In Google Cloud Platform (GCP), every **VM instance** can have two types of IP addresses, depending on its configuration:

**1. Internal IP (Private IP)**

* **Definition**: An IP address assigned from the VPC’s subnet range.
* **Scope**: Reachable **only within the same VPC network**, and also across connected VPCs (via VPC peering, Shared VPC, or VPN/Interconnect).
* **Usage**: Used for communication between VMs/services inside your GCP network without exposing them to the public internet.
* **Example**: 10.128.0.5

**2. External IP (Public IP)**

* **Definition**: An internet-routable IP address assigned to the VM.
* **Scope**: Reachable **from the public internet** (if firewall rules allow it).
* **Usage**: Required if you want to SSH/RDP directly from the internet, or expose apps/services (like a web server) publicly.
* **Types**:
  + **Ephemeral**: Temporary, assigned when VM starts, can change if stopped/restarted.
  + **Static**: Reserved and permanent, doesn’t change even if the VM restarts.
* **Example**: 34.120.55.77

✅ **Key Notes:**

* A VM **must** have an internal IP, but an external IP is **optional**.
* For security best practices:
  + Use only **internal IPs** for communication between services in your VPC.
  + Place services behind a **load balancer** or **Cloud NAT** instead of giving each VM an external IP.

**Instance Groups**

**1. Instance Groups**

* A collection of VM instances that you manage as a single unit.
* Two types:
  + **Managed instance groups (MIGs)** → All VMs are identical, created from an instance template. Great for scaling apps and auto-healing.
  + **Unmanaged instance groups** → Different VMs grouped for convenience, no automatic scaling.
* **Example**: A web app with 5 identical VMs in a load balancer backend → use a **MIG**.

**2. Health Checks**

* Tests used by load balancers or instance groups to check if a VM is **healthy and serving traffic**.
* If a VM fails health checks, traffic is redirected to healthy ones.
* **Example**: HTTP health check on port 80 for a web server.

**VM Manager**

**3. Patch**

* Helps keep VM OS and software up to date by automating patch management.
* Schedules and applies updates across multiple VMs at once.
* **Example**: Apply security updates to 50 VMs at midnight automatically.

**4. OS Policies**

* Policies to enforce **OS configurations** (like required packages, firewall rules, or agent installs).
* Ensures all VMs meet compliance or security standards.
* **Example**: Enforce that all Linux VMs must have the nginx package installed.

**Bare Metal Solution**

This is for **special workloads that can’t easily run on virtual machines**, like Oracle databases or highly regulated enterprise apps. Google provides **dedicated physical servers (bare metal)** in their data centers but connected to GCP services.

**5. Servers**

* Physical bare metal servers you provision (instead of virtual VMs).
* Used for workloads like Oracle databases, SAP, or apps that need dedicated hardware.

**6. Networks**

* Networking setup for bare metal servers (similar to VPC for VMs).
* Lets bare metal servers communicate securely with GCP resources.

**7. VRFs (Virtual Routing and Forwarding) 🆕**

* Network virtualization feature that allows multiple isolated routing tables on the same physical network.
* Useful for multi-tenant setups or separating traffic between workloads.
* **Example**: Isolating database network traffic from app server traffic.

**8. Volumes**

* Storage volumes attached to bare metal servers.
* Similar to persistent disks for VMs, but optimized for bare metal workloads.

**9. NFS Shares**

* Network File System (NFS) storage shares that bare metal servers can access.
* Useful for shared storage across multiple servers.

**10. Procurements 🆕**

* Used for ordering, tracking, and managing bare metal resources (servers, storage, networking).
* Basically the "billing + ordering" section for Bare Metal Solution.

**11. Maintenance Events**

* Shows planned maintenance or outages for your bare metal servers.
* Helps you plan downtime or migrations.

⚡ **In summary**:

* **Instance Groups** = group of VMs + auto-healing/load balancing.
* **VM Manager** = automate OS updates and enforce compliance.
* **Bare Metal Solution** = dedicated physical servers + storage + networking for enterprise workloads (Oracle, SAP, etc.).

**Settings Section**

**1. Metadata**

* Key–value data you can assign at **project level** or **VM level**.
* Often used for configuration, startup scripts, or passing information to VMs.
* **Example**: Store an SSH key in project metadata → all VMs in the project allow that SSH login.

**2. Zones**

* Each GCP region has multiple **zones** (isolated data centers).
* When creating a VM, you pick a **zone** (like us-central1-a).
* Knowing zones is important for availability and failover planning.
* **Example**: Run one VM in us-central1-a and another in us-central1-b for redundancy.

**3. Network Endpoint Groups (NEGs)**

* A collection of network endpoints (IP + port pairs) used by Google Cloud Load Balancers.
* Types:
  + **Zonal NEG** → points to VMs in a zone.
  + **Internet NEG** → points to external services outside GCP.
  + **Serverless NEG** → points to Cloud Run, Cloud Functions, or App Engine services.
* **Example**: A load balancer uses a **serverless NEG** to send traffic to a Cloud Run backend.

**4. Operations**

* Logs of **all tasks and activities** performed in Compute Engine (like creating a VM, snapshot, disk, etc.).
* Shows status (pending, running, done, failed).
* Useful for debugging or tracking what happened in your project.
* **Example**: Check if your “disk snapshot creation” finished successfully.

**5. Settings**

* General **Compute Engine service-wide settings**.
* Includes things like:
  + Default service account used by VMs
  + Default network settings
  + Resource usage reports
* **Example**: Change the default project-wide service account that all new VMs use.

⚡ **In summary**:

* **Metadata** → pass configs to VMs.
* **Zones** → physical locations for your VMs.
* **NEGs** → backend groups for load balancers (VMs, serverless, external).
* **Operations** → task logs/history.
* **Settings** → global Compute Engine defaults.