Networking in GCP

Topics to cover:

- Global vs. Regional vs. Zonal resources.
- Networking tiers
- Virtual Private Clouds (VPCs).
- IP addressing & Subnets CIDRs (optional).
- Firewall rules.

Global vs. Regional vs. Zonal resources

Global	Regional	Zonal
Global resources are accessible by any resource in any zone within the same project	Regional resources are accessible by any resources within the same region	Resources that are unique to a zone and are only usable by other resources in the same zone
• Addresses • Images • Snapshots • Instance templates • VPC network • Firewalls • Routes	• Addresses • Subnets • Regional MIGs • Regional disks	• Instances • Persistent disks • Machine types • Zonal MIGs

Global, regional, and zonal resources | Compute Engine Documentation | Google Cloud

Networking tiers

Optimize cloud network for performance or price.
GCP is the first major public cloud to offer a tiered cloud network.

Premium
Give users exceptional high performing network experience by using Google's global network.

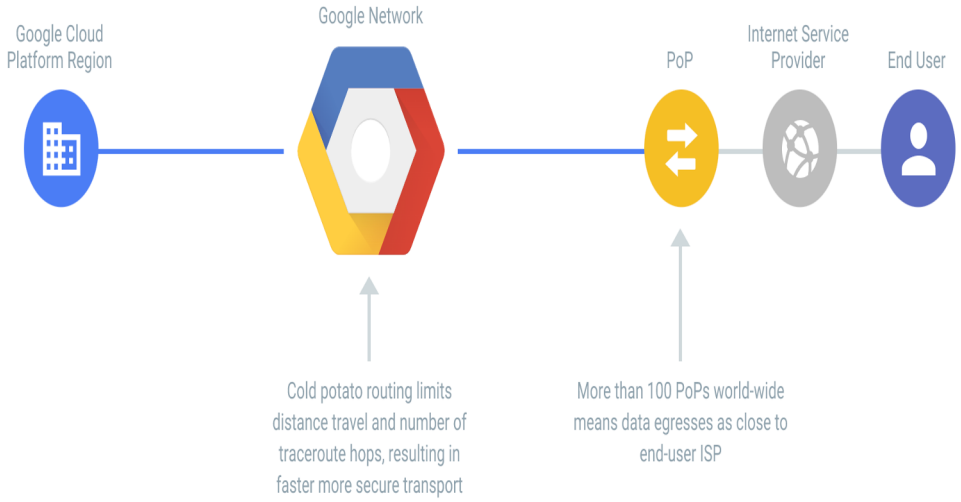
VS

Standard
Get control over network costs, while still delivering performance comparable with other cloud providers.

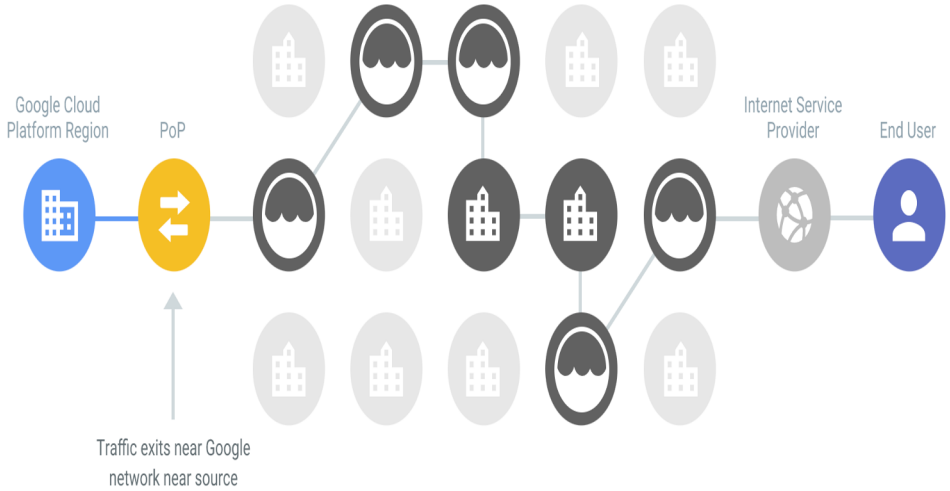
Category	Premium	Standard
Network	High performance routing	Lower performance network
Network Services	Network services such as Cloud Load Balancing are global (single VIP for backends in multiple regions)	Network services such as Cloud Load Balancing are regional (one VIP per region)
Service Level	High performance and reliability	Performance and availability comparable to other public cloud providers (lower than premium)
Use Case	Performance, reliability, global footprint and user experience are your main considerations	Cost is your main consideration, and you're willing to trade-off some network performance

Networking tiers - Cont.

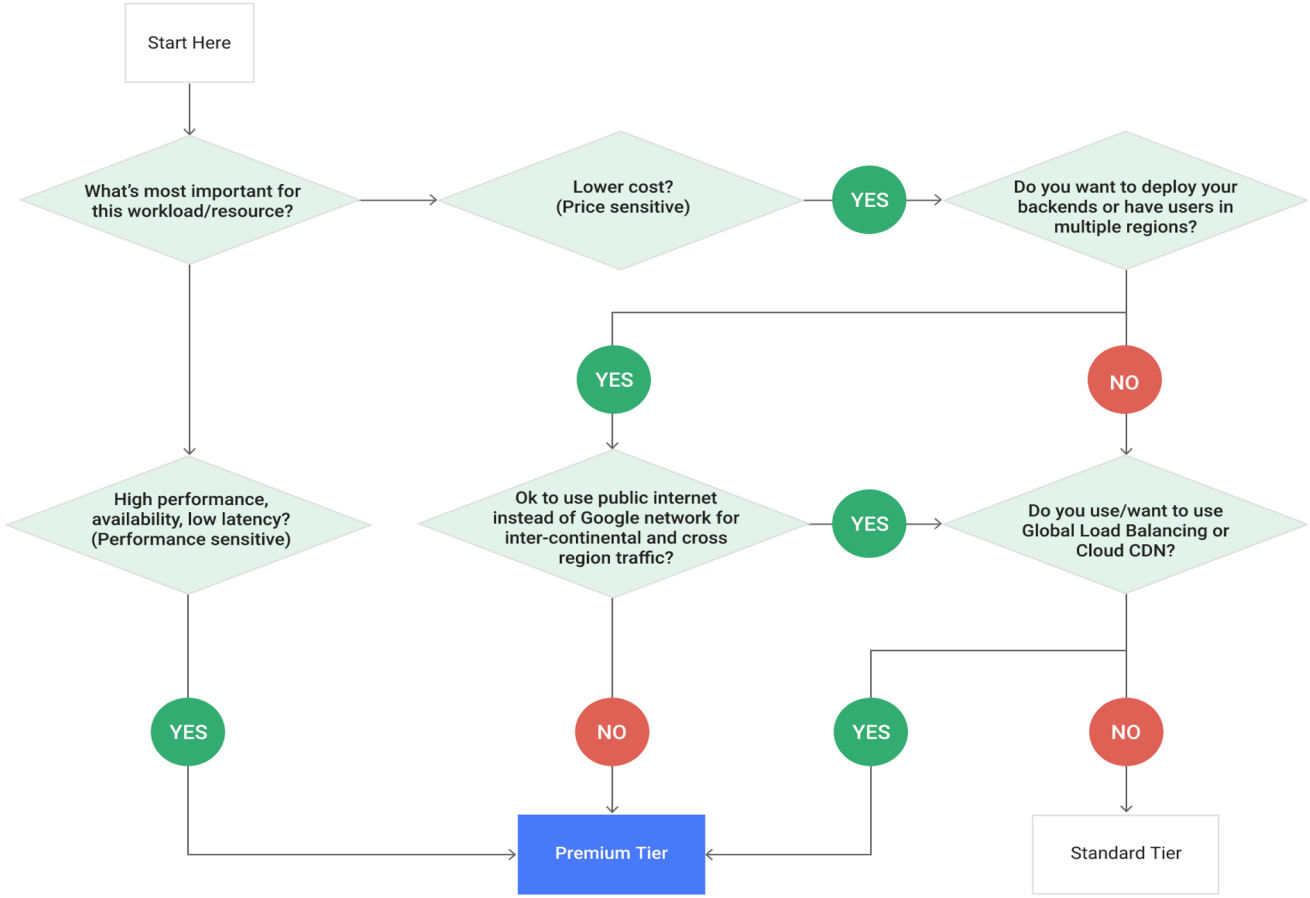
Premium Tier delivers GCP traffic over Google's well-provisioned, low latency, highly reliable global network. This network consists of an extensive global private fiber network with over 100 points of presence (POPs) across the globe. By this measure, Google's network is the largest of any public cloud provider.



Standard Tier delivers GCP traffic over a transit ISP's network with the latency and reliability typical of transit ISPs, and with a network quality comparable to that of other public clouds, at a lower price than our Premium Tier. Also provide only regional network services in Standard tier, such as the new regional Cloud Load Balancing service.



Networking tiers - Cont.

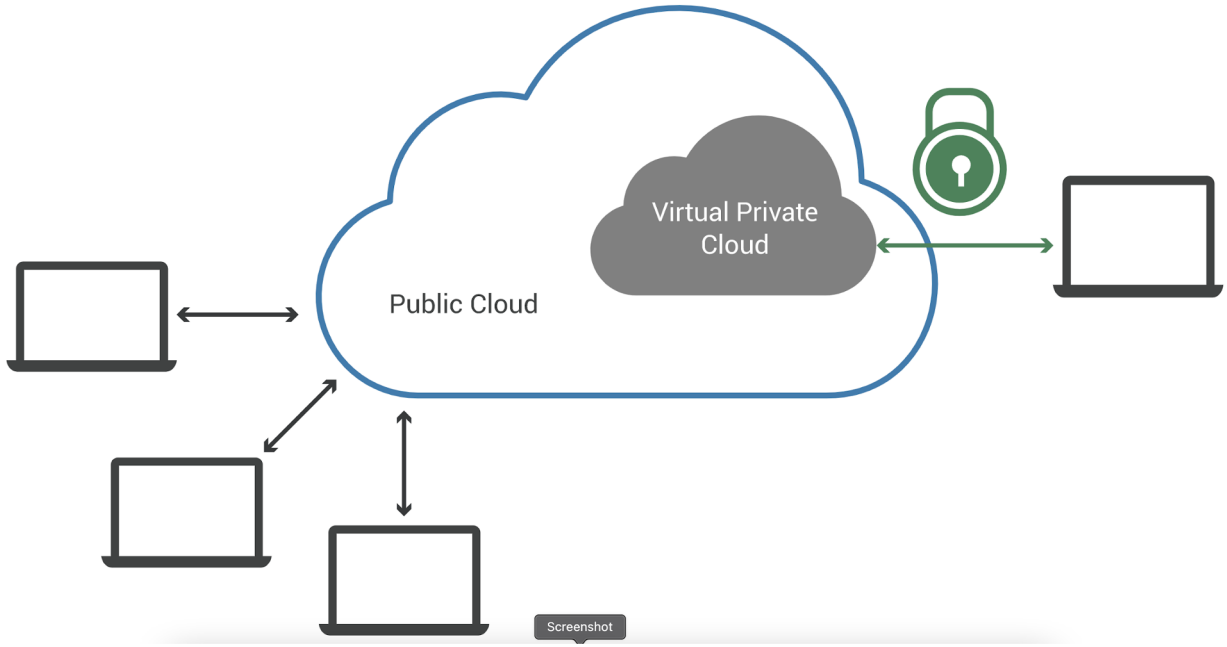


Networking tiers - Cont.

Features	Premium Tier	Standard Tier
Plain VM instance	Yes - Regional	Yes - Regional
HTTP(S) Load Balancing (LB)	Yes - Global only	Yes - Regional
TCP/SSL Proxy LB (non-HTTP traffic)	Yes - Global only	Yes - Regional
Network / Internal LB	Yes - Regional VIP (+ Client can be anywhere)	Yes - Regional VIP (+ Client needs to be in same region)
Google Cloud Storage, Google Kubernetes Engine	Yes	Yes - Regional but only via LB
Cloud CDN	Yes	No
Cloud VPN/Cloud Router	Yes	No

Virtual Private Clouds (VPCs)

A VPC isolates computing resources from the other computing resources available in the public cloud.



Virtual Private Clouds (VPCs)

```
gcloud compute networks create NETWORK \
    --subnet-mode=auto \
    --bgp-routing-mode=DYNAMIC_ROUTING_MODE \
    --mtu=MTU
```

```
gcloud compute networks subnets create SUBNET \
    --network=NETWORK \
    --range=PRIMARY_RANGE \
    --region=REGION
```

[Using VPC networks | Google Cloud]

Firewall rules

Ingress (inbound) rule					
Priority	**Action**	**Enforcement**	**Target (defines the destination)**	**Source**	**Protocols and ports**
Integer from `0` to `65535`, inclusive; default `1000`	`allow` or `deny`	`enabled` (default) or `disabled`	The *target* parameter specifies the destination; it can be one of the following: • All instances in the VPC network • Instances by network tag • Instances by service account	One of the following: • Range of IPv4 addresses; default is any (`0.0.0.0/0`) • Instances by network tag • Instances by service account • Range of IPv4 address *and* instances by network tag • Range of IPv4 address *and* instances by service account	Specify a protocol or a protocol and a destination port. If not set, the rule applies to all protocols and destination ports.

Egress (outbound) rule					
Priority	**Action**	**Enforcement**	**Target (defines the source)**	**Destination**	**Protocols and ports**
Integer from `0` to `65535`, inclusive; default `1000`	`allow` or `deny`	`enabled` (default) or `disabled`	The *target* parameter specifies the source; it can be one of the following: • All instances in the VPC network • Instances by network tag • Instances by service account	Any network or a specific range of IPv4 addresses; default is any (`0.0.0.0/0`)	Specify a protocol or a protocol and a destination port. If not set, the rule applies to all protocols and destination ports.

Firewall rules

EXAMPLES

To create a firewall rule allowing incoming TCP traffic on port 8080, run:

```
$ gcloud compute firewall-rules create FooService --allow=tcp:8080
    --description="Allow incoming traffic on TCP port 8080" --direction=INGRESS
```

To create a firewall rule that allows TCP traffic through port 80 and determines a list of specific IP address blocks that are allowed to make inbound connections, run:

```
$ gcloud compute firewall-rules create "tcp-rule" --allow=tcp:80
    --source-ranges="10.0.0.0/22,10.0.0.0/14" --description="Narrowing TCP traffic"
```

To list existing firewall rules, run:

```
$ gcloud compute firewall-rules list
```

Notes on Firewall rules priorities

- Priority is a numerical value which determines whether the rule is applied. Only the highest priority (lowest priority number) rule whose other components match traffic is applied; conflicting rules with lower priorities are ignored.

- The highest priority rule applicable to a target for a given type of traffic takes precedence. Target specificity does not matter. For example, a higher priority ingress rule for certain destination ports and protocols intended for all targets overrides a similarly defined rule with lower priority for the same destination ports and protocols intended for specific targets.

- A rule with a deny action overrides another with an allow action *only if the two rules have the same priority.* Using relative priorities, it is possible to build allow rules that override deny rules, and deny rules that override allow rules.

Notes on Firewall rules priorities - cont.

Consider the following example where two firewall rules exist:

- An ingress rule from sources `0.0.0.0/0` (anywhere) applicable to all targets, all protocols, and all destination ports, having a `deny` action and a priority of `1000`.
- An ingress rule from sources `0.0.0.0/0` (anywhere) applicable to specific targets with the tag `webserver`, for traffic on TCP 80, with an `allow` action.

The priority of the second rule determines whether TCP traffic to port 80 is allowed for the `webserver` targets:

- If the priority of the second rule is set to a number *greater than* `1000`, it has a *lower* priority, so the first rule denying all traffic applies.
- If the priority of the second rule is set to `1000`, the two rules have identical priorities, so the first rule denying all traffic applies.
- If the priority of the second rule is set to a number *less than* `1000`, it has a *higher* priority, thus allowing traffic on TCP 80 for the `webserver` targets. Absent other rules, the first rule would still deny other types of traffic to the `webserver` targets, and it would also deny all traffic, including TCP 80, to instances *without* the `webserver` tag.

The previous example demonstrates how you can use priorities to create selective `allow` rules and global `deny` rules to implement a security best practice of least privilege.

Google Compute Engine (VMs)

Topics to cover:

- What is Google Compute Engine
- Create and manage VMs from cloud console and using gcloud tool.
- Accessing a VM.
- VMs types.
- Attaching additional disks and GPUs.
- Creating snapshots and custom images.
- Instance groups (IGs)

What is Compute Engine?

Customizable virtual machines in Google Cloud

Compute Engine - cont.

- Create and manage VMs from cloud console and using gcloud tool.
- Accessing a VM.
- VMs types.
- Attaching additional disks and GPUs.
- Creating snapshots and custom images.

gcloud compute instances create | Cloud SDK Documentation

Instance groups (IGs):

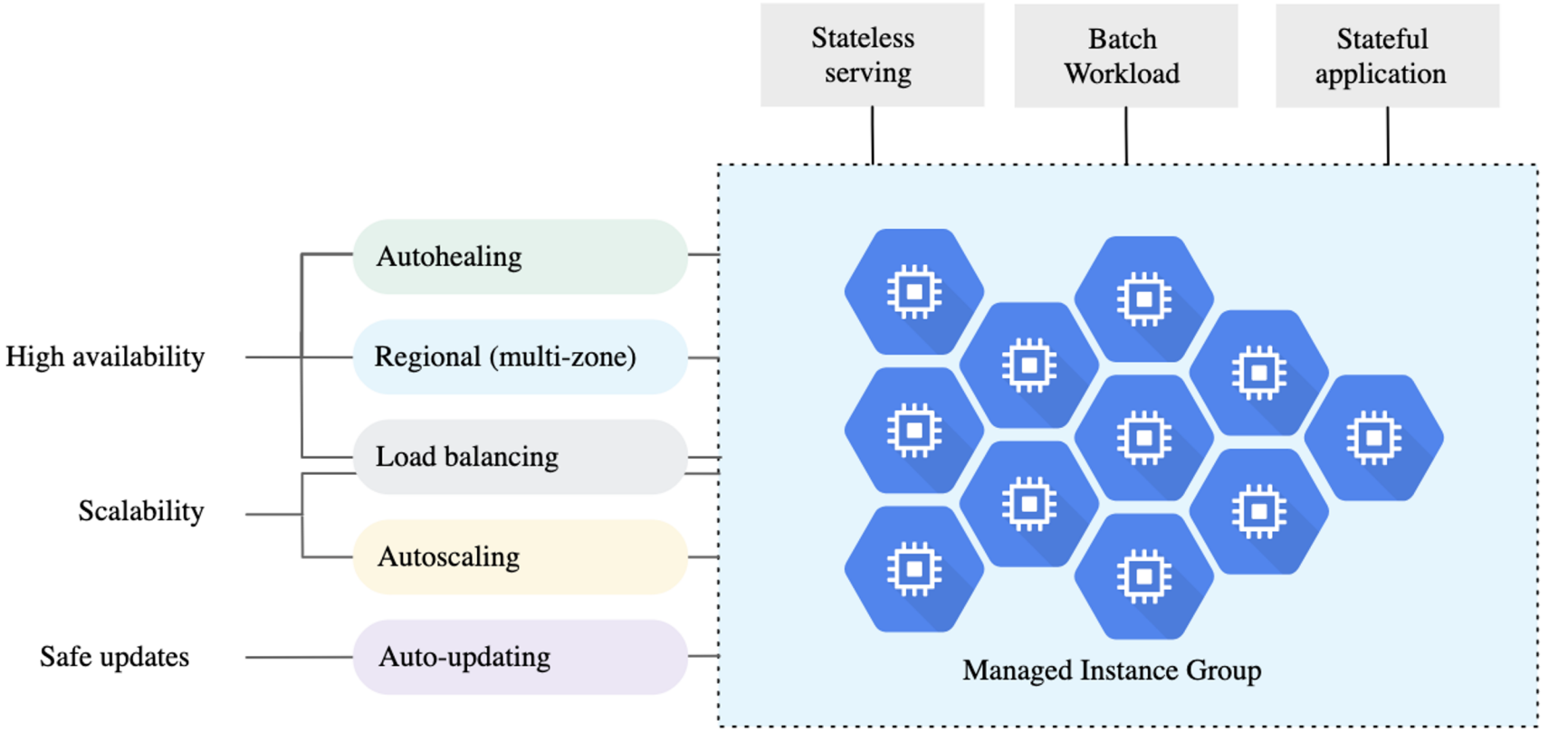
Types of Instance Groups:

- Managed instance groups (MIGs)
- Unmanaged instance groups

`

Creating MIGs - Demo

Managed instance groups (MIGs):



Lab 2.1

1. From Cloud console, create a VPC named “auto-vpc” with auto-mode enabled,
   How many subnets created?
2. From Cloud console, create a VPC named “custom-vpc” with auto-mode disabled and create two subnets.
3. Using gcloud tool list all available VPCs and list subnets of each VPC.
4. Block internet access from you VPC using firewall rules.
5. Create a firewall rule to allow incoming SSH requests from internet to all instances in your vpc.
6. Modify the previous firewall rule to allow only ssh requests coming through Google’s IAP servers.

Lab 2.2

1. Create a VM with public ip then:
   - In two different ways, SSH into this VM.
   - Enforce SSH into this VM to be IAP protected.
2. Create a VM **without** public ip then:
   - SSH into this vm.
   - update system packages (is it possible?)
3. Create a VM **with** public ip then:
   - SSH into this vm
   - Update system packages.
   - Setup Nginx Web Server and test your setup.
   - Create a custom image from this VM named “custom-img-nginx”.
4. Create MIG (min 3 and max 5) of a template using the custom image “custom-img-nginx”.