

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

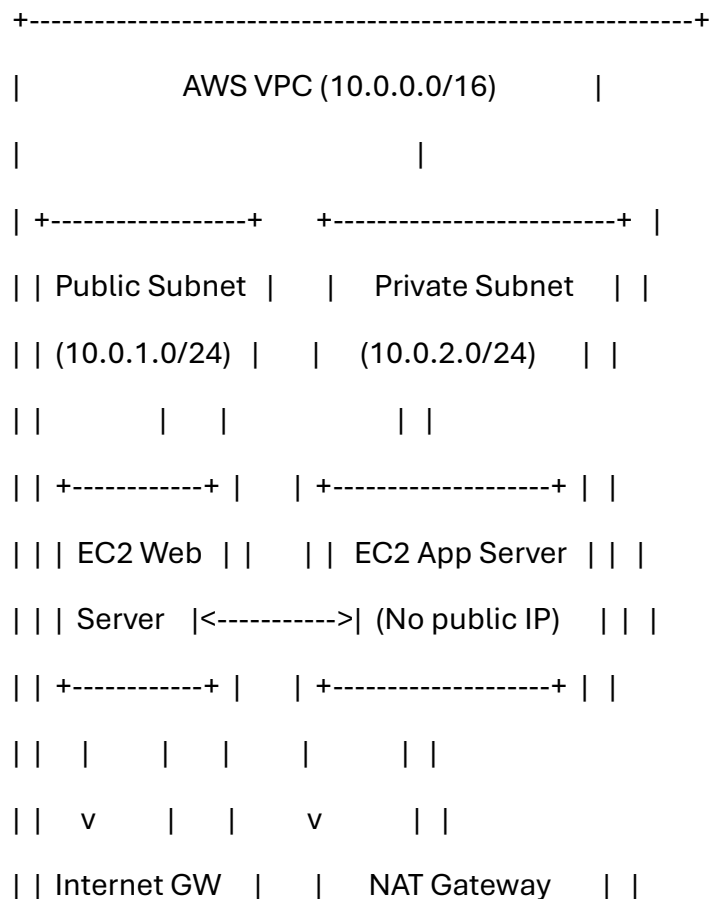
### Why Use a VPC?

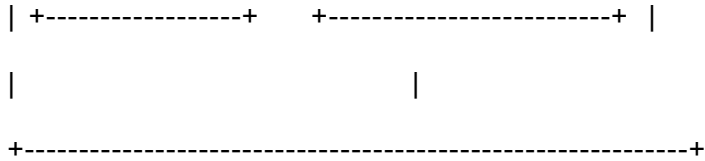
- **Security:** Full control over inbound and outbound traffic.
  - **Isolation:** Keep different applications or environments (dev/test/prod) separate.
  - **Scalability:** Easily expand your IP range, subnets, or route tables as your infrastructure grows.
- 

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





---

## Components Explained

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