VPC networks

Archived: 2026-04-05 18:13:24 UTC

VPC networks Stay organized with collections Save and categorize content based on your preferences.

A Virtual Private Cloud (VPC) network is a virtual version of a physical network that is implemented inside of Google's production network by using Andromeda.

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.

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

Networks and subnets

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 resources in Google Cloud.

For more information, see Subnets.

Virtual machine instances

A Compute Engine virtual machine (VM) instance is a virtual machine that is hosted on Google's infrastructure. The terms *Compute Engine instance*, *VM instance*, and *VM* are synonymous. They are used interchangeably in the Google Cloud console, the Google Cloud CLI reference, and the API documentation.

VM instances include Google Kubernetes Engine (GKE) clusters, App Engine flexible environment instances, and other Google Cloud products built on Compute Engine VMs.

For more information, see Virtual machine instances in the Compute Engine documentation.

Specifications

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.
- Each subnet defines the following IP address ranges:
    - IPv4-only and dual-stack subnets both define a range of IPv4 addresses, while dual-stack subnets also define a range of IPv6 addresses.
    - IPv6-only subnets define a range of IPv6 addresses.

  For more information, see Types of subnets.
- Traffic to and from instances can be controlled with network firewall rules. Rules are implemented on the VMs themselves, so traffic can only be controlled and logged as it leaves or arrives at a VM.
- Resources within a VPC network can communicate with one another by using internal IPv4 addresses, internal IPv6 addresses, or external IPv6 addresses, subject to applicable network firewall rules. For more information, see communication within the network.
- Instances with internal IPv4 or IPv6 addresses can communicate with Google APIs and services. For more information, see Private access options for services.
- Network administration can be secured by using Identity and Access Management (IAM) roles.
- An organization can use Shared VPC to keep a VPC network in a common host project. Authorized IAM principals 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 networks can be connected to on-premises networks or other cloud providers by using Cloud VPN or Cloud Interconnect.
- VPC networks support version 0 of the GRE protocol with the following limitations:
    - Cloud NAT doesn't support GRE.
    - Application Load Balancers and proxy Network Load Balancers don't support GRE.
    - Passthrough Network Load Balancers and protocol forwarding support GRE when using the `L3_DEFAULT` forwarding rule protocol. For more information, see Internal passthrough Network Load Balancer overview, Backend service-based external passthrough Network Load Balancer overview, and Protocol forwarding overview.

    - Cloud Customer Care doesn't provide configuration or troubleshooting assistance for overlay networks.

- VPC networks support IPv4 and IPv6 unicast addresses. VPC networks do **not** support broadcast or multicast addresses *within* the network.

  For more information about IPv6 subnet ranges, see Subnets.

VPC network example

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


VPC network example (click to enlarge).

- *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-b 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-b zone and a second instance in the us-east1-c 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.

Organization policy constraints

- Each new project starts with a default VPC network. You can disable the creation of default networks by creating an organization policy with the `compute.skipDefaultNetworkCreation` constraint. Projects that inherit this policy won't have a default network.

- You can control the following IPv6 configurations using organization policies:

    - **Disable VPC External IPv6 usage**: If set to true, the `constraints/compute.disableVpcExternalIpv6` constraint prevents you from configuring subnets with external IPv6 ranges.

    - **Disable VPC Internal IPv6 usage**: If set to true, the `constraints/compute.disableVpcInternalIpv6` constraint prevents you from configuring subnets with internal IPv6 ranges.

    - **Disable All IPv6 usage**: If set to true, the `constraints/compute.disableAllIpv6` constraint disables the creation of, or update to, any subnets or other networking resources involved in IPv6 usage.

For more information about constraints, see Organization policy constraints.

Subnet creation mode

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

- When an *auto mode* VPC network is created, one subnet from each region is automatically created within it. These automatically created subnets use a set of predefined IPv4 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 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, 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. 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.

Default network

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 IPv4 firewall rules. The default network does not have pre-populated IPv6 firewall rules.

Considerations for auto mode VPC networks

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 your VPC network to another network:
    - 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.
    - Because the auto mode `10.128.0.0/9` CIDR range is part of the commonly used RFC 1918 address space, networks outside of Google Cloud might use an overlapping CIDR range.
- You want to create subnets with IPv6 ranges. Auto mode VPC networks don't support subnets with IPv6 ranges.

IPv4 subnet ranges

Each subnet has a *primary IPv4 address range*. The primary internal addresses for the following resources come from the subnet's primary range: VM instances, internal load balancers, and internal protocol forwarding. You can optionally add *secondary IP address ranges* to a subnet, which are only used by alias IP ranges. However, you can configure alias IP ranges for instances from the primary or secondary range of a subnet.

Each primary or secondary IPv4 range for all subnets in a VPC network must be a unique valid CIDR block. Refer to the per network limits for the number of secondary IP ranges you can define.

Your IPv4 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.

When you create a subnet in a custom mode VPC network, you choose what IPv4 range to use. For more information, see valid ranges, prohibited subnet ranges, and Limitations for IPv4 subnet ranges.

There are four unusable IP addresses in every primary IPv4 subnet range. For more information, see Unusable addresses in IPv4 subnet ranges.

Auto mode VPC networks are created with one subnet per region at creation time and automatically receive new subnets in new regions. The subnets have IPv4 ranges only, and all subnet ranges fit inside the `10.128.0.0/9` CIDR block. Unused portions of `10.128.0.0/9` are reserved for future Google Cloud use. For information about what IPv4 range is used in which region, see Auto mode IPv4 subnet ranges.

IPv6 subnet ranges

When you create a subnet with an IPv6 range in a custom mode VPC network, you choose whether the subnet is configured with an internal IPv6 subnet range or an external IPv6 subnet range.

- Internal IPv6 subnet ranges are used for VM-to-VM communication within VPC networks. They can't be reached from the internet and aren't publicly routable.
- External IPv6 subnet ranges can be used for VM-to-VM communication, and they are also publicly routable.

For more information about IPv6 subnet ranges, see Subnets.

Networks that support subnets with IPv6 address ranges

You can create subnets with IPv6 address ranges in a custom mode VPC network. For more information, see Work with subnets.

Subnets with IPv6 address ranges aren't supported in the following:

- Auto mode VPC networks, including the default network
- Legacy networks

If you have an auto mode VPC network that you would like to add subnets with IPv6 address ranges to, you can do the following:

1. Convert the auto mode network to custom mode.
2. Create new dual-stack or IPv6-only subnets. You can also convert existing IPv4-only subnets to dual-stack.

Routes and firewall rules

Routes

Routes define paths for packets leaving instances (egress traffic). For details about Google Cloud route types, see Routes.

Dynamic routing mode

Each VPC network has an associated *dynamic routing mode* that controls the behavior of all of its Cloud Routers. Cloud Routers manage BGP sessions for Google Cloud products that use Cloud Router.

For a description of dynamic routing mode options, see Dynamic routing mode in the Cloud Router documentation.

Route advertisements and internal IP addresses

The following IP addresses are advertised within a VPC network:

- Regional internal IPv4 addresses

  Used for primary and secondary IPv4 subnet address ranges

- Regional internal and external IPv6 addresses

  Used for internal and external IPv6 subnet address ranges

- Global internal IPv4 addresses

  Used for Private Service Connect endpoints for Google APIs

If you connect VPC networks using VPC Network Peering, subnet ranges using private IPv4 addresses are always exchanged. You can control whether subnet ranges using privately used public IPv4 addresses are exchanged and whether internal and external IPv6 subnet ranges are exchanged. Global internal IPv4 addresses are never exchanged using peering. For additional details, see the VPC Network Peering documentation.

When you connect a VPC network to another network, such as an on-premises network, using a Google Cloud connectivity product like Cloud VPN, Cloud Interconnect, or Router appliance:

- You can advertise the VPC network's internal IP addresses to another network (such as an on-premises network).
- Though connectivity between a VPC network and another network (such as an on-premises network) can use private routing provided by a Google Cloud connectivity product, the other network's IP addresses might also be publicly routable. Keep this in mind if an on-premises network uses publicly routable IP addresses.
- VM instances in a VPC network containing subnet ranges with privately used public IP addresses are not able to connect to external resources which use those same public IP addresses.
- Take extra care when advertising privately used public IP addresses to another network (such as an on-premises network), especially when the other network can advertise those public IP addresses to the internet.

Firewall rules

Both hierarchical firewall policies and VPC firewall rules apply to packets sent to and from VM instances (and resources that depend on VMs, such as Google Kubernetes Engine nodes). Both types of firewalls control traffic even if it is between VMs in the same VPC network.

To monitor which firewall rule allowed or denied a particular connection, see Firewall Rules Logging.

Communications and access

Communication within the network

The system-generated subnet routes define the paths for sending traffic among instances within the network by using internal 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 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 Google Cloud console.

Internet access requirements

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` for IPv4, `::/0` for IPv6). This route defines the path to the internet. For more information, see Route types.
- 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 or after it has been created.
    - The instance must be able to use Cloud NAT or an instance-based proxy that is the target for a static `0.0.0.0/0` or `::/0` route.

Communications and access for App Engine

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: 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: 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.

Traceroute to external IP addresses

For internal reasons, Google Cloud increases the TTL counter of packets that traverse next hops in Google's network. Tools like `traceroute` and `mtr` might provide incomplete results because the TTL doesn't expire on some of the hops. Hops that are inside of Google's network might be hidden when you send packets from Compute Engine instances to destinations on the internet.

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` or `mtr` result don't mean that outbound traffic is dropped.

There is no workaround for this behavior. You must take it into account if you configure third-party monitoring that connects to an external IP address associated with a VM.

Egress throughput limits

Network throughput information is available on the Network bandwidth page in the Compute Engine documentation.

Packet size

You can find information about packet size in Maximum transmission unit.

Maximum transmission unit

For more information about the maximum transmission unit (MTU) setting for a VPC network and its connected VMs, see Maximum transmission unit.

For information about changing the MTU of a VPC network, or migrating VMs between VPC networks with different MTU settings, see Change the MTU setting of a VPC network.

Supported protocols

Google Cloud supports only the following protocols and extension headers:

- **IPv4 data packets between VMs:** all IPv4 protocols.
- **IPv4 data packets between VMs and the internet:** the ICMP, IPIP, TCP, UDP, GRE, ESP, AH, and SCTP protocols.
- **IPv6 data packets between VMs *and* between VMs and the internet:** the AH, ESP, GRE, ICMP, ICMPv6, IPIP, SCTP, TCP, and UDP protocols and the Destination Options and Fragments extension headers. However, placing the Destination Options header after the Fragment header in an IPv6 data packet is not supported.
- **Protocol forwarding:** the AH, ESP, GRE, ICMP, ICMPv6, SCTP, TCP, and UDP protocols

To allow data packets of the supported protocols, you need to configure firewall rules or protocol forwarding rules based on your requirements.

Network profiles for specific use cases

Google Cloud uses the network profile resource to pre-configure certain properties in a VPC network for a specific use case. You can optionally specify a network profile provided by Google Cloud when you create your network.

The use case supported by network profiles is running AI workloads on machines with network interfaces (NICs) that support remote direct memory access (RDMA). Google Cloud provides RDMA network profiles that let you create Virtual Private Cloud (VPC) networks that support RDMA connectivity.

For more information, see the network profiles overview.

For more information about running AI workloads in Google Cloud, see the AI Hypercomputer documentation.

Network performance

Latency

The measured inter-region latency for Google Cloud networks can be found in our live dashboard. The dashboard shows Google Cloud's median inter-region latency and throughput performance metrics and methodology to reproduce these results using PerfKit Benchmarker.

Google Cloud typically measures round-trip latencies less than 55 µs at the 50th percentile and tail latencies less than 80µs at the 99th percentile between c2-standard-4 VM instances in the same zone.

Google Cloud typically measures round-trip latencies less than 45µs at the 50th percentile and tail latencies less than 60µs at the 99th percentile between c2-standard-4 VM instances in the same low-latency network ("compact" placement policy). A compact placement policy lowers the network latency by ensuring that the VMs are located physically within the same low-latency network.

Methodology: Intra-zone latency is monitored via a blackbox prober that constantly runs netperf TCP_RR benchmark between a pair of c2-types VMs in every zone c2 instances are available. It collects P50 and P99 results for setup with and without compact placement policy. TCP_RR benchmark measures request/response performance by measuring the transaction rate. If your applications require best possible latency, c2 instances are recommended.

Packet loss

Google Cloud tracks cross-region packet loss by regularly measuring round-trip loss between all regions. We target the global average of those measurements to be lower than 0.01% .

Methodology: A blackbox vm-to-vm prober monitors the packet loss for every zone pair using pings and aggregates the results into one global loss metric. This metric is tracked with a one-day window.

What's next

- To learn about using VPC networks and subnets, see Create, modify, or delete VPC networks and subnets.
- To learn about best practices for deploying VPC networks, see Best practices and reference architectures for VPC design.
- To learn about deploying VPC networks as part of Cross-Cloud Network, see Cross-Cloud Network for distributed applications.

Try it for yourself

If you're new to Google Cloud, create an account to evaluate how VPC performs in real-world scenarios. New customers also get $300 in free credits to run, test, and deploy workloads.

Try VPC free
