

Networking and Security

Identity and Access Management

Networking basics

VPC networking and security

Design a VPC

Build your own VPC and Launch a Web Server

Identity and Access Management

1. Identity and Access Management

The *Identity and access management (IAM)* mechanism encompasses the components and policies necessary to control and track user identities and access privileges for IT resources, environments, and systems.

Specifically, IAM mechanisms exist as systems comprised of four main components:

Authentication – Username and password combinations remain the most common forms of user authentication credentials managed by the IAM system, which also can support digital signatures, digital certificates, biometric hardware (fingerprint readers), specialized software (such as voice analysis programs), and locking user accounts to registered IP or MAC addresses.

Aut
con
sys
bio
voi
ad

Authorization – The authorization component defines the correct granularity for access controls and oversees the relationships between identities, access control rights, and IT resource availability.

Au
for
co

User Management – Related to the administrative capabilities of the system, the user management program is responsible for creating new user identities, the u and access groups, resetting passwords, defining password policies, and managing privileges.

User
the u
and
defin
and
man

Credential Management – The credential management system establishes **Cred** identities and access control rules for defined user accounts, which ident mitigates the threat of insufficient authorization.

Cred
ident
mitig

The IAM mechanism is primarily used to counter the insufficient authorization, denial of service, and overlapping trust boundaries threats.

2. Networking basics

Networking basics

Open system:

A system which is connected to the network and is ready for communication.

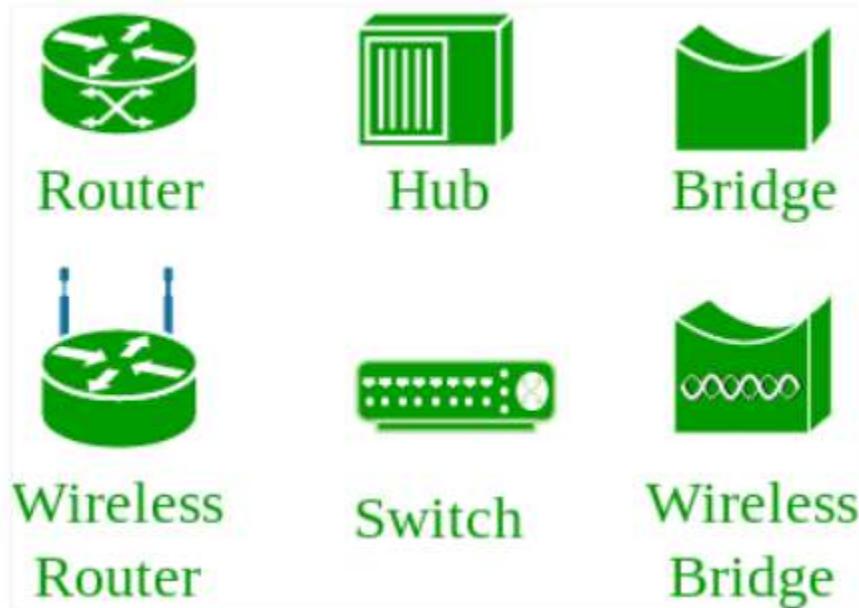
Closed system:

A system which is not connected to the network and can't be communicated with.

Computer Network:

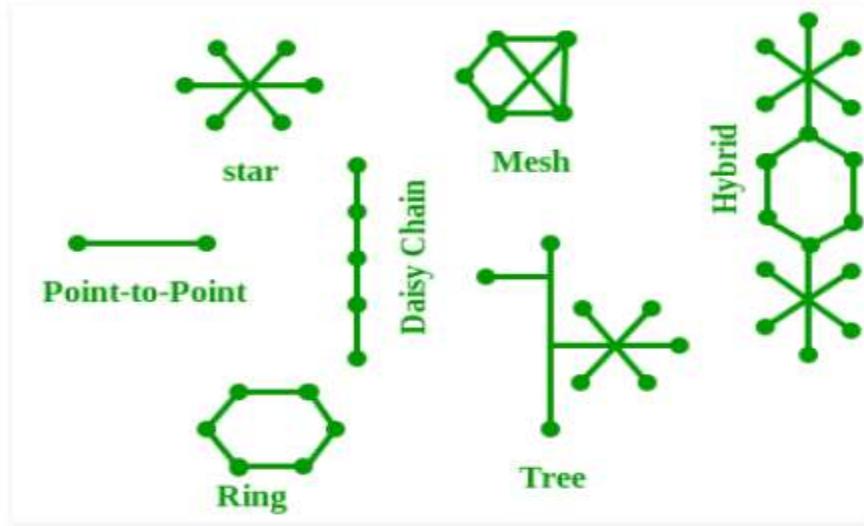
An interconnection of multiple devices, also known as hosts, that are connected using multiple paths for the purpose of sending/receiving data or media.

Computer networks can also include multiple devices/mediums which help in the communication between two different devices; these are known as **Network devices** and include things such as routers, switches, hubs, and bridges.



Network Topology:

The layout arrangement of the different devices in a network. Common examples include: Bus, Star, Mesh, Ring, and Daisy chain.



OSI:

OSI stands for **Open Systems Interconnection**. It is a reference model that specifies standards for communications protocols and also the functionalities of each layer.

Protocol:

A protocol is the set of rules or algorithms which define the way how two entities can communicate across the network and there exists different protocol defined at each layer of the OSI model. Few of such protocols are TCP, IP, UDP, ARP, DHCP, FTP and so on.

UNIQUE IDENTIFIERS OF NETWORK

Host name:

Each device in the network is associated with a unique device name known as Hostname.

Command: `hostname`

IP Address (Internet Protocol address):

Also known as the Logical Address, the IP Address is the network address of the system across the network.

Command: `ipconfig`

MAC Address (Media Access Control address):

Also known as physical address, the MAC Address is the unique identifier of each host and is associated with its NIC (Network Interface Card).

A MAC address is assigned to the NIC at the time of manufacturing.

Command: `ipconfig/all`

DNS Server:

DNS stands for **Domain Name system**.

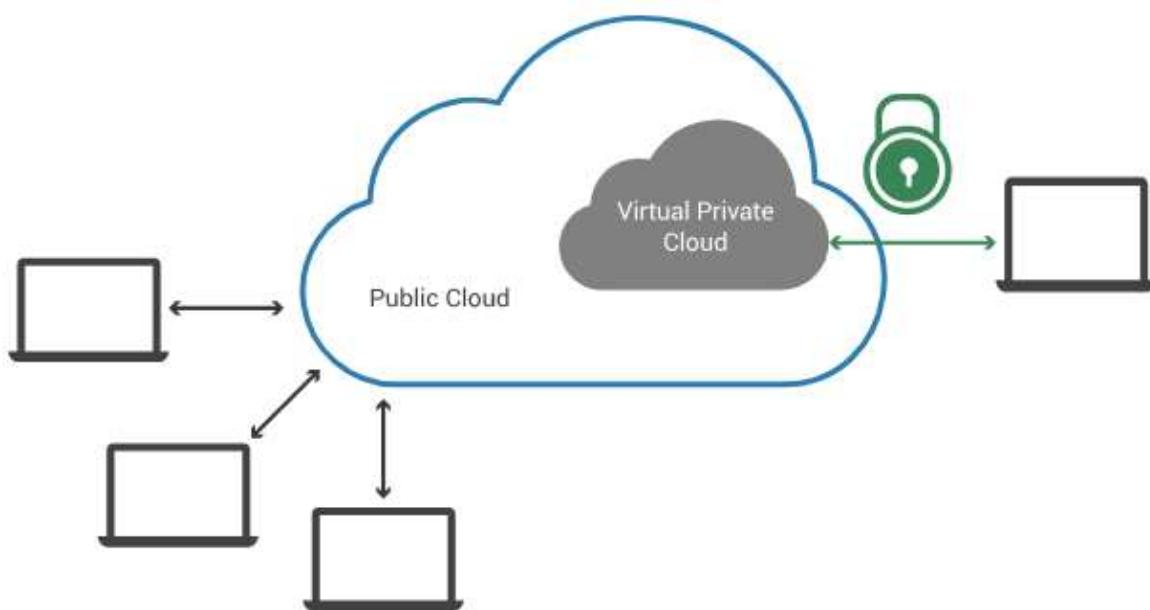
DNS is basically a server which translates web addresses or URLs (ex: www.google.com) into their corresponding IP addresses. We don't have to remember all the IP addresses of each and every website.

Command: **nslookup**

3. VPC networking and security

networking and security

A virtual private cloud (VPC) is a secure, isolated [private cloud](#) hosted within a [public cloud](#). VPC customers can run code, store data, host websites, and do anything else they could do in an ordinary private cloud, but the private cloud is hosted remotely by a public cloud provider.



Features of VPC:

1. Agility

A VPC provides full control over the size of your network and the ability to deploy and scale resources at any time.

2. Security

Although a VPC is part of a public cloud, VPCs are logically isolated networks so your data and applications are entirely separate from your provider's other clients. Access is limited to your resources, unless you grant this.

3. Affordability

VPCs are cost-effective. You'll save money on hardware, labor, and other related cloud resources. The cloud provider will be responsible for all maintenance and upkeep for all physical servers and software.

4. Availability

A virtual private cloud offers redundancy and fault-tolerant availability zone architectures to decrease downtime and keep applications and workloads available every moment.

Virtual Private Clouds: Cloud inside cloud. Subnet portion inside cloud is dedicated to specific user only. In that specific portion only admin has specific authority to Add or Delete resource according to requirements. It is most secure as per application point of view.

Design a VPC

Create VPC: VPCs → Create VPC

The screenshot shows the 'Create VPC' configuration page in the AWS Management Console. The top navigation bar includes the AWS logo, Services dropdown, Resource Groups dropdown, and account information (Region: Oregon, Devops, Ohio). The main title is 'Create VPC'. Below it, a sub-header says 'Create VPC'. A descriptive text explains that a VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. It requires specifying an IPv4 address range, which must be a Classless Inter-Domain Routing (CIDR) block (e.g., 10.0.0.0/16). The CIDR block cannot be larger than /16 and can optionally be associated with an IPv6 CIDR block. The configuration fields include:

- Name tag:** Test vpc
- IPv4 CIDR block***: 10.0.0.0/26
- IPv6 CIDR block**:
 - No IPv6 CIDR Block
 - Amazon provided IPv6 CIDR block
 - IPv6 CIDR owned by me
- Tenancy**: Default

At the bottom left, a note says '* Required'. At the bottom right, there are 'Cancel' and 'Create' buttons.

IPv4 CIDR Block: 10.0.0.0 / 26 (26 is Subnet Mask)

$$32 - 26 = 6$$

$2^6 = 64$ so 64 IP addresses are available in this range.

Create VPC



Build your own VPC and Launch a Web Server

Create Route Table for VPC: Click on option Route Tables

The screenshot shows the AWS VPC Route Tables page. On the left sidebar, under the "Route Tables" section, the "Route Tables" link is highlighted with a yellow box. The main content area displays a table of route tables. One row is selected, showing the "Name" as "test-vpc-rt" and the "Route Table ID" as "rtb-0c6c90dd09ddc065f". A yellow box highlights the "Route Table: rtb-0c6c90dd09ddc065f" label. Below the table, there are tabs for "Summary", "Routes", and "Subnets". At the bottom, the "Route Table ID" is again shown as "rtb-0c6c90dd09ddc065f".

Create Subnet:

We will Create Two Public Subnets and Two Private Subnets.
So IPv4 CIDR Block address should be 10.0.0.0 / 28

10.0.0.16 / 28

10.0.0.32 / 28
10.0.0.48 / 28

It will Looks Like:

Name	Subnet ID	State	VPC	IPv4 CIDR	Available IPv4	IPv6 CIDR
private subnet 1a	subnet-0c7329654e35d103c	available	vpc-09f18b3feb77cec15 ...	10.0.0.32/28	11	-
public-subnet-2b	subnet-0e20b9c830b914fdc	available	vpc-09f18b3feb77cec15 ...	10.0.0.16/28	11	-
public-subnet-1a	subnet-0ec6649178e6bf226	available	vpc-09f18b3feb77cec15 ...	10.0.0.0/28	11	-
private subnet 2b	subnet-0f21e70dd6629b7e6	available	vpc-09f18b3feb77cec15 ...	10.0.0.48/28	11	-

Again go to Route Tables and associate that route tables to Subnets

The screenshot shows the AWS Route Tables interface. A red number '1' highlights the 'test-vpc-rt' route table in the list. A red number '2' highlights the 'Subnet Associations' tab. A red number '3' highlights the 'Edit subnet associations' button. The table below lists two subnets associated with the route table.

Subnet ID	IPv4 CIDR	IPv6 CIDR
subnet-0e20b9c830b914fdc	10.0.0.16/28	-
subnet-0ec6649178e6bf226	10.0.0.0/28	-

Now we have created two Route Tables (Public and Private) but How it can be Public and Private logically? → It can be done by using **Internet Gateway**

Internet Gateway is attached to VPC via Route Table.

Create Internet Gateway → Attach to VPC → Route Tables → Click on **Routes** → Edit Routes → Add Route

Route Tables > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/26 0.0.0.0/0	local IGW-070ee77471301e301	active No	No

Add route

* Required

Cancel Save routes

Build Your Own VPC and Launch a Web Server

As we have already created VPC above. Following is structure of VPC we designed.



AWS Management Console → VPC console → Launch VPC wizard

console.aws.amazon.com/console/home?region=us-east-1#

BWS Services Search for services, features, blogs, docs, and more [Option+S]

IAM

AWS Management Console

AWS services

Recently visited services

- IAM
- S3
- EC2
- Cloud9
- Billing
- AWS Cost Explorer
- Route 53
- VPC
- RDS
- CloudFormation
- CloudWatch
- Amazon EventBridge
- Lambda
- Simple Notification Service

All services

New VPC Experience [Learn more](#)

VPC Dashboard [EC2 Global View](#)

Filter by VPC: Select a VPC

VIRTUAL PRIVATE CLOUD

- Your VPCs
- Subnets
- Route Tables
- Internet Gateways
- Egress Only Internet Gateways
- Carrier Gateways
- DHCP Options Sets

Use Amazon VPC IP Addr

Launch VPC Wizard

Note: Your Instances will launch in:

Resources by Region

You are using the following Amazon Regions:

VPCs [See all regions](#)

Subnets [See all regions](#)

Step 1: Select a VPC Configuration

VPC with a Single Public Subnet

VPC with Public and Private Subnets **Selected**

VPC with Public and Private Subnets and Hardware VPN Access

VPC with a Private Subnet Only and Hardware VPN Access

In addition to containing a public subnet, this configuration adds a private subnet whose instances are not addressable from the internet. Instances in the private subnet can establish outbound connections to the internet via the public subnet using Network Address Translation (NAT).

Creates:

A /16 network with two /24 subnets. Public subnet instances use elastic IPs to access the internet. Private subnet instances access the internet via Network Address Translation (NAT). (Hourly charges for NAT devices apply.)

Important:
If you are using a Local Zone with your VPC follow this link to create your VPC.

Select

