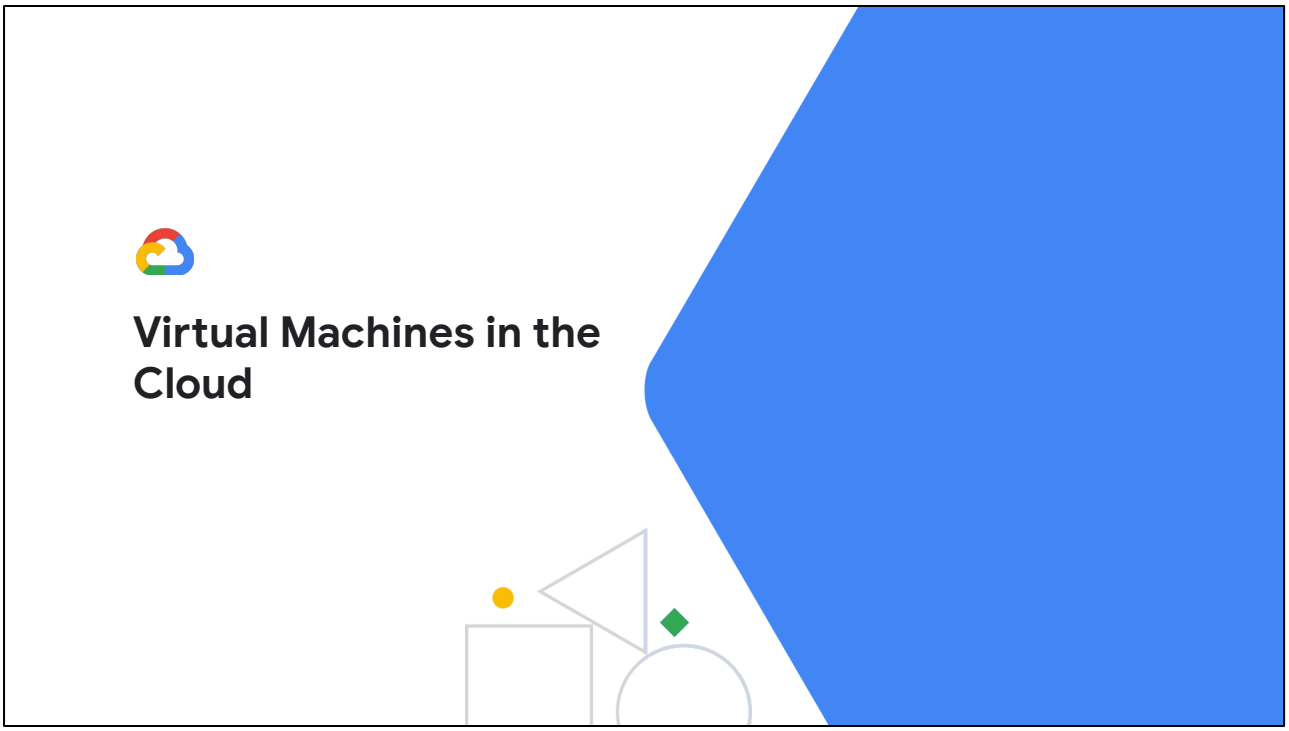


Of all the ways you can run workloads in the cloud, virtual machines may be the most familiar. Compute Engine lets you run virtual machines on Google's global infrastructure. In this module, we'll learn how Compute Engine works, with a focus on Google virtual networking.

One of the nice things about virtual machines is that they have the power and generality of a full-fledged operating system in each. You configure a virtual machine much like you build out a physical server: by specifying its amounts of CPU power and memory, its amounts and types of storage, and its operating system. You can flexibly reconfigure them. And a VM running on Google's cloud has unmatched worldwide network connectivity.

Agenda

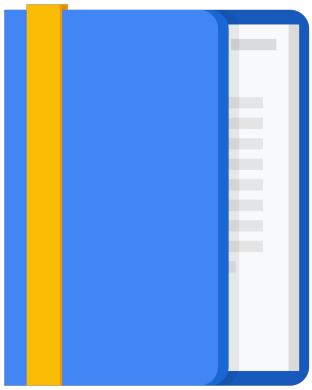
Virtual Private Cloud (VPC) Network

Compute Engine

Important VPC Capabilities

Quiz and Lab

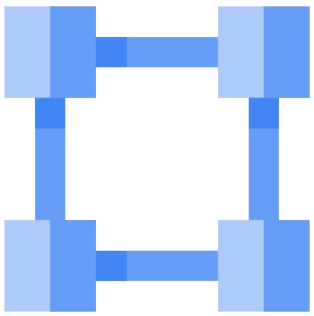
Resources



Google Cloud

Virtual Private Cloud Networking

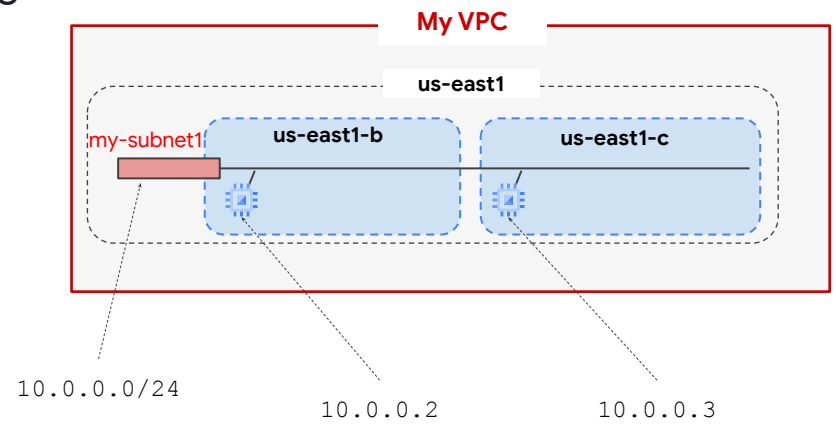
- Each VPC network is contained in a Google Cloud project.
- You can provision Google Cloud resources, connect them to each other, and isolate them from one another.



Google Cloud

The way a lot of people get started with Google Cloud is to define their own Virtual Private Cloud inside their first Google Cloud project. Or they can simply choose the default VPC and get started with that. Regardless, your VPC networks connect your Google Cloud resources to each other and to the internet. You can segment your networks, use firewall rules to restrict access to instances, and create static routes to forward traffic to specific destinations.

Google Cloud VPC networks are global; subnets are regional



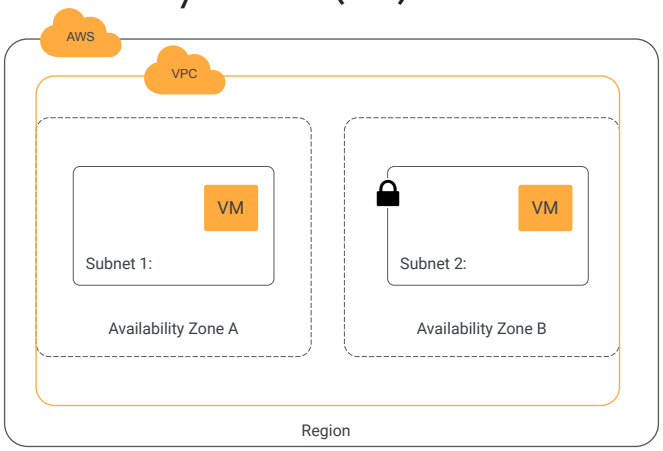
Google Cloud

Here's something that surprises a lot of people who are new to Google Cloud. The Virtual Private Cloud networks that you define have global scope. They can have subnets in any Google Cloud region worldwide. Subnets can span the zones that make up a region. This architecture makes it easy for you to define your own network layout with global scope. You can also have resources in different zones on the same subnet.

You can dynamically increase the size of a subnet in a custom network by expanding the range of IP addresses allocated to it. Doing that doesn't affect already configured VMs.

In this example, your VPC has one network. So far, it has one subnet defined, in Google Cloud's us-east1 region. Notice that it has two Compute Engine VMs attached to it. They're neighbors on the same subnet even though they are in different zones! You can use this capability to build solutions that are resilient but still have simple network layouts.

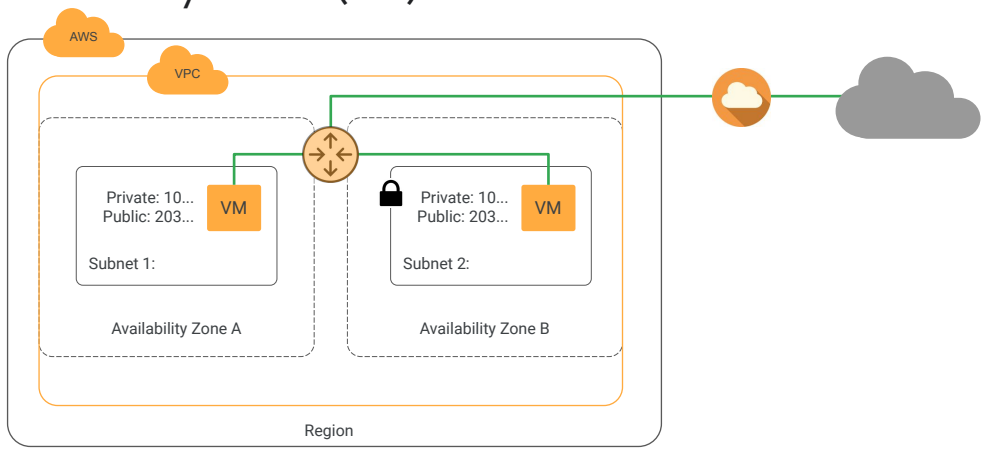
AWS VPCs are built within a region using subnets on availability zones (1/4)



Google Cloud

The VPC is made up of subnets, which must be built on availability zones in the region.

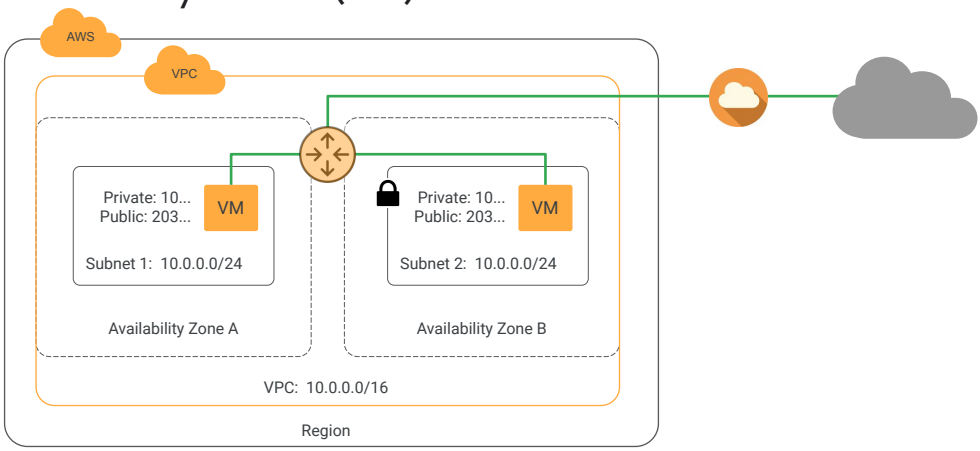
AWS VPCs are built within a region using subnets on availability zones (2/4)



Google Cloud

A subnet can be either public or private. A public subnet can route traffic to the internet. A private subnet's traffic never leaves the VPC.

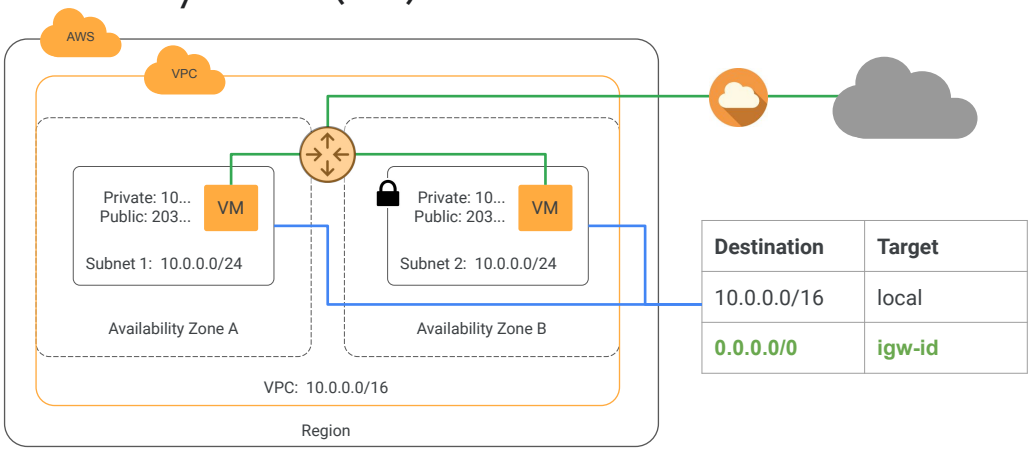
AWS VPCs are built within a region using subnets on availability zones (3/4)



Google Cloud

A VPC must be built with a CIDR, or Classless Inter-Domain Routing, range of private IP addresses that conform to RFC 1918. All subnets in a VPC must have private IP ranges that are part of the VPC CIDR range.

AWS VPCs are built within a region using subnets on availability zones (4/4)



Google Cloud

Route tables are built for each VPC that configure paths for traffic. Traffic cannot flow outside the network without Security groups, which are firewalls that can be applied to the virtual machine or network. Network Access Control Lists can be configured to allow and deny traffic to a subnet, but they are not stateful.

Summary of differences between Google and AWS VPC

	Google Cloud VPC	AWS VPC
Virtual networks	VPC networks (global)	VPCs (regional)
IP address ranges	Subnets (regional)	Subnets (Availability Zone)
Routing entries	Routes (global)	Routes (regional)
Security boundaries	Firewall rules (global)	NACLs, Security Groups (global)

The networking services that Google and Amazon provide can vary in terms of their scope, as indicated here in parentheses.

The key takeaways are that Google VPC networks are global and subnets span regions, not Availability Zone as in the case of AWS. Global networks offer farther, out-of-the-box reach, which means you can create a single private network that is global, without having to connect multiple private networks and manage those spaces separately.

You can also define multiple networks per project for added flexibility. We'll cover this in more detail later.

Agenda

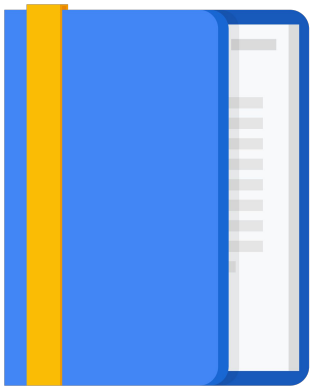
Virtual Private Cloud (VPC) Network

Compute Engine

Important VPC Capabilities

Quiz and Lab

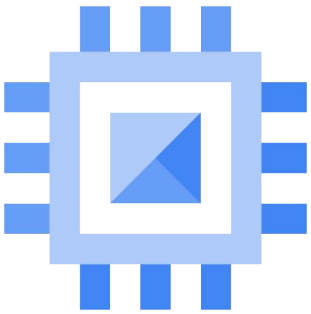
Resources



Google Cloud

Compute Engine offers managed virtual machines

- High CPU, high memory, standard and shared-core machine types.
- Persistent disks .
- Standard, SSD, and local SSD.
- Snapshots
- Resize disks with no downtime.
- Instance metadata and startup scripts.



Google Cloud

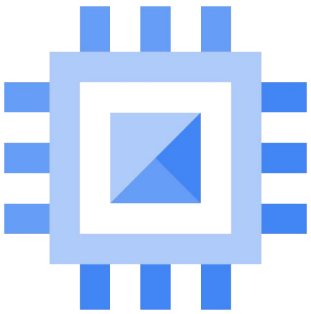
Virtual machines have the power and generality of a full-fledged operating system in each. You configure a virtual machine much like you build out a physical server: by specifying its amounts of CPU power and memory, its amounts and types of storage, and its operating system. Compute Engine lets you create and run virtual machines on Google infrastructure. There are no upfront investments, and you can run thousands of virtual CPUs on a system that is designed to be fast and to offer consistent performance.

You can flexibly reconfigure Compute Engine virtual machines. And a VM running on Google's cloud has unmatched worldwide network connectivity.

You can create a virtual machine instance by using the Cloud Console or the gcloud command-line tool. A Compute Engine instance can run Linux and Windows Server images provided by Google or any customized versions of these images. You can also build and run images of other operating systems.

Compute Engine offers customer friendly pricing

- Per-second billing, sustained use discounts, committed use discounts.
- Preemptible instances.
- High throughput to storage at no extra cost.
- Custom machine types: Only pay for the hardware you need.



Google Cloud

Compute Engine bills by the second for use of virtual machines, with a one-minute minimum. And discounts apply automatically to virtual machines that run for substantial fractions of a month. For each VM that you run for more than 25% of a month, Compute Engine automatically gives you a discount for every incremental minute. You can get up to a 30% net discount for VMs that run the entire month.

Compute Engine offers the ability to purchase committed use contracts in return for deeply discounted prices for VM usage. These discounts are known as committed use discounts. If your workload is stable and predictable, you can purchase a specific amount of vCPUs and memory for up to a 57% discount off of normal prices in return for committing to a usage term of 1 year or 3 years.

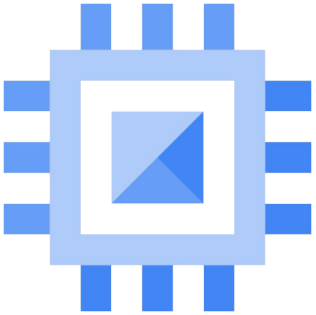
Suppose you have a workload that no human being is sitting around waiting to finish. Say, a batch job analyzing a large dataset. You can save money by choosing Preemptible VMs to run the job. A Preemptible VM is different from an ordinary Compute Engine VM in only one respect: you've given Compute Engine permission to terminate it if its resources are needed elsewhere. You can save a lot of money with preemptible VMs, although be sure to make your job able to be stopped and restarted.

You don't have to select a particular option or machine type to get high throughput

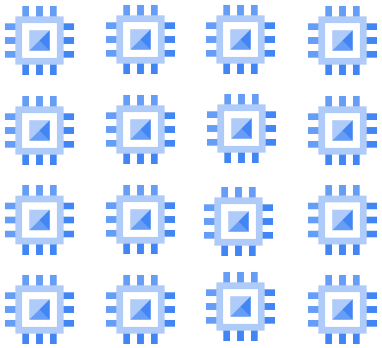
between your processing and your persistent disks. That’s the default.

You can choose the machine properties of your instances, such as the number of virtual CPUs and the amount of memory, by using a set of predefined machine types or by creating your own custom machine types.

Scale up or scale out with Compute Engine



Use big VMs for memory- and compute-intensive applications



Use Autoscaling for resilient, scalable applications

Google Cloud

You can make very large VMs in Compute Engine. At the time this deck was produced, the maximum number of virtual CPUs in a VM was zone-dependent and at 96, and the maximum memory size was at 624 GB (6.5 GB per CPU).

You can also use a mega-memory machine type that scales to 1.4 TB memory.

Check the Google Cloud website to see where these maximums are today.

These huge VMs are great for workloads like in-memory databases and CPU-intensive analytics. But most Google Cloud customers start off with scaling out, not up. Compute Engine has a feature called Autoscaling that lets you add and take away VMs from your application based on load metrics. The other part of making that work is balancing the incoming traffic among the VMs. And Google VPC supports several different kinds of load balancing! We'll consider those in the next section.

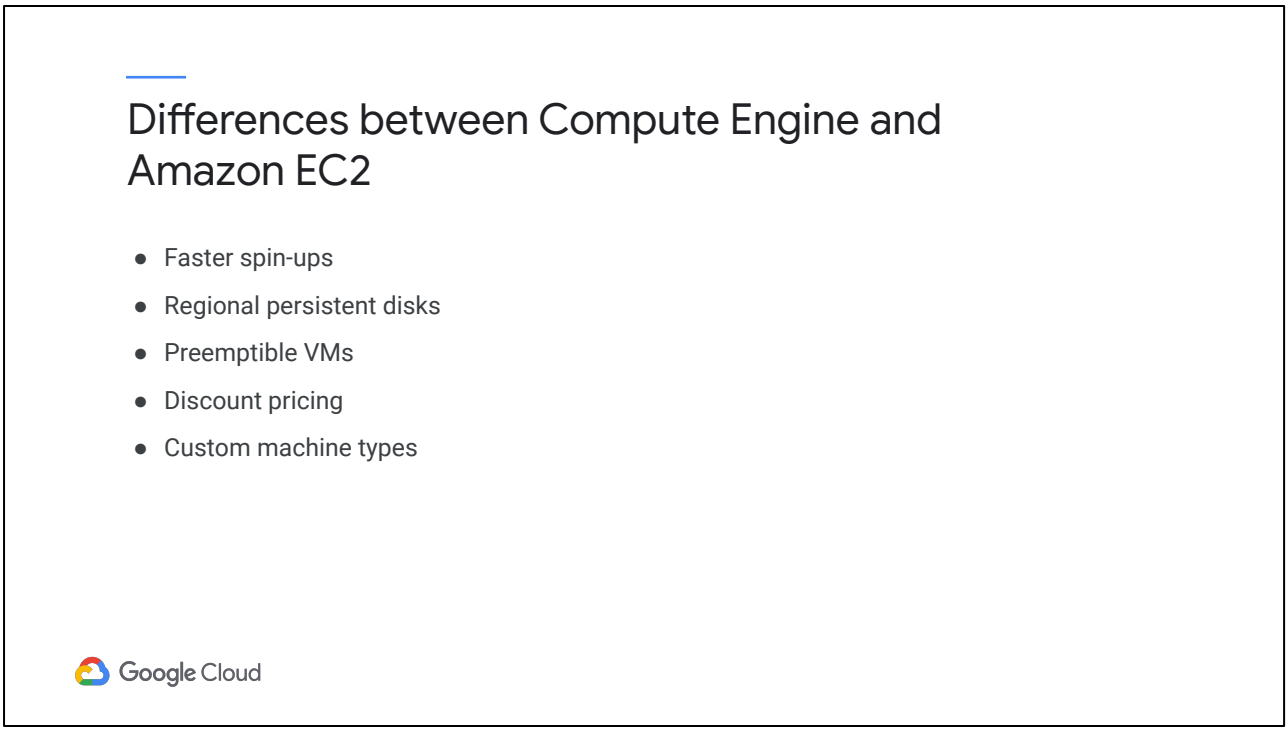
Similarities between Compute Engine and Amazon EC2

- RAM, CPU, and GPU
- Boot disk and operating system
- Additional disks
- IP addresses
- Startup scripts with metadata

Google Cloud

AWS and Google virtual machines have a lot in common.

- Both Compute Engine and Amazon Elastic Compute Cloud (Amazon EC2) allow you to choose and configure RAM, CPU, and GPU.
- Both offer a wide range of operating systems.
- Additional virtual disks can be added to both virtual machine types.
- Ephemeral and static public and private IP addresses can be used for both types of instances.
- Both instance types can use metadata and scripts for bootstrapping.



Let's look at some of the key differences between Compute Engine and Amazon EC2.

- It typically takes a Compute Engine instance about 30 seconds to start. An Amazon EC2 instance can take minutes.
- Google Cloud and AWS both offer block storage options as part of their compute services. Compute Engine provides persistent disks, and Amazon EC2 provides Elastic Block Store (EBS). Each service has several block storage types that cover a range of price and performance characteristics. Regional persistent disks provide durable storage and replication of data between two zones in the same region. If you are designing robust systems on Compute Engine, consider using regional persistent disks to maintain high availability for resources across multiple zones. AWS does not offer a regional persistent disk option.
- Amazon EC2 offers temporary instances called *spot instances,* and Compute Engine offers similar instances called *preemptible VMs*. Both provide similar functionality, but they have different pricing models. With Compute Engine, preemptible VMs pricing is fixed, although depending on the machine type, preemptible VM prices can be discounted to nearly 80% of the on-demand rate. If not reclaimed by Compute Engine, preemptible VMs run for a maximum of 24 hours and then are automatically terminated. Also, if you use a premium operating system with a license fee, you will be charged the full cost of the license while using that preemptible VM.
- Compute Engine and Amazon EC2 have similar on-demand pricing models for running instances. Both charge by the second with a minimum charge of one minute. In Amazon EC2, discounted pricing can be obtained by committing to

- provision reserved instances for one or three years. The more you pay up front, the greater the discount, subject to conditions. For Compute Engine instances, discounted pricing is obtained through sustained use, and the discount is automatically applied. The longer you use an instance in a given month, the greater the discount, with potential savings of as much as 30% of the standard on-demand rate.
- Although AWS has an extensive list of Amazon Machine Image, or AMI, options, you have to select a predefined instance to use. Google Cloud allows you to build a custom machine to your exact specification.

Summary of Compute Engine and Amazon EC2 differences

	Compute Engine	Amazon EC2
Machine RAM and CPU	Machine types	Instance types
Machine images	Images	Amazon Machine Images
Block storage	Persistent disks	Elastic Block Store
Local attached disk	Local SSD	Ephemeral drives
Discounts	Preemptible VMs, Sustained-use discounts Committed-use discounts	Spot Instances, Reserved Instances Savings Plan

Google Cloud

Let's summarize the key differences between Compute Engine and Amazon EC2.

- Compute Engine and Amazon EC2 both offer a variety of predefined instance configurations with specific amounts of virtual CPU, RAM, and network. Compute Engine refers to them as machine types; Amazon EC2 refers to these configurations as instance types.
- Compute Engine and Amazon EC2 both use machine images to create new instances. Amazon calls these images Amazon Machine Images (AMIs), and Compute Engine simply calls them images.
- Both offer block storage options as part of their compute services. Compute Engine provides persistent disks, and Amazon EC2 provides Elastic Block Store (EBS).
- On Compute Engine, local disks are referred to as local SSD and can be attached to almost any machine type. A maximum of 24 can be attached to a single instance. On Amazon EC2, local disks are called instance store or ephemeral store. These disks can be either HDD or SSD, depending on the instance type family. The number and size of these disks depends on the specific instance type and is not adjustable.
- Compute Engine and Amazon EC2 approach discount pricing in very different ways. Compute Engine offers discounts for preemptible VMs, sustained-use instances, and committed-use contracts. Amazon EC2 offers discounts for their temporary spot instances and by by provisioning reserved instances. Savings Plans offer substantial savings in exchange for a commitment to consistent amount of usage for a 1- or 3-year term but is more flexible than

- Reserved Instances.

Agenda

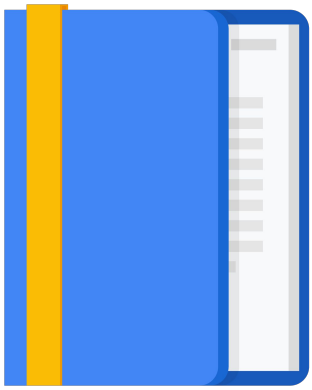
Virtual Private Cloud (VPC) Network

Compute Engine

Important VPC Capabilities

Quiz and Lab

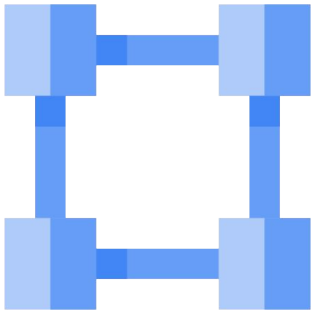
Resources



Google Cloud

You control the topology of your VPC network

- Use its route table to forward traffic within the network, even across subnets.
- Use its firewall to control what network traffic is allowed.
- Use Shared VPC to share a network, or individual subnets, with other Google Cloud projects.
- Use VPC Peering to interconnect networks in Google Cloud projects.



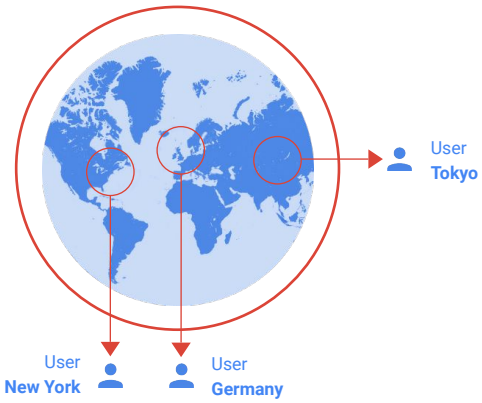
Google Cloud

Much like physical networks, VPCs have routing tables. These are used to forward traffic from one instance to another instance within the same network, even across subnetworks and even between Google Cloud zones, without requiring an external IP address. VPCs' routing tables are built in; you don't have to provision or manage a router.

Another thing you don't have to provision or manage for Google Cloud: a firewall. VPCs give you a global distributed firewall you can control to restrict access to instances, both incoming and outgoing traffic. You can define firewall rules in terms of metadata tags on Compute Engine instances, which is really convenient. For example, you can tag all your web servers with, say, "WEB," and write a firewall rule saying that traffic on ports 80 or 443 is allowed into all VMs with the "WEB" tag, no matter what their IP address happens to be.

Recall that VPCs belong to Google Cloud projects. But what if your company has several Google Cloud projects, and the VPCs need to talk to each other? If you simply want to establish a peering relationship between two VPCs, so that they can exchange traffic, configure VPC Peering does. On the other hand, if you want to use the full power of IAM to control who and what in one project can interact with a VPC in another, configure Shared VPC.

With global Cloud Load Balancing, your application presents a single front-end to the world



- Users get a single, global anycast IP address.
- Traffic goes over the Google backbone from the closest point-of-presence to the user.
- Backends are selected based on load.
- Only healthy backends receive traffic.
- No pre-warming is required.

Google Cloud

A few slides back, we talked about how virtual machines can autoscale to respond to changing load. But how do your customers get to your application when it might be provided by four VMs one moment and forty VMs at another? Cloud Load Balancing is the answer.

Cloud Load Balancing is a fully distributed, software-defined, managed service for all your traffic. And because the load balancers don't run in VMs you have to manage, you don't have to worry about scaling or managing them. You can put Cloud Load Balancing in front of all of your traffic: HTTP(S), other TCP and SSL traffic, and UDP traffic too.

With Cloud Load Balancing, a single anycast IP front-ends all your backend instances in regions around the world. It provides cross-region load balancing, including automatic multi-region failover, which gently moves traffic in fractions if backends become unhealthy. Cloud Load Balancing reacts quickly to changes in users, traffic, network, backend health, and other related conditions.

And what if you anticipate a huge spike in demand? Say, your online game is already a hit; do you need to file a support ticket to warn Google of the incoming load? No. No so-called "pre-warming" is required.

Google VPC offers a suite of load-balancing options

Global HTTP(S)	Global SSL Proxy	Global TCP Proxy	Regional	Regional internal
Layer 7 load balancing based on load.	Layer 4 load balancing of non-HTTPS SSL traffic based on load.	Layer 4 load balancing of non-SSL TCP traffic.	Load balancing of any traffic (TCP, UDP).	Load balancing of traffic inside a VPC.
Can route different URLs to different back ends.	Supported on specific port numbers.	Supported on specific port numbers.	Supported on any port number.	Use for the internal tiers of multi-tier applications.

Google Cloud

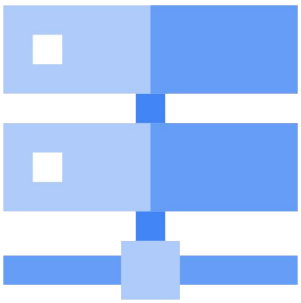
If you need cross-regional load balancing for a Web application, use HTTP(S) load balancing. For Secure Sockets Layer traffic that is not HTTP, use the Global SSL Proxy load balancer. If it's other TCP traffic that does not use Secure Sockets Layer, use the Global TCP Proxy load balancer.

Those two proxy services only work for specific port numbers, and they only work for TCP. If you want to load balance UDP traffic, or traffic on any port number, you can still load balance across a Google Cloud region with the Regional load balancer.

Finally, what all those services have in common is that they're intended for traffic coming into the Google network from the Internet. But what if you want to load balance traffic inside your project, say, between the presentation layer and the business layer of your application? For that, use the Internal load balancer. It accepts traffic on a Google Cloud internal IP address and load balances it across Compute Engine VMs.

Cloud DNS is highly available and scalable

- Create managed zones, then add, edit, delete DNS records.
- Programmatically manage zones and records using RESTful API or command-line interface.



Google Cloud

One of the most famous Google services that people don't pay for is 8.8.8.8, which provides a public Domain Name Service to the world. DNS is what translates Internet hostnames to addresses, and as you would imagine, Google has a highly developed DNS infrastructure. It makes 8.8.8.8 available so that everybody can take advantage of it.

But what about the Internet hostnames and addresses of applications you build in Google Cloud?
Google Cloud offers Cloud DNS to help the world find them. It's a managed DNS service running on the same infrastructure as Google. It has low latency and high availability, and it's a cost-effective way to make your applications and services available to your users. The DNS information you publish is served from redundant locations around the world.

Cloud DNS is also programmable. You can publish and manage millions of DNS zones and records using the Cloud Console, the command-line interface, or the API.

Cloud CDN

- Use Google's globally distributed edge caches to cache content close to your users.
- Or use CDN Interconnect if you'd prefer to use a different CDN.

Google Cloud

Google has a global system of edge caches. You can use this system to accelerate content delivery in your application using Cloud CDN. Your customers will experience lower network latency, the origins of your content will experience reduced load, and you can save money too. Once you've set up HTTP(S) Load Balancing, simply enable Cloud CDN with a single checkbox.

There are lots of other CDNs out there, of course. If you are already using one, chances are, it is a part of Google Cloud's CDN Interconnect partner program, and you can continue to use it.

Google Cloud offers many interconnect options

Connection	Provides	Capacity	Requirements	Access type
IPsec VPN tunnels	Encrypted tunnel to VPC networks through the public internet	1.5 - 3.0 Gbps per tunnel	On-premises VPN gateway	Internal IP addresses
Direct Peering	Dedicated, direct connection to Google's network	10 Gbps per link	Connection in Google Cloud PoPs	Public IP addresses
Carrier Peering	Peering through a service provider to Google's public network	Varies based on partner offering	Service provider	Public IP addresses
Dedicated Interconnect	Dedicated, direct connection to VPC networks	8 x 10 Gbps circuits, or 2 x 100 Gbps circuits per connection	Connection in a colocation facility	Internal IP addresses
Partner Interconnect	Dedicated bandwidth, connection to VPC network through a service provider	50 Mbps – 10 Gbps per connection	Service provider	Internal IP addresses

Google Cloud

Lots of Google Cloud customers want to interconnect their other networks to their Google VPCs. such as on-premises networks or their networks in other clouds. There are many good choices.

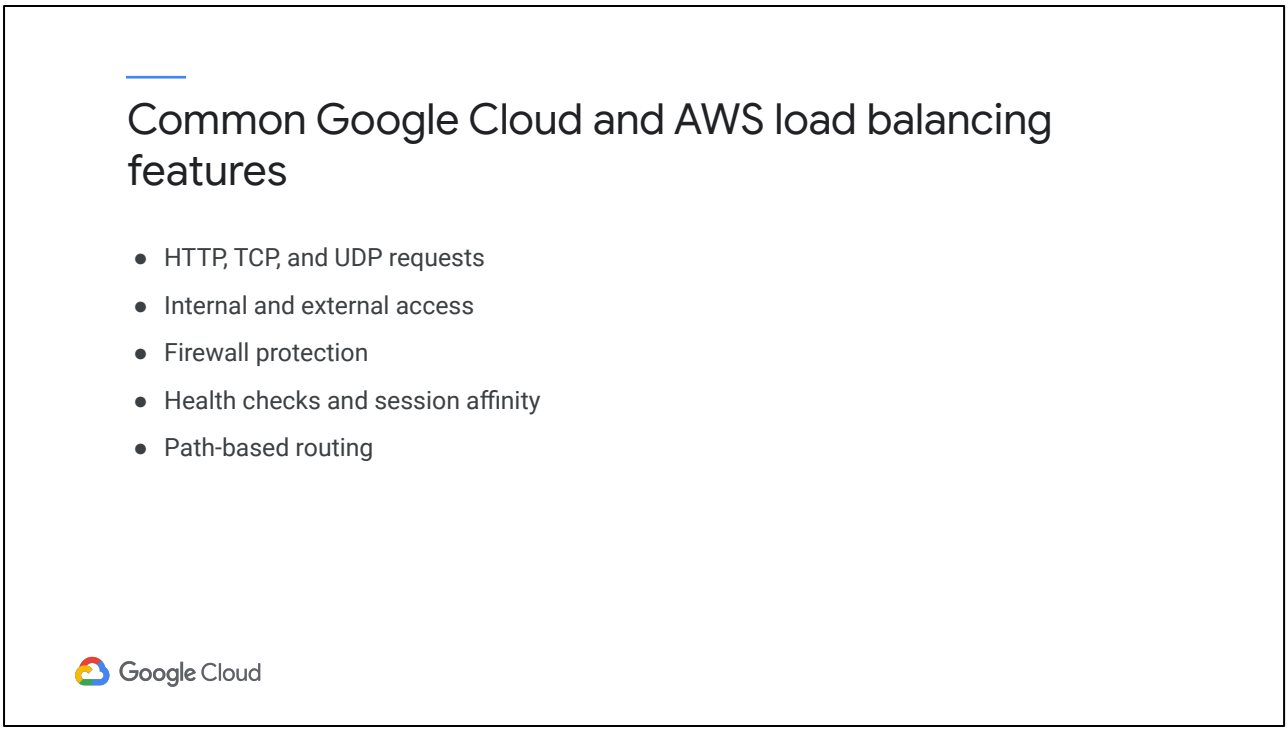
Many customers start with a Virtual Private Network connection over the Internet, using the IPsec protocol. To make that dynamic, they use a Google Cloud feature called Cloud Router. Cloud Router lets your other networks and your Google VPC exchange route information over the VPN using the Border Gateway Protocol. For instance, if you add a new subnet to your Google VPC, your on-premises network will automatically get routes to it.

But some customers don't want to use the Internet, either because of security concerns or because they need more reliable bandwidth. They can consider peering with Google using Direct Peering. Peering means putting a router in the same public datacenter as a Google point of presence and exchanging traffic. Google has more than 100 points of presence around the world. Customers who aren't already in a point of presence can contract with a partner in the Carrier Peering program to get connected.

One downside of peering, though, is that it isn't covered by a Google Service Level Agreement. Customers who want the highest uptimes for their interconnection with

Google should use Dedicated Interconnect, in which customers get one or more direct, private connections to Google. If these connections have topologies that meet Google’s specifications, they can be covered by up to a 99.99% SLA. These connections can be backed up by a VPN for even greater reliability.

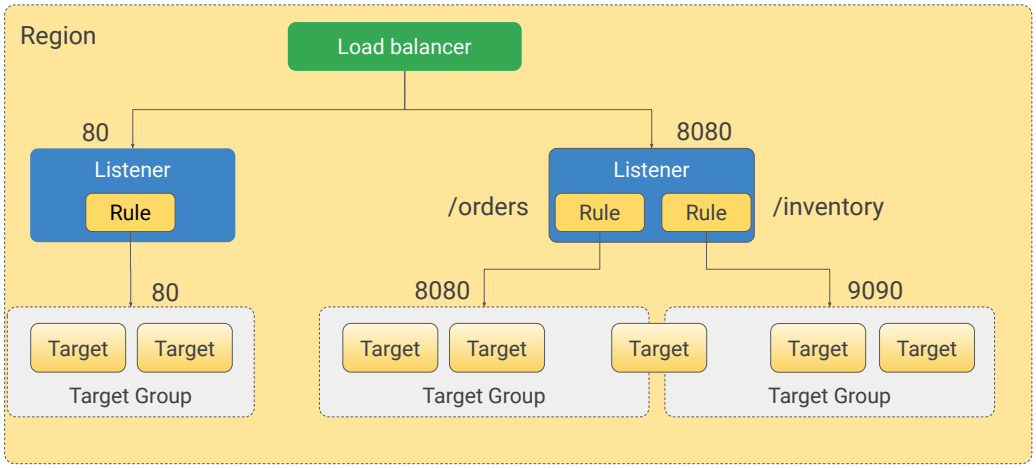
Partner Interconnect provides connectivity between your on-premises network and your VPC network through a supported service provider. A Partner Interconnect connection is useful if your data center is in a physical location that can't reach a Dedicated Interconnect colocation facility or if your data needs don't warrant an entire 10 Gbps connection. Depending on your availability needs, you can configure Partner Interconnect to support mission-critical services or applications that can tolerate some downtime. As with Dedicated Interconnect, if these connections have topologies that meet Google’s specifications, they can be covered by up to a 99.99% SLA, but note that Google is not responsible for any aspects of Partner Interconnect provided by the third party service provider nor any issues outside of Google's network.



There are several common features between Google and Amazon when comparing load balancing.

- Requests can be in HTTP, TCP, or UDP.
- Requests can come from the internet or from internal resources.
- The load balancers can be positioned behind a firewall to limit client access.
- Both support health checks and session affinity.
- In addition, Google's HTTP load balancer and Amazon's Application Load Balancer support path-based routing to micro-services, so we'll focus on them.

AWS load balancers



Google Cloud

AWS load balancers are a managed service that is instance-based, and the load balancer can take minutes to scale. They are built within a region and can only distribute traffic in the region. So, if you are expecting a significant amount of traffic to reach the load balancer at a predetermined time, it may be necessary to open a ticket with AWS support to pre-warm a load balancer to handle the traffic.

Elastic Load Balancers can handle HTTP(S), TCP, and application traffic.

Summary of Google Cloud and AWS load balancing approaches

	Google Cloud load balancers	AWS load balancers
Service type	Software-based	Instance-based
Managed service	Global	Regional
Request routing	URL map (HTTP only)	Listener, listener rule
Service health check	Instance group, Backend service (capacity)	Target group
Load balanced scope	Global	Region*

Google Cloud

Let's summarize the key differences between the approaches taken by Google Cloud and AWS to load balancing.

- Google Cloud load balancers are software-based. AWS load balancers run on EC2 instances. The difference in service type is the reason that you don't have to pre-warm load balancers in Google Cloud. The service can spin up quickly because a virtual machine is not used. An EC2 instance can take minutes to start. So if you are expecting a heavy amount of traffic in a short period of time on AWS, you may have to open a support ticket to manually start more load balancer virtual machines.
- Google Cloud load balancers are a global managed service. The load balancer can survive a region outage because the service is not built on a region. The service is built on the network edges. AWS load balancers are built in the regions. So, a region that goes down can take the load balancer with it.
- A Google Cloud load balancer can route traffic based on a URL for HTTP only. The URL can direct traffic to a particular backend service. AWS can route traffic based on a listener and a listener rule for HTTP and TCP.
- Both Google Cloud and AWS leverage health checks to ensure that traffic is only sent to healthy instances and can autoheal, which is an option to destroy virtual machines with new instances. But Google Cloud health checks can be used on both the load balancer, via the backend service, and the instance group. AWS only applies the health check on the target group.

- Google Cloud offers load balancers that can distribute traffic both globally and within a region. AWS load balancers can only distribute traffic within a region. To achieve load balancing at a global scale in AWS you would need to combine regional load balancers with AWS Global Accelerator

Agenda

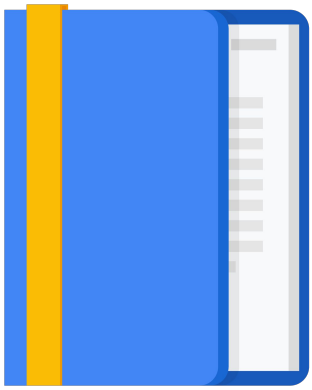
Virtual Private Cloud (VPC) Network

Compute Engine

Important VPC Capabilities

Quiz and Lab

Resources



Google Cloud

Quiz 1

Name 3 robust networking services available to your applications on Google Cloud.

Quiz 1

Name 3 robust networking services available to your applications on Google Cloud.

- Cloud Virtual Network
- Cloud Interconnect
- Cloud DNS
- Cloud Load Balancing
- Cloud CDN

Quiz 2

Name three Compute Engine pricing features.

Quiz 2

Name three Compute Engine pricing features.

- Per-second billing
- Custom machine types
- Preemptible instances

Quiz 3

True or False: Cloud Load Balancing lets you balance HTTP traffic across multiple Compute Engine regions.

A. True

B. False

Quiz 3

True or False: Cloud Load Balancing lets you balance HTTP traffic across multiple Compute Engine regions.

A. True

B. False

The objectives for this lab are to:

- Create a Compute Engine virtual machine using the Cloud Console.
- Create a Compute Engine virtual machine using the gcloud command-line interface.
- Connect between the two instances.

Agenda

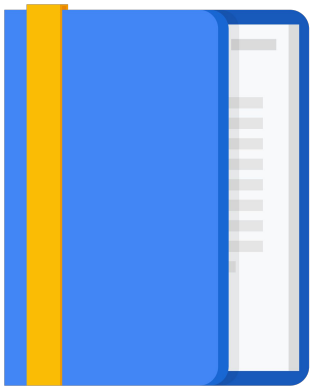
Virtual Private Cloud (VPC) Network

Compute Engine

Important VPC Capabilities

Quiz and Lab

Resources



Google Cloud

Resources

Compute Engine https://cloud.google.com/compute/docs/

Virtual Private Cloud https://cloud.google.com/compute/docs/vpc/

Google Cloud operations suite https://cloud.google.com/stackdriver/docs/

gcloud tool guide https://cloud.google.com/source-repositories/docs/


Google Cloud