A Virtual Private Cloud (VPC) network is a virtual version of a physical network, such as a data center network. It provides connectivity for your Compute Engine virtual machine (VM) instances (/compute/docs/instances/), Google Kubernetes Engine (GKE) clusters (/kubernetes-engine/docs/concepts/cluster-architecture), App Engine flexible environment (/appengine/docs/flexible/) instances, and other resources in your project (/resource-manager/docs/cloud-platform-resource-hierarchy#projects).

Projects can contain multiple VPC networks. Unless you create an organizational policy that prohibits it, new projects start with a default network that has one subnetwork (subnet) in each region (an auto mode VPC network).

tant: This page describes *VPC networks*, which are different from legacy networks (/vpc/docs/legacy). Although you legacy networks by using the `gcloud` command-line tool or the REST API, they are *not* recommended for production se they do not support advanced networking features. You *cannot* convert a legacy network to a VPC network. To vi xisting network's type, see Viewing networks (/vpc/docs/using-vpc#viewing-networks).

VPC networks have the following properties:

- VPC networks, including their associated routes and firewall rules, are global resources (/compute/docs/regions-zones/global-regional-zonal-resources#globalresources). They are *not* associated with any particular region or zone.
- Subnets are regional resources (/compute/docs/regions-zones/global-regional-zonal-resources#regionalresources). Each subnet defines a range of IP addresses.
- Traffic to and from instances can be controlled with network firewall rules (#firewall_rules).
- Resources within a VPC network can communicate with one another by using internal (private) IPv4 addresses, subject to applicable network firewall rules. For more information, see communication within the network (#intra_vpc_reqs).
- Instances with internal IP addresses can communicate with Google APIs and services (https://developers.google.com/apis-explorer/). For more information, see Private access options for services (/vpc/docs/private-access-options).

- Network administration can be secured by using Cloud Identity and Access Management (Cloud IAM) (/iam/docs/) roles.
- An organization (/resource-manager/docs/cloud-platform-resource-hierarchy#organizations) can use Shared VPC (/vpc/docs/shared-vpc) to keep a VPC network in a common host project. Authorized Cloud IAM members from other projects in the same organization can create resources that use subnets of the Shared VPC network.
- VPC networks can be connected to other VPC networks in different projects or organizations by using VPC Network Peering (/vpc/docs/vpc-peering).
- VPC networks can be securely connected in hybrid environments by using Cloud VPN (/vpn/docs/concepts/overview) or Cloud Interconnect (/interconnect/docs/concepts/overview).
- VPC networks only support IPv4 unicast (https://wikipedia.org/wiki/Unicast) traffic. They do **not** support broadcast (https://wikipedia.org/wiki/Broadcasting_(networking)), multicast (https://wikipedia.org/wiki/IP_multicast), or IPv6 traffic *within* the network; VMs in the VPC network can only send to IPv4 destinations and only receive traffic from IPv4 sources. However, it is possible to create an IPv6 address for a global load balancer (/load-balancing/docs/ipv6).

The terms *subnet* and *subnetwork* are synonymous. They are used interchangeably in the Google Cloud Console, `gcloud` commands, and API documentation.

A subnet is *not* the same thing as a (VPC) network. Networks and subnets are *different* types of objects in Google Cloud.

Each VPC network consists of one or more useful IP range partitions called subnets. Each subnet is associated with a region. VPC networks do not have any IP address ranges associated with them. IP ranges are defined for the subnets (#manually_created_subnet_ip_ranges).

A network must have at least one subnet before you can use it. Auto mode VPC networks create subnets in each region automatically. Custom mode VPC networks start with no subnets, giving you full control over subnet creation. You can create more than one subnet per region. For information about the differences between auto mode and custom mode VPC networks, see types of VPC networks (#subnet-ranges).

When you create a resource in Google Cloud, you choose a network and subnet. For resources other than instance templates, you also select a zone

(/compute/docs/regions-zones/global-regional-zonal-resources#zoneresource) or a region. Selecting a zone implicitly selects its parent region. Because subnets are regional objects, the region that you select for a resource determines the subnets that it can use:

- The process of creating an instance (/compute/docs/instances/create-start-instance) involves selecting a zone, a network, and a subnet. The subnets available for selection are restricted to those in the selected region. Google Cloud assigns the instance an IP address from the range of available addresses in the subnet.
- The process of creating a managed instance group (/compute/docs/instance-groups/creating-groups-of-managed-instances) involves selecting a zone or region, depending on the group type, and an instance template. The instance templates available for selection are restricted to those whose defined subnets are in the same region selected for the managed instance group.
    - Instance templates are global resources. The process of creating an instance template (/compute/docs/instance-templates/create-instance-templates) involves selecting a network and a subnet. If you select an auto mode VPC network, you can choose to use auto subnets to defer subnet selection to one that is available in the selected region of any managed instance group that would use the template. Auto mode VPC networks have a subnet in every region by definition.
- The process of creating a Kubernetes container cluster (/kubernetes-engine/docs/how-to/creating-a-container-cluster) involves selecting a zone or region (depending on the cluster type), a network, and a subnet. The subnets available for selection are restricted to those in the selected region.

Google Cloud offers two types of VPC networks, determined by their *subnet creation mode*:

- When an *auto mode* VPC network is created (/vpc/docs/using-vpc#create-auto-network), one subnet from each region is automatically created within it. These automatically created subnets use a set of predefined IP ranges (#ip-ranges) that fit within the `10.128.0.0/9` CIDR block. As new Google Cloud regions become available, new subnets in those regions are automatically added to auto mode VPC networks by using an IP range from that block. In addition to the automatically created subnets, you can add more subnets manually (#manually_created_subnet_ip_ranges) to auto mode VPC networks in regions that you choose by using IP ranges outside of `10.128.0.0/9`.

- When a *custom mode* VPC network is created (/vpc/docs/using-vpc#create-custom-network), no subnets are automatically created. This type of network provides you with complete control over its subnets and IP ranges. You decide which subnets to create in regions that you choose by using IP ranges that you specify.

You can switch a VPC network from auto mode to custom mode (/vpc/docs/using-vpc#switch-network-mode). This is a one-way conversion; custom mode VPC networks cannot be changed to auto mode VPC networks. To help you decide which type of network meets your needs, see the considerations for auto mode VPC networks (#auto-mode-considerations).

Unless you choose to disable it, each new project starts with a default network. The default network is an auto mode VPC network with pre-populated firewall rules (/vpc/docs/firewalls#more_rules_default_vpc).

You can disable the creation of default networks by creating an organization policy (/resource-manager/docs/organization-policy/creating-managing-policies) with the `compute.skipDefaultNetworkCreation` constraint (/resource-manager/docs/organization-policy/org-policy-constraints). Projects that inherit this policy won't have a default network.

Auto mode VPC networks are easy to set up and use, and they are well suited for use cases with these attributes:

- Having subnets automatically created in each region is useful.
- The predefined IP ranges of the subnets do not overlap with IP ranges that you would use for different purposes (for example, Cloud VPN connections to on-premises resources).

However, custom mode VPC networks are more flexible and are better suited to production. The following attributes highlight use cases where custom mode VPC networks are recommended or required:

- Having one subnet automatically created in each region isn't necessary.
- Having new subnets automatically created as new regions become available could overlap with IP addresses used by manually created subnets or static routes, or could interfere with your overall network planning.

- You need complete control over the subnets created in your VPC network, including regions and IP address ranges used.
- You plan to connect VPC networks by using VPC Network Peering or Cloud VPN. Because the subnets of every auto mode VPC network use the same predefined range of IP addresses, you cannot connect auto mode VPC networks to one another.

tant: Production networks should be planned in advance. We recommend that you use custom mode VPC networks ction.

When you create a subnet, you must define a *primary IP address range*. You can optionally define *secondary IP address ranges*:

- *Primary IP address range*: You can choose any private RFC 1918 (https://tools.ietf.org/html/rfc1918) CIDR block for the primary IP address range of the subnet. These IP addresses can be used for VM primary internal IP addresses, VM alias IP addresses (/vpc/docs/alias-ip), and the IP addresses of internal load balancers.
- *Secondary IP address ranges*: You can define one or more secondary IP address ranges, which are separate RFC 1918 CIDR blocks. These IP address ranges are used only for alias IP addresses. The per network limits (/vpc/docs/quota#per_network) describe the maximum number of secondary ranges that you can define for each subnet.

Your subnets don't need to form a predefined contiguous CIDR block, but you can do that if desired. For example, auto mode VPC networks do create subnets that fit within a predefined auto mode IP range.

For more information, see working with subnets (/vpc/docs/using-vpc#subnet-rules).

Every subnet has four reserved IP addresses in its primary IP range. There are no reserved IP addresses in the secondary IP ranges.

Reserved IP address	Description	Example
Network	First address in the primary IP range for the subnet	`10.1.2.0` in `10.1.2.0/24`

Default gateway	Second address in the primary IP range for the subnet	`10.1.2.1` in `10.1.2.0/24`
Second-to-last address	Second-to-last address in the primary IP range for the subnet that is reserved by Google Cloud for potential future use	`10.1.2.254` in `10.1.2.0/24`
Broadcast	Last address in the primary IP range for the subnet	`10.1.2.255` in `10.1.2.0/24`

Google Cloud software-defined networking reserves a virtual gateway IP address for the primary IP ranges of each s
C network. However, virtual gateways do **not** respond to ICMP traffic or decrement IP TTL headers.

t secondary IP ranges don't have a reserved virtual gateway IP address. Thus, a default gateway doesn't respond to
esn't appear when you run `traceroute` from a VM instance.

hat ping the gateway IP address as a connectivity test must be configured so that they don't consider the inability to
al gateway to be a failure condition.

This table lists the IP ranges for the automatically created subnets in an auto mode VPC network. IP ranges for these subnets fit inside the `10.128.0.0/9` CIDR block. Auto mode VPC networks are built with one subnet per region at creation time and automatically receive new subnets in new regions. Unused portions of `10.128.0.0/9` are reserved for future Google Cloud use.

Region	IP range (CIDR)	Default gateway	Usable addresses (inclusive)
asia-east1	10.140.0.0/20	10.140.0.1	10.140.0.2 to 10.140.15.253
asia-east2	10.170.0.0/20	10.170.0.1	10.170.0.2 to 10.170.15.253
asia-northeast1	10.146.0.0/20	10.146.0.1	10.146.0.2 to 10.146.15.253
asia-northeast2	10.174.0.0/20	10.174.0.1	10.174.0.2 to 10.174.15.253
asia-northeast3	10.178.0.0/20	10.178.0.1	10.178.0.2 to 10.178.15.253
asia-south1	10.160.0.0/20	10.160.0.1	10.160.0.2 to 10.160.15.253
asia-southeast1	10.148.0.0/20	10.148.0.1	10.148.0.2 to 10.148.15.253
australia-southeast1	10.152.0.0/20	10.152.0.1	10.152.0.2 to 10.152.15.253

europe-north1	10.166.0.0/20	10.166.0.1	10.166.0.2 to 10.166.15.253
europe-west1	10.132.0.0/20	10.132.0.1	10.132.0.2 to 10.132.15.253
europe-west2	10.154.0.0/20	10.154.0.1	10.154.0.2 to 10.154.15.253
europe-west3	10.156.0.0/20	10.156.0.1	10.156.0.2 to 10.156.15.253
europe-west4	10.164.0.0/20	10.164.0.1	10.164.0.2 to 10.164.15.253
europe-west6	10.172.0.0/20	10.172.0.1	10.172.0.2 to 10.172.15.253
northamerica-northeast1	10.162.0.0/20	10.162.0.1	10.162.0.2 to 10.162.15.253
southamerica-east1	10.158.0.0/20	10.158.0.1	10.158.0.2 to 10.158.15.253
us-central1	10.128.0.0/20	10.128.0.1	10.128.0.2 to 10.128.15.253
us-east1	10.142.0.0/20	10.142.0.1	10.142.0.2 to 10.142.15.253
us-east4	10.150.0.0/20	10.150.0.1	10.150.0.2 to 10.150.15.253
us-west1	10.138.0.0/20	10.138.0.1	10.138.0.2 to 10.138.15.253
us-west2	10.168.0.0/20	10.168.0.1	10.168.0.2 to 10.168.15.253

Routes define paths for packets leaving instances (egress traffic). Routes in Google Cloud are divided into two categories: system-generated and custom.

Every new network starts with two types of system-generated routes:

- The default route (/vpc/docs/routes#routingpacketsinternet) defines a path for traffic to leave the VPC network. It provides general internet access to VMs that meet the internet access requirements (#internet_access_reqs). It also provides the typical path for Private Google Access.
- A subnet route (/vpc/docs/routes#subnet-routes) is created for each of the IP ranges associated with a subnet. Every subnet has at least one subnet route for its primary IP range. Additional subnet routes are created for a subnet if you add secondary IP ranges to it. Subnet routes define paths for traffic to reach VMs that use the subnets. You cannot remove subnet routes manually.

Custom routes are either static routes that you create manually or dynamic routes maintained automatically by one or more of your Cloud Routers. For more information, see custom routes (/vpc/docs/routes#custom-routes).

For complete details about routing in Google Cloud, see the routes overview (/vpc/docs/routes).

Each VPC network has an associated *dynamic routing mode* that controls the behavior of all of its Cloud Routers (/router/docs/concepts/overview). Cloud Routers share routes to your VPC network and learn custom dynamic routes from connected networks when you connect your VPC network to another network by using a Cloud VPN tunnel that uses dynamic routing (/vpn/docs/concepts/choosing-networks-routing#dynamic-routing), or by using Dedicated Interconnect (/interconnect/docs/how-to/choose-type#dedicated) or Partner Interconnect (/interconnect/docs/how-to/choose-type#partner).

- *Regional dynamic routing* is the default. In this mode, routes to on-premises resources learned by a given Cloud Router in the VPC network only apply to the subnets in the same region as the Cloud Router. Unless modified by custom advertisements, each Cloud Router only shares the routes to subnets in its region with its on-premises counterpart.
- *Global dynamic routing* changes the behavior of all Cloud Routers in the network such that the routes to on-premises resources that they learn are available in all subnets in the VPC network, regardless of region. Unless modified by custom advertisements, each Cloud Router shares routes to all subnets in the VPC network with its on-premises counterpart.

For information about how the set of routes shared by a Cloud Router can be customized, see custom advertisements (/router/docs/how-to/advertising-overview).

The dynamic routing mode can be set when you create or modify a VPC network. You can change the dynamic routing mode from regional to global and vice-versa without restriction. For instructions, see Changing the dynamic routing mode (/vpc/docs/using-vpc#switch-dynamic-routing).

n: Changing the dynamic routing mode has the potential to interrupt traffic within the network or enable or disable r
xpected ways. Carefully review the role of each Cloud Router before changing the dynamic routing mode.

Firewall rules apply to both outgoing (egress) and incoming (ingress) traffic in the network. Firewall rules control traffic even if it is entirely within the network, including communication among VM instances.

Every VPC network has two implied firewall rules (/vpc/docs/firewalls#default_firewall_rules). One implied rule allows most egress traffic, and the other denies all ingress traffic. You cannot delete the implied rules, but you can override them with your own. Google Cloud always blocks some traffic, regardless of firewall rules; for more information, see blocked traffic (/vpc/docs/firewalls#blockedtraffic).

To monitor which firewall rule allowed or denied a particular connection, see Firewall Rules Logging (/vpc/docs/firewall-rules-logging).

The system-generated subnet routes define the paths for sending traffic among instances within the network by using internal (private) IP addresses. For one instance to be able to communicate with another, appropriate firewall rules must also be configured because every network has an implied deny firewall rule for ingress traffic.

Except for the default network, you must explicitly create higher priority ingress firewall rules (/vpc/docs/firewalls#priority_order_for_firewall_rules) to allow instances to communicate with one another. The default network includes several firewall rules in addition to the implied ones, including the `default-allow-internal` rule, which permits instance-to-instance communication within the network. The default network also comes with ingress rules allowing protocols such as RDP and SSH.

Rules that come with the default network are also presented as options for you to apply to new auto mode VPC networks that you create by using the Cloud Console.

The following criteria must be satisfied for an instance to have outgoing internet access:

- The network must have a valid *default internet gateway* route or custom route whose destination IP range is the most general (`0.0.0.0/0`). This route defines the path to the internet. For more information, see Routes (/vpc/docs/routes#types_of_routes).
- Firewall rules must allow egress traffic from the instance. Unless overridden by a higher priority rule, the implied allow rule for egress traffic permits outbound traffic from all instances.
- One of the following must be true:

- The instance must have an external IP address. An external IP address can be assigned to an instance when it is created (/compute/docs/ip-addresses/reserve-static-external-ip-address#assign_new_instance) or after it has been created (/compute/docs/ip-addresses/reserve-static-external-ip-address#IP_assign).
- The instance must be able to use Cloud NAT (/nat/docs) or an instance-based proxy that is the target for a static `0.0.0.0/0` route.

VPC firewall rules apply to resources running in the VPC network, such as Compute Engine VMs. For App Engine instances, firewall rules work as follows:

- App Engine standard environment (/appengine/docs/standard/python/creating-firewalls): Only App Engine firewall rules apply to ingress traffic. Because App Engine standard environment instances do not run inside your VPC network, VPC firewall rules do not apply to them.
- App Engine flexible environment (/appengine/docs/flexible/python/creating-firewalls): Both App Engine and VPC firewall rules apply to ingress traffic. Inbound traffic is only permitted if it is allowed by both types of firewall rules. For outbound traffic, VPC firewall rules apply.

For more information about how to control access to App Engine instances, see App security (/appengine/docs/standard/python/application-security).

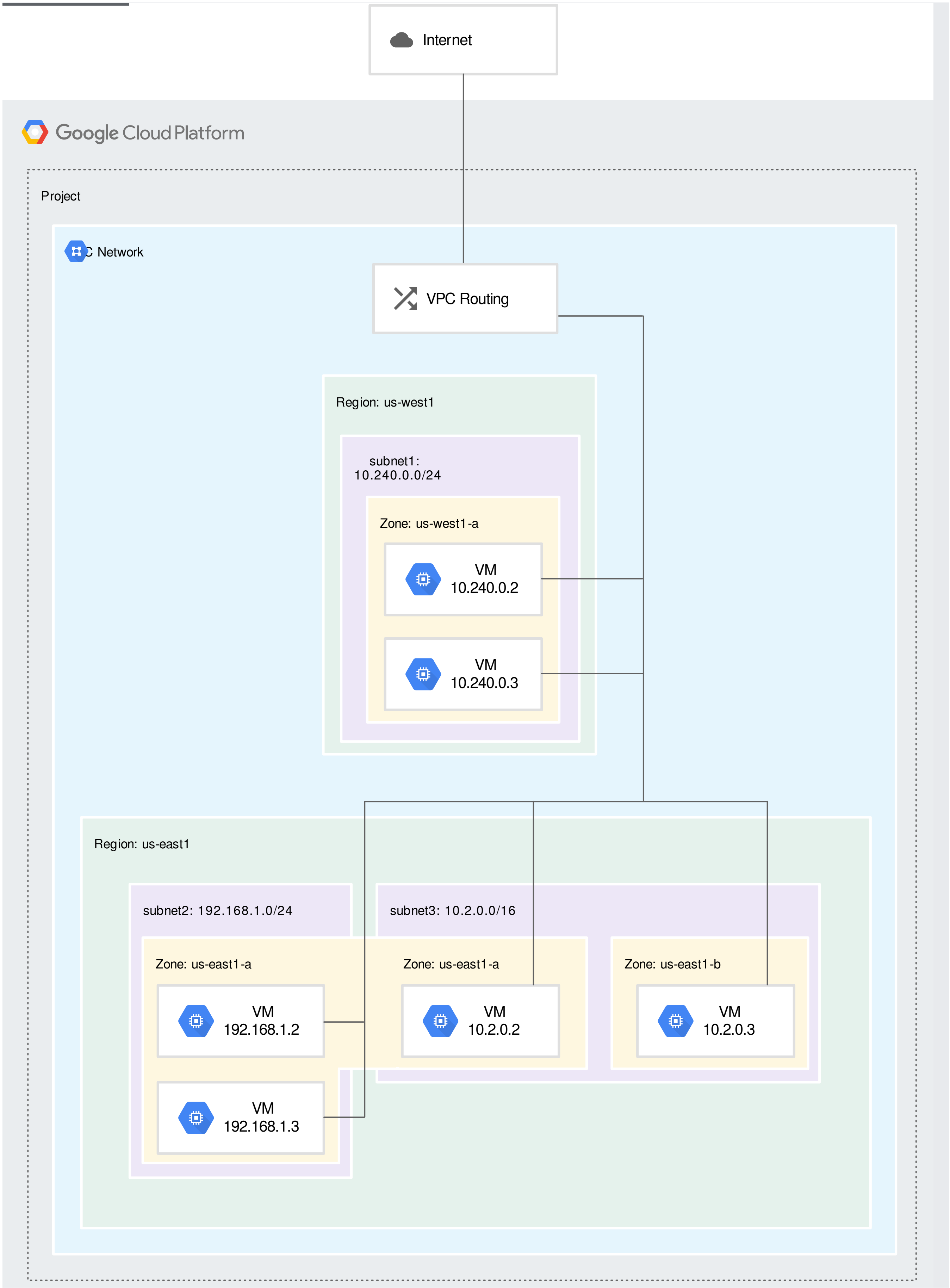
For internal reasons, Google Cloud increases the TTL counter of packets leaving Compute Engine instances for the internet. Tools like `traceroute` might provide incomplete results because the TTL doesn't expire on some of the hops. Hops that are inside and outside of Google's network might be hidden.

The number of hidden hops varies based on the instance's Network Service Tiers, region, and other factors. If there are only a few hops, it's possible for all of them to be hidden. Missing hops from a `traceroute` result don't mean that outbound traffic is dropped.

There is no workaround for this behavior.

The following example illustrates a custom mode VPC network with three subnets in two regions:

VPC network example (click to enlarge)


Internet
Google Cloud Platform
Project
C Network
VPC Routing
Region: us-west1
subnet1:
10.240.0.0/24
Zone: us-west1-a
VM
10.240.0.2
VM
10.240.0.3
Region: us-east1
subnet2: 192.168.1.0/24
subnet3: 10.2.0.0/16
Zone: us-east1-a
Zone: us-east1-a
Zone: us-east1-b
VM
192.168.1.2
VM
10.2.0.2
VM
10.2.0.3
VM
192.168.1.3

(/vpc/images/vpc-overview-example.svg)

- *Subnet1* is defined as `10.240.0.0/24` in the us-west1 region.
    - Two VM instances in the us-west1-a zone are in this subnet. Their IP addresses both come from the available range of addresses in *subnet1*.
- *Subnet2* is defined as `192.168.1.0/24` in the us-east1 region.
    - Two VM instances in the us-east1-a zone are in this subnet. Their IP addresses both come from the available range of addresses in *subnet2*.
- *Subnet3* is defined as `10.2.0.0/16`, also in the us-east1 region.
    - One VM instance in the us-east1-a zone and a second instance in the us-east1-b zone are in *subnet3*, each receiving an IP address from its available range. Because subnets are regional resources, instances can have their network interfaces associated with any subnet in the same region that contains their zones.

- To create, modify, or delete VPC networks, see Using VPC networks (/vpc/docs/using-vpc).