VPC Short Notes

In **AWS (Amazon Web Services)**, a **VPC (Virtual Private Cloud)** is a **logically isolated network** that you define within the AWS cloud. It allows you to launch and manage AWS resources (like EC2 instances, RDS databases, etc.) in a **customized virtual network** environment.

🔧 Key Concepts of a VPC

Component	Description
Subnets	Segments of the VPC where you can place resources. Can be **public** (internet-facing) or **private** (internal only).
Route Tables	Rules that determine how traffic moves between subnets and external networks.
Internet Gateway (IGW)	Allows communication between instances in your VPC and the internet.
NAT Gateway/Instance	Allows private subnets to access the internet (for updates, etc.) without being exposed to incoming traffic.
Security Groups	Virtual firewalls that control **inbound/outbound traffic** to AWS resources.
Network ACLs	Optional stateless firewall rules at the subnet level.
CIDR Block	IP address range for your VPC (e.g., 10.0.0.0/16).

🛠️ Example Use Case

Let’s say you’re deploying a web application:

1. **Public subnet**: Hosts the web server (e.g., EC2 + Load Balancer), accessible via the Internet.
2. **Private subnet**: Hosts the backend database (e.g., RDS), not directly accessible from the internet.

3. **NAT Gateway**: Lets backend servers access the internet for updates without exposing them.

🧠 Why Use a VPC?

- **Security**: Full control over inbound and outbound traffic.
- **Isolation**: Keep different applications or environments (dev/test/prod) separate.
- **Scalability**: Easily expand your IP range, subnets, or route tables as your infrastructure grows.

Here's a **simple diagram and explanation** of a typical AWS VPC setup for a **web application** with both public and private subnets.

🧭 Typical VPC Architecture

```
+-------------------------------------------------------------+
|                    AWS VPC (10.0.0.0/16)             |
|                                              |
| +------------------+      +--------------------------+  |
| | Public Subnet  |      |    Private Subnet     |  |
| | (10.0.1.0/24)  |      |    (10.0.2.0/24)      |  |
| |              |      |                   |  |
| | +------------+ |      | +--------------------+ |  |
| | | EC2 Web   | |      | |  EC2 App Server  | |  |
| | | Server    |<----------->|  (No public IP)     | |  |
| | +------------+ |      | +--------------------+ |  |
| |    |         |      |         |          |  |
| |    v         |      |         v          |  |
| | Internet GW    |      |     NAT Gateway       |  |
| +------------------+     +--------------------------+  |
|                                      |
+--------------------------------------------------------------+
```

🧱 Components Explained

Component	Purpose
VPC (10.0.0.0/16)	Entire virtual network space. You define its IP range.
Public Subnet (10.0.1.0/24)	Subnet with direct internet access via **Internet Gateway (IGW)**.
Private Subnet (10.0.2.0/24)	No direct internet access. Used for backend services like databases.
EC2 Web Server	Deployed in the public subnet to serve traffic from the internet.
EC2 App Server	In the private subnet, communicates only with internal services.
Internet Gateway	Enables outbound/inbound internet access for public subnet.
NAT Gateway	Allows resources in the private subnet to **initiate** outbound traffic to the internet (e.g., for software updates), but **blocks inbound traffic** from the internet.

🔐 Security Layering

- **Security Groups**: Control traffic **to/from individual instances** (e.g., allow HTTP on port 80).
- **Network ACLs**: Control traffic at the **subnet level** (stateless).