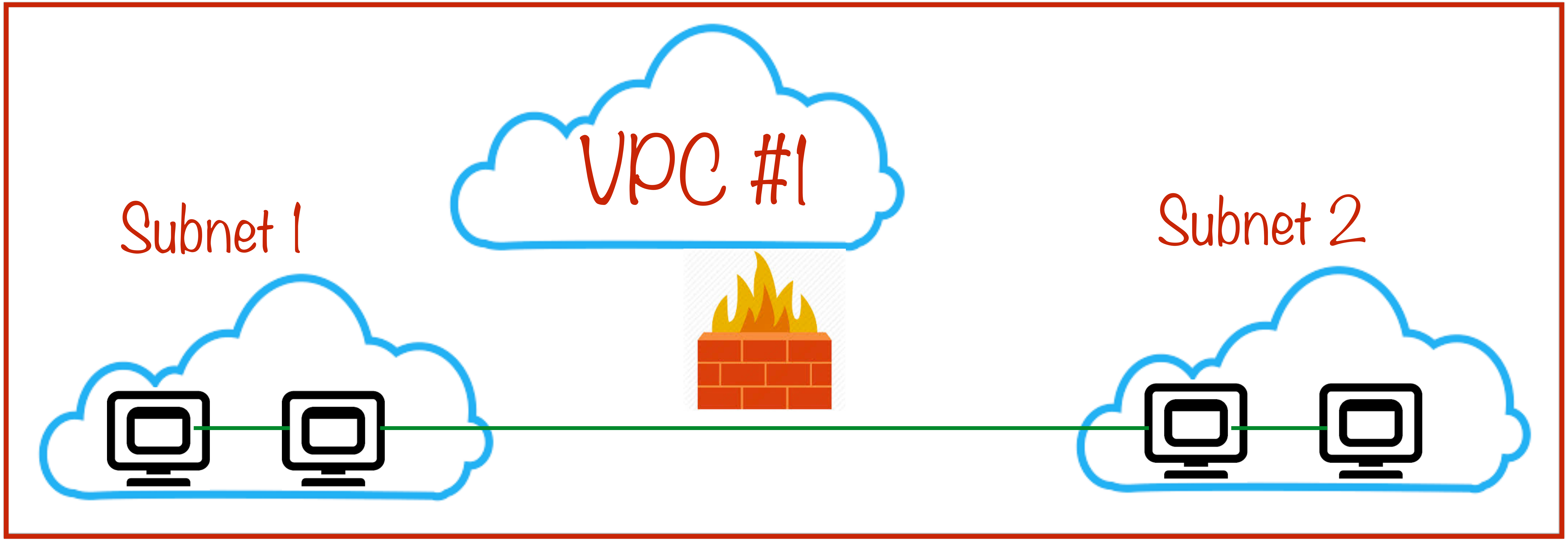


Routes



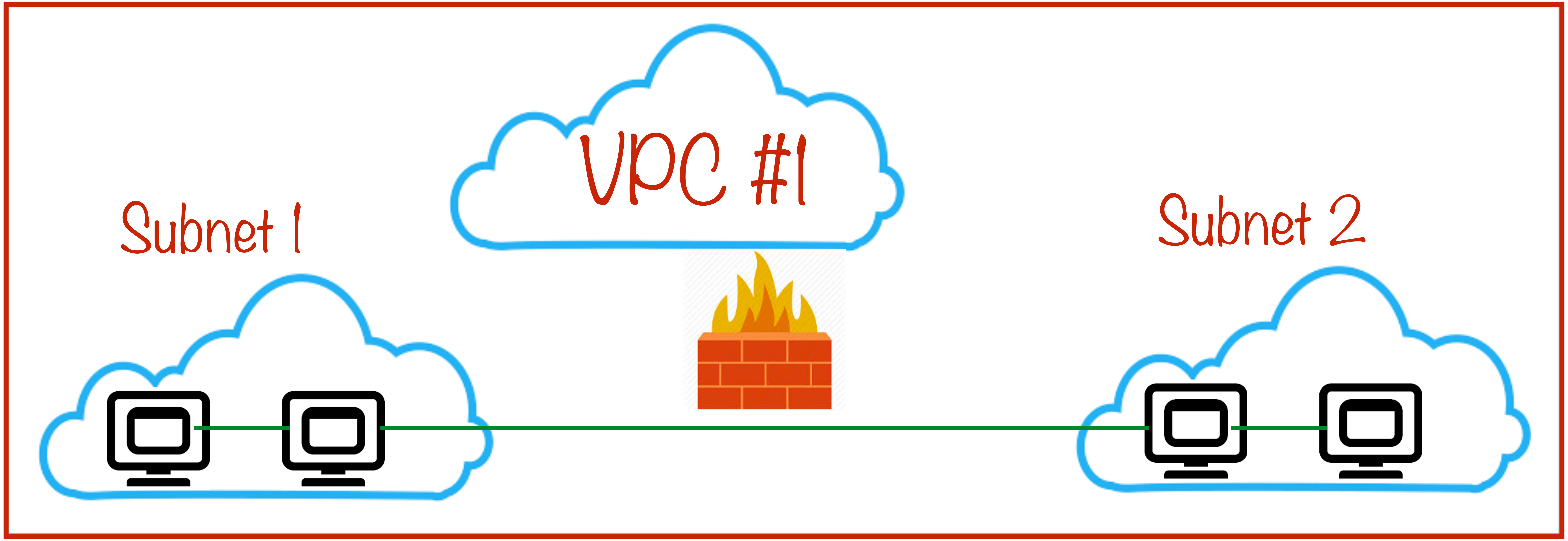
Firewall Rule

Configure firewall rules for packets to traverse this route



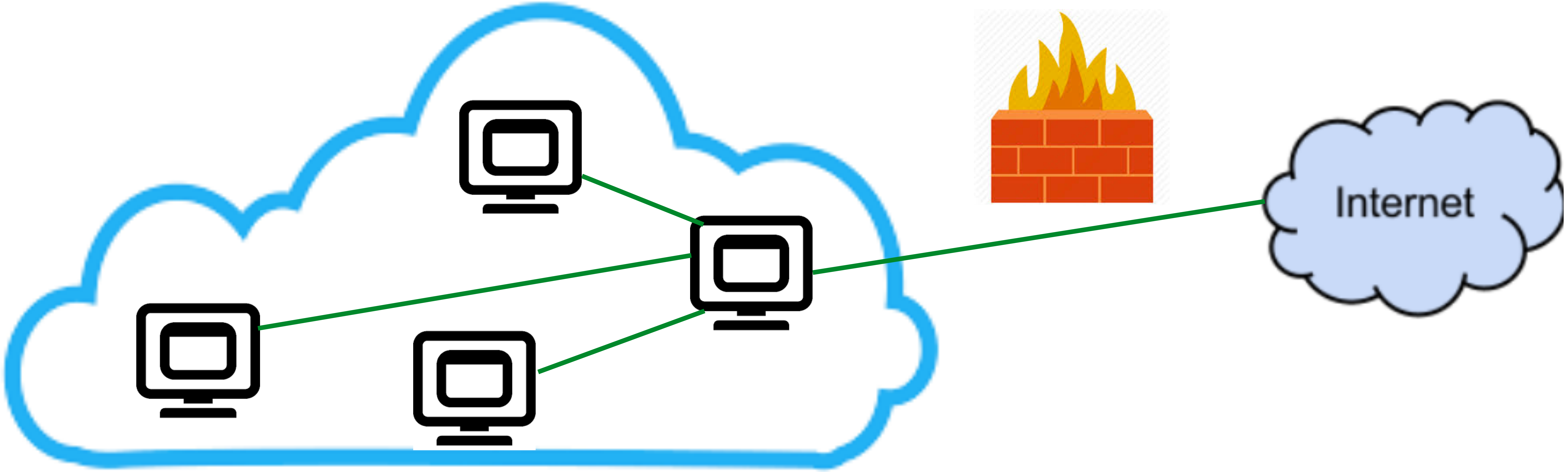
Firewall Rule

Firewall rules exist between instances in the same network



Firewall Rule

As well as between instances and other networks



Firewall Rules

- **Action:** allow or deny
- **Direction:** ingress or egress
- Source IPs (ingress), Destination IPs (egress)
- Protocol and port
- Specific instance names
- Priorities and tiebreakers

GCP firewall rules are stateful

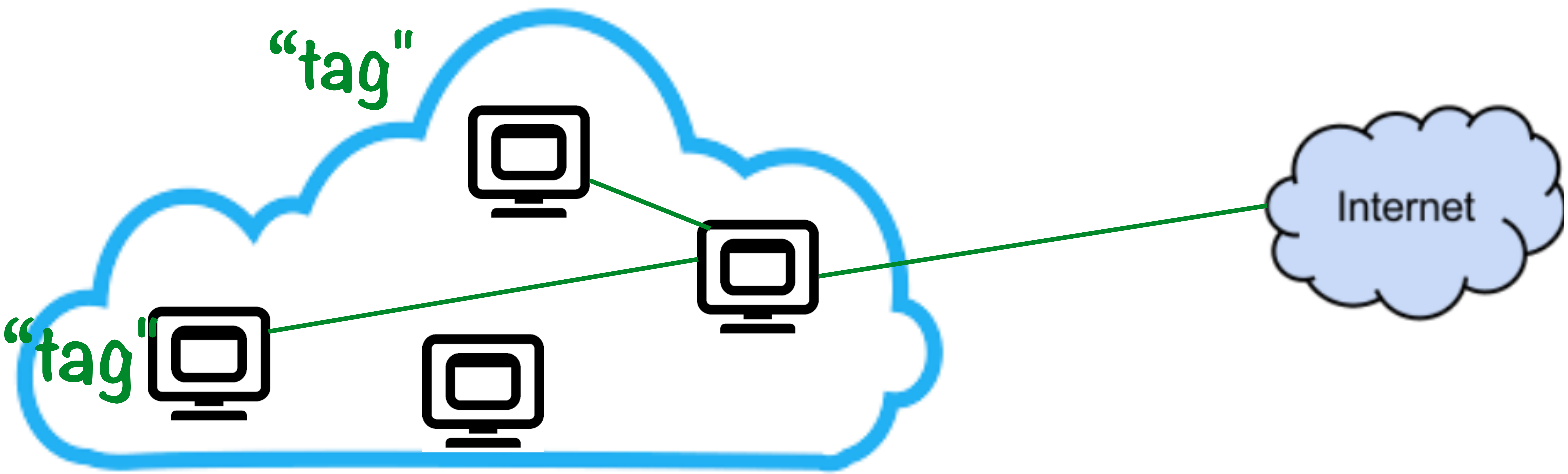
If a connection is allowed, all traffic in the flow is also allowed, in both directions

Rule Assignment

- Every rule is assigned to every instance in a network
- Rule assignment can be restricted using tags or service accounts
    - Allow traffic from instances with source tag "backend"
    - Deny traffic to instances running as service account "blah@appspot.gcpserviceaccount.com"

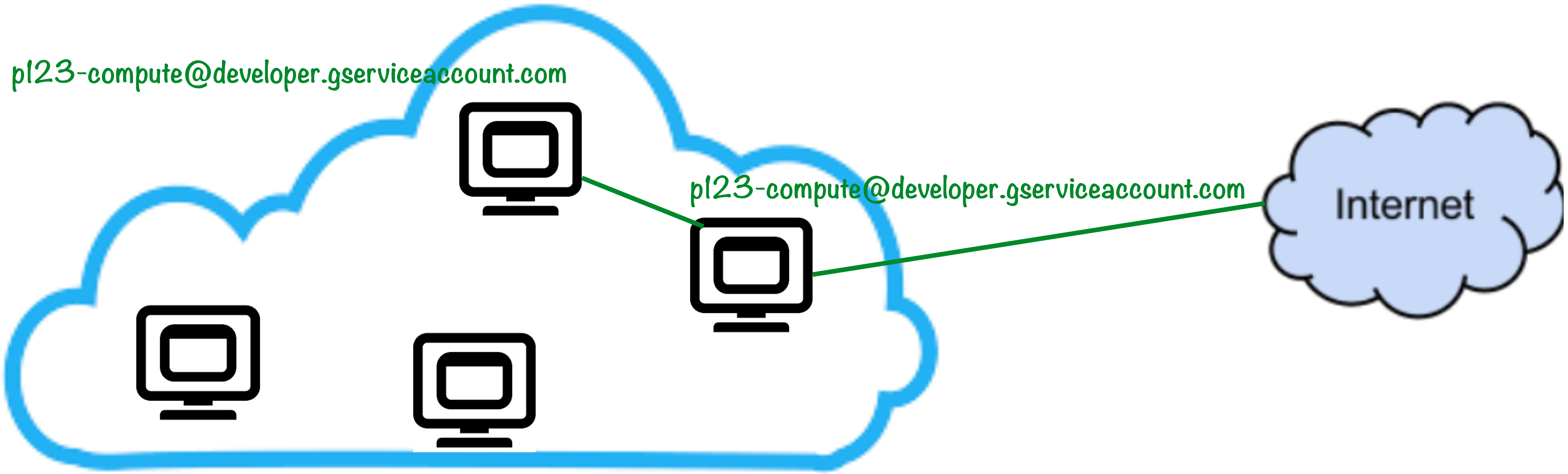
Service Accounts vs. Tags

Used to control which instances of a network the firewall rule applies to



Service Accounts vs. Tags

Used to control which instances of a network the firewall rule applies to


p123-compute@developer.gserviceaccount.com
p123-compute@developer.gserviceaccount.com
Internet

Service Accounts vs. Tags

Service Accounts	Tags
Represents the identity that the instance runs with	Logically group resources for billing or applying firewalls
An instance can have just one service account	An instance can have any number of tags
Restricted by IAM permissions, permissions to start an instance with a service account has to be explicitly given	Tags can be changed by any user who can edit an instance
Changing a service account requires stopping and restarting an instance	Changing tags is metadata update and is a much lighter operation

Prefer service accounts to tags to group instances so that firewall rules can be applied

Firewall Rules

- Only IPv4 addresses are supported in a firewall rule
- Firewall rules are specific to a network. They cannot be shared between networks
- Tags and service accounts cannot be used together in the same firewall rule

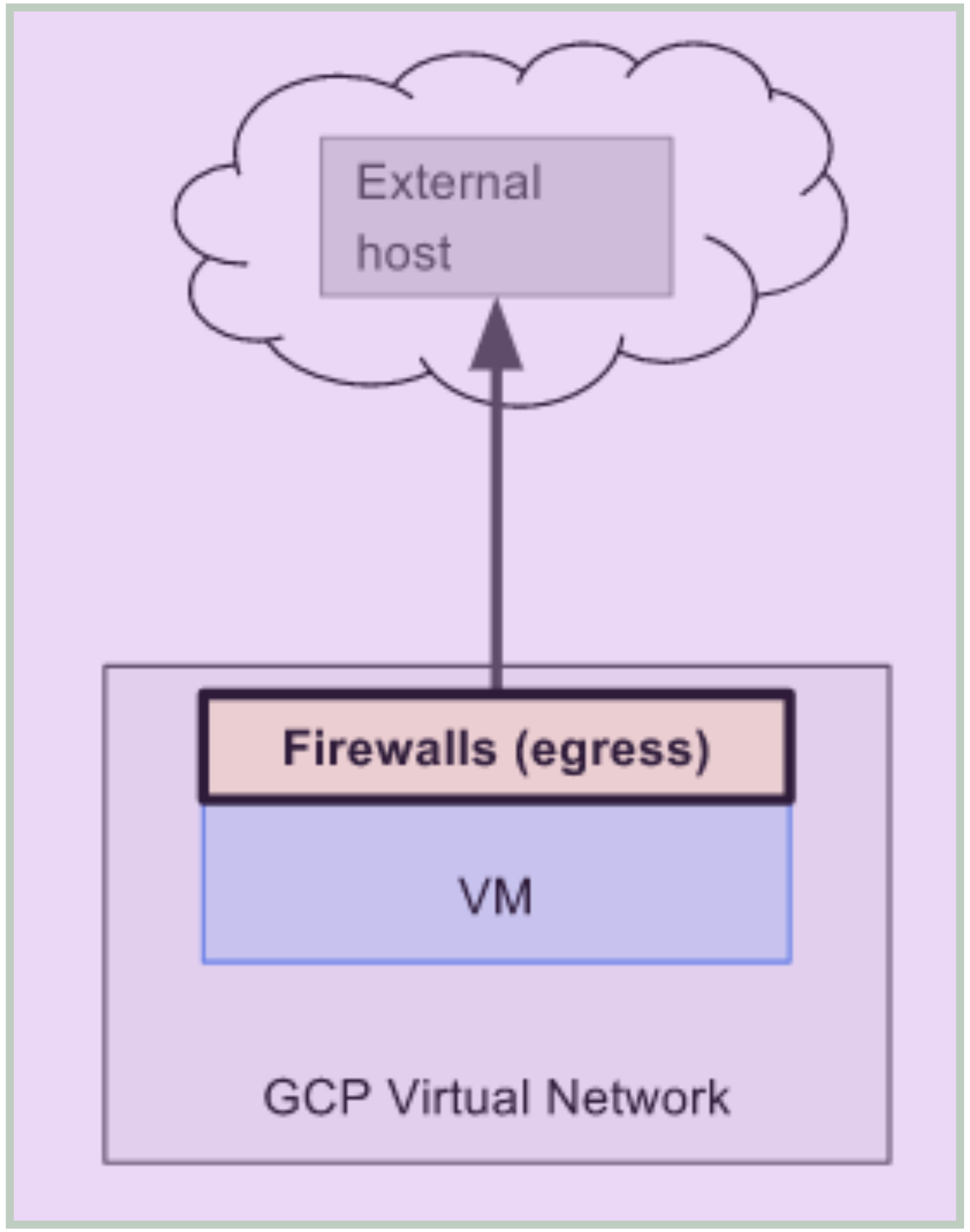
Implied Rules

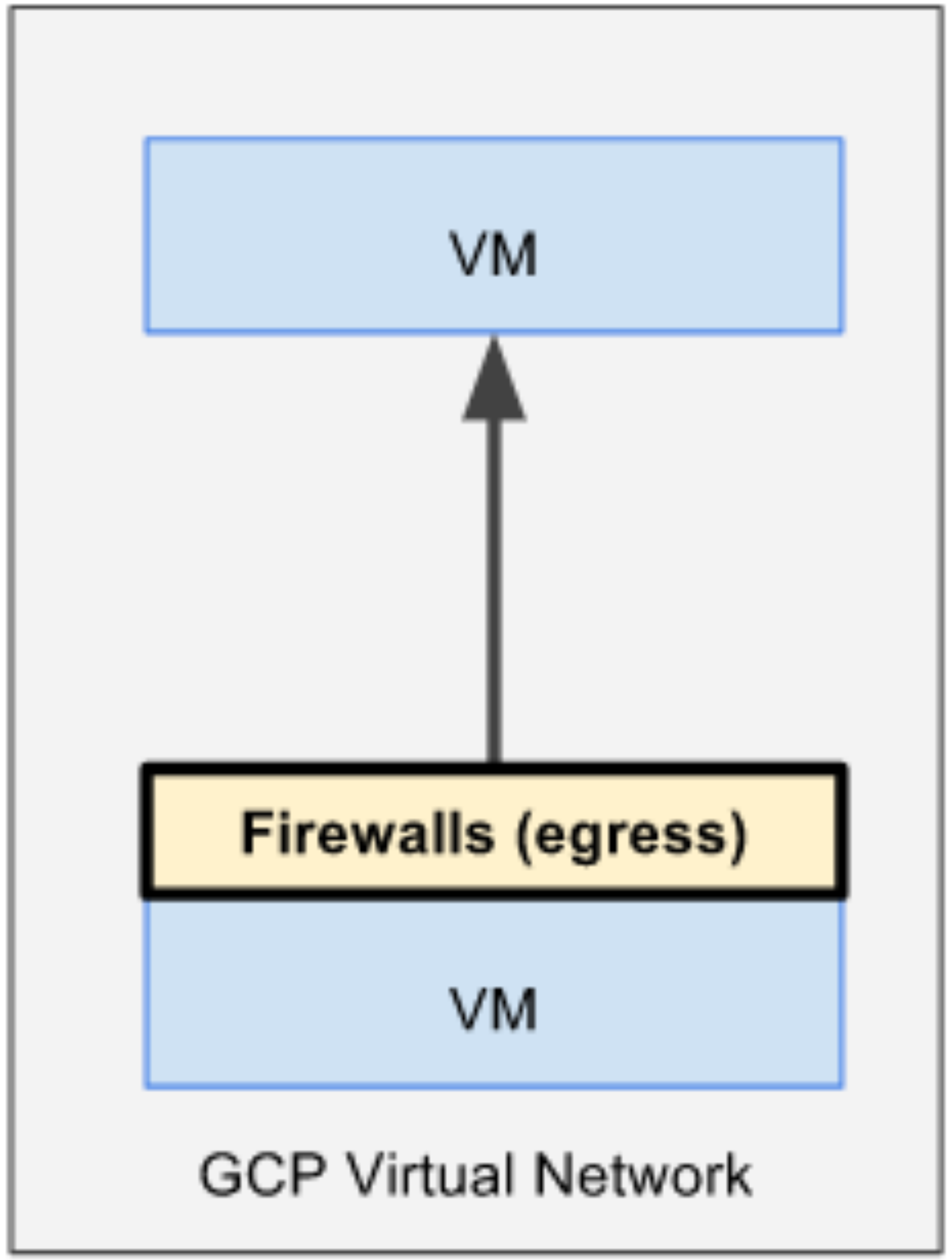
- A default "allow egress" rule.
  - Allows all egress connections. Rule has a priority of 65535.
- A default "deny ingress" rule.
  - Deny all ingress connection. Rule has a priority of 65535

Firewall Rules for the "default" network

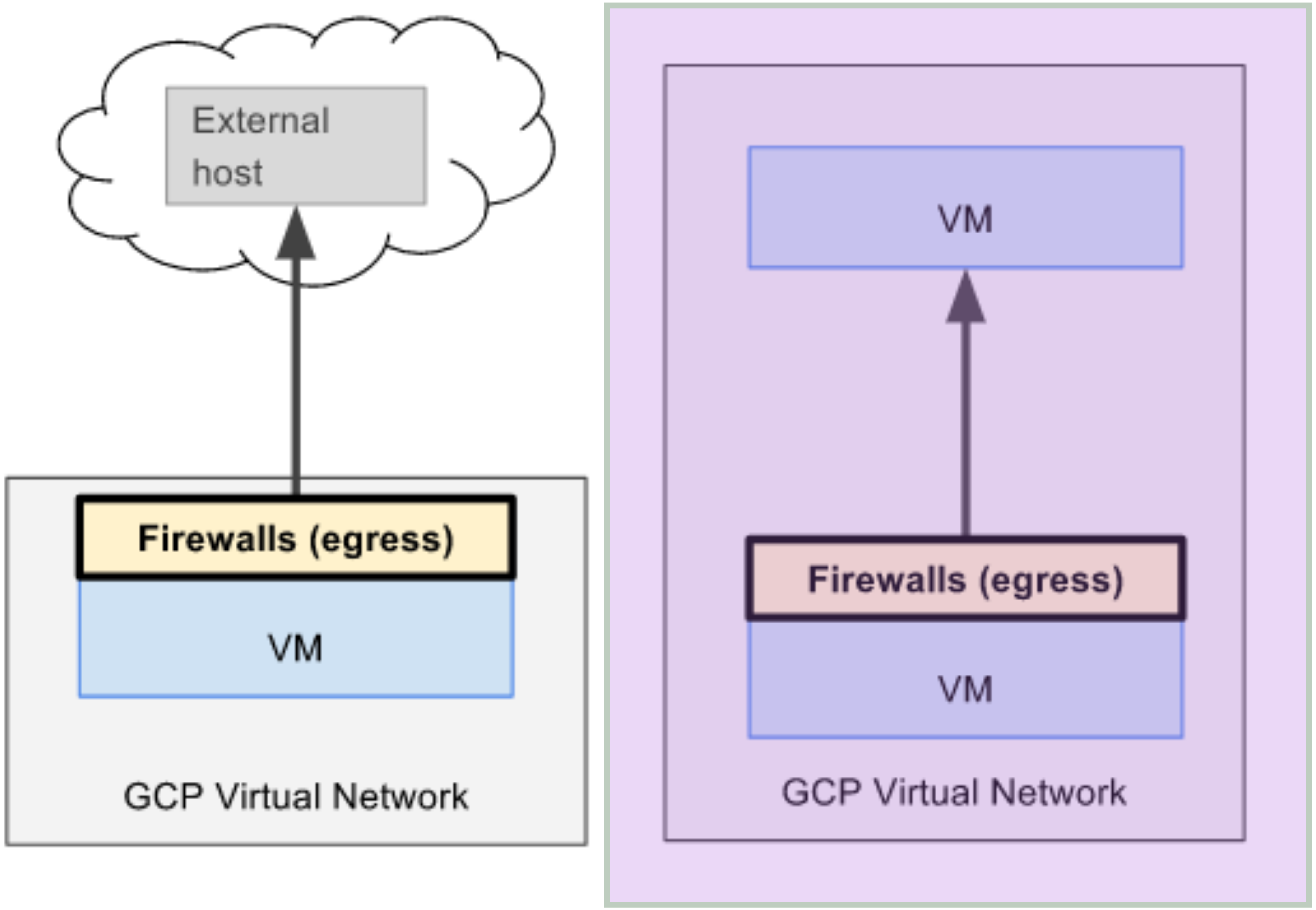
- **default-allow-internal**
  - Allows ingress network connections of any protocol and port between VM instances on the network
- **default-allow-ssh**
  - Allows ingress TCP connections from any source to any instance on the network over port 22
- **default-allow-icmp**
  - Allows ingress ICMP traffic from any source to any instance on the network.
- **default-allow-rdp**
  - Allows ingress remote desktop protocol traffic to TCP port 3389

Egress Connections

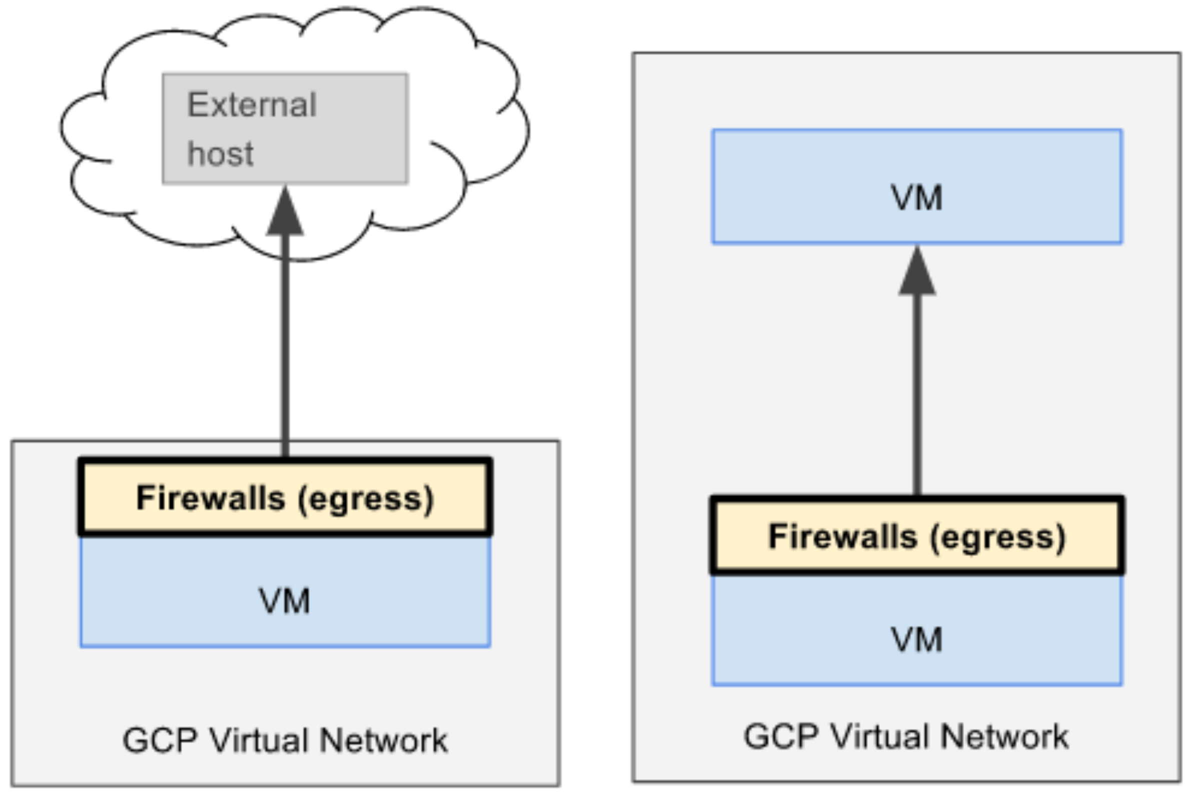




Egress Connections

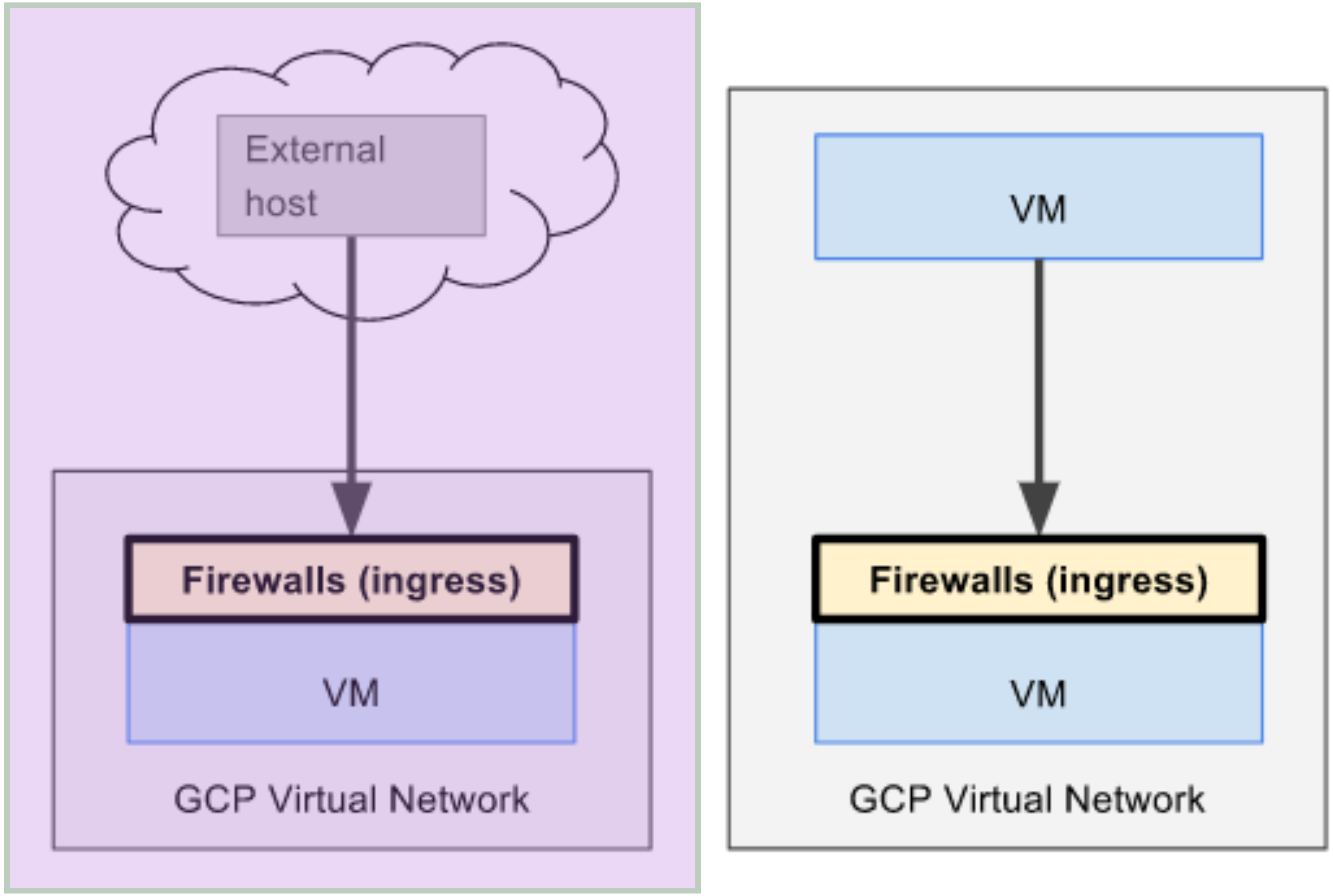


Egress Connections

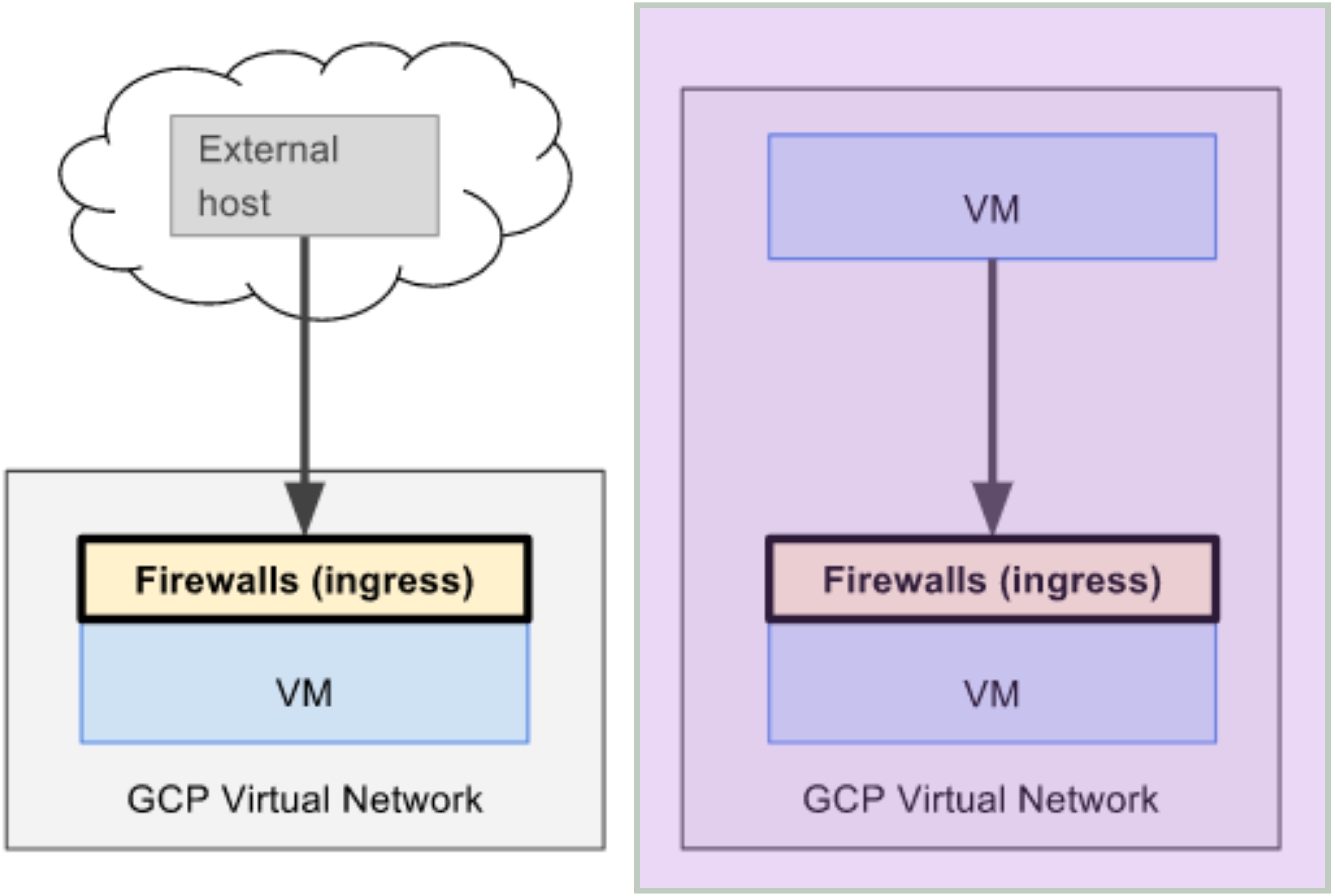


- Destination CIDR ranges, Protocols, Ports
- Destinations with specific tags or service accounts
  - Allow: Permit matching egress connections
  - Deny: Block the matching egress connections

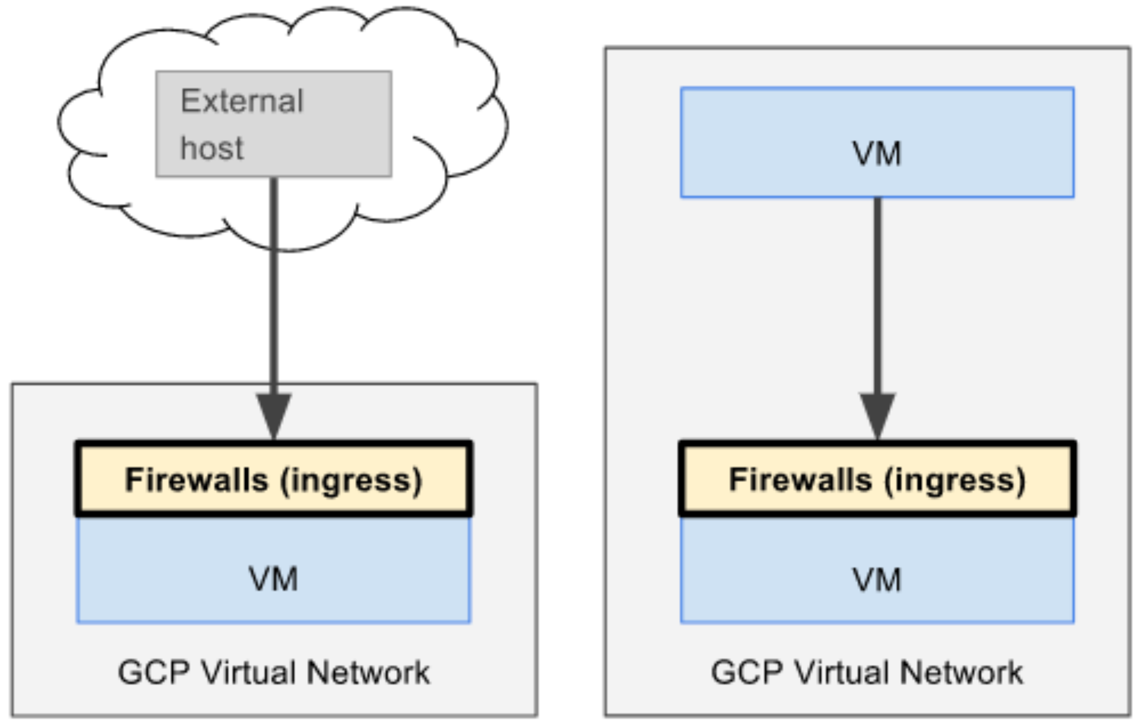
Ingress Connections



Ingress Connections



Ingress Connections



- Source CIDR ranges, Protocols, Ports
- Sources with specific tags or service accounts
  - Allow: Permit matching ingress connections
  - Deny: Block the matching ingress connections

Interconnecting Networks

3 Interconnection Options

Virtual Private Networks (VPNs) using Cloud Router

Dedicated Interconnect

Direct and Carrier Peering

3 Interconnection Options

Virtual Private Networks (VPNs) using Cloud Router

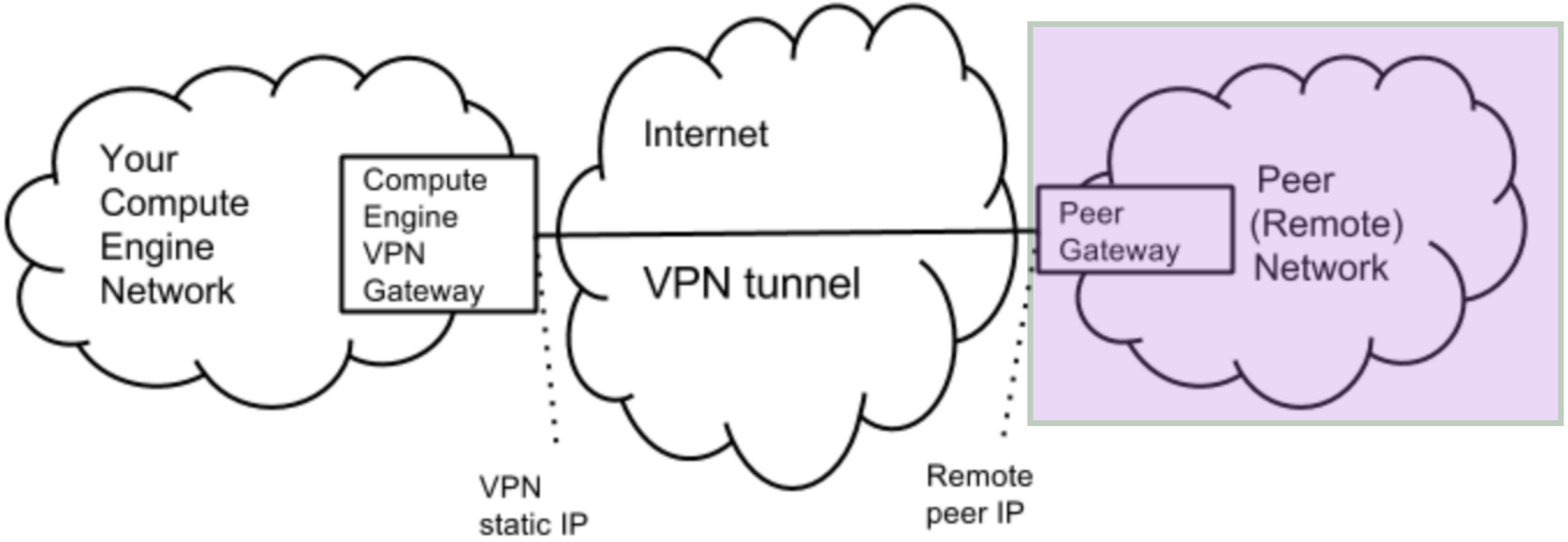
Dedicated Interconnect

Direct and Carrier Peering

VPN

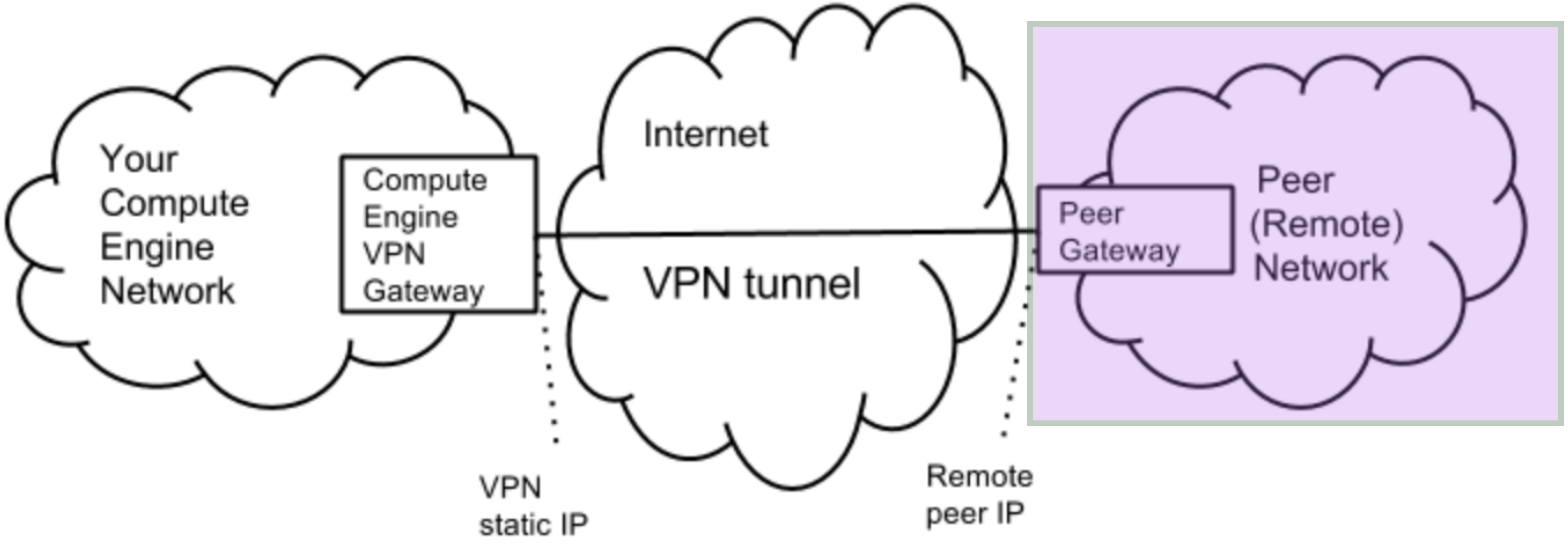
- Connects your on premise network to the Google Cloud VPC
- Offers 99.9% service availability
- Traffic is encrypted by one VPN gateway and then decrypted by another VPN gateway
- Supports both static and dynamic routes for traffic between on-premise and cloud

VPN



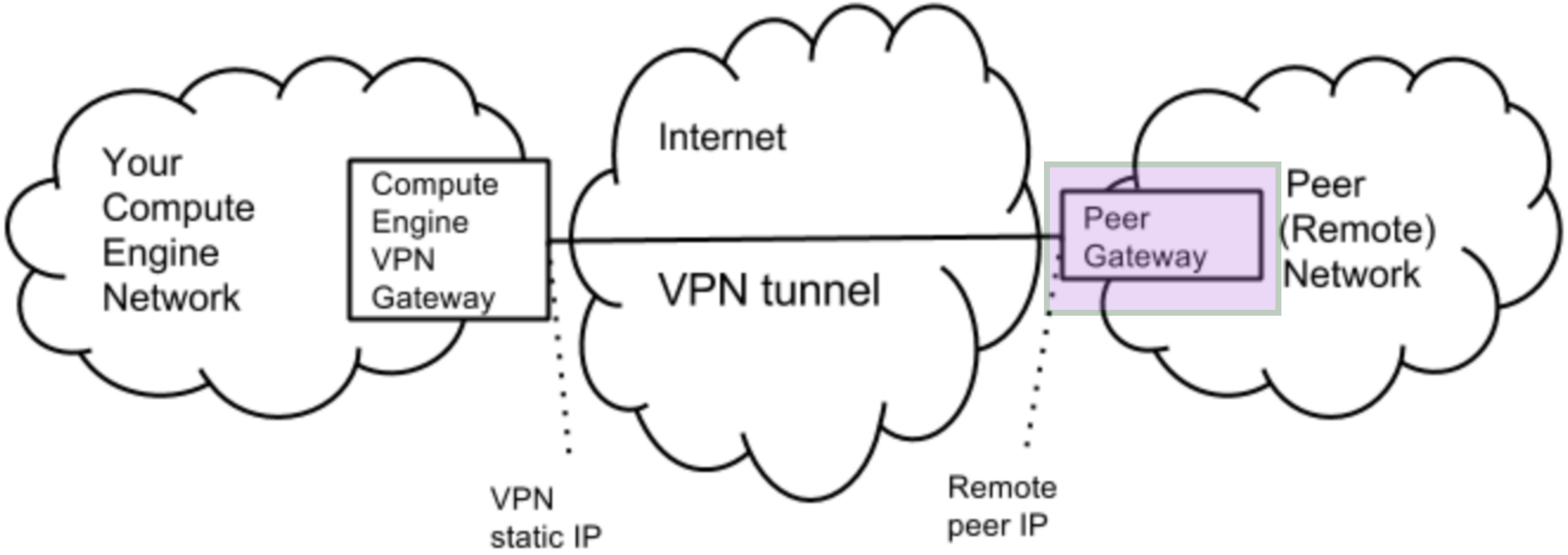
The on-premise network to be connected to the cloud

VPN



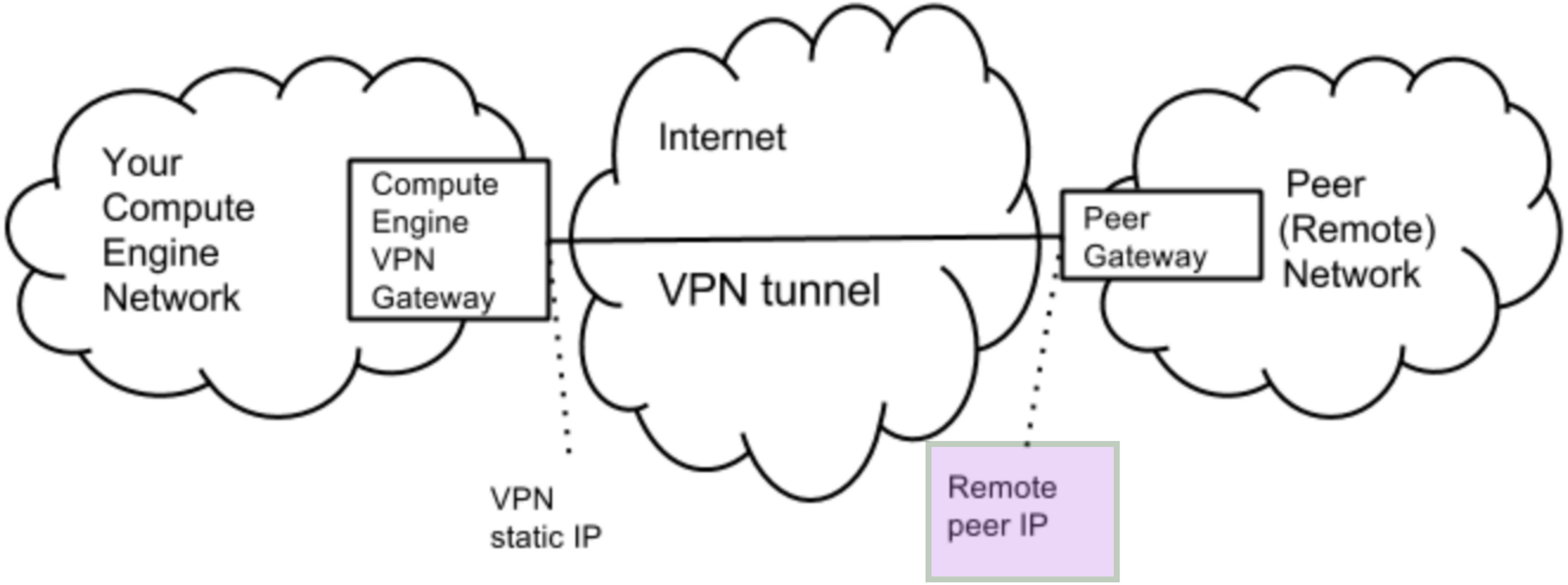
Can also be another cloud VPC

VPN



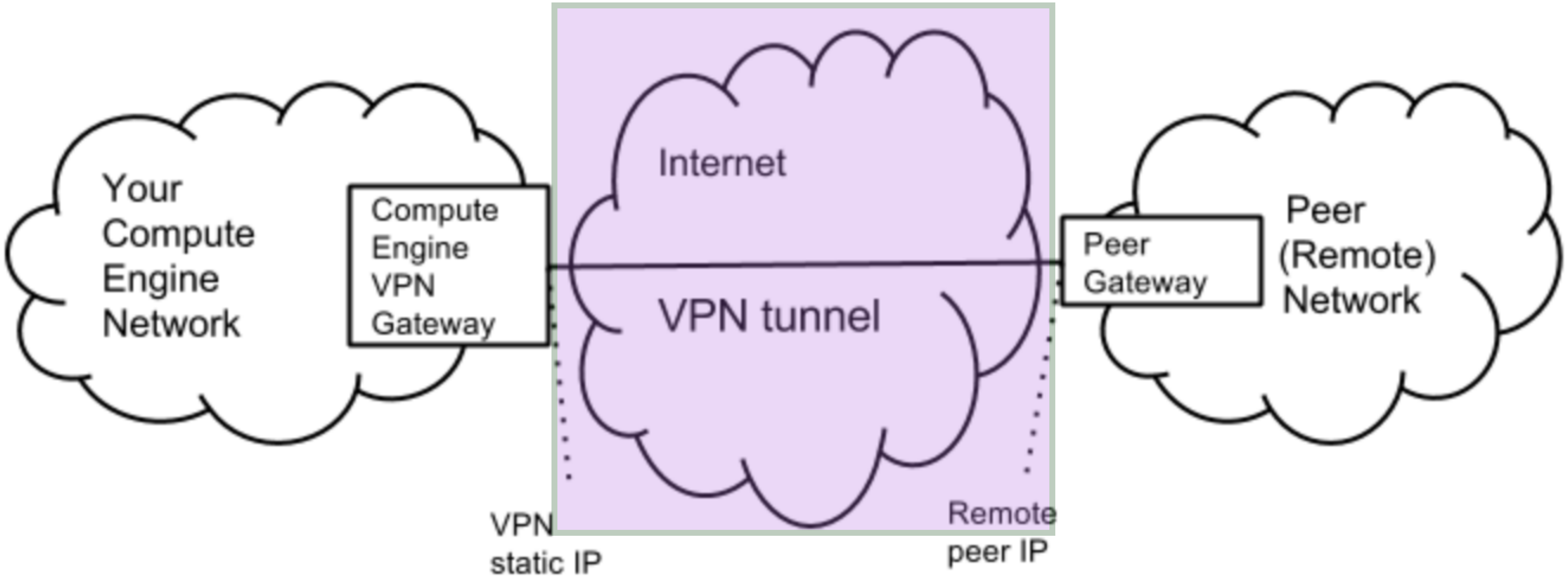
Only IPSec gateway to gateway scenarios are supported, does not work with client software on a laptop

VPN



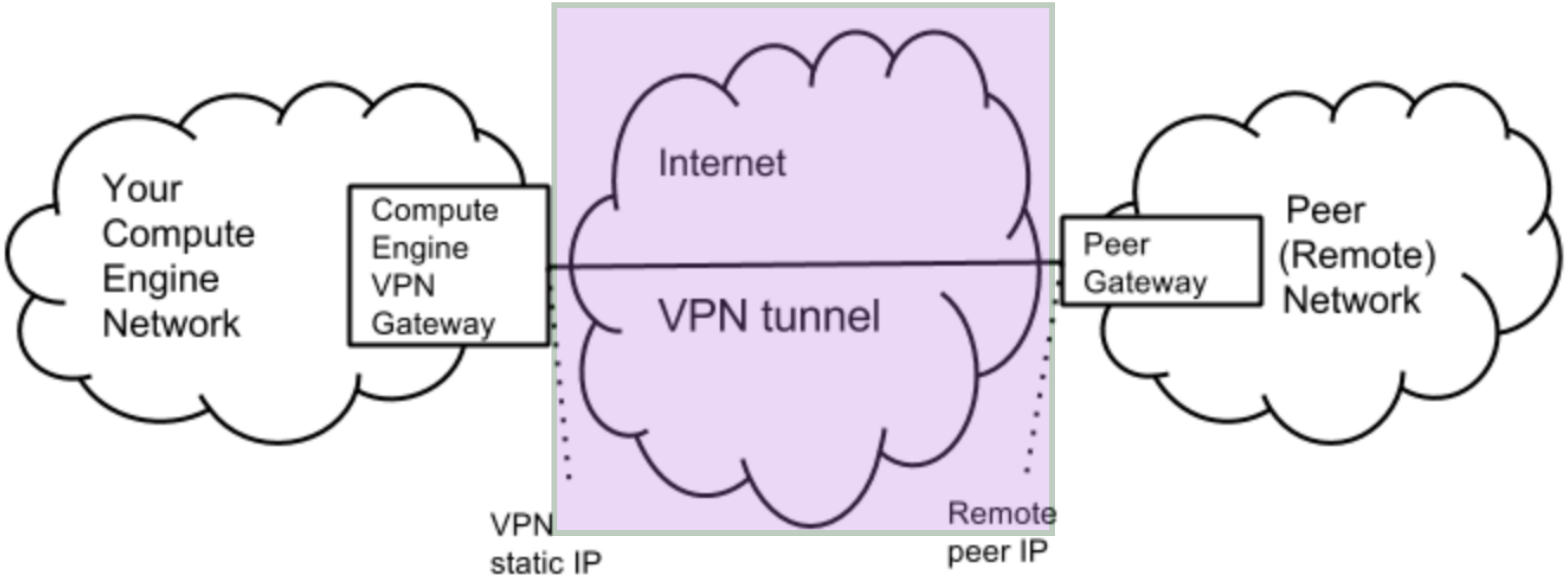
Must have a static external IP address

VPN



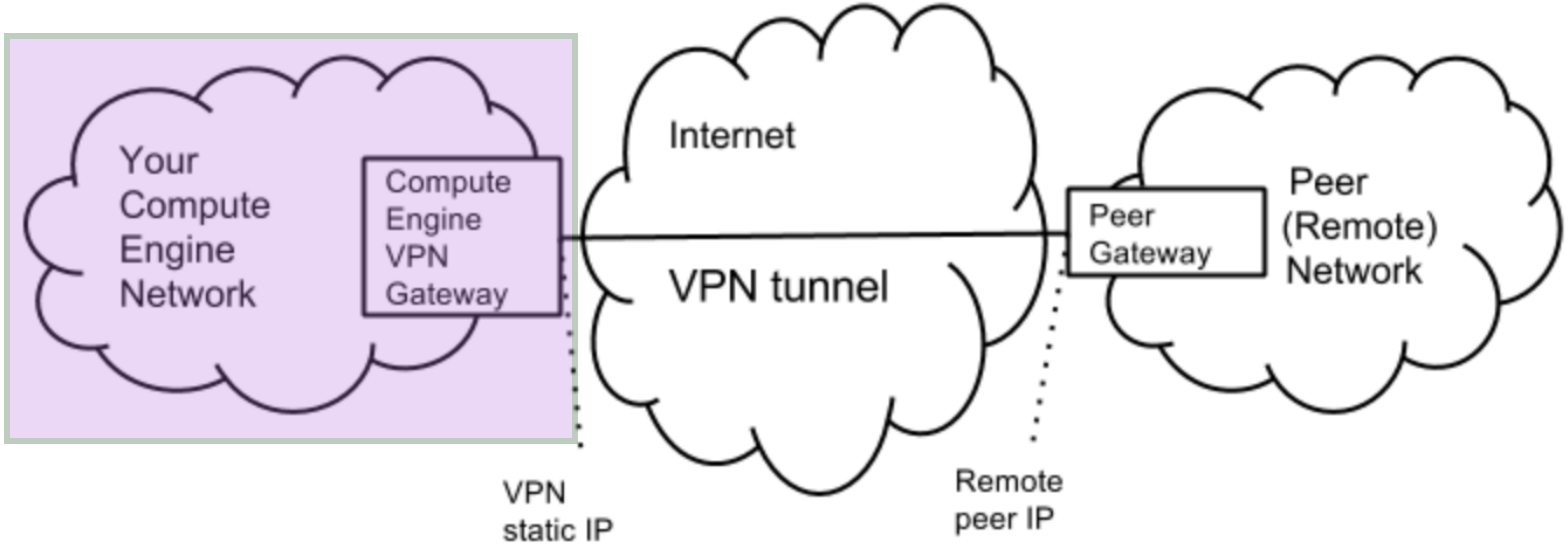
Needs to know what destination IPs are allowed and create routes to forward packets to those IPs

VPN



Can have multiple tunnels to a single VPN gateway, site-to-site VPN

VPN



The cloud VPC to connect to the on-premise network

VPN traffic has to traverse the internet

VPN will have higher latency and lower throughput as compared with dedicated interconnect and peering options

Cloud Router

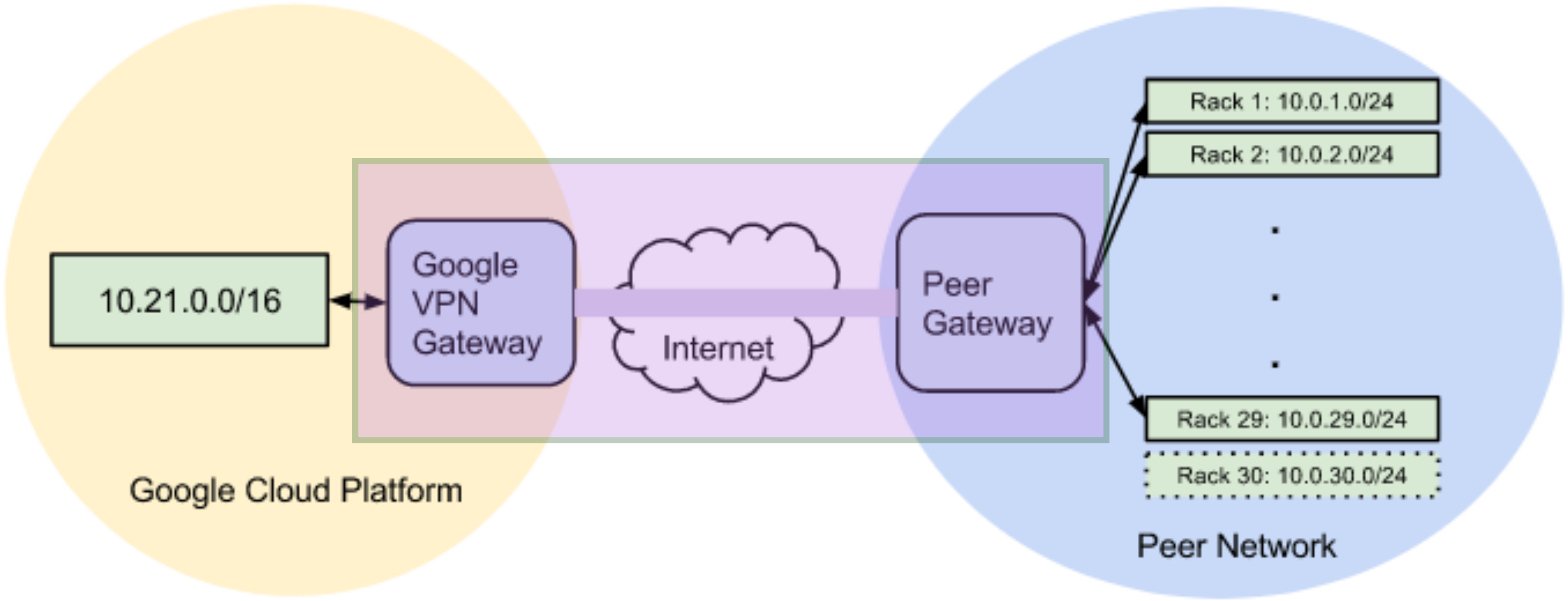
Cloud Router

- Dynamically exchange routes between Google VPCs and on premise networks
- Fully distributed and managed Google cloud service
- Peers with on premise gateway or router to exchange route information
- Uses the BGP or Border Gateway Protocol

Static Routes

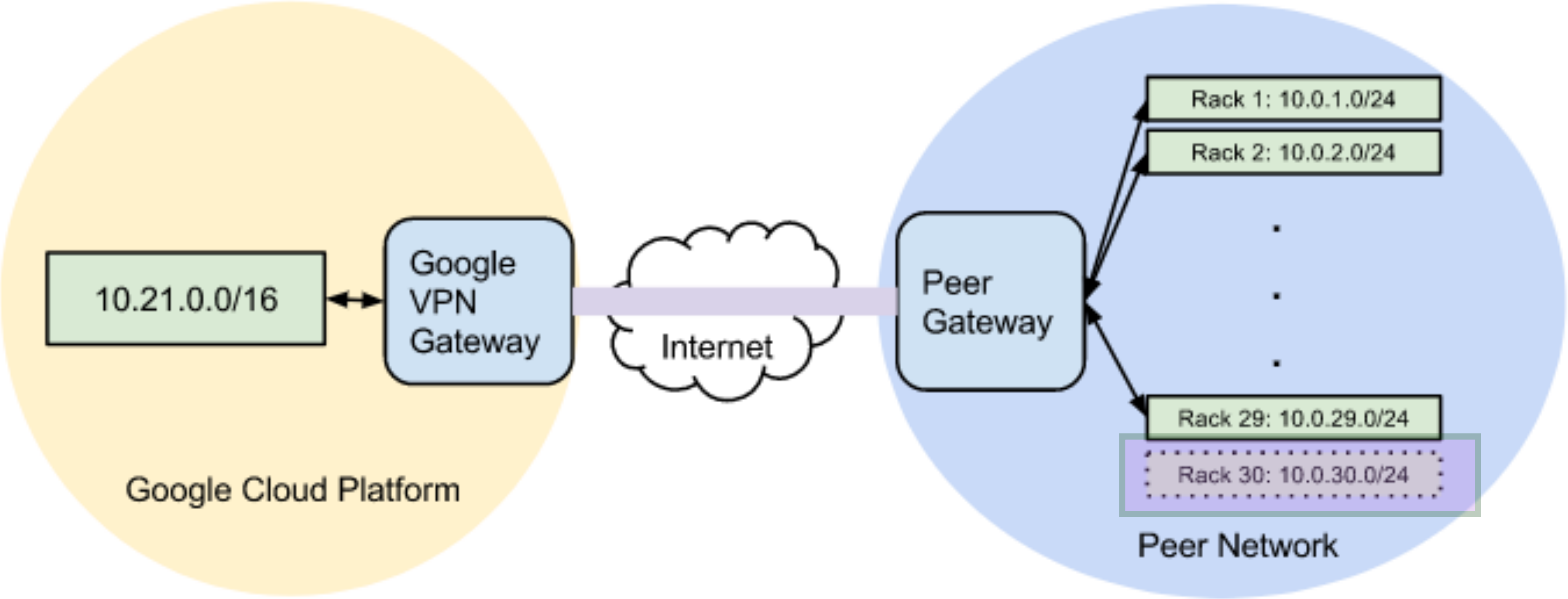
- Create and maintain a routing table
- A topology change in the network requires routes to be manually updated
- Cannot re-route traffic automatically if a link fails
- Suitable for small networks with stable topologies
- Routers do not advertise routes

Static Routing for VPN tunnels



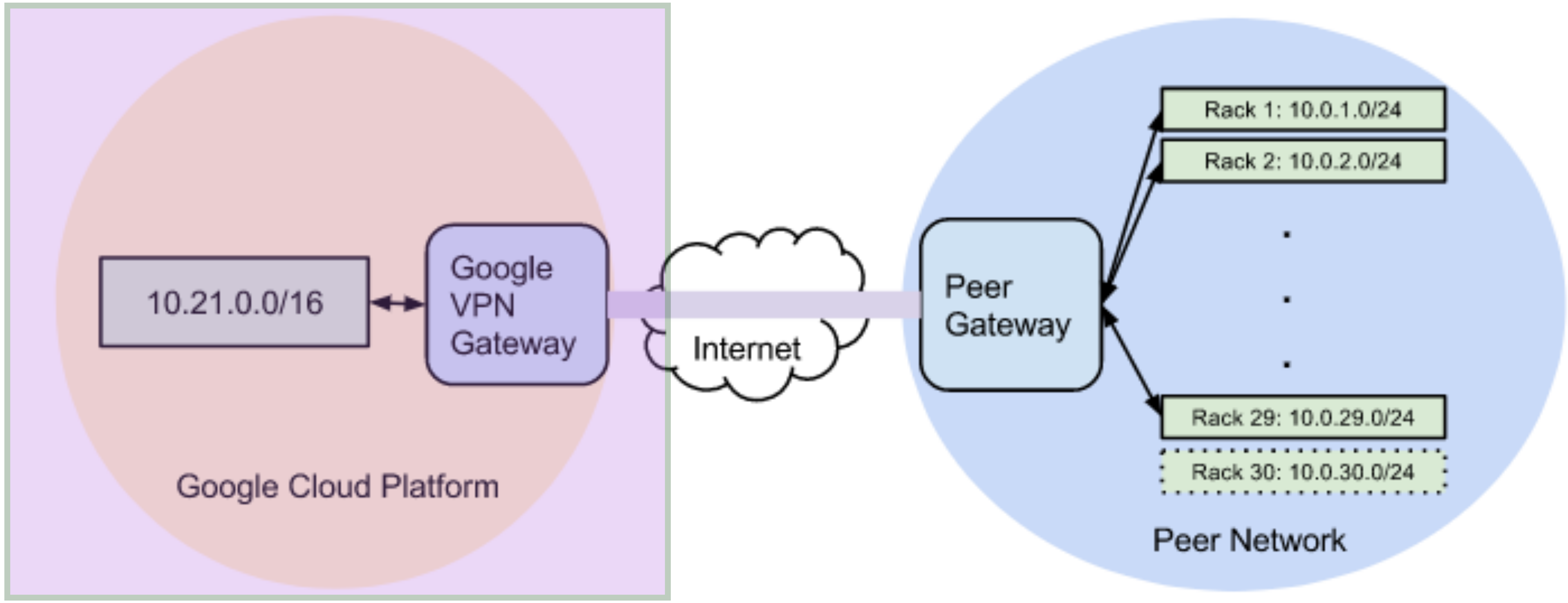
A VPN tunnel connecting a gateway at either end

Static Routing for VPN tunnels



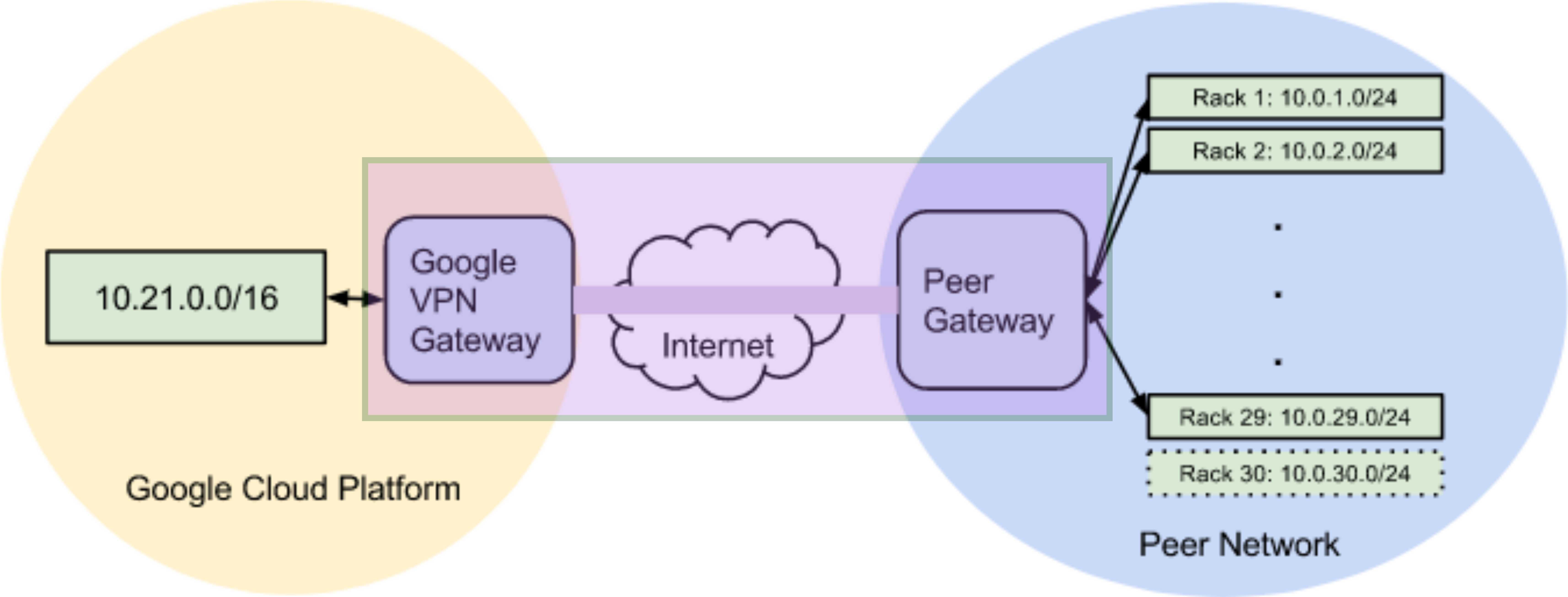
A new subnet added to the on premise network

Static Routing for VPN tunnels



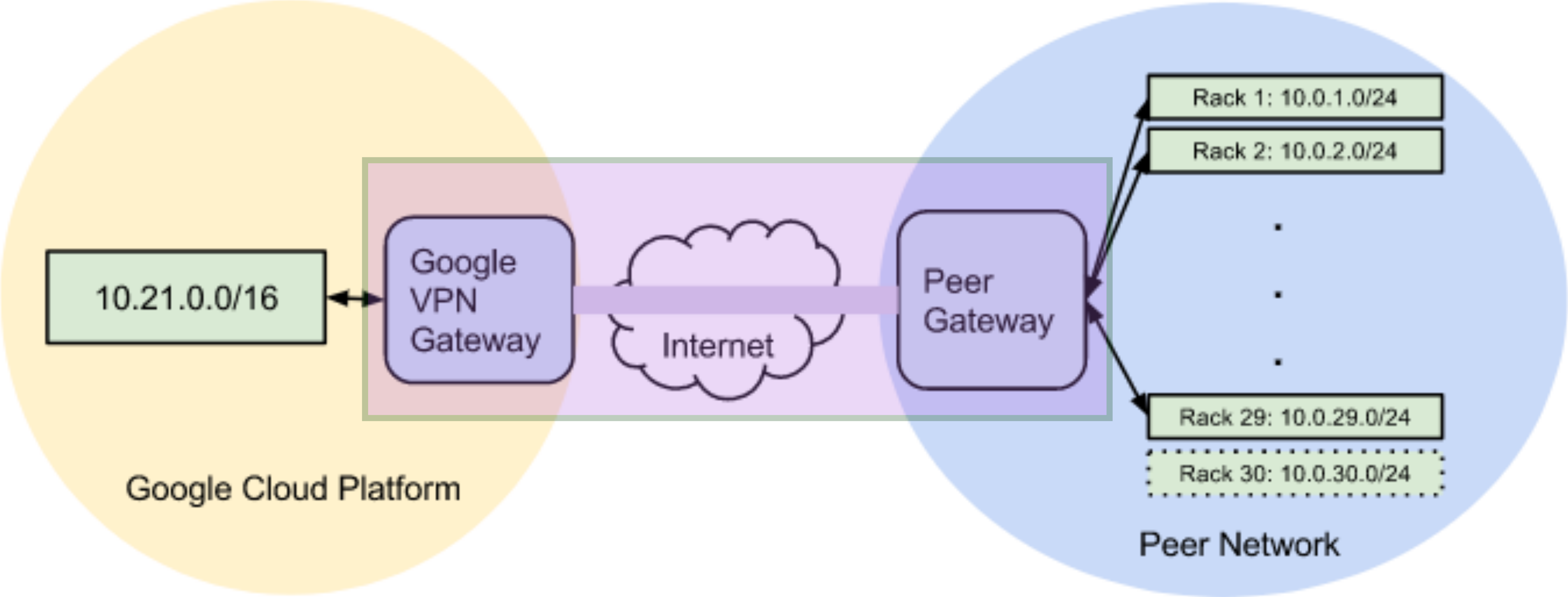
New routes need to be added to the cloud VPC to reach the new subnet

Static Routing for VPN tunnels



VPN tunnel will need to be torn down and re-established to include the new subnet

Static Routing for VPN tunnels

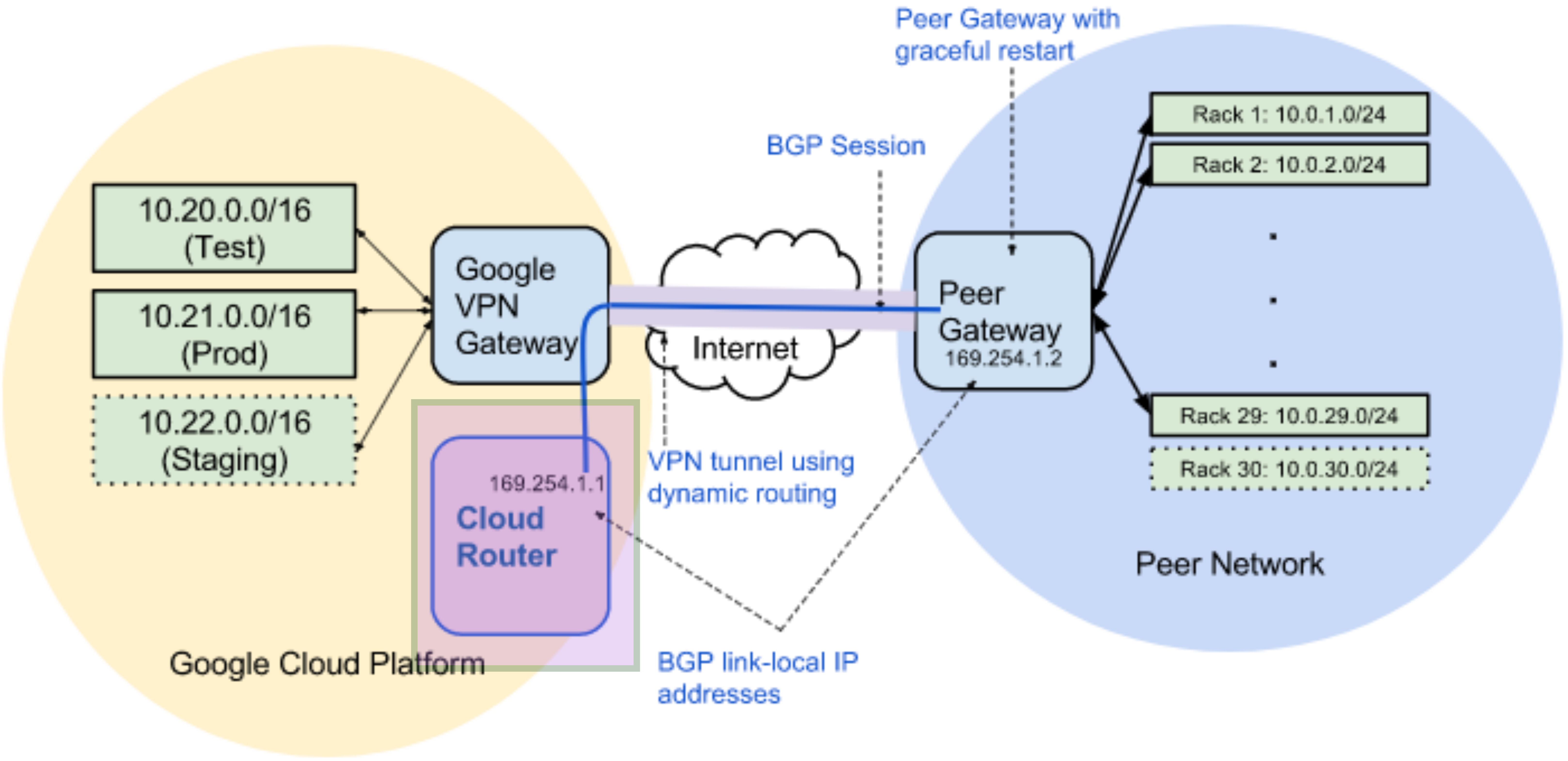


Static routes are slow to converge as updates are manual

Dynamic Routes

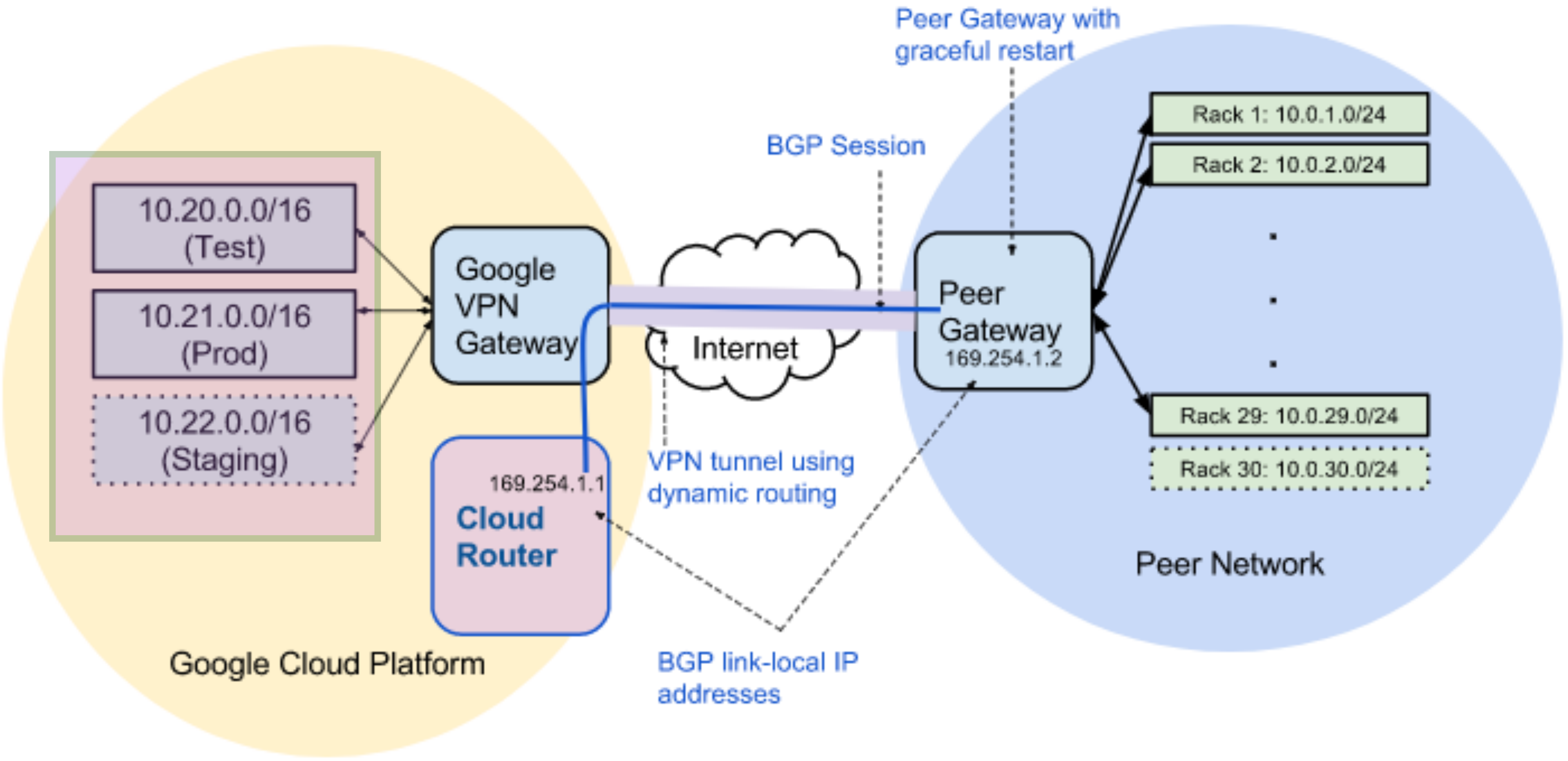
- Can be implemented using Cloud Router on the GCP
- Uses BGP to exchange route information between networks
- Networks automatically and rapidly discover changes
- Changes implemented without disrupting traffic

Dynamic Routing for VPN tunnels



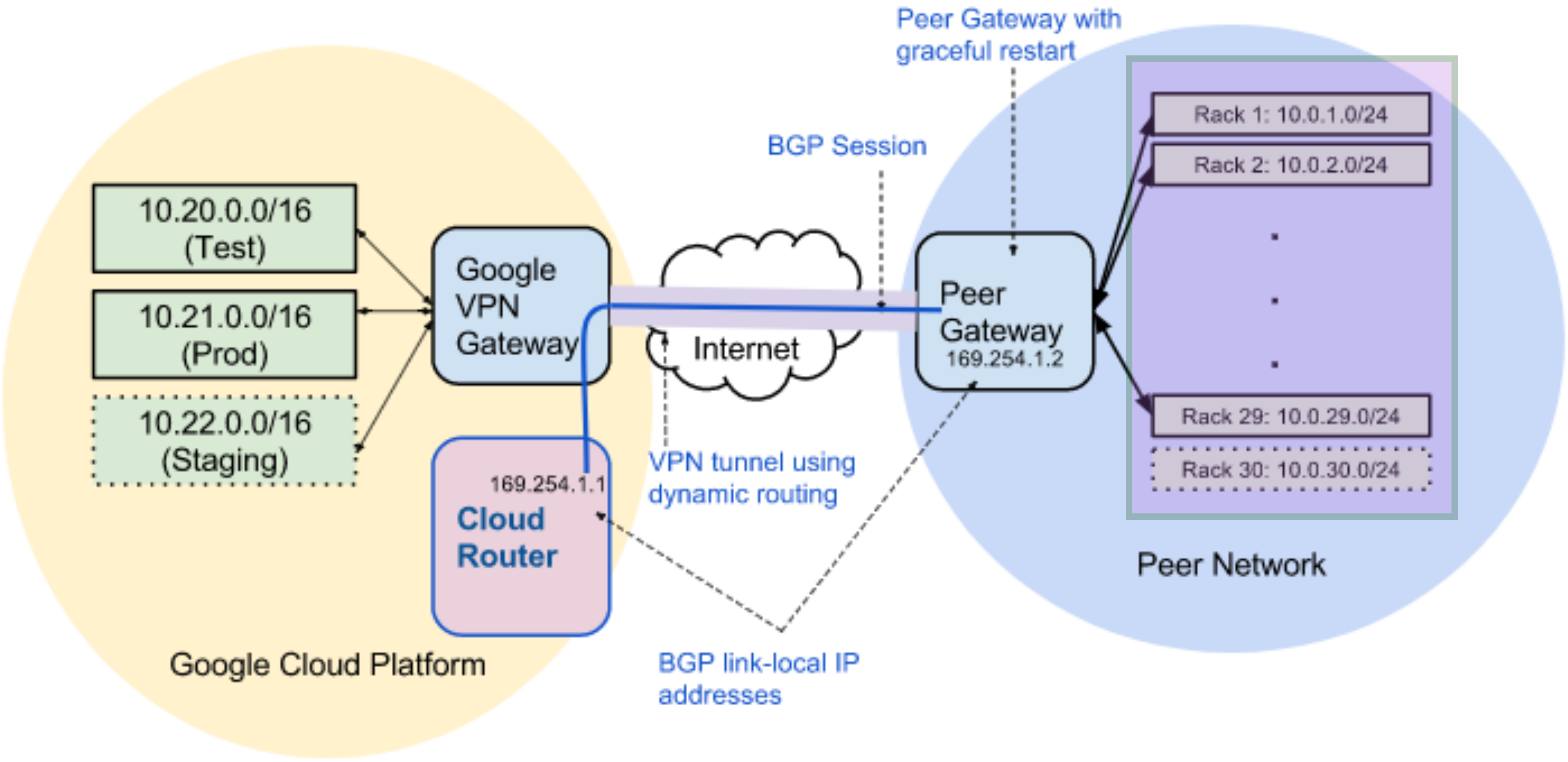
A Cloud Router belongs to a particular network and a particular region

Dynamic Routing for VPN tunnels



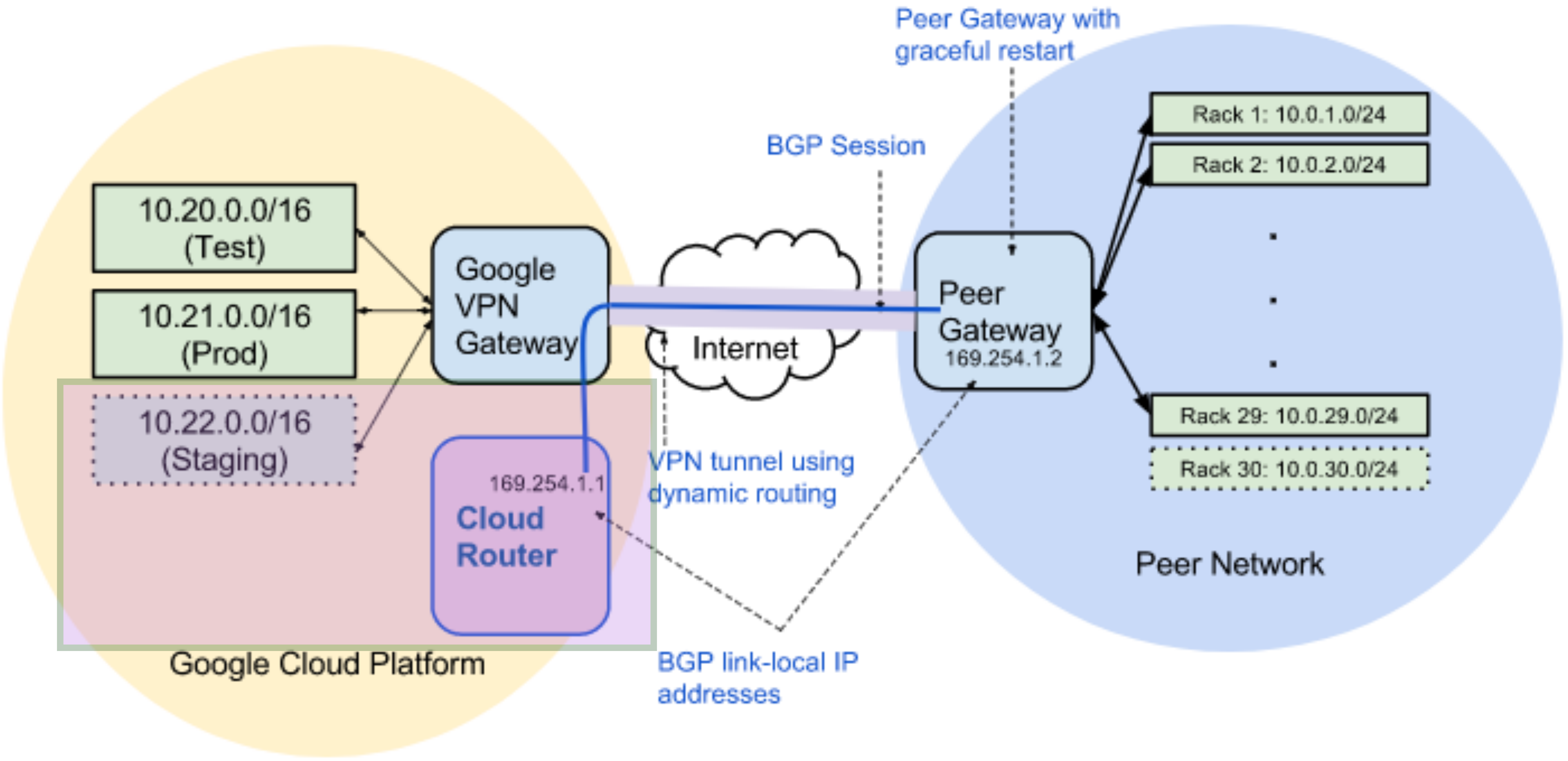
Subnets segmenting the network IP space

Dynamic Routing for VPN tunnels



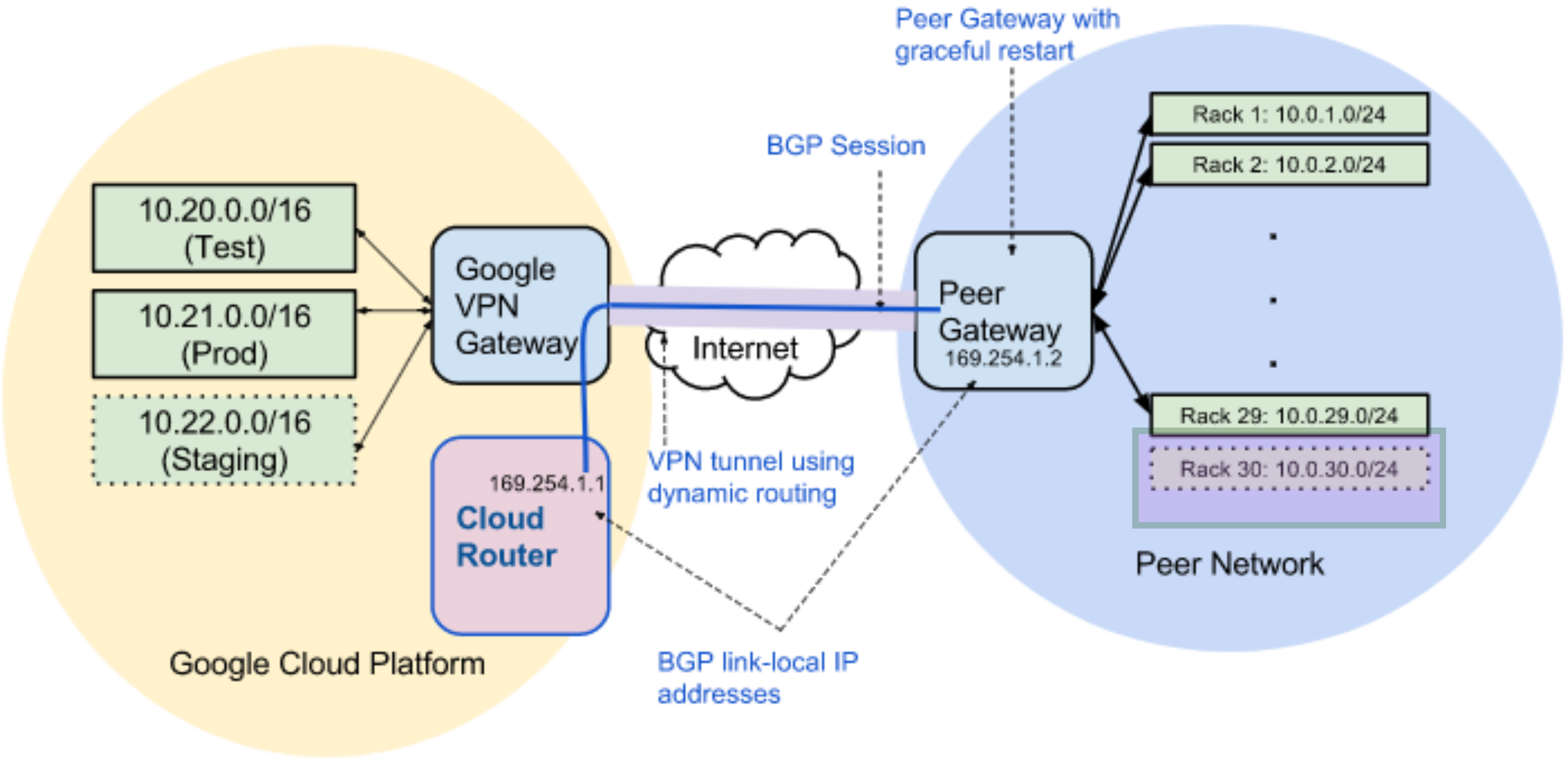
Subnets segmenting the network IP space

Dynamic Routing for VPN tunnels



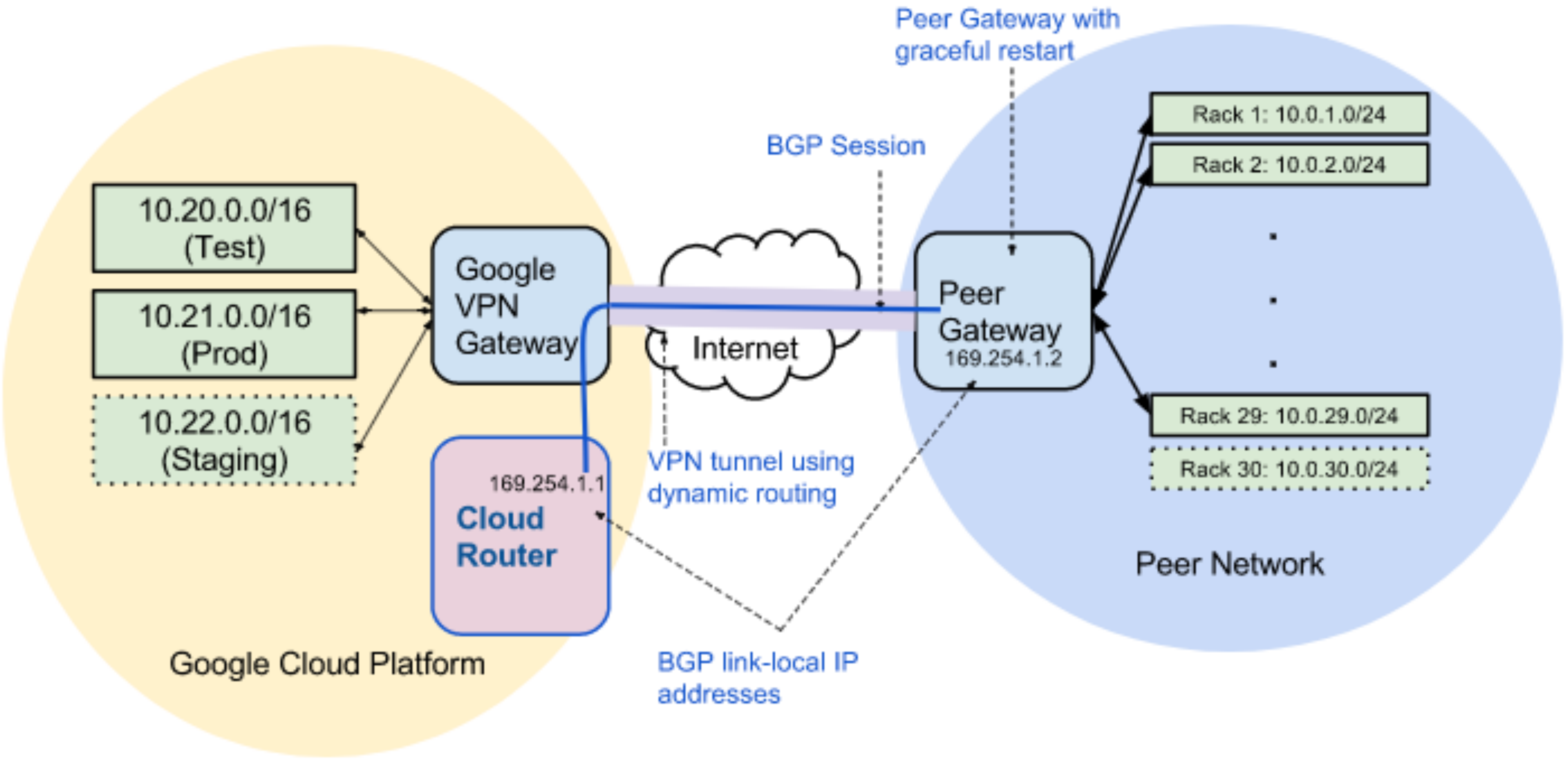
Advertises subnet changes using the BGP

Dynamic Routing for VPN tunnels



Also learns about subnet changes in the on premise network through BGP

Dynamic Routing for VPN tunnels



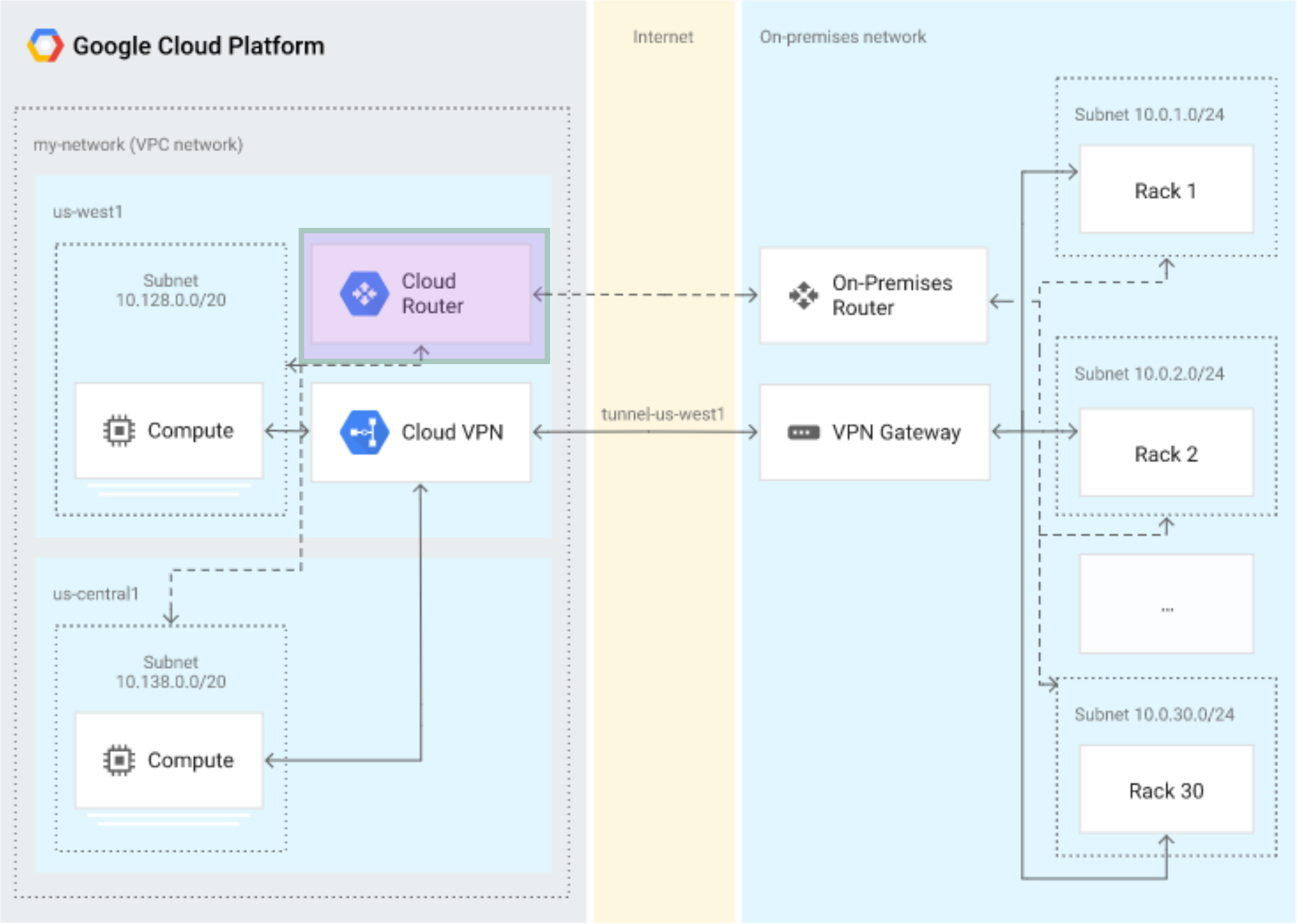
The IP address of the Cloud Router and the gateway router should both be link local IP addresses (valid only for communication within the network link)

Dynamic Routing Mode

- Determines which subnets are visible to Cloud Routers
- **Global dynamic routing**
  - Cloud router advertises all subnets in the VPC network to the on-premise router
- **Regional dynamic routing**
  - Advertises and propagates only those routes in its local region

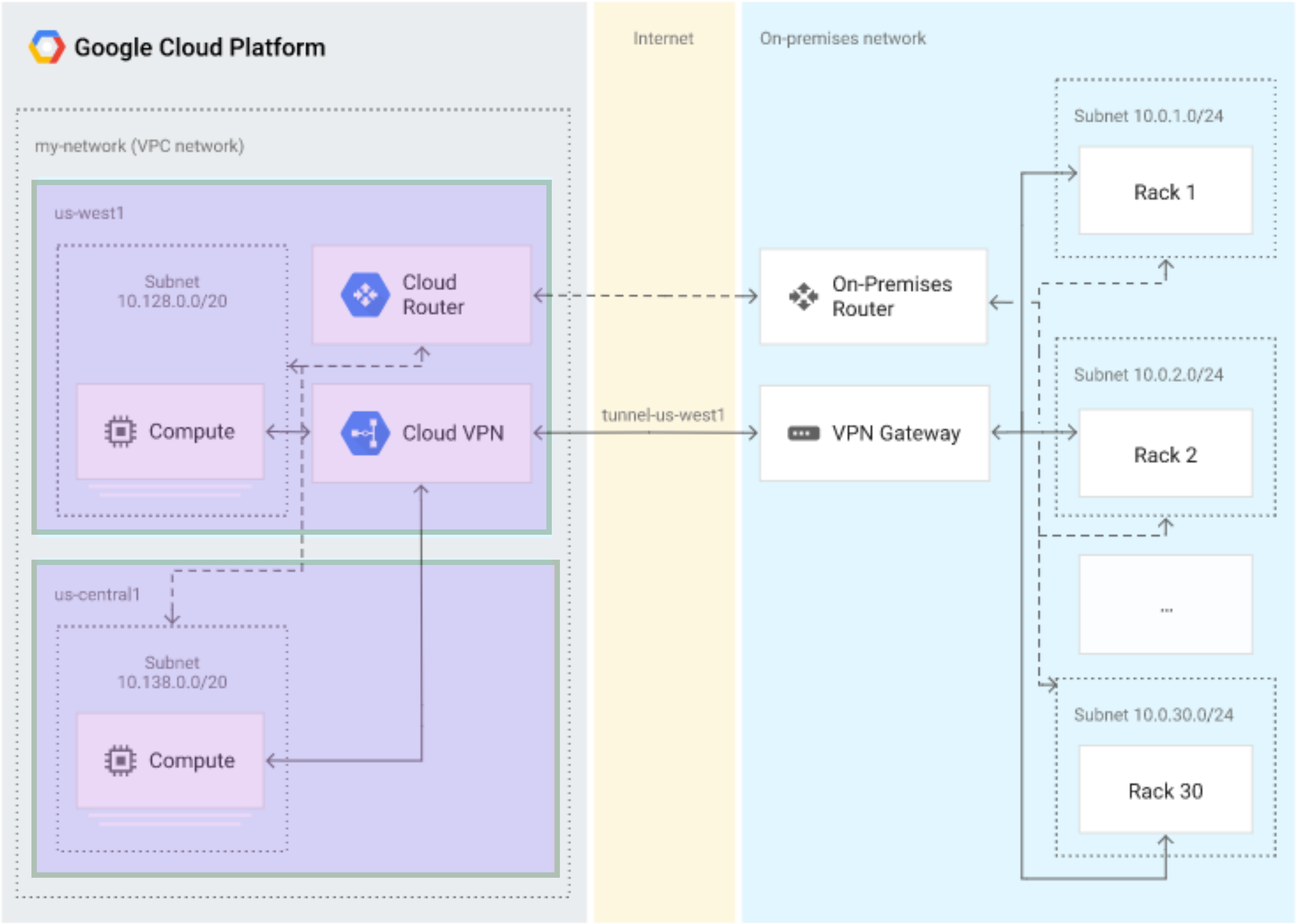
Global Dynamic Routing

Cloud router in the us-west region



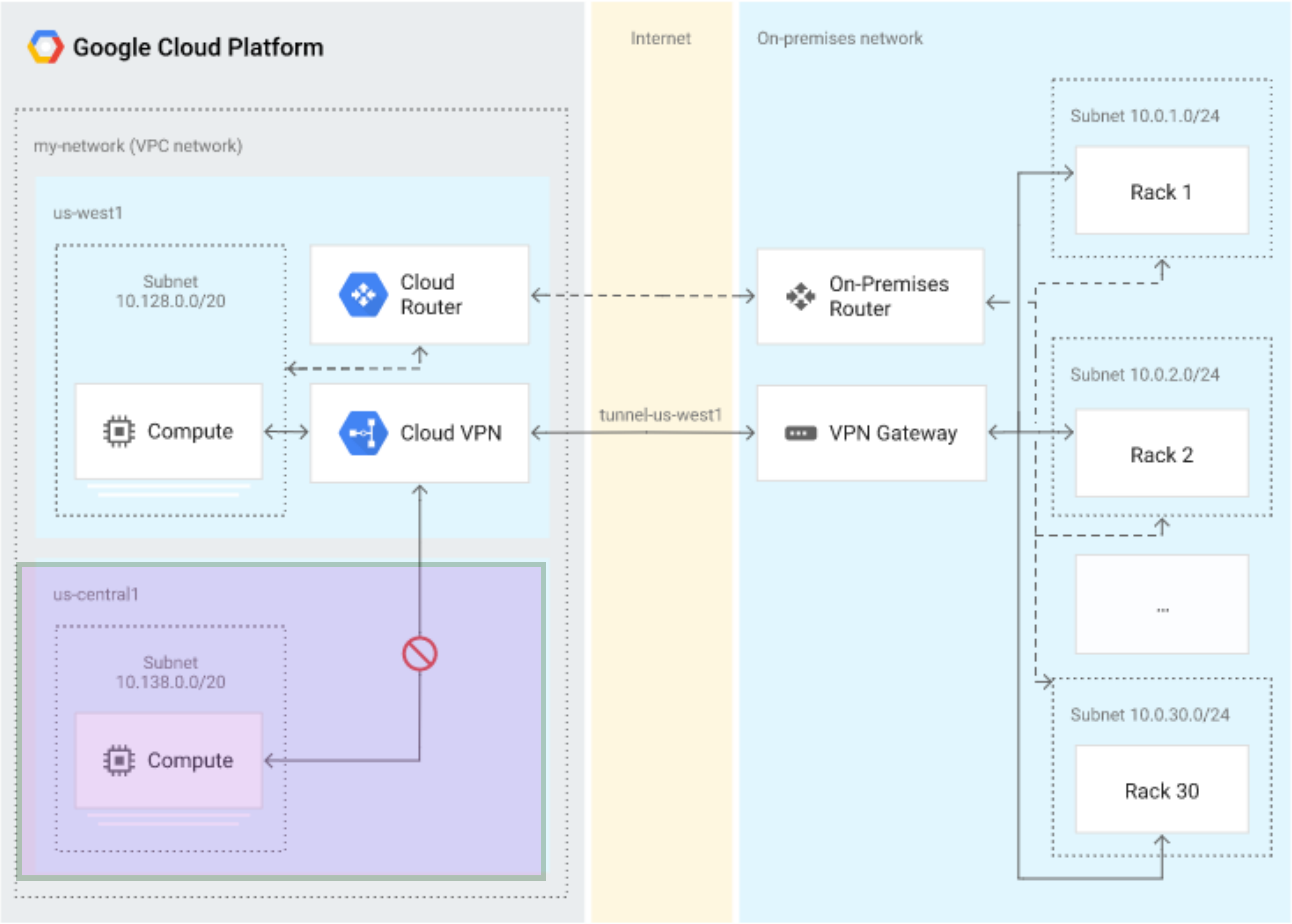
Global Dynamic Routing

Routes in both regions are advertised to the connected network



Regional Dynamic Routing

Routes in the us-centrall region are **not** advertised



3 Interconnection Options

Virtual Private Networks (VPNs) using Cloud Router

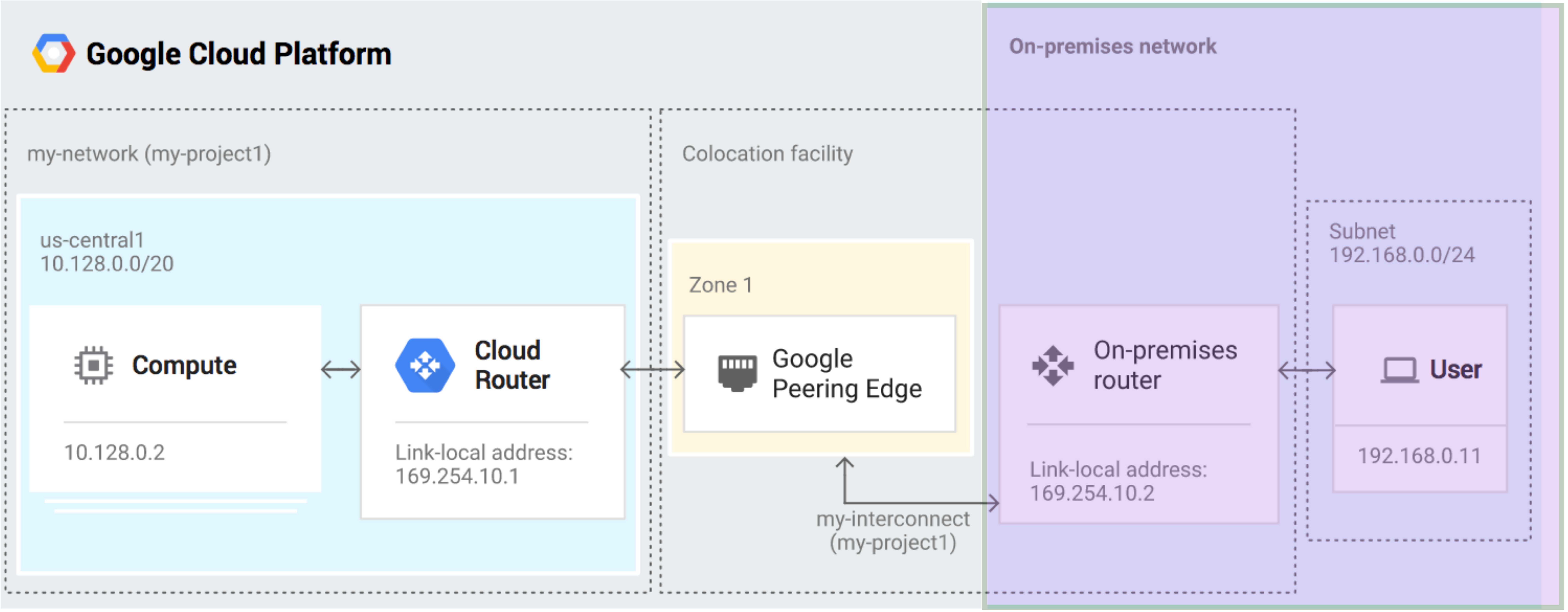
Dedicated Interconnect

Direct and Carrier Peering

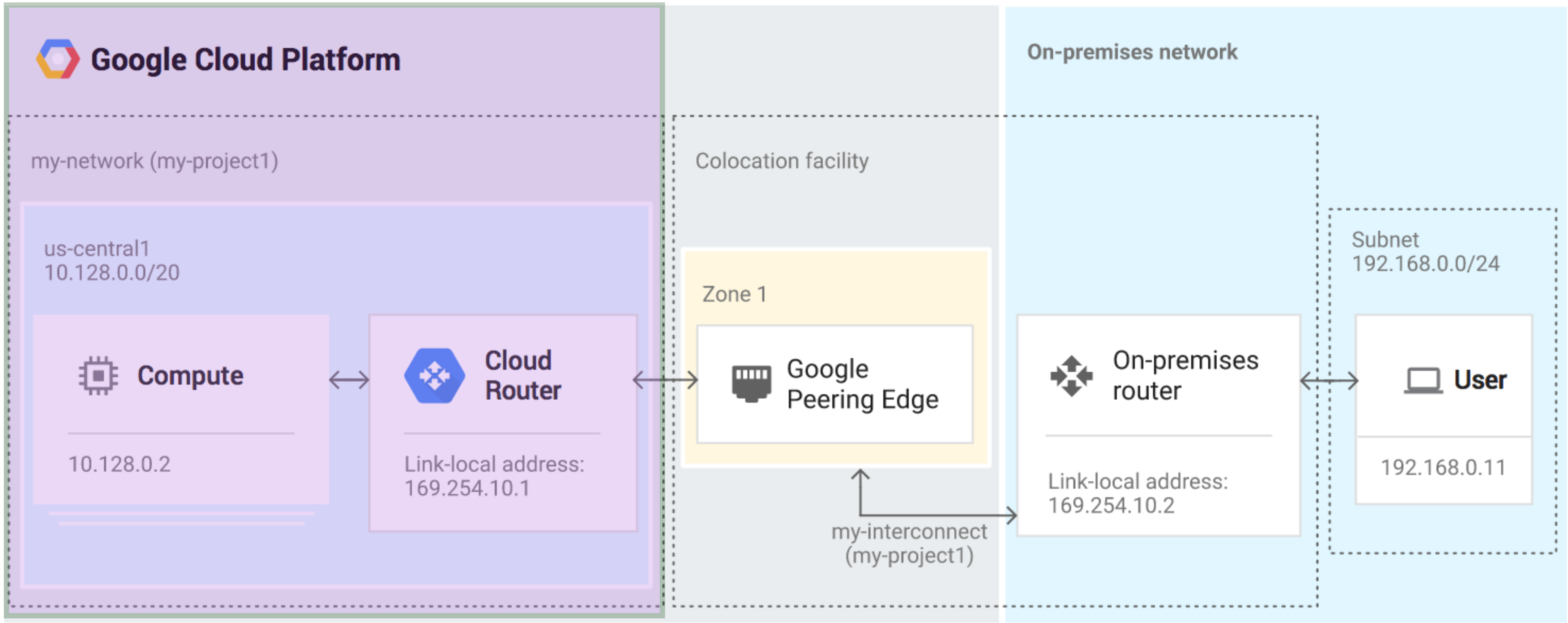
Dedicated Interconnect

- Direct physical connection and RFC 1918 communication between on-premise network and cloud VPC
- Can transfer large amounts of data between networks
- More cost effective than using high bandwidth internet connections or using VPN tunnels
- Capacity of a single connection is 10Gbps
- A maximum of 8 connections supported

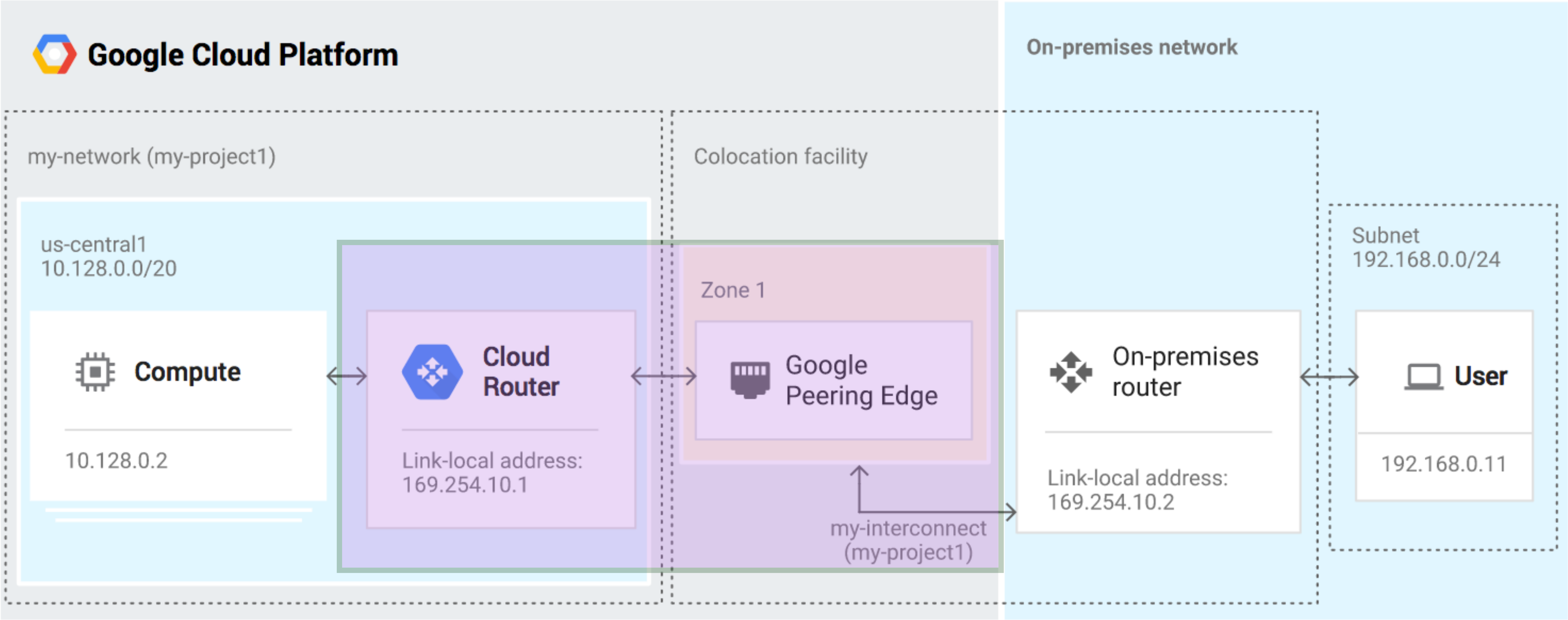
Dedicated Interconnect



Dedicated Interconnect

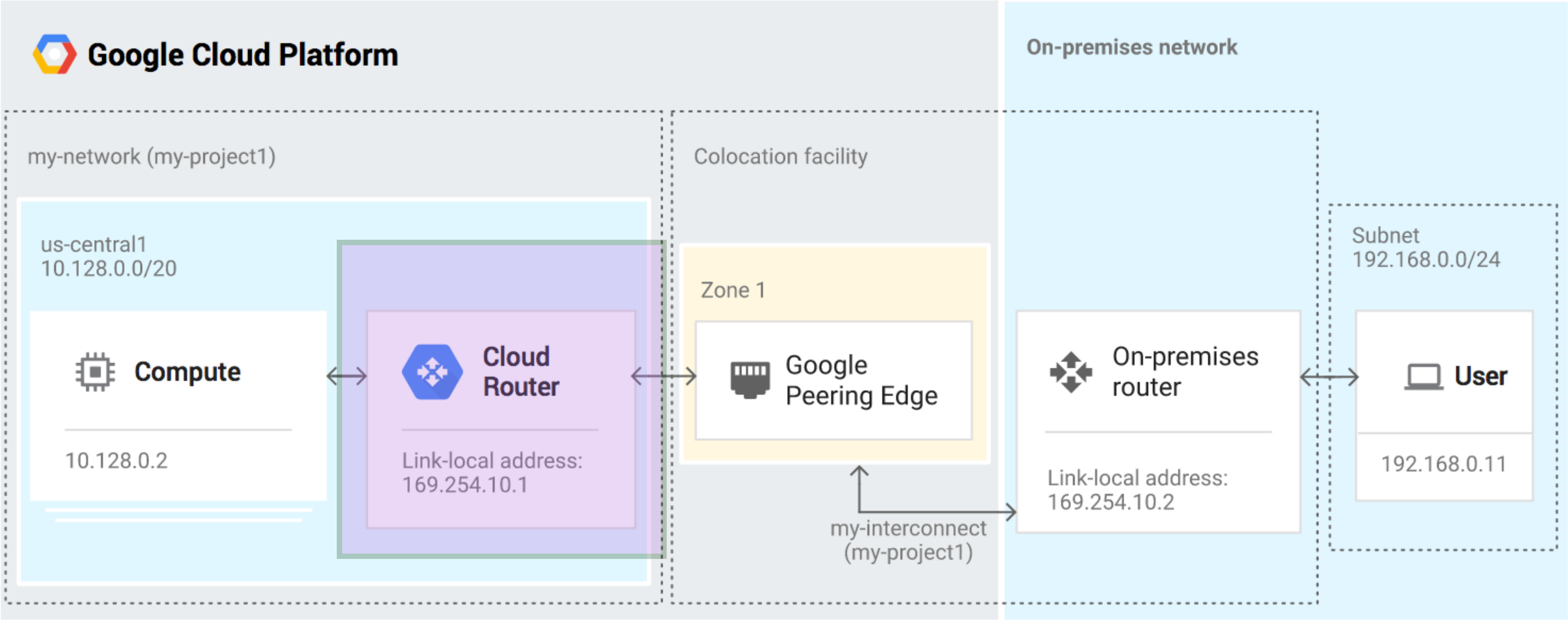


Dedicated Interconnect



Cross connect between the Google network and the on premise router in a common colocation facility

Dedicated Interconnect



Cross connect between the Google network and the on premise router in a common colocation facility

Dedicated Interconnect Benefits

- Does not traverse the public internet. Fewer hops between points so fewer points of failure
- Can use internal IP addresses over a dedicated connection
- Scale connection based on needs up to 80Gbps
- Cost of egress traffic from VPC to on-premise network reduced

3 Interconnection Options

Virtual Private Networks (VPNs) using Cloud Router

Dedicated Interconnect

Direct and Carrier Peering

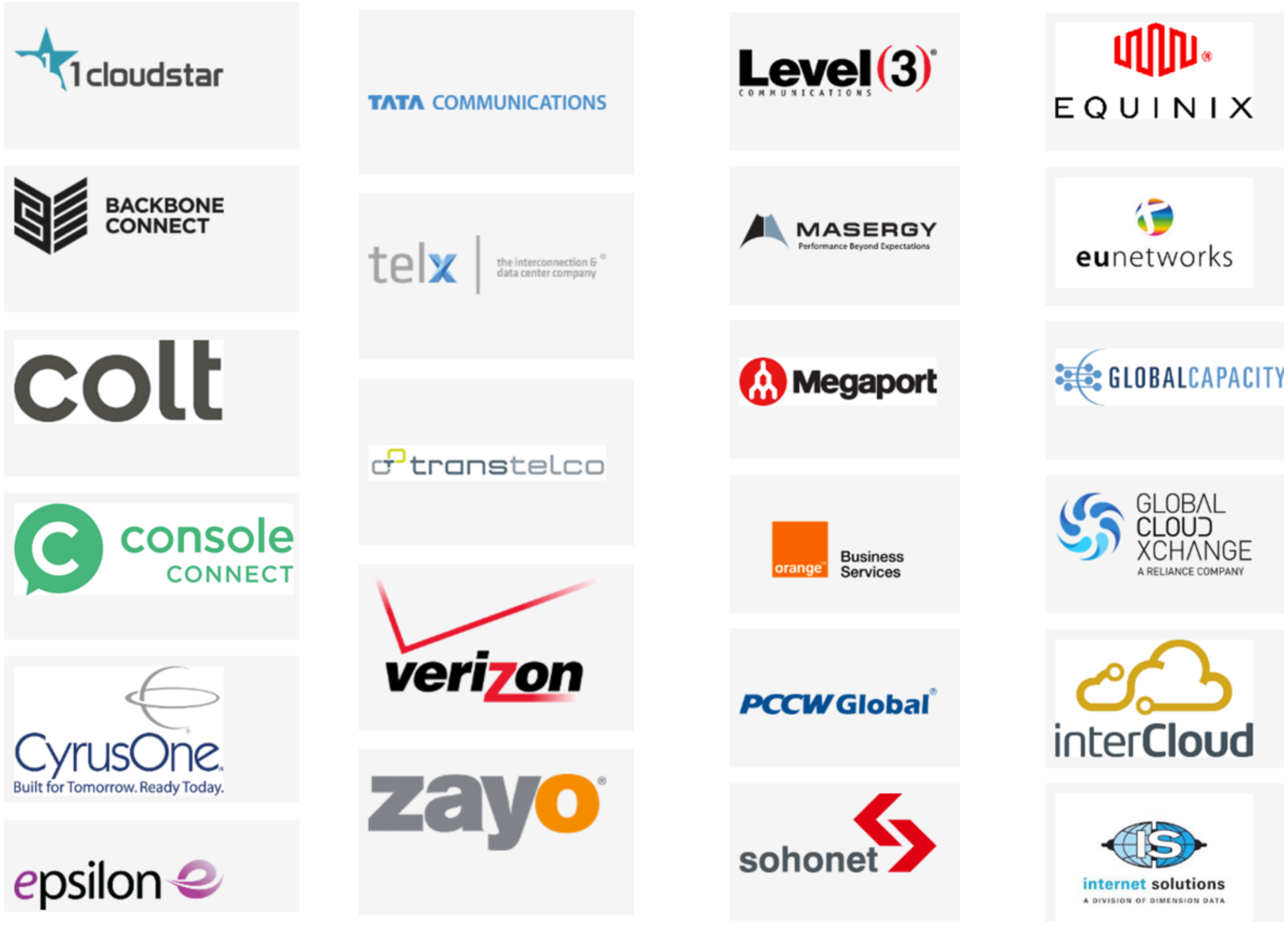
Direct Peering

- Direct connection between on-premise network and Google at Google's **edge network locations**
- BGP routes exchanged for dynamic routing
- Direct peering can be used to reach all of Google's services include the full suite of GCP products
- Special billing rate for GCP egress traffic, other traffic billed at standard GCP rates

Carrier Peering

- Enterprise grade network services connecting your infrastructure to Google using a service provider
- Can get high availability and lower latency using one or more links
- No Google SLA, the SLA depends on the carrier
- Special billing rate for GCP egress traffic, other traffic billed at standard GCP rates

Carrier Peering Providers



Connecting VPC networks

Shared VPC

VPC Network Peering

Connecting VPC networks

Shared VPC

VPC Network Peering

Shared VPC

Used to be called XPN (Cross-Project Networking)

So far - one project, multiple VPCs

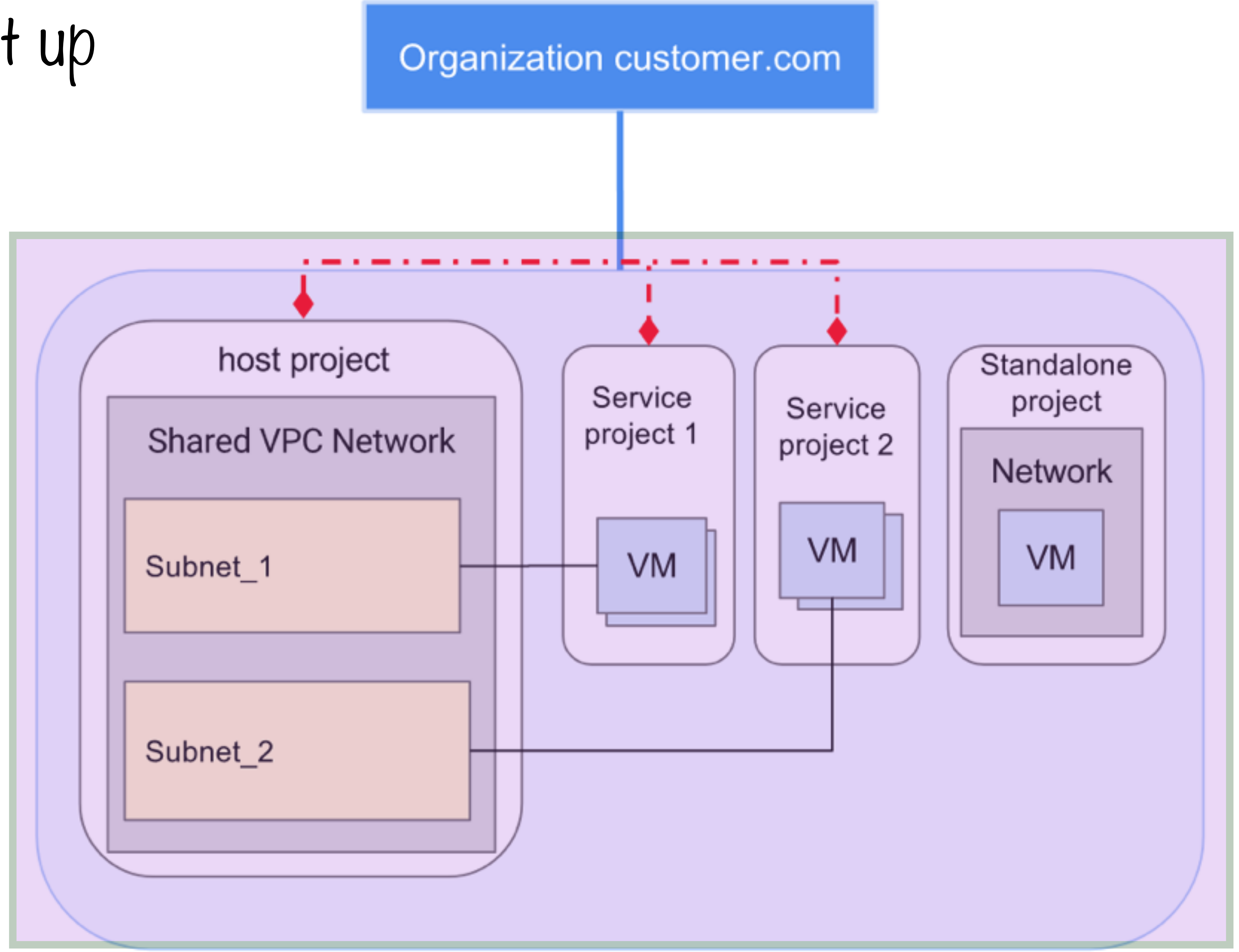
Shared VPC - multiple projects, one VPC

Shared VPC

- So far one project, multiple networks
- Shared VPCs allow cross project networking i.e. multiple projects one network
  - Also called XPN
- Creates a VPC network of RFC1918 IP spaces that associated projects can use
- Firewall rules and policies apply to all projects on the network

Shared VPC

4 projects in this set up

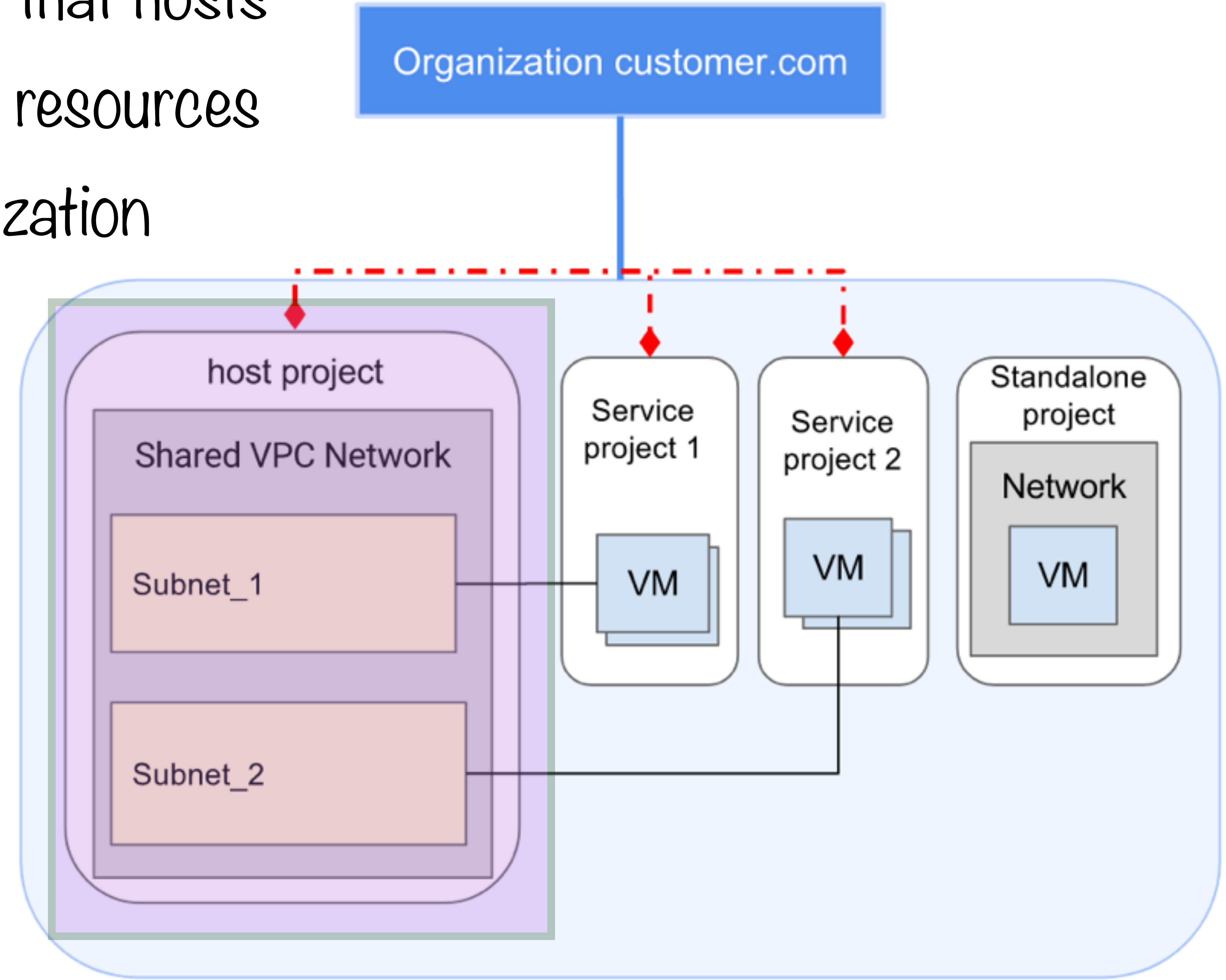


◆-·-◆ Service project connected to host projects and using a shared VPC network

—— VM created in a subnetwork

Shared VPC

Host Project — Project that hosts sharable VPC networking resources within a Cloud Organization

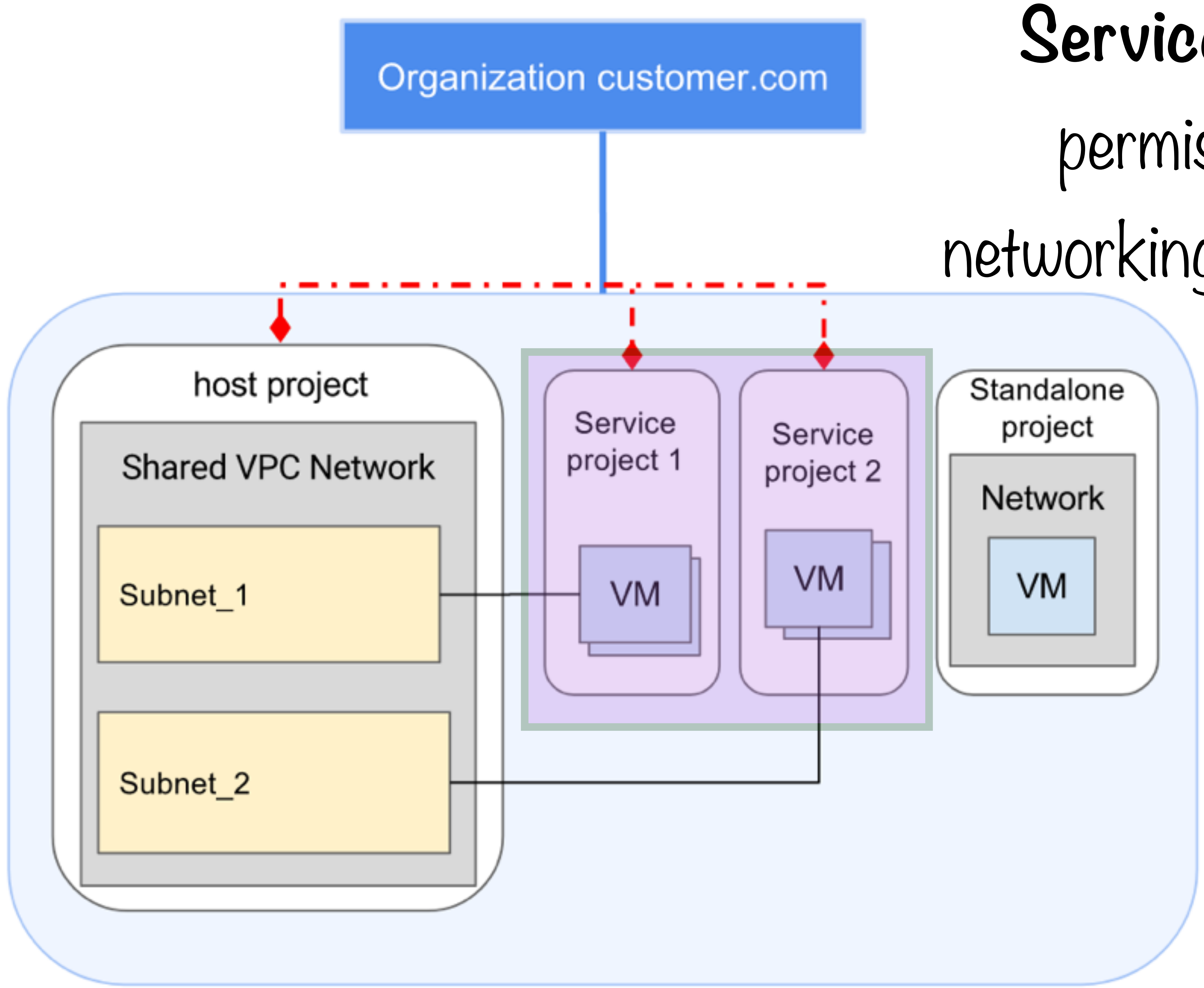


◆-·-◆ Service project connected to host projects and using a shared VPC network

—— VM created in a subnetwork

Shared VPC

Service project — Project that has permission to use the shared VPC networking resources from the host project

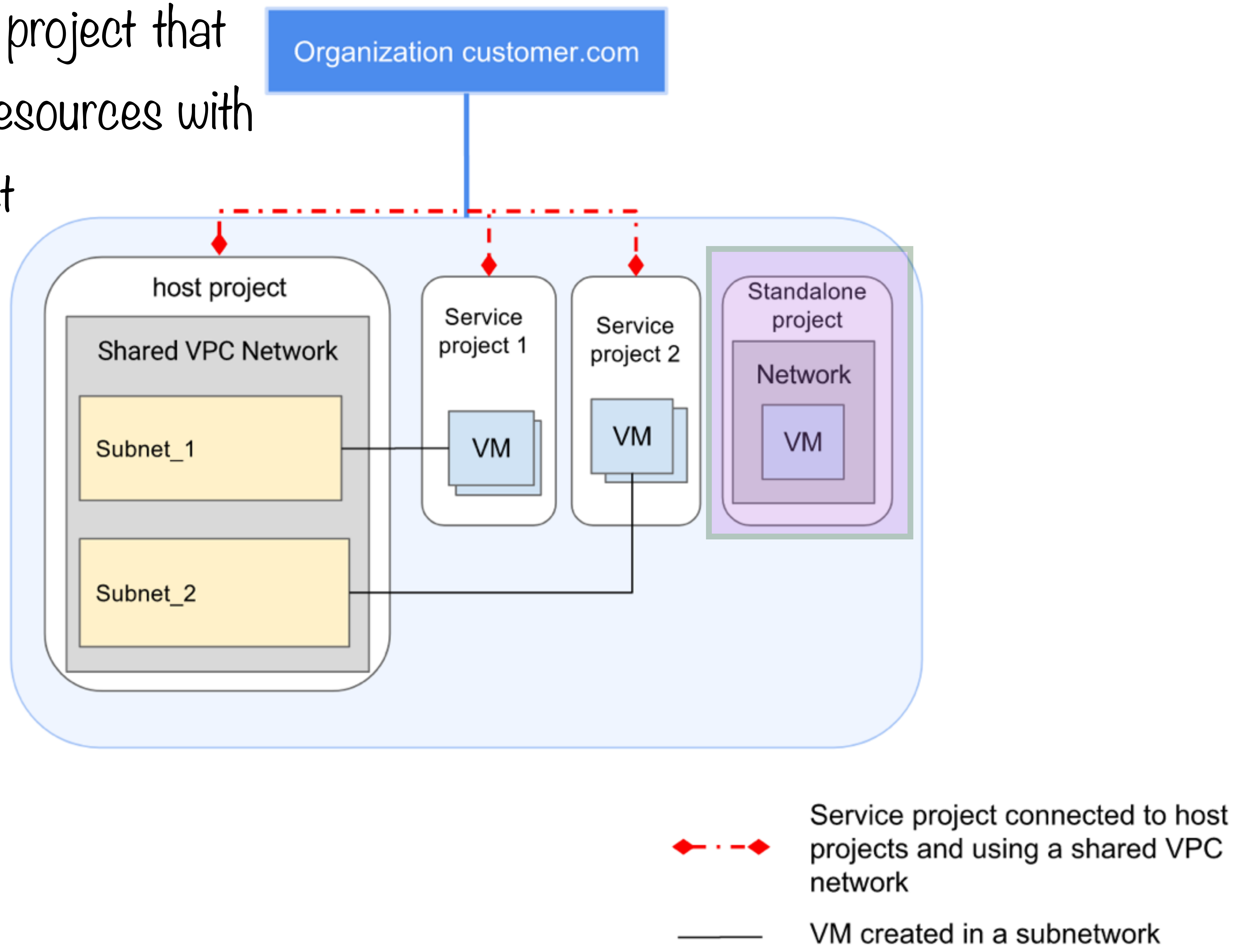


◆-·-◆ Service project connected to host projects and using a shared VPC network

—— VM created in a subnetwork

Shared VPC

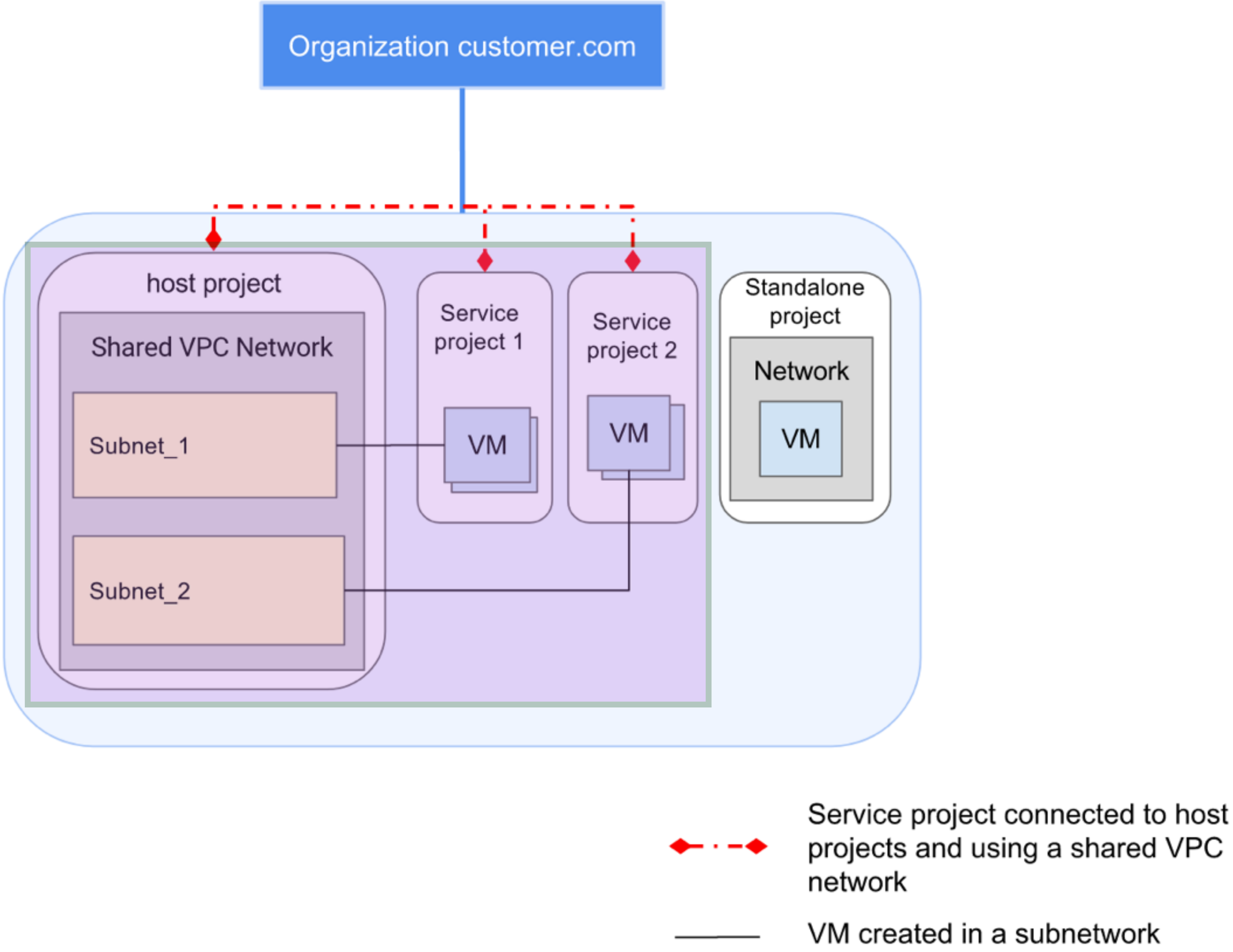
Standalone project — A project that does not share networking resources with any other project



Organization customer.com
host project
Shared VPC Network
Subnet_1
Subnet_2
Service project 1
VM
Service project 2
VM
Standalone project
Network
VM
Service project connected to host projects and using a shared VPC network
VM created in a subnetwork

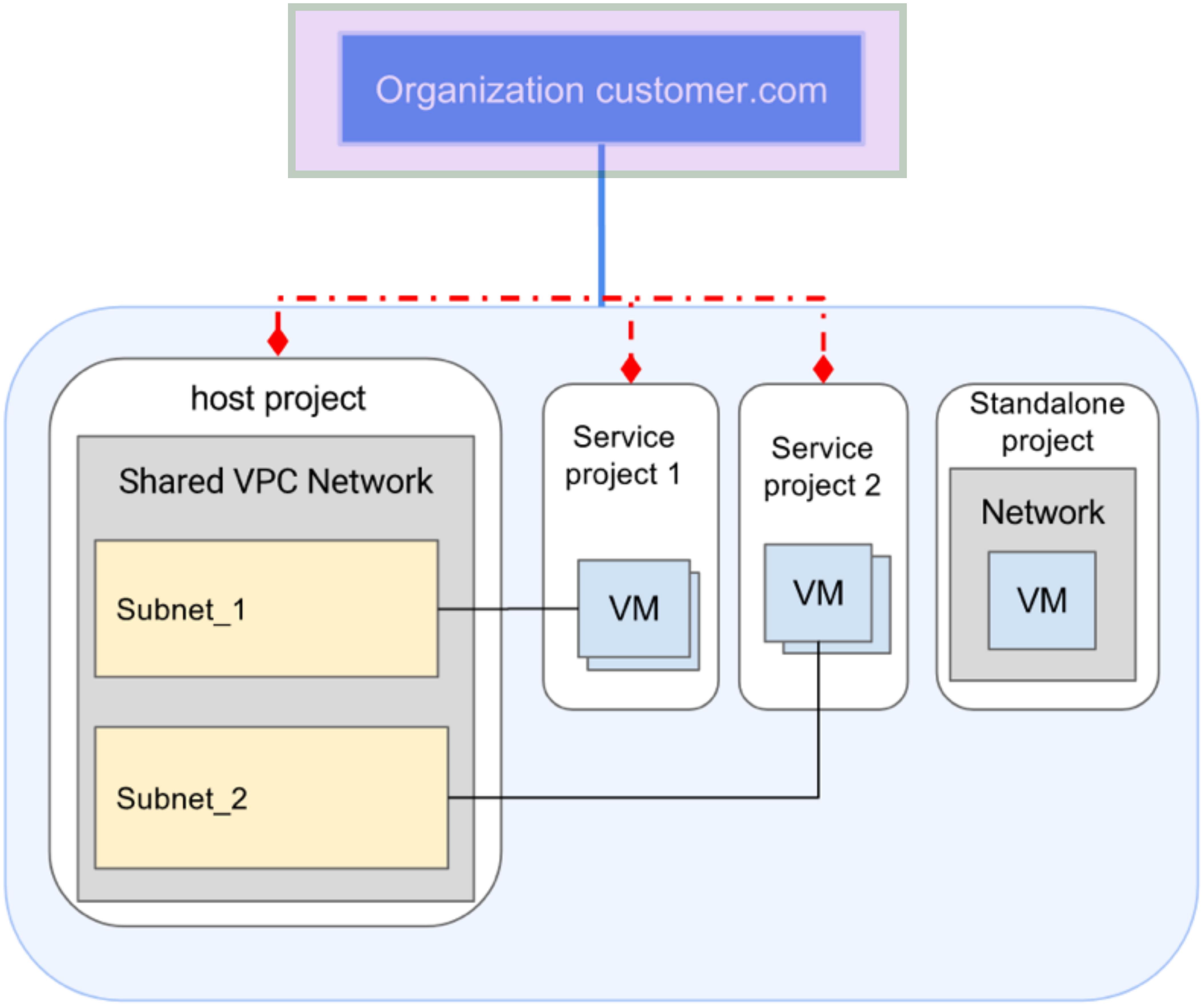
Shared VPC

Shared VPC network — A VPC network owned by the host project and shared with one or more service projects in the Cloud Organization


Organization customer.com
host project
Shared VPC Network
Subnet_1
Subnet_2
Service project 1
VM
Service project 2
VM
Standalone project
Network
VM
Service project connected to host projects and using a shared VPC network
VM created in a subnetwork

Shared VPC

Organization — The Cloud Organization is the top level in the Cloud Resource Hierarchy and the top-level owner of all the projects and resources created under it. A given host project and its service projects must be under the same Cloud Organization.



A given host project and its service projects must be under the **same Cloud Organization.**

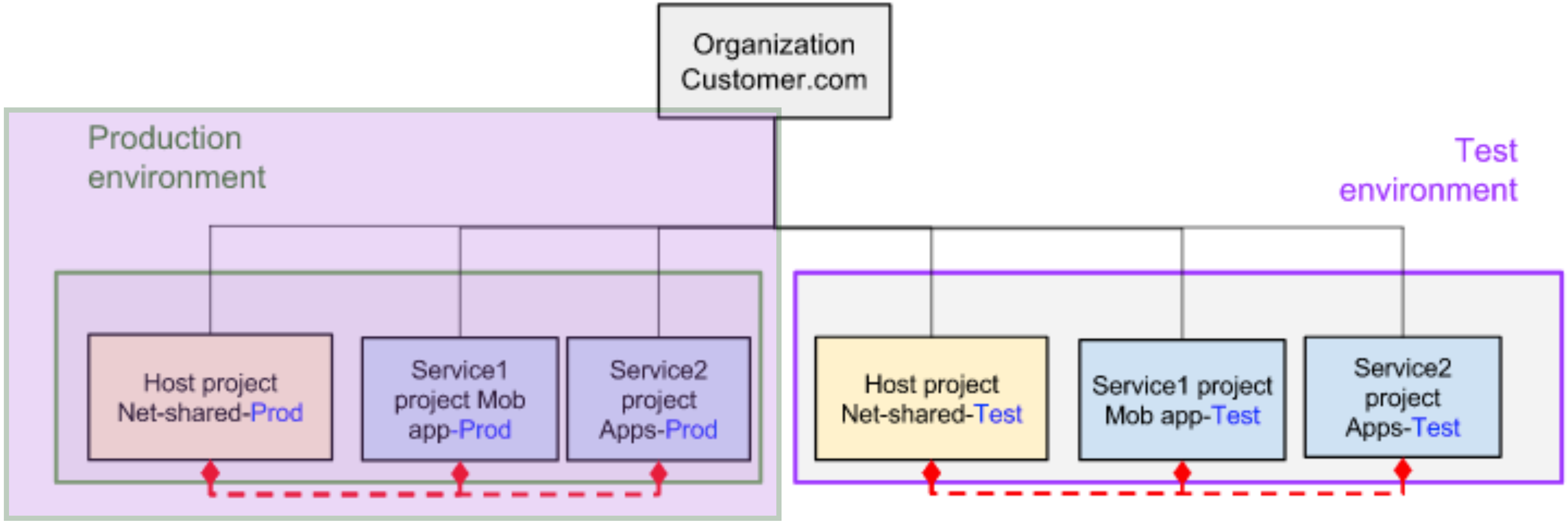
◆- · -◆ Service project connected to host projects and using a shared VPC network

—— VM created in a subnetwork

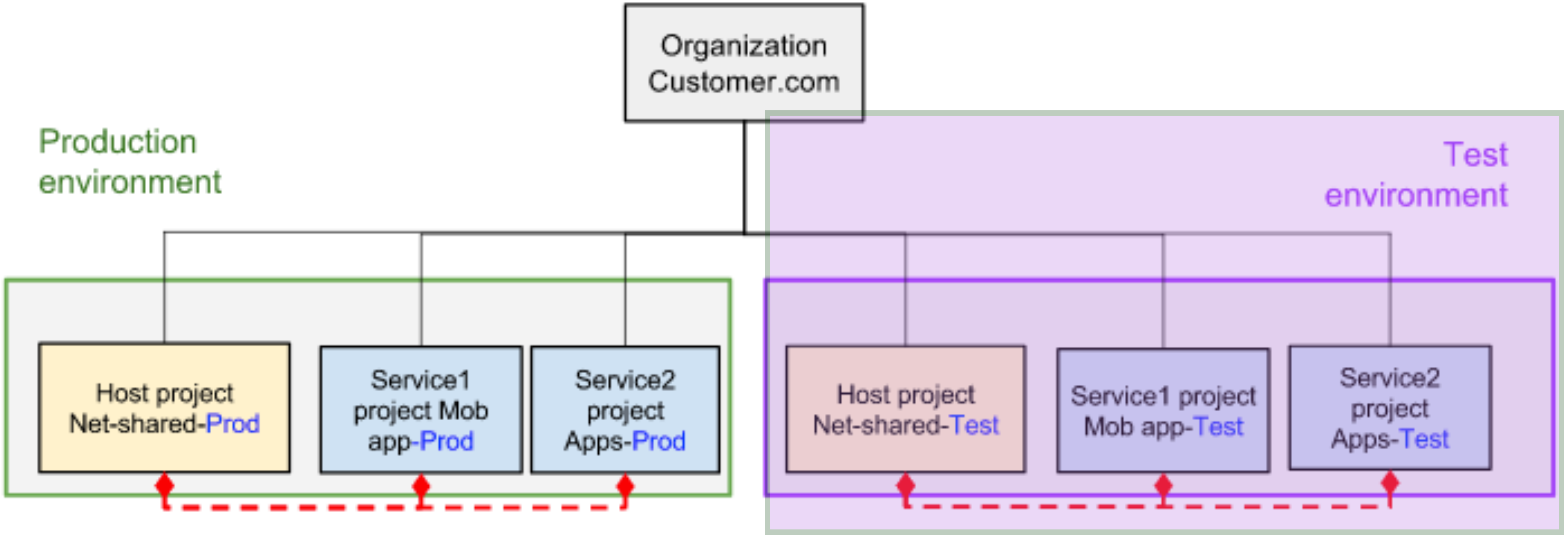
Host and Service projects

- A service project can only be associated with a single host
- A project cannot be a host as well as a service project at the same time
- Instances in a project can only be assigned external IPs from the same project
- Existing projects can use shared VPC networks
- Instances on a shared VPC need to be created explicitly for the VPC

Multiple Shared VPCs



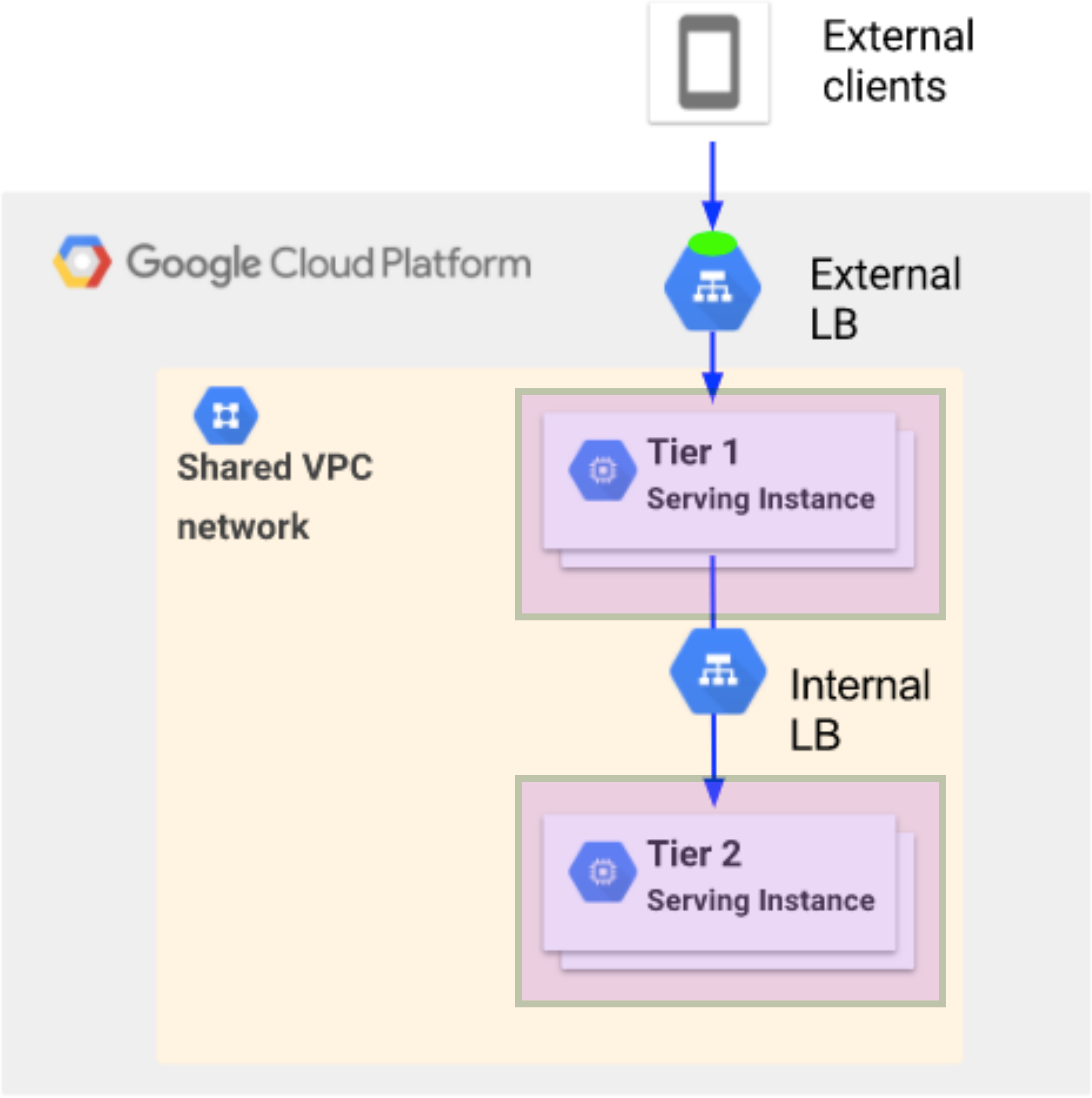
Multiple Shared VPCs



Use Cases: Two Tier Web Service

A different team owns the Tier 1 and Tier 2 services

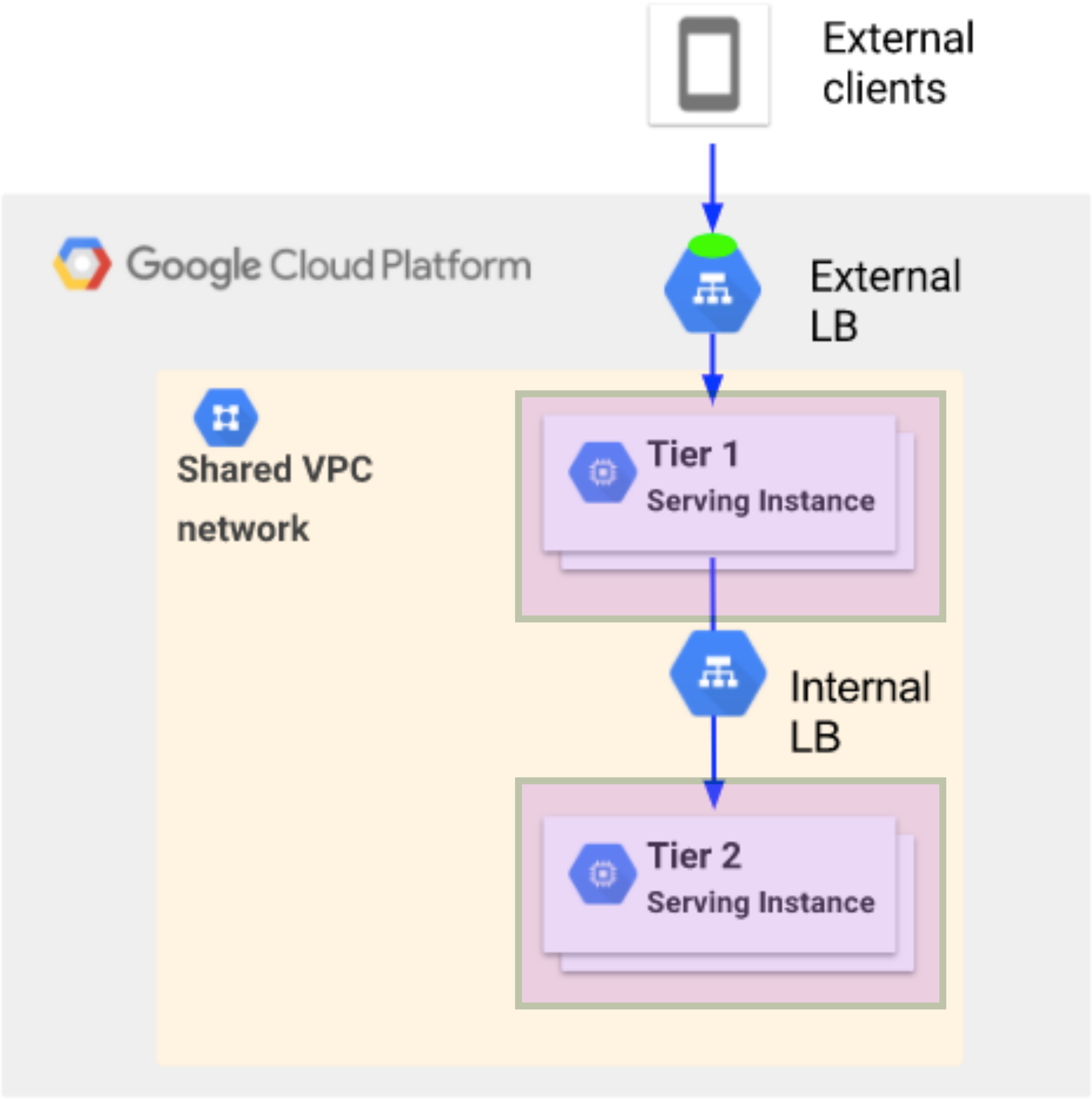
Each team can deploy and operate its services independently



Use Cases: Two Tier Web Service

Each project is billed separately

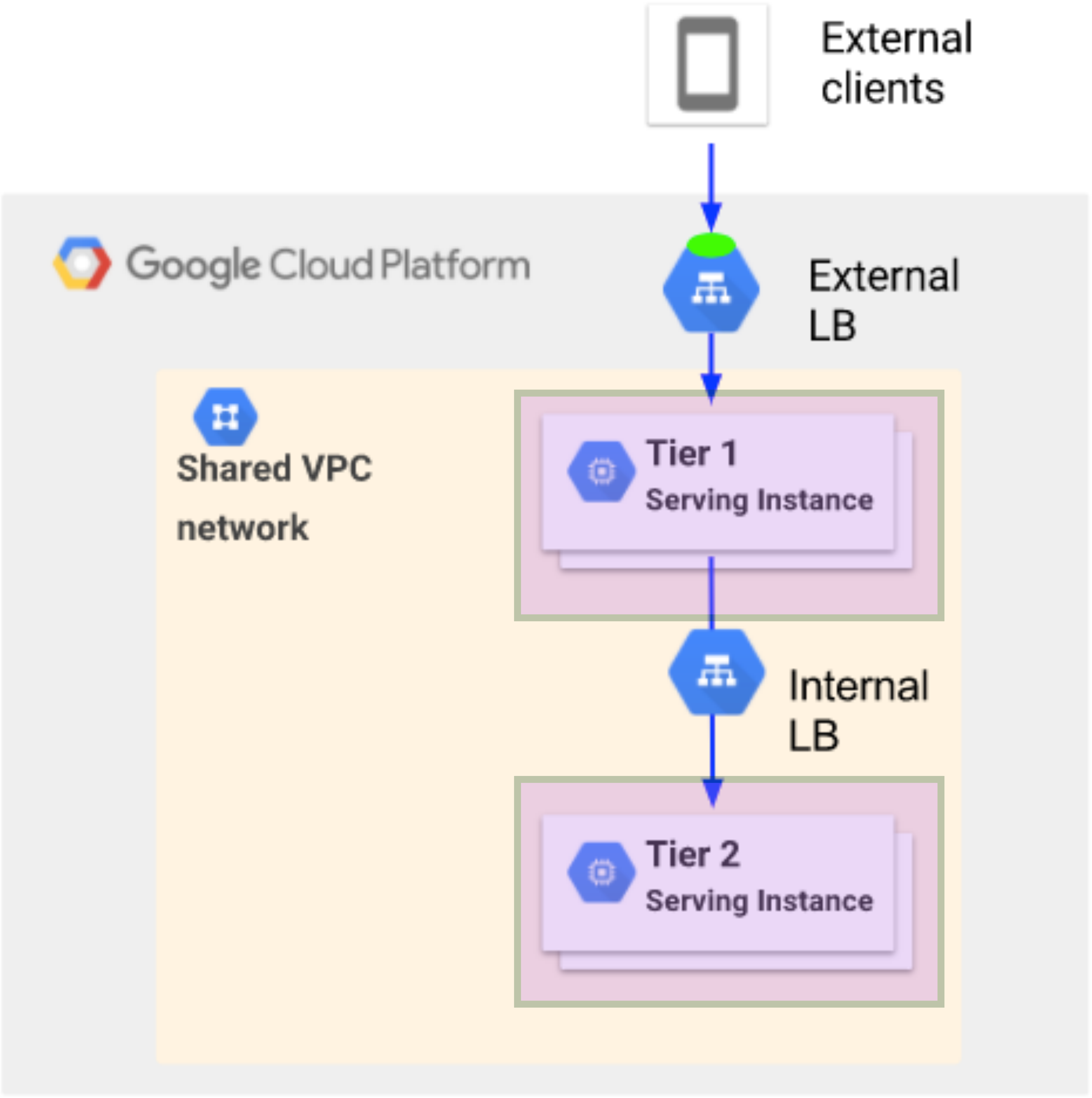
Each project admin can manage their own resources



Use Cases: Two Tier Web Service

A single group of network and security admins can be responsible for the shared VPC

They are in charge of network connectivity and security rules for the organization as a whole



Connecting VPC networks

Shared VPC

VPC Network Peering

VPC Network Peering

- Allows private RFC1918 connectivity across two VPC networks
- Networks can be in the same or in different projects
- Build SaaS ecosystems in GCP, services can be made available privately across different VPC networks
- Useful for organizations:
  - With several network administrative domains
  - Which want to peer with other organizations on the GCP

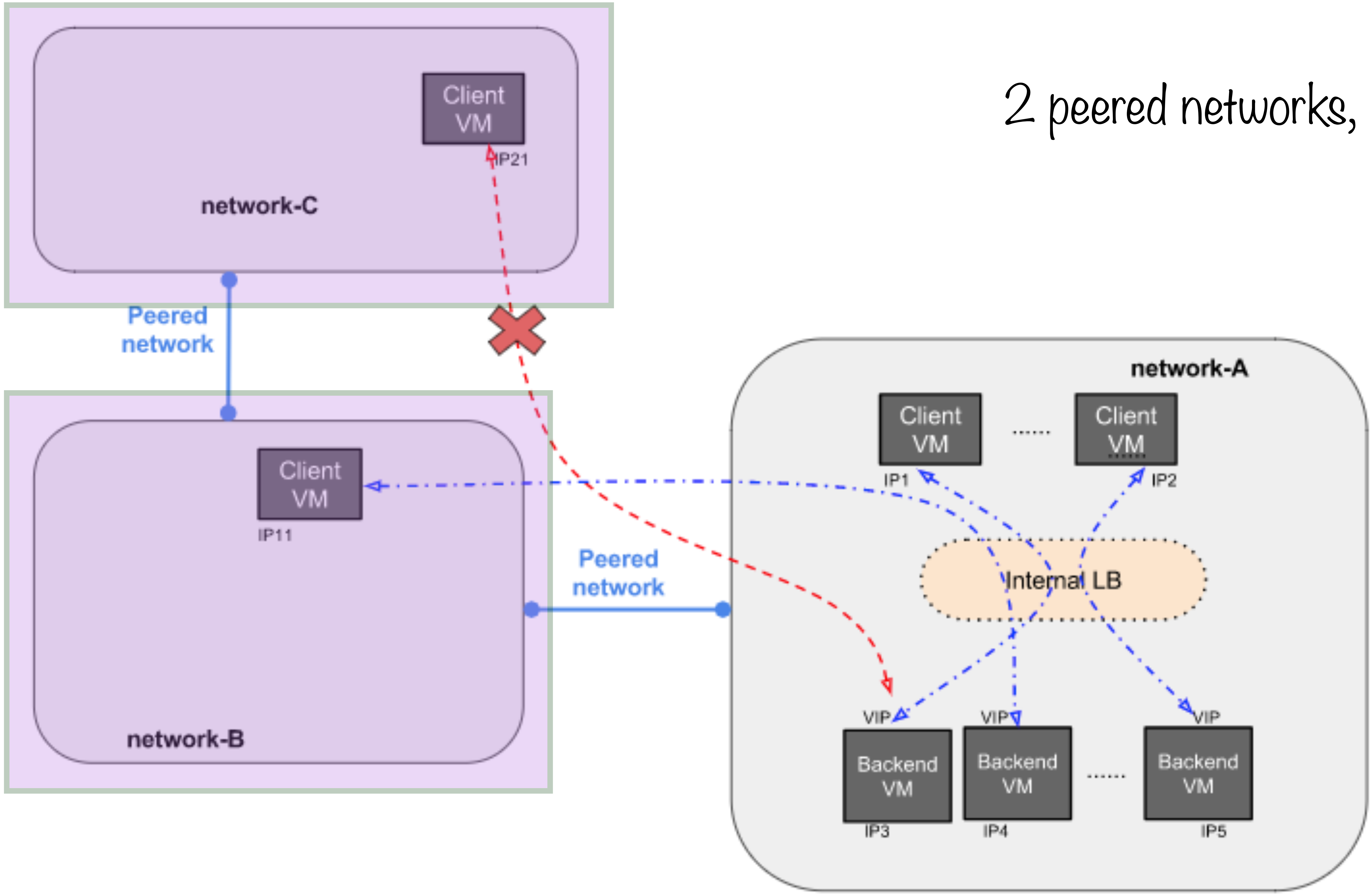
VPC Network Peering Benefits

- Lower latency as compared with public IP networking
- Better security since services need not expose an external IP address
- Using internal IPs for traffic avoids egress bandwidth pricing on the GCP

VPC Network Peering Properties

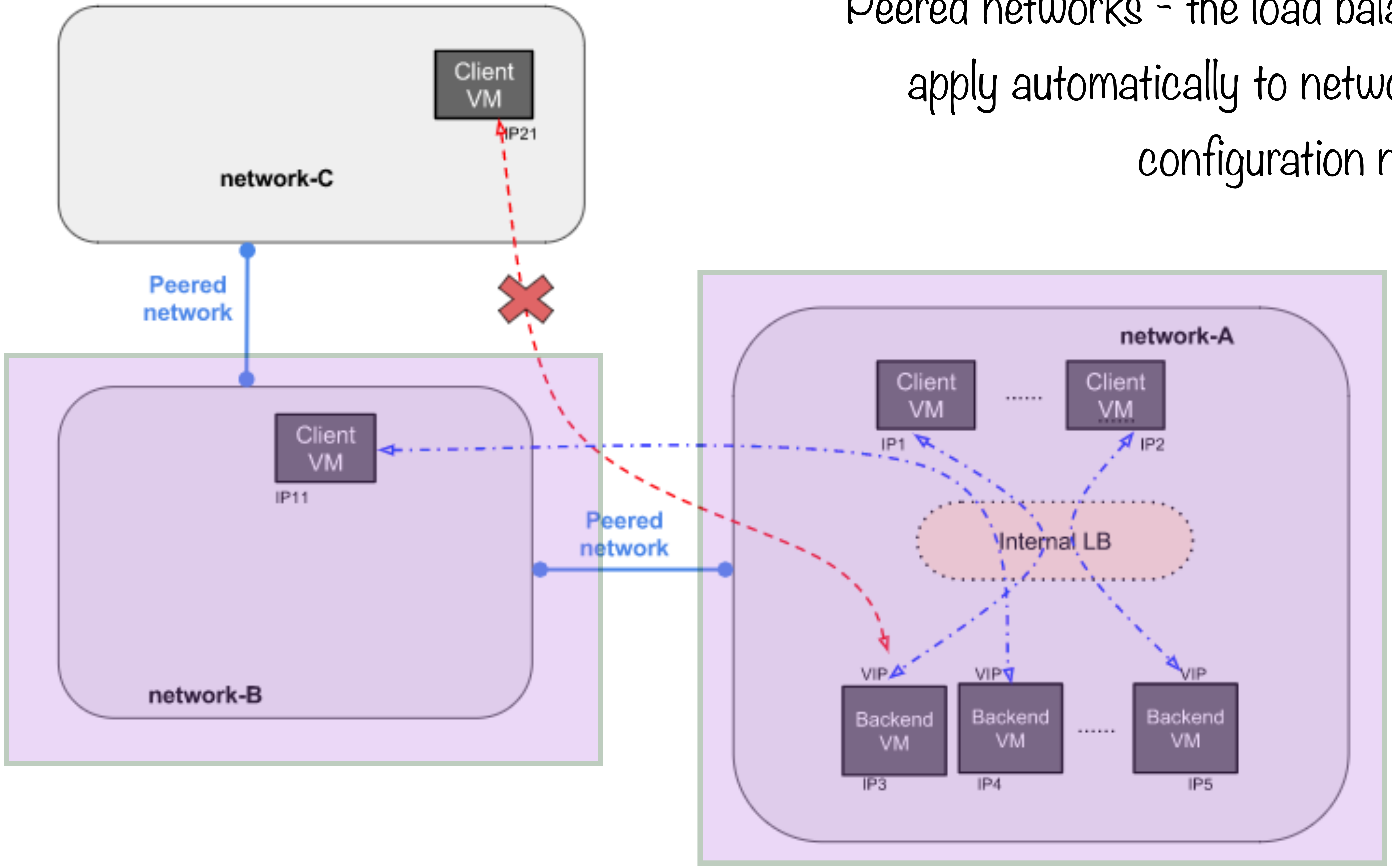
- Peered networks are **administratively separate** - routes, firewalls, VPNs and traffic management applied independently
- One VPC can peer with multiple networks with a limit of 25
- Only directly peered networks can communicate

Peered Networks and Internal Load Balancing



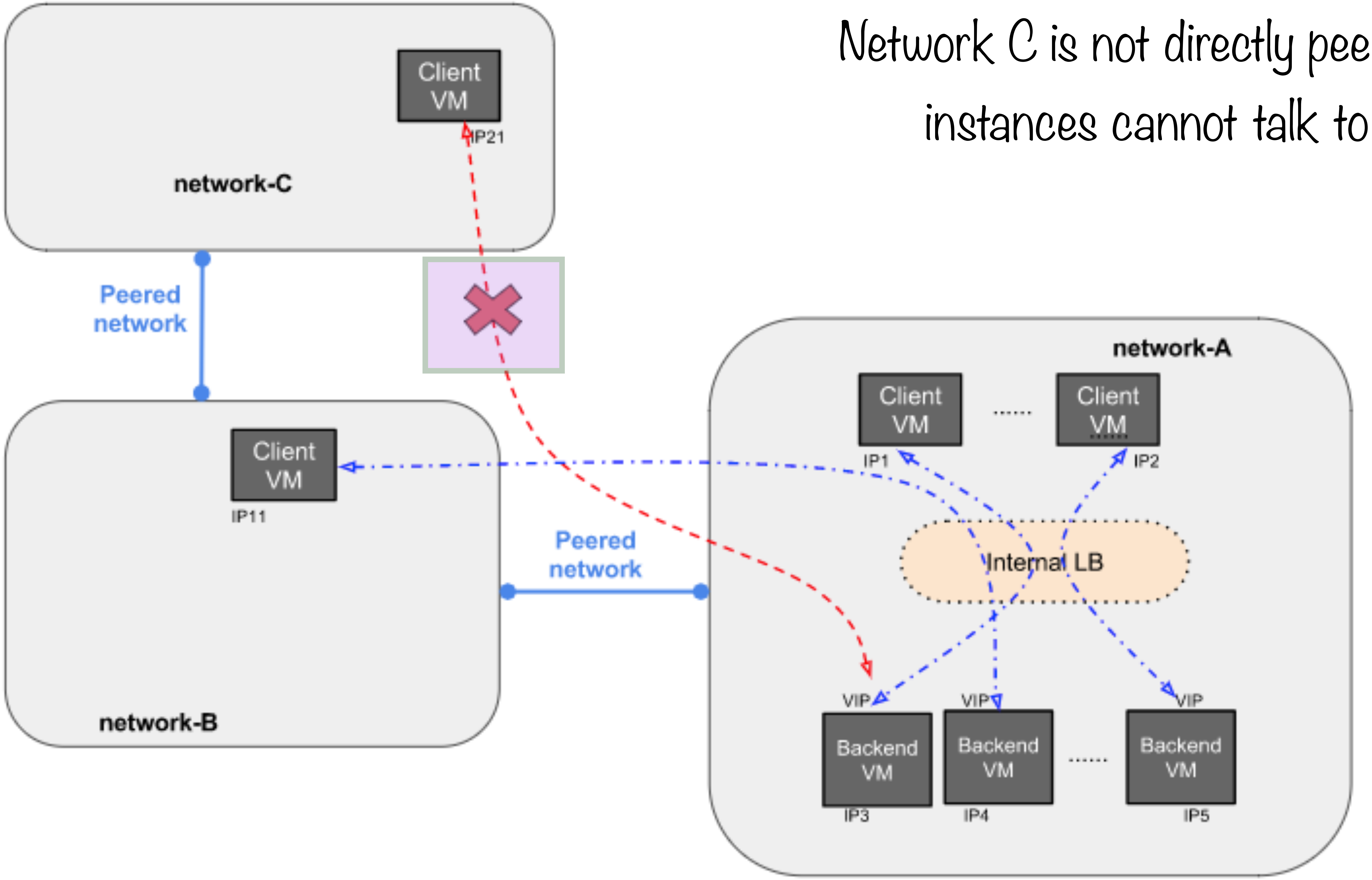
2 peered networks, no load balancer

Peered Networks and Internal Load Balancing



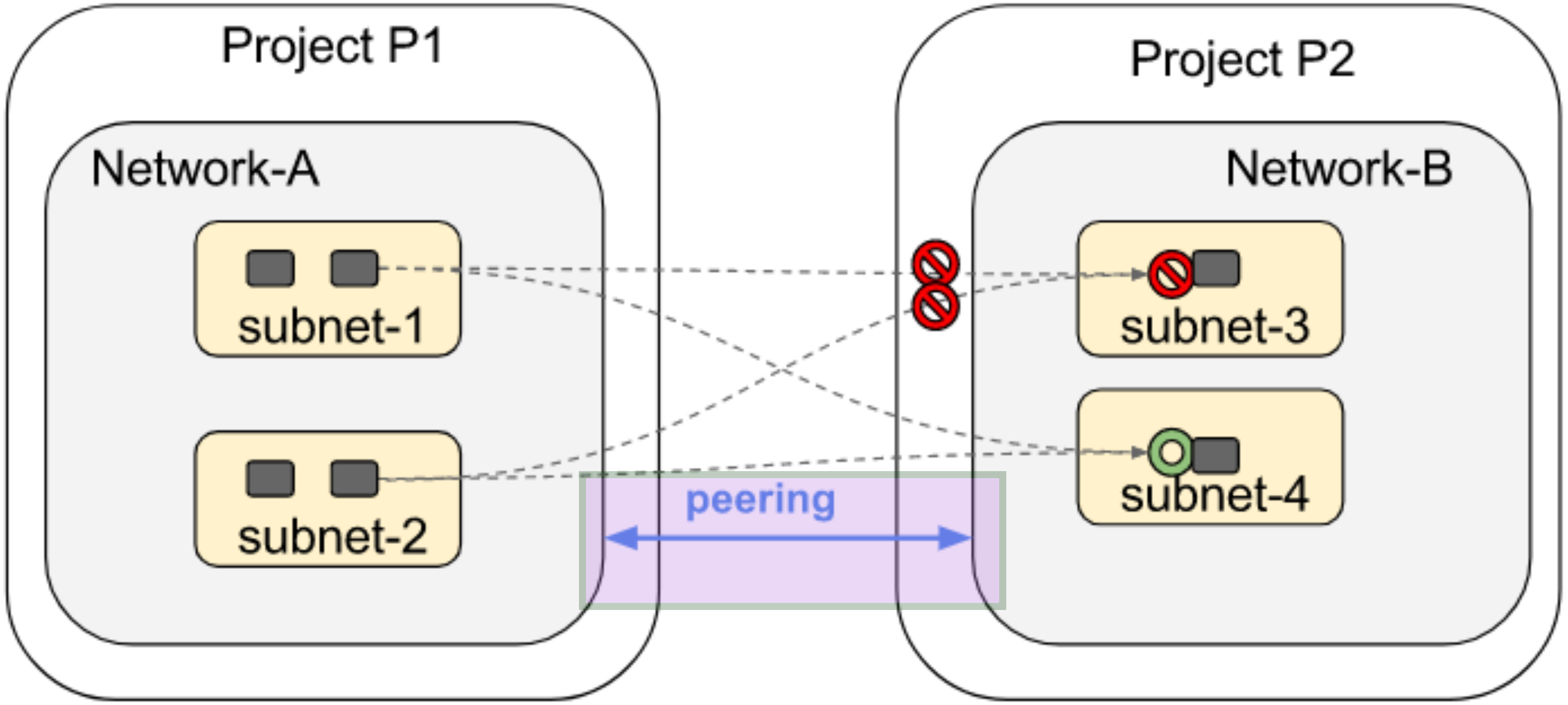
Peered networks - the load balancer in network A will apply automatically to network B, no additional configuration needed

Peered Networks and Internal Load Balancing



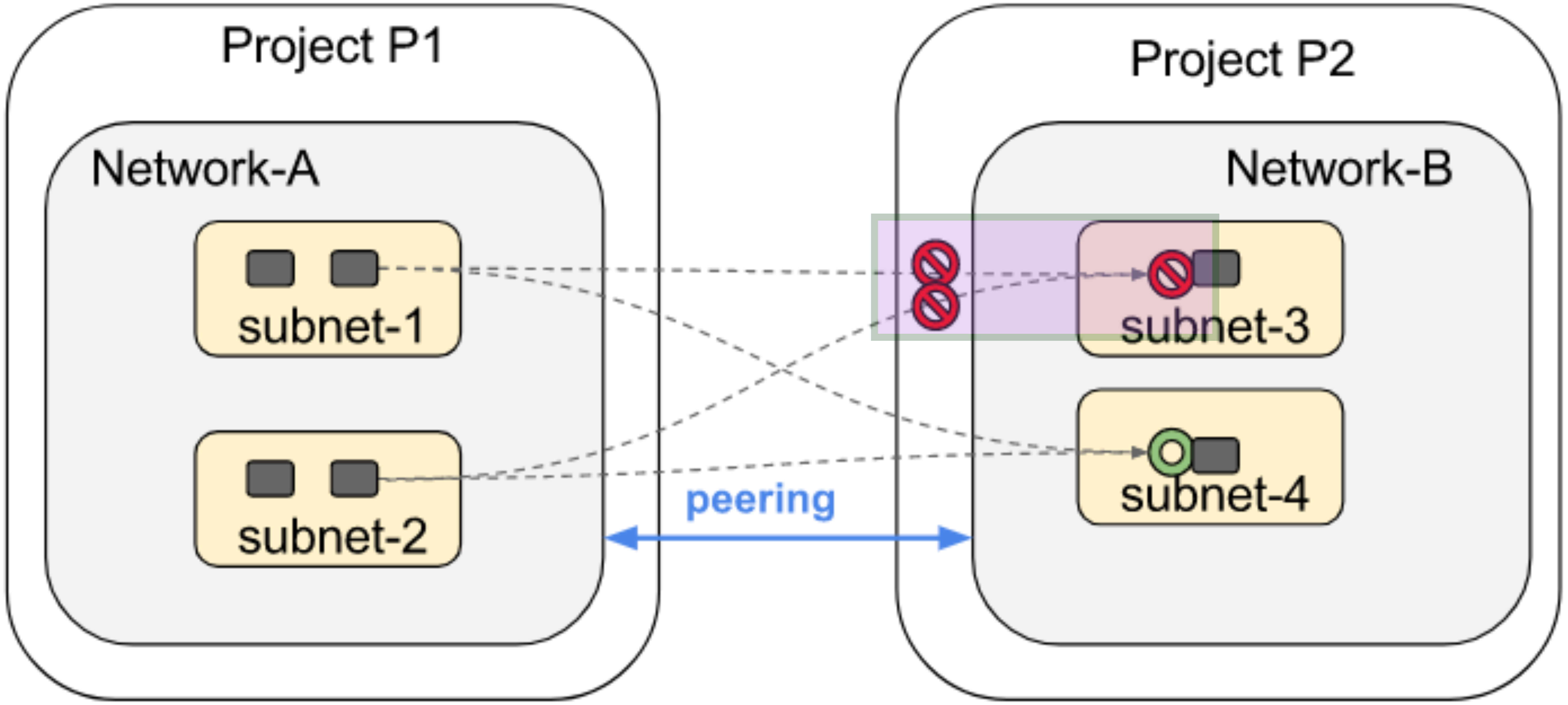
Network C is not directly peered with network A, the instances cannot talk to each other directly

Peered Networks and Firewalls



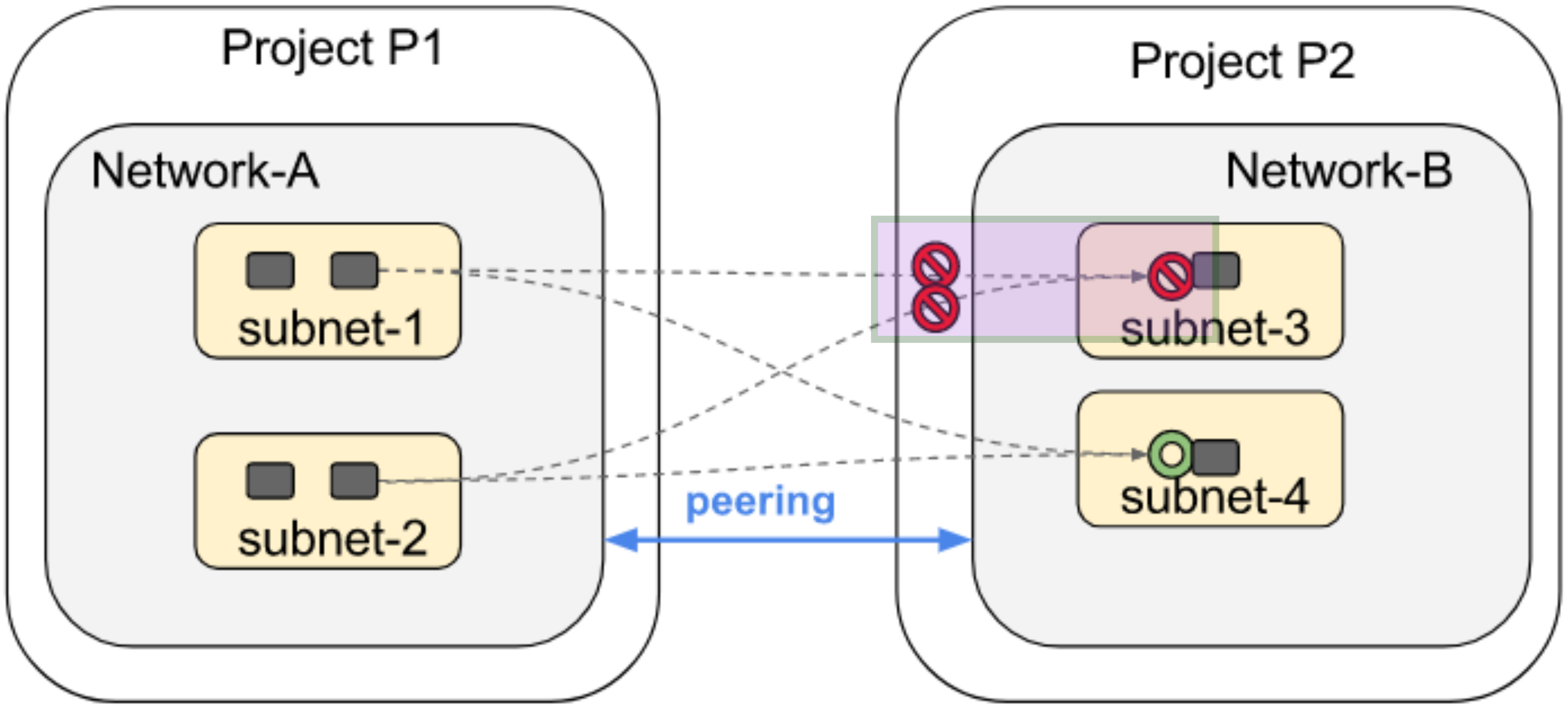
Firewall rules are configured separately in each network

Peered Networks and Firewalls



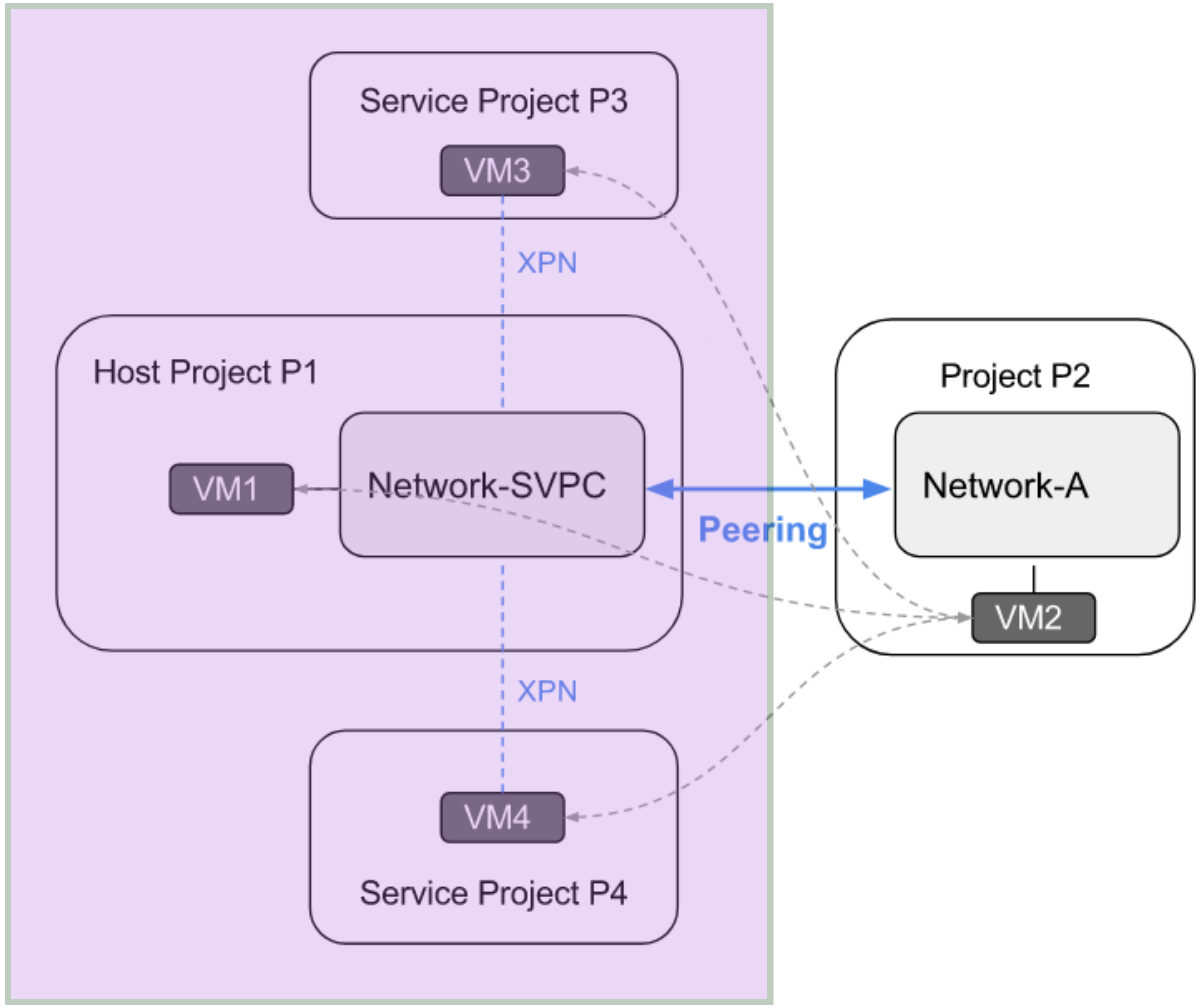
Ingress firewalls can **prevent** traffic from subnet-1 and subnet-2 from reaching subnet-3

Peered Networks and Firewalls



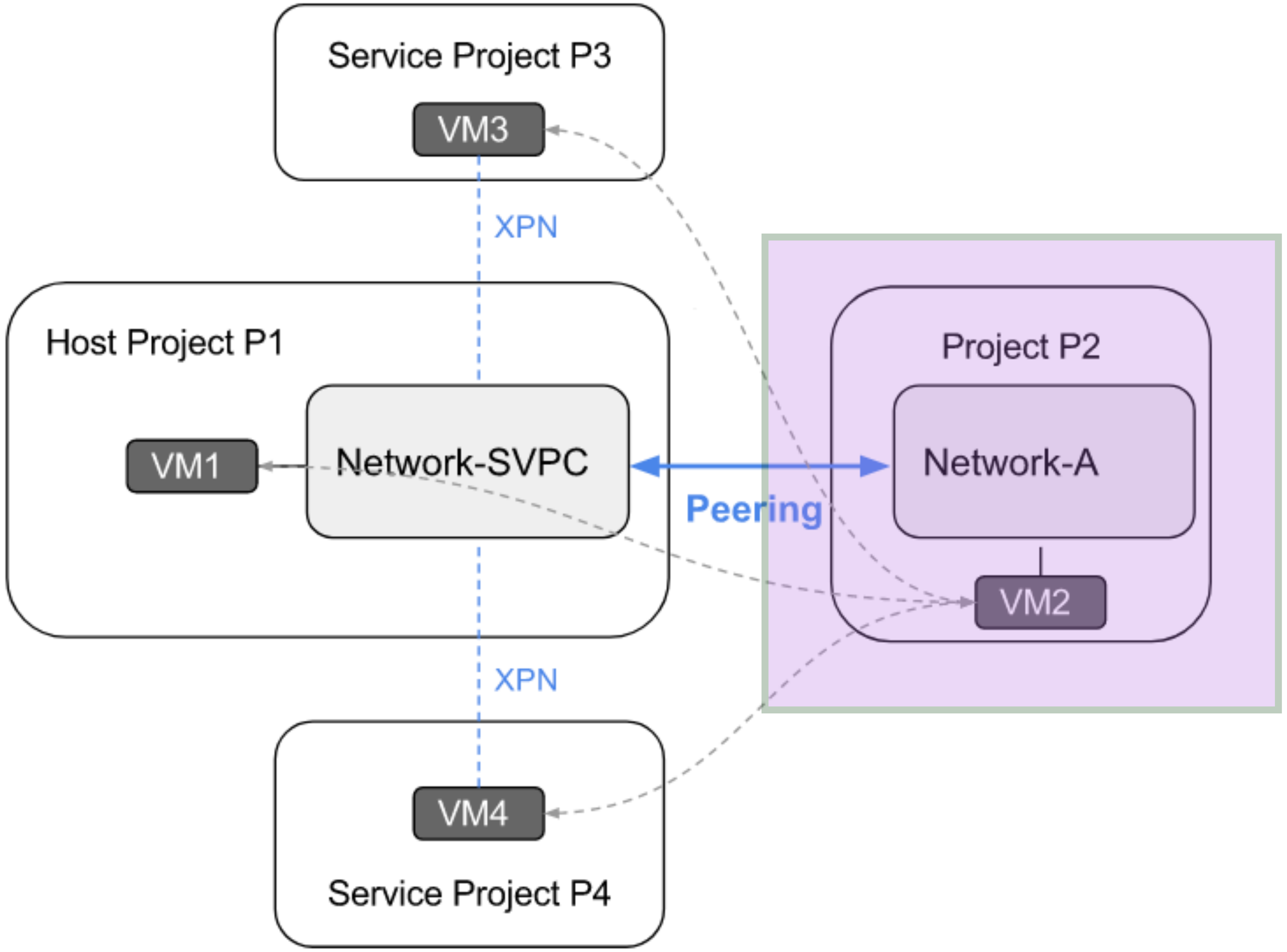
Peering networks allows access to all instances in the network - firewalls are the only way to block access to certain instances

Peered Networks and Shared VPCs



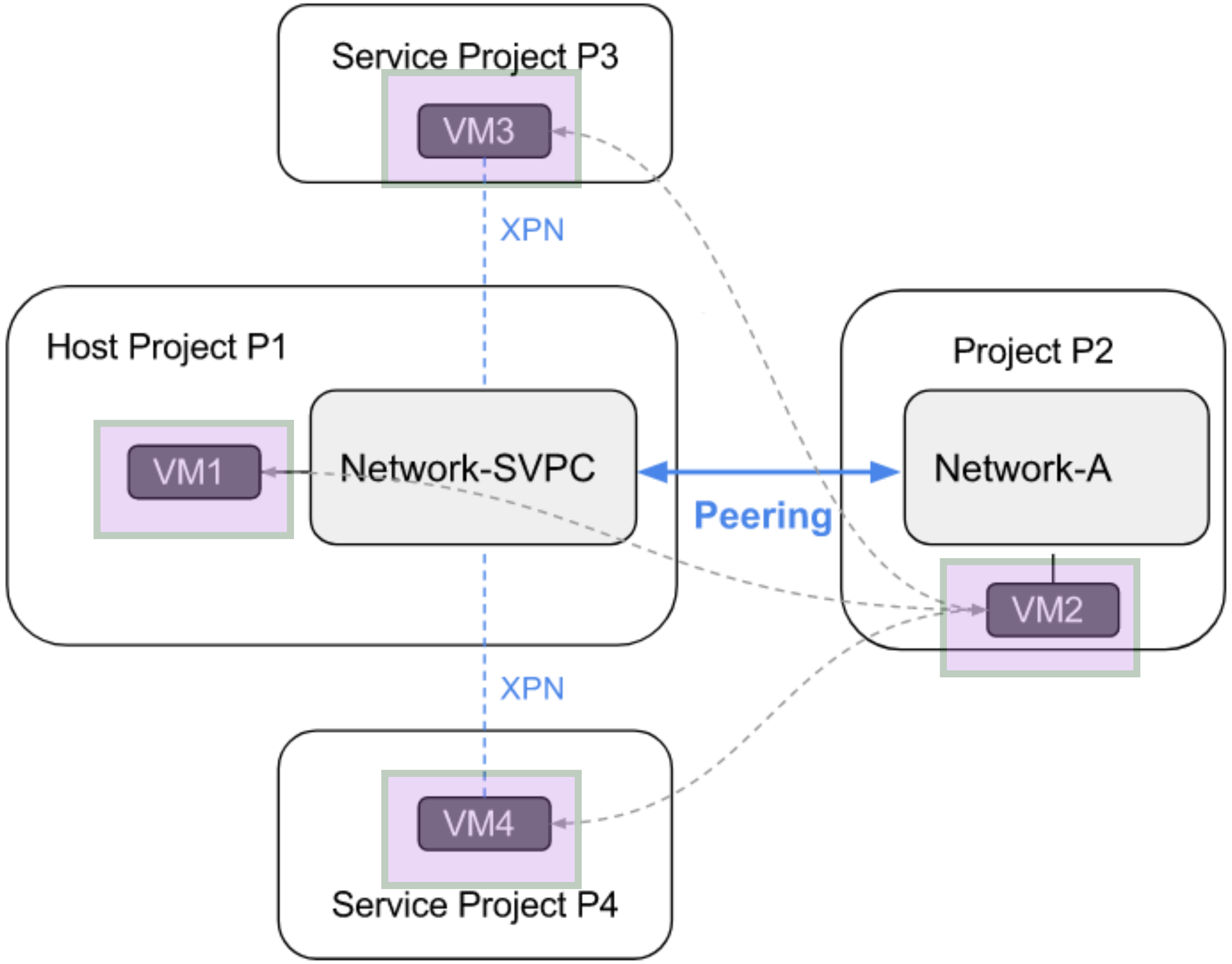
A shared VPC with one host and 2 service projects

Peered Networks and Shared VPCs



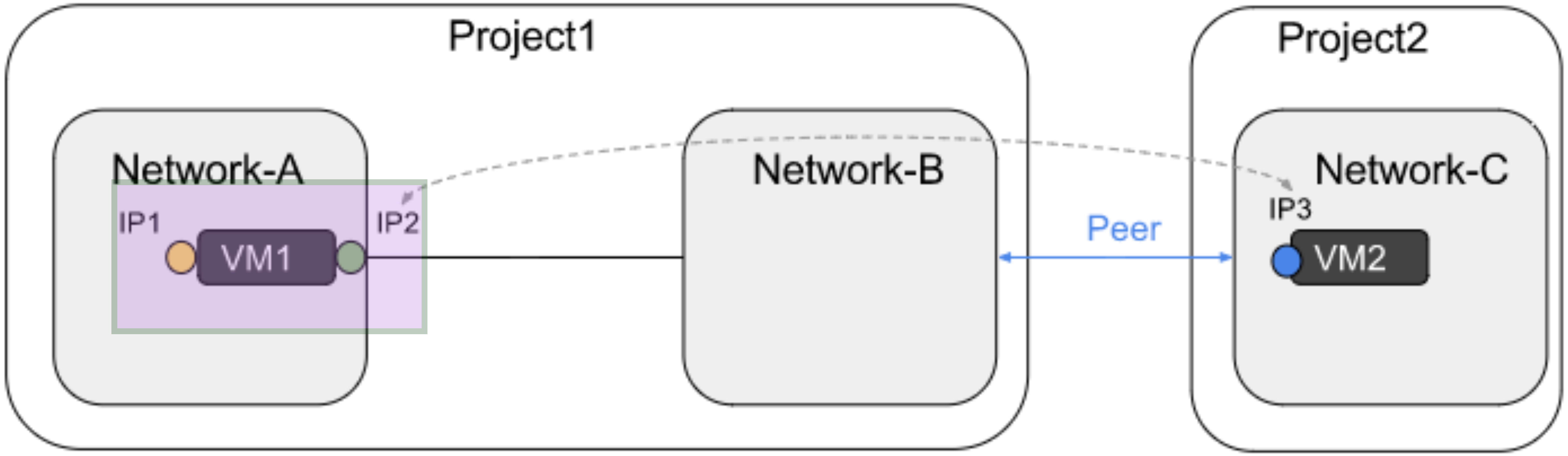
Peered with a simple VPC

Peered Networks and Shared VPCs



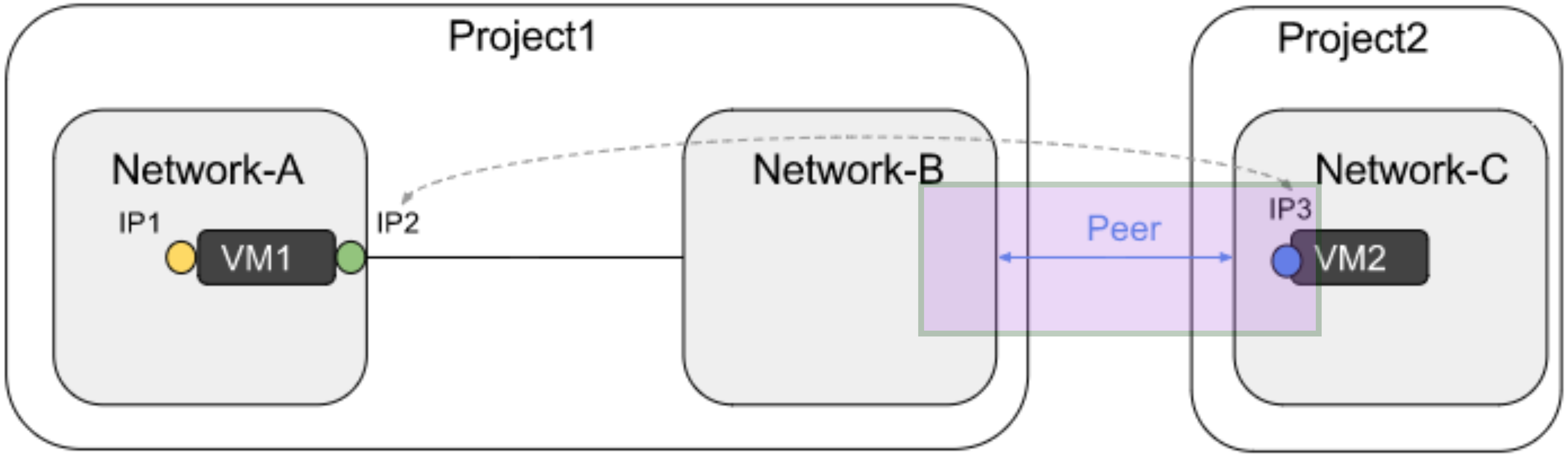
All VMs can communicate with each other via internal IP addresses

Peered Networks and Multiple NICs



The VM has two network interfaces - one in network A and one in network B

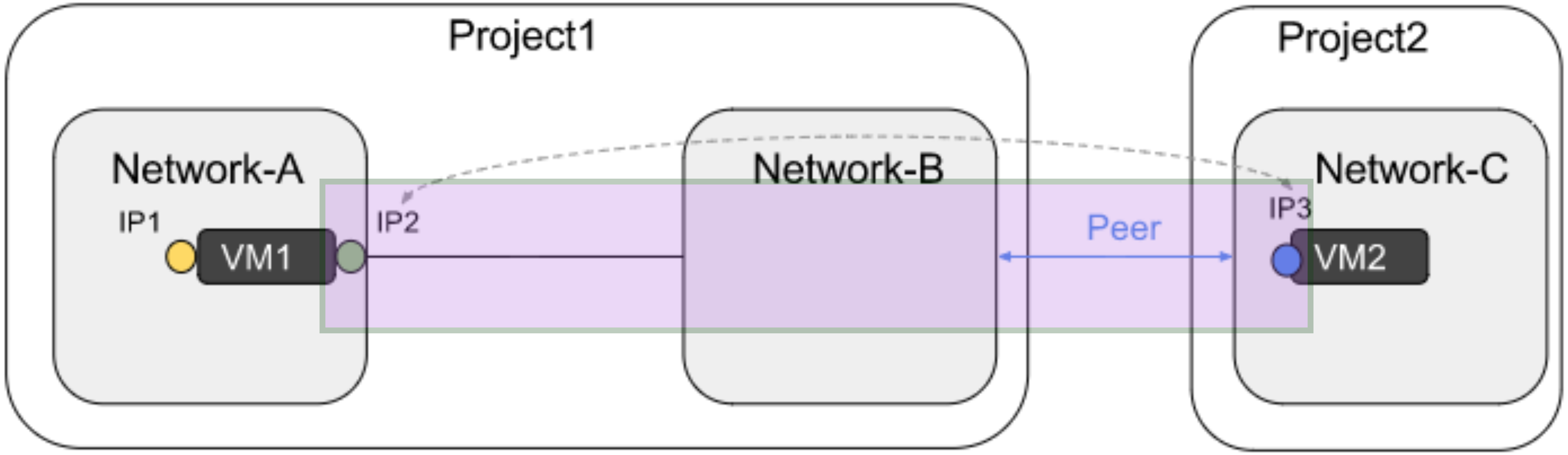
Peered Networks and Multiple NICs



Network B and C are peered with each other

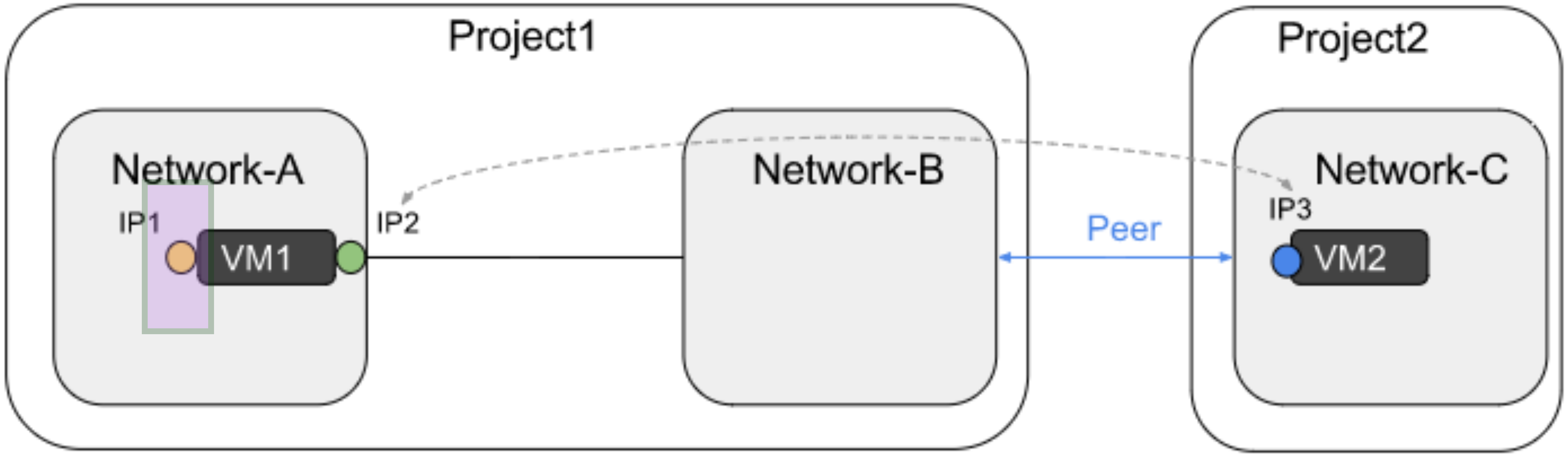
Network A is standalone i.e. not peered

Peered Networks and Multiple NICs



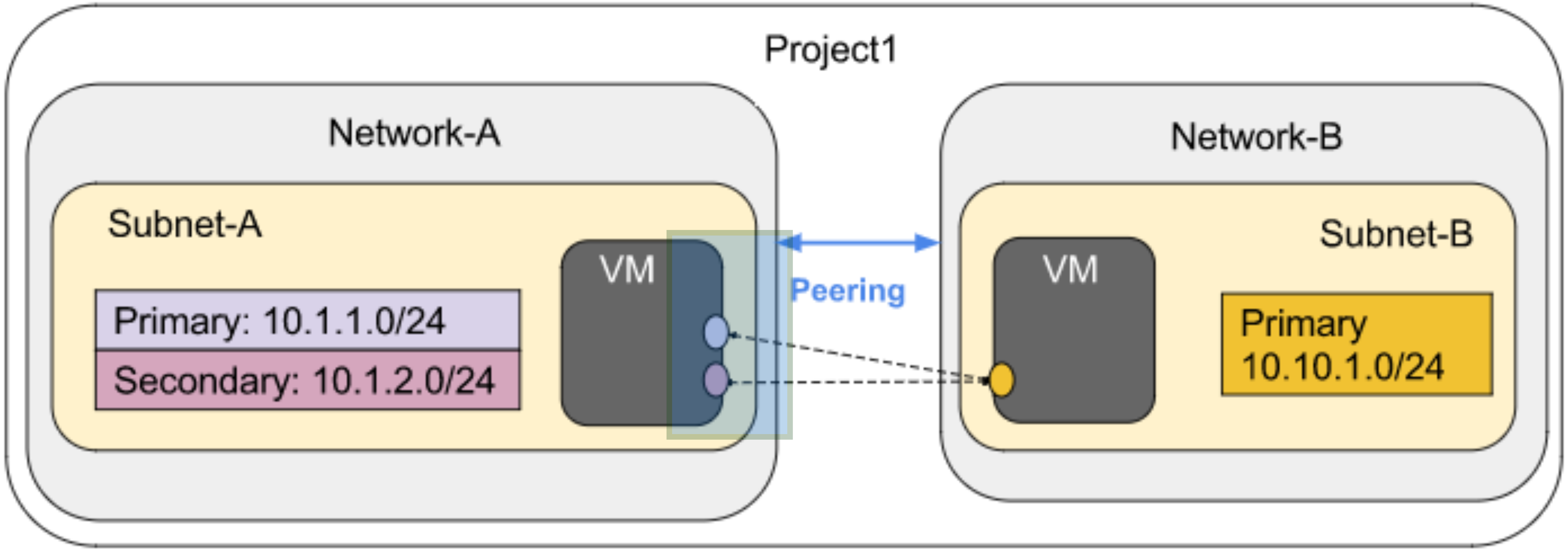
IP3 and IP2 can see and communicate with each other

Peered Networks and Multiple NICs



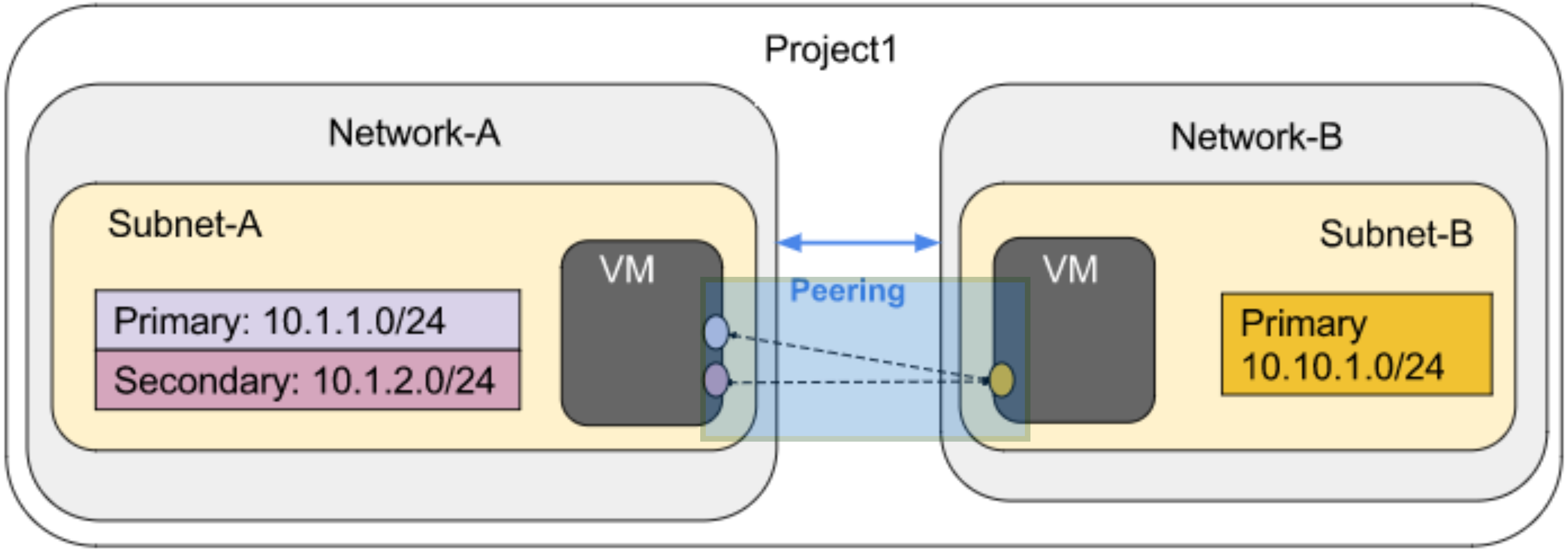
IP1 on network-A cannot see any instances in network-B or network-C

Peered Networks and IP Aliasing



The VM has IP aliased IP addresses, one from the primary range and one from the secondary range

Peered Networks and IP Aliasing



With peering both the IP addresses are visible to the instances in the peered network

Cloud DNS

Cloud DNS

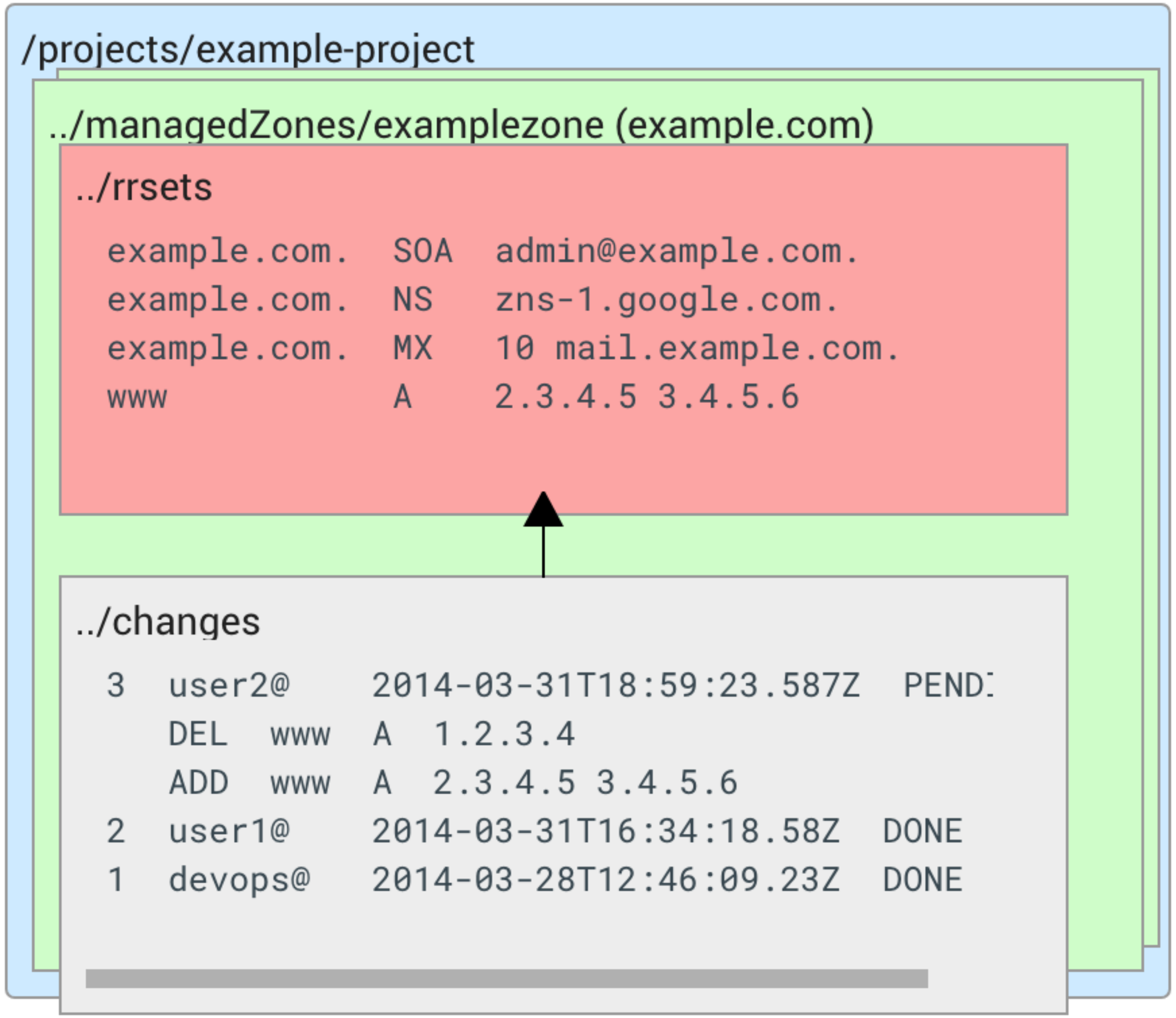
Google Cloud DNS is a high-performance, resilient, global Domain Name System (DNS) service that publishes your domain names to the global DNS in a cost-effective way.

https://cloud.google.com/dns/overview

Cloud DNS

- Hierarchical distributed database that lets you store IP addresses and other data and look them up by name
- Publish zones and records in the DNS
- No burden of managing your own DNS server

Cloud DNS


/projects/example-project
../managedZones/examplezone (example.com)
../rrsets

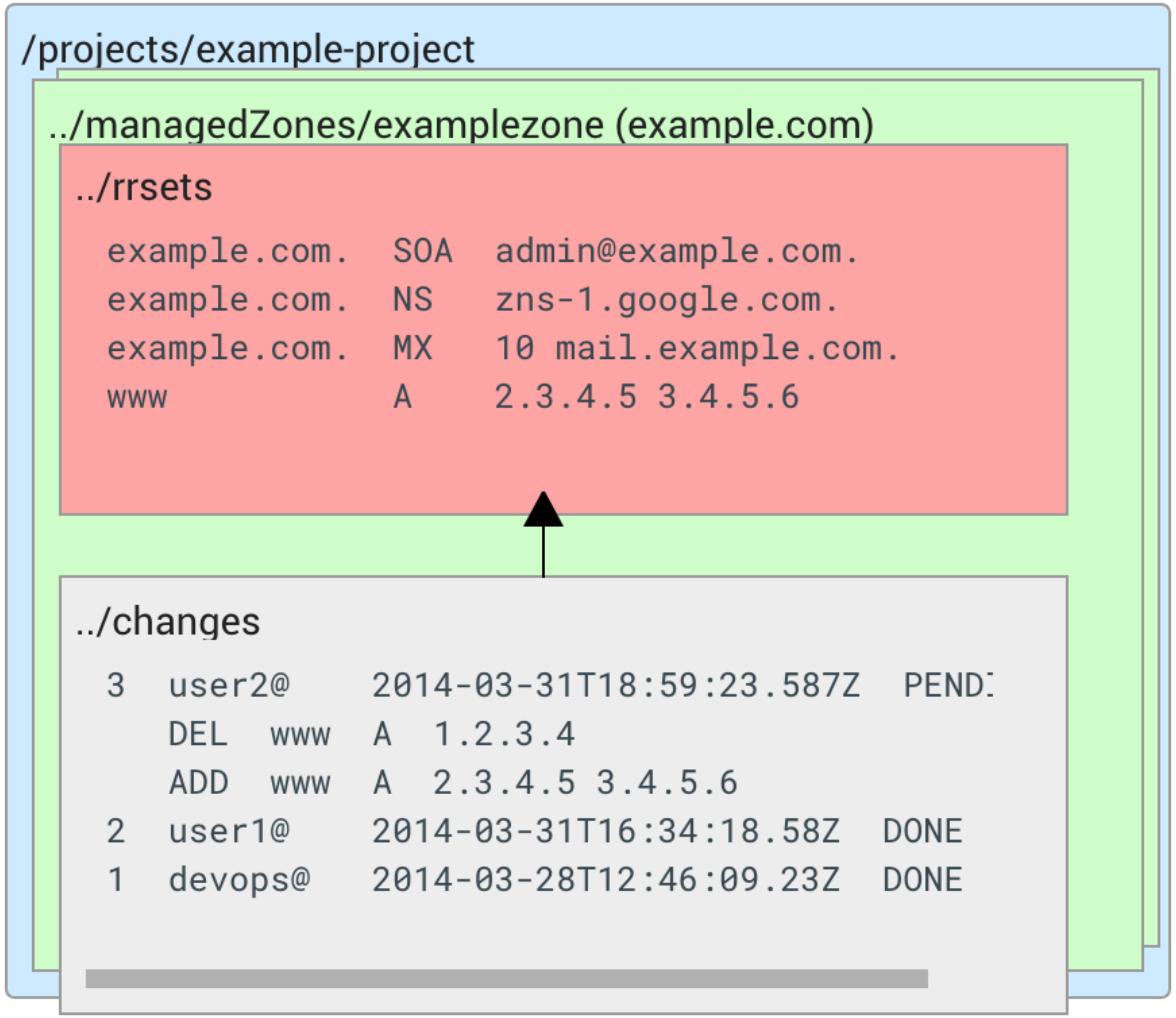
```
example.com.  SOA  admin@example.com.
example.com.  NS   zns-1.google.com.
example.com.  MX   10 mail.example.com.
www           A    2.3.4.5 3.4.5.6
```

```
3  user2@    2014-03-31T18:59:23.587Z  PENDI
   DEL  www  A  1.2.3.4
   ADD  www  A  2.3.4.5 3.4.5.6
2  user1@    2014-03-31T16:34:18.58Z  DONE
1  devops@   2014-03-28T12:46:09.23Z  DONE
```

Managed Zone

- Entity that manages DNS records for a given suffix (example.com)
- Maintained by Cloud DNS

Cloud DNS



/projects/example-project

../managedZones/examplezone (example.com)

../rrsets

```
example.com.  SOA  admin@example.com.
example.com.  NS   zns-1.google.com.
example.com.  MX   10 mail.example.com.
www           A    2.3.4.5 3.4.5.6
```

../changes

```
3  user2@    2014-03-31T18:59:23.587Z  PEND:
   DEL  www  A  1.2.3.4
   ADD  www  A  2.3.4.5 3.4.5.6
2  user1@    2014-03-31T16:34:18.58Z  DONE
1  devops@   2014-03-28T12:46:09.23Z  DONE
```

Record types -

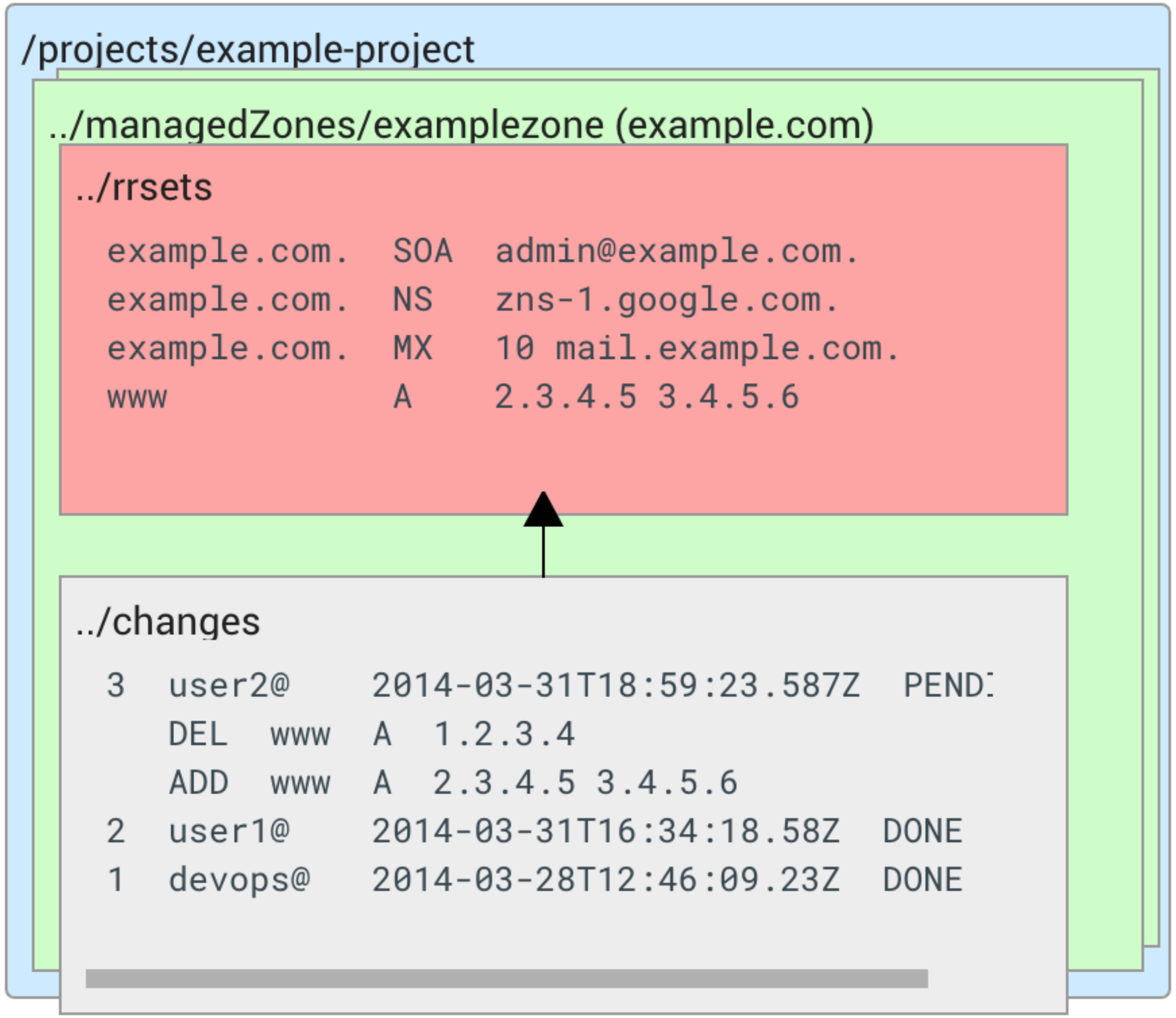
A - Address record, maps hostnames to IPv4 addresses

SOA - Start of authority - specifies authoritative information on a managed zone

MX - Mail exchange used to route requests to mail servers

NS - Name Server record, delegates a DNS zone to an authoritative server

Cloud DNS



/projects/example-project

../managedZones/examplezone (example.com)

../rrsets

```
example.com.  SOA  admin@example.com.
example.com.  NS   zns-1.google.com.
example.com.  MX   10 mail.example.com.
www           A    2.3.4.5 3.4.5.6
```

../changes

```
3   user2@    2014-03-31T18:59:23.587Z  PEND
    DEL  www  A  1.2.3.4
    ADD  www  A  2.3.4.5 3.4.5.6
2   user1@    2014-03-31T16:34:18.58Z  DONE
1   devops@   2014-03-28T12:46:09.23Z  DONE
```

Resource Record Changes

The changes are first made to the authoritative servers and is then picked up by the DNS resolvers when their cache expires

Legacy GCP Networks

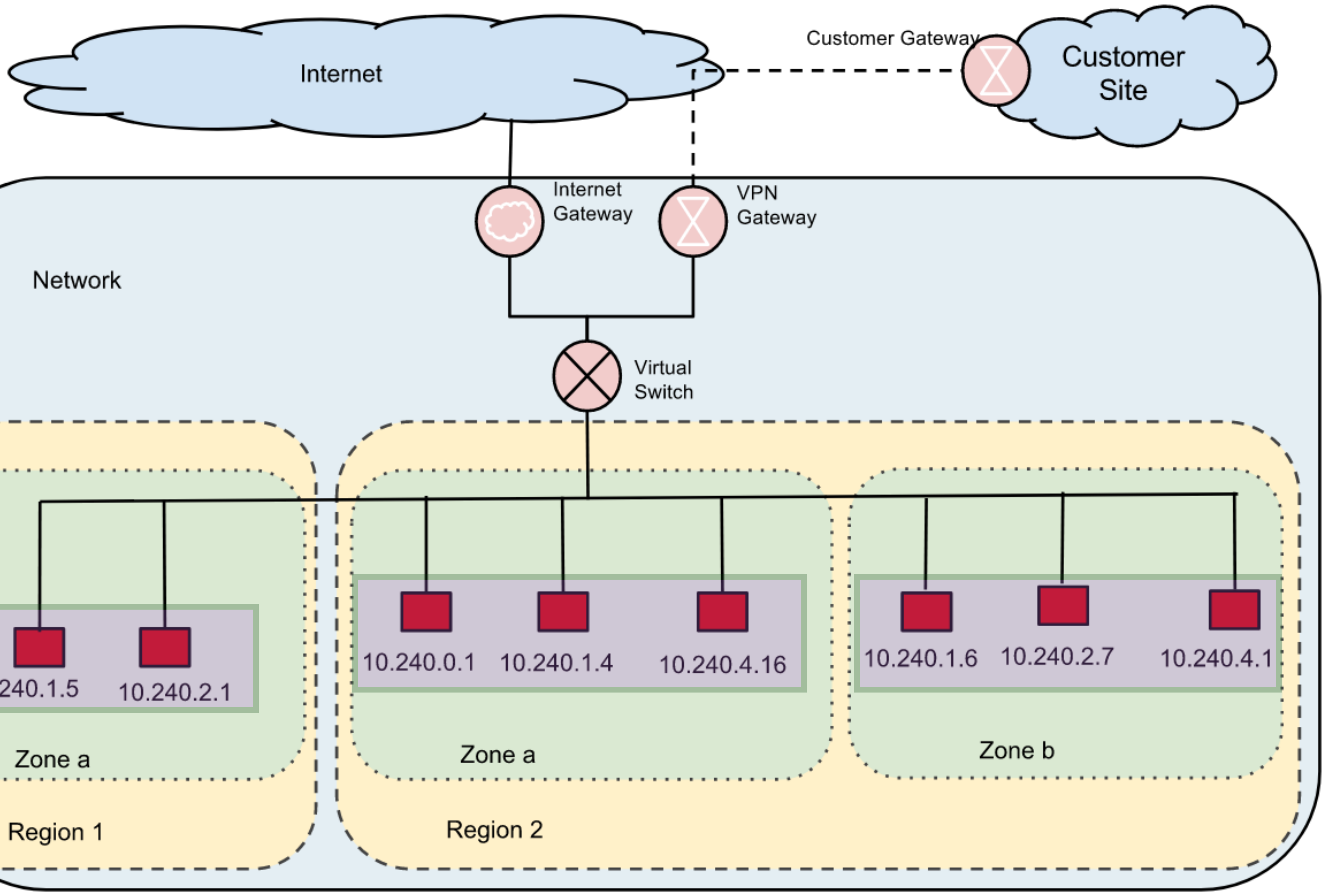
Not recommended :-)

- Instance IP addresses are not grouped by region or zone
- No subnets
- Random and non-contiguous IP addresses

It is still possible to create legacy networks through the gcloud command-line tool and the REST API. It is not possible to create legacy networks using the Google Cloud Platform Console.

Legacy GCP Networks

As shown in the example, instances from 10.240.0.0/16 are spread unpredictably across regions 1 and 2.


Internet
Customer Gateway
Customer Site
Internet Gateway
VPN Gateway
Network
Virtual Switch
10.240.1.5
10.240.2.1
Zone a
Region 1
10.240.0.1
10.240.1.4
10.240.4.16
Zone a
10.240.1.6
10.240.2.7
10.240.4.1
Zone b
Region 2