1. VPC (Virtual Private Cloud)

What It Is:

A **VPC** is your **own isolated network within AWS**, similar to your **own private data center**.

Core Concepts:

- You define **IP range** (via CIDR block, e.g., `10.0.0.0/16`)
- Inside this VPC, you can create **subnets** (smaller slices of your IP range)
- It's fully **isolated**, but you can open access as needed (to internet, other VPCs, etc.)

Why Use It:

- Complete control over your networking (IP ranges, routing, firewalls)
- Secure and isolated cloud environment

---

2. Subnet

What It Is:

A **Subnet** is a **smaller segment of your VPC's IP range**—a way to **organize** and **isolate resources**.

Types:

- **Public Subnet**: Can talk to the **internet**
- **Private Subnet**: **No direct internet access**

Why It Matters:

- Helps you group resources (e.g., DBs in private subnet, web servers in public)
- IP allocation and routing control

---

3. Internet Gateway (IGW)

What It Is:

An **Internet Gateway** is what allows traffic to/from the internet into your VPC.

How It Works:

- Attach it to your VPC
- Add a **route** in your **subnet's route table** pointing to it (`0.0.0.0/0 → IGW`)
- Only then can instances **in public subnets** talk to the internet

---

4. NAT Gateway / NAT Instance

What It Is:

NAT = Network Address Translation Allows **instances in private subnets** to **initiate outbound connections** (e.g., to download updates from the internet) but **blocks inbound traffic**.

Why Use NAT:

- Private instances need software updates or make API calls—NAT helps without exposing them.

Types:

- **NAT Gateway**: Managed, scalable, easier
- **NAT Instance**: EC2-based, cheaper, more customizable but more operational work

---

5. Security Group (SG)

What It Is:

Virtual firewall attached to **instances** (EC2, RDS, etc.)

Key Traits:

- **Stateful**: If you allow **inbound**, return traffic is automatically allowed
- Applied **at instance level**

- Works with **allow rules only** (no deny)

Example Rule:

Allow inbound TCP on port 22 (SSH) from your IP only.

---

6. NACL (Network Access Control List)

What It Is:

Firewall at the subnet level

Key Traits:

- **Stateless**: Must explicitly allow both **inbound and outbound** rules
- Supports **allow and deny**
- Rules are evaluated **top to bottom**, first match wins

When to Use NACL vs SG:

- Use **Security Groups** for day-to-day instance-level control
- Use **NACLs** for **broad subnet-wide rules**, like blocking a bad IP range

---

7. Route Table

What It Is:

Determines **how traffic flows** inside/outside your subnets.

Example:

- Public Subnet → Route `0.0.0.0/0` to **IGW**
- Private Subnet → Route `0.0.0.0/0` to **NAT Gateway**

Each subnet must be associated with one route table.

---

8. DHCP Options Set

What It Is:

Configures things like **domain name servers (DNS)** inside your VPC.

Defaults:

AWS provides default DNS, but you can customize (e.g., your own internal DNS servers).

---

9. VPC Peering

What It Is:

Connects **two VPCs**, enabling traffic between them.

Caveat:

No **transitive peering** (A connected to B and B connected to C does **not** mean A connects to C)

---

10. Transit Gateway

What It Is:

Central hub to connect **multiple VPCs and on-prem networks**

Use Case:

If you have 10+ VPCs, use Transit Gateway instead of complex mesh of peering connections.

---

11. Endpoints (Interface & Gateway)

What They Are:

Let your VPC talk to AWS services **without going over the public internet**

Types:

- **Interface Endpoint**: Connects to most AWS services via **ENI (Elastic Network Interface)**
- **Gateway Endpoint**: Only for S3 and DynamoDB; adds route in route table

---

12. Elastic IP

What It Is:

A **static public IP** you can assign to EC2 or NAT Gateway

Use Case:

Your EC2 needs a **fixed public IP**

---

13. VPC Flow Logs

What It Is:

Logs of **network traffic** going in/out of your VPC/subnets/instances

Use Case:

Audit, troubleshoot, or analyze traffic patterns

---

14. Elastic Network Interface (ENI)

What It Is:

A virtual **network interface** you can attach to EC2 (can have multiple ENIs)

Use Case:

For high availability, failover, or running multiple services on different subnets

---

Recap Table:

Concept	Scope	Stateful?	Allows/Deny	Use Case
VPC	Whole network	N/A	N/A	Your isolated AWS network
Subnet	Subdivision of VPC	N/A	N/A	Group resources by access
IGW	VPC-wide	N/A	N/A	Internet access for public subnet
NAT	Private subnets	N/A	N/A	Outbound internet access
SG	Instance-level	Yes	Allow only	Instance protection
NACL	Subnet-level	No	Allow & Deny	Broad subnet rules
Route Table	Subnet routing	N/A	N/A	Controls traffic paths
VPC Peering	Between VPCs	N/A	N/A	VPC-VPC comm
Transit Gateway	Org-wide	N/A	N/A	Large network hub
Endpoints	VPC to AWS Service	N/A	N/A	Private AWS comm