AWS

NETWORKING

FUNDAMENTALS

@CODEBYHP

REGIONS & AVAILABILITY ZONES

REGION

A Region is a geographic location where AWS operates a set of isolated infrastructure.

Ex. ap-south-1 → Mumbai (India)

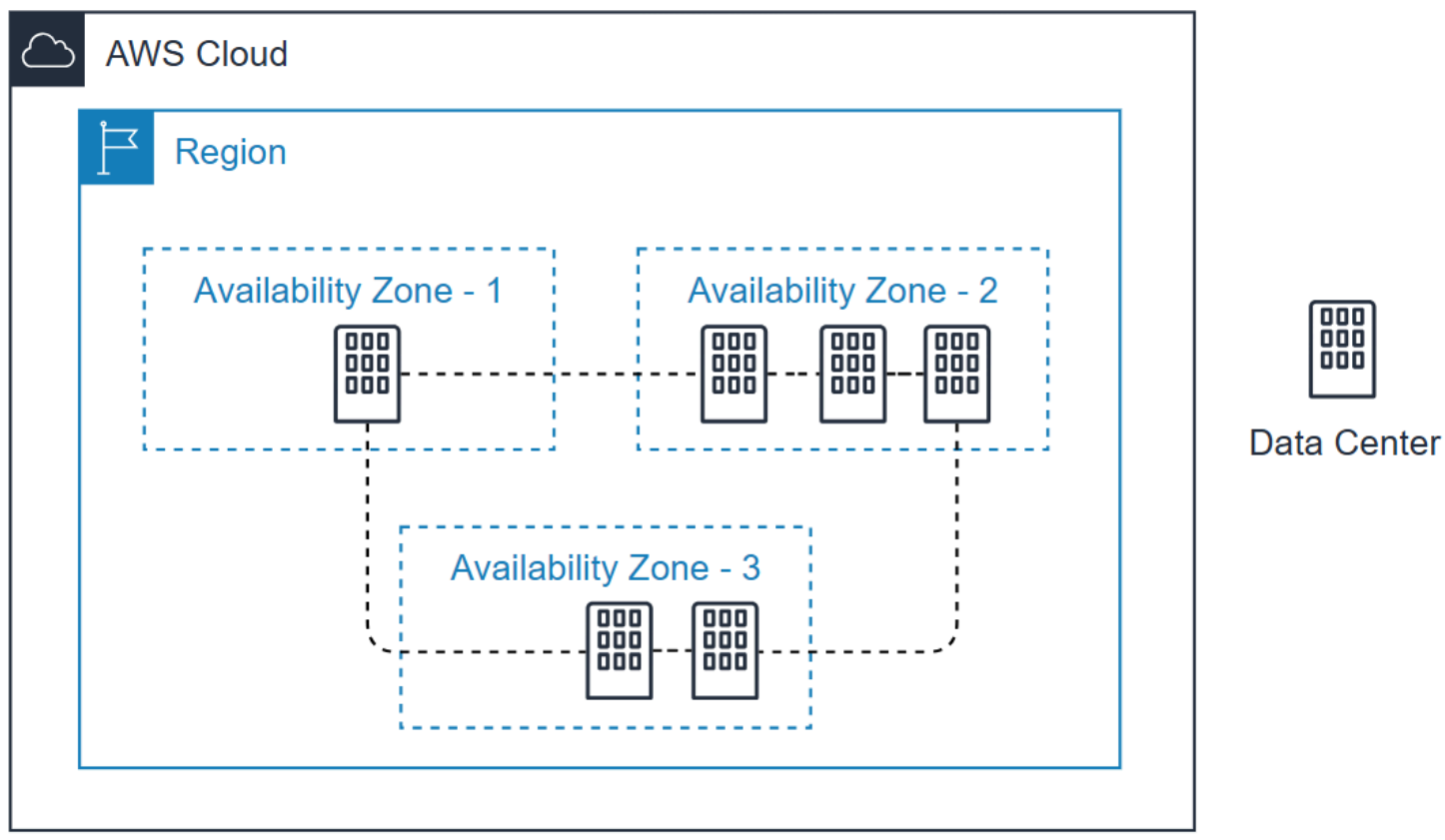
- Regions are physically separated from each other
- Each Region operates independently
- Designed to limit the blast radius of failures
- You typically deploy applications in one Region

AVAILABILITY ZONE

An Availability Zone is one or more physically separate data centers within a Region.

Ex. ap-south-1a
ap-south-1b
ap-south-1c (AZs in ap-south-1 Region)

- Each AZ has independent power, cooling, and networking
- AZs are isolated from failures in other AZs
- AZs are connected using low-latency, high-bandwidth private links
- High availability in AWS is achieved by running applications across multiple AZs (e.g., ap-south-1a, ap-south-1b) to tolerate single-AZ failures.



VPCS & SUBNETS

VPC (VIRTUAL PRIVATE CLOUD)

A VPC is a logically isolated virtual network in AWS where you run your cloud resources.

- A VPC is created inside a single Region
- You define an IP address range (CIDR block) for the VPC
- A VPC spans all Availability Zones in the Region
- Resources inside a VPC communicate using private IP addresses
- You control routing, security, and connectivity

SUBNET

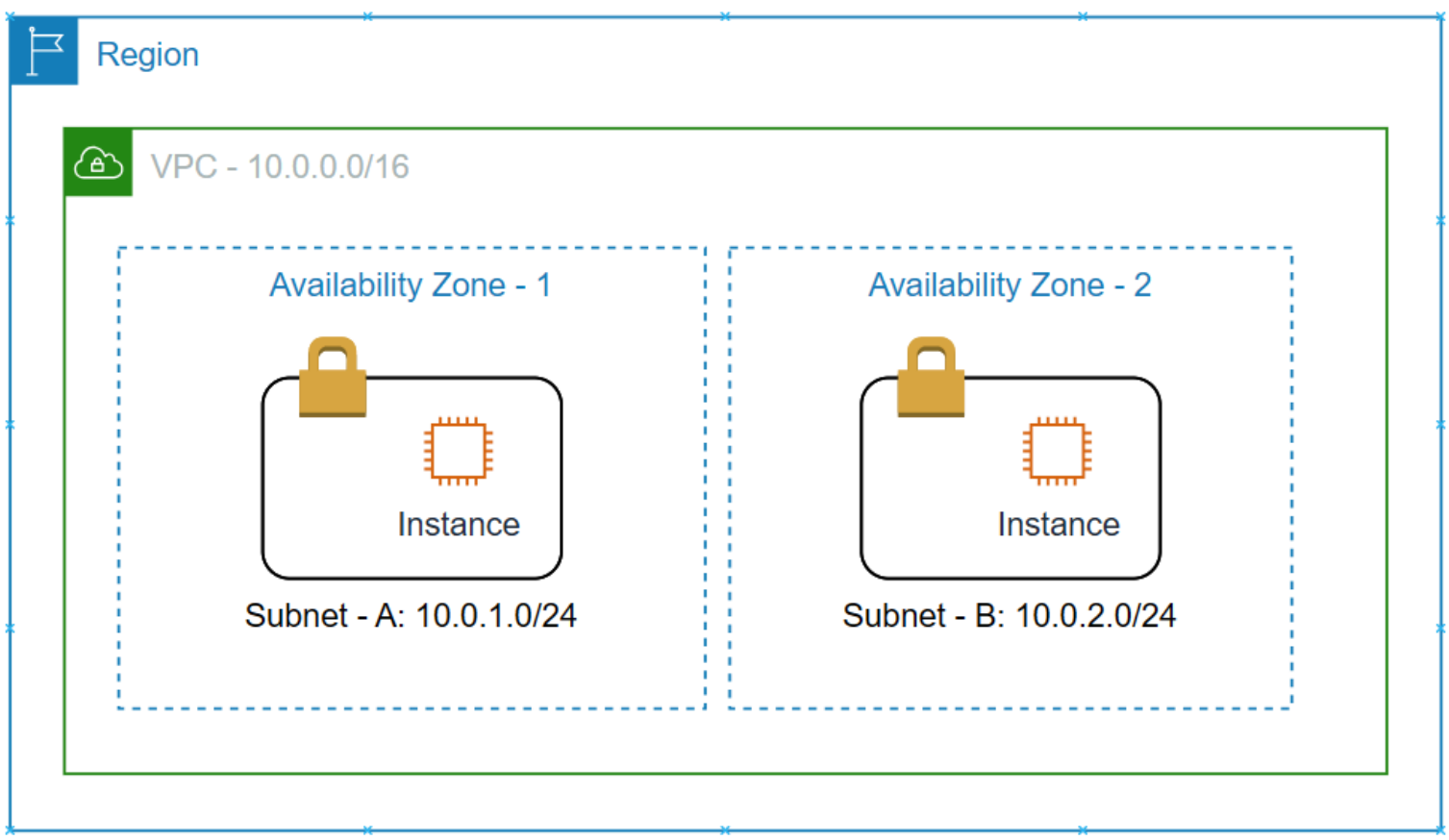
A subnet is a range of IP addresses within a VPC that exists in exactly one Availability Zone.

- A subnet is created inside a VPC
- Each subnet belongs to one and only one AZ
- Subnets are defined using CIDR blocks
- Subnet CIDR ranges must not overlap
- AWS resources (EC2, RDS, ALB targets) are placed inside subnets

TYPES OF SUBNET

Public Subnet: A subnet whose route table includes a route to an Internet Gateway (IGW), allowing resources to be reachable from the internet.

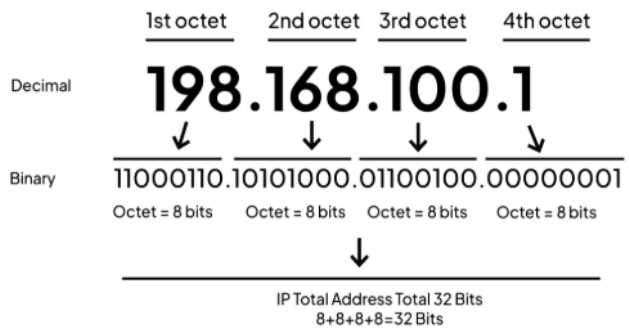
Private Subnet: A subnet without a direct route to an Internet Gateway, preventing direct internet access to its resources.



IPv4 & CIDR

IPV4 ADDRESS

A 32-bit numeric address used to identify a device or resource on a network.
Written as **x.x.x.x**, where each x is an octet (8 bits, 0–255).



CIDR BLOCK (CLASSLESS INTER-DOMAIN ROUTING)

A CIDR block defines a range of IP addresses.

Written as: **x.x.x.x/n**
n → number of fixed **network bits** (the "frozen" part)
32 - n → number of **variable host bits** (assignable to devices)

CIDR Example: 10.0.1.0/24
/24 → first **24 bits** are network, last **8 bits** are host

Network bits(frozen) | Host bits(can be assigned to device)
11111111 . 11111111 . 11111111 . 00000000

- Last 8 bits could form → 2^8 = 256 IP addresses
- AWS reserves 5 IPs → usable IPs = 251

ASSIGNING CIDR RANGES TO VPC AND SUBNETS

The **VPC** is assigned a CIDR block defining its entire IP address space.
VPC CIDR: 10.0.0.0/16 → range 10.0.0.0 – 10.0.255.255 (Total IPs: 65,536)
11111111 . 11111111 . 00000000 . 00000000

Subnets are smaller (non-overlapping) slices of the VPC's CIDR block.
Subnet A: 10.0.1.0/24 → range 10.0.1.0 – 10.0.1.255 (Total IPs: 256)
Subnet B: 10.0.2.0/24 → range 10.0.2.0 – 10.0.2.255 (Total IPs: 256)
Subnet C: 10.0.3.0/24 → range 10.0.3.0 – 10.0.3.255 (Total IPs: 256)

11111111 . 11111111 . 11111111 . 00000000

Route Table

A Route Table is a set of rules that determine where outbound network traffic from a subnet is directed based on destination IP addresses.

Each route has two parts:

- **Destination** → Where the traffic wants to go (CIDR range)
- **Target** → Where AWS sends the traffic next (gateway or connection)

Destination	Target(next hop)
10.0.0.0/16	local
0.0.0.0/0	igw-123456

Any traffic within the VPC (10.0.0.0/16) stays local

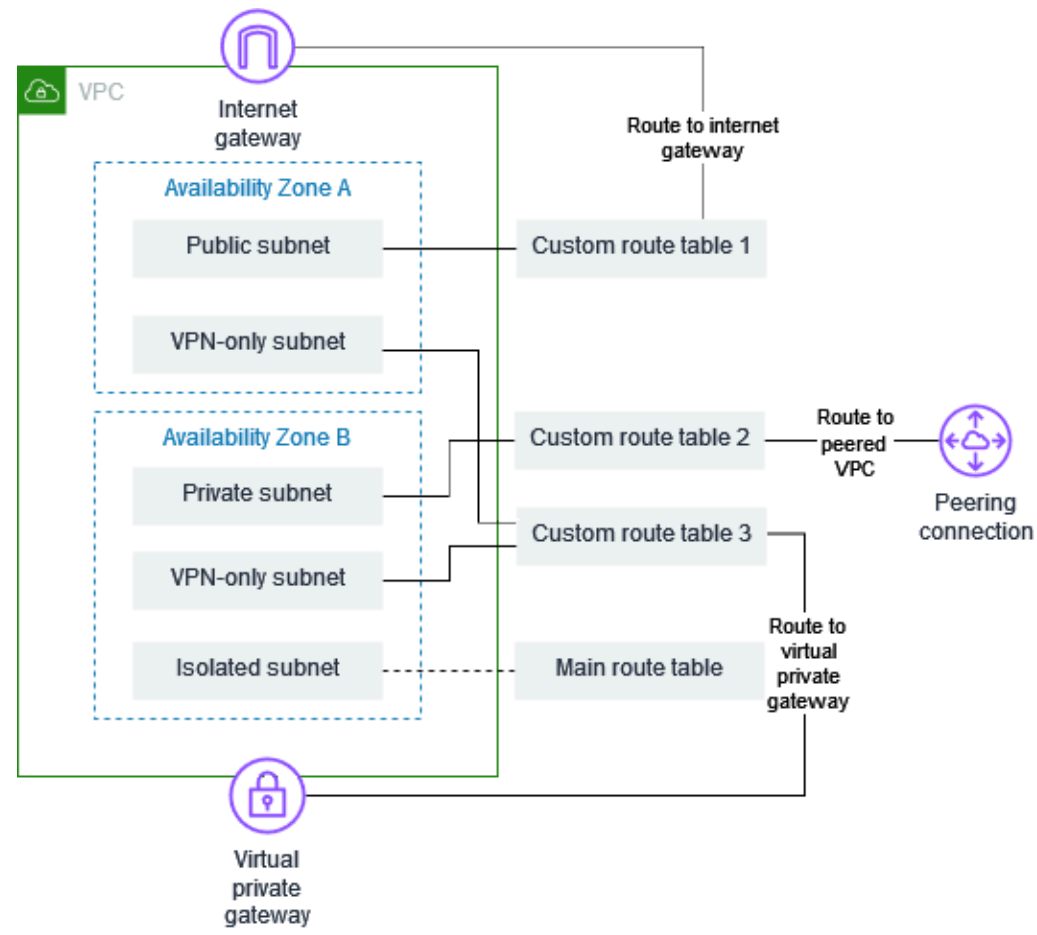
Any traffic to the internet (0.0.0.0/0) is sent to the Internet Gateway

- Route tables belong to a VPC
- Route tables are associated with subnets
- One subnet uses one route table
- One route table can be used by multiple subnets

Default (Main) Route Table: Automatically created by AWS; used by subnets not explicitly associated with a custom table.

Custom Route Table: Manually created route table; can be associated with one or more subnets to control traffic independently.

```
VPC
├─ Main Route Table (Default)
│   └─ Used by Subnet1, Subnet2
│      (if no custom table)
└─ Custom Route Table
    └─ Used by Subnet3 (e.g., private)
    └─ Used by Subnet4 (e.g., public)
```



From Aws Docs

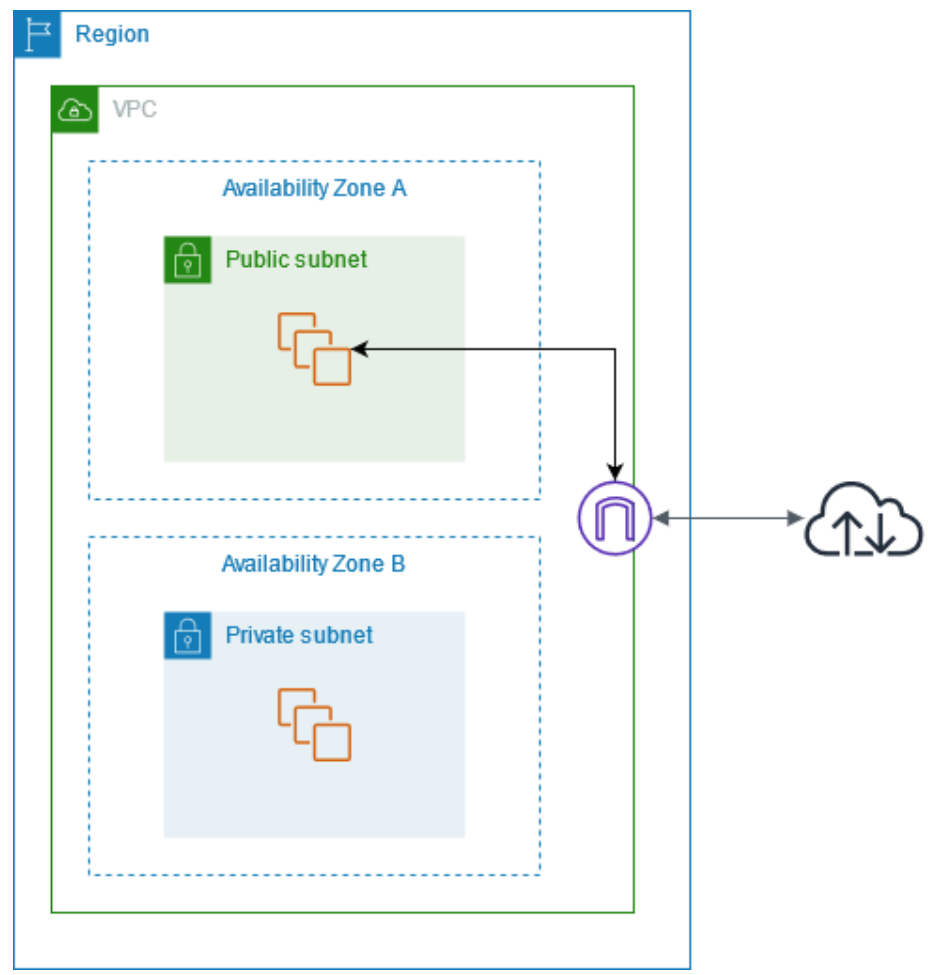
AWS Gateways

INTERNET GATEWAY (IGW)

A managed AWS gateway that allows communication between a VPC and the public internet.

- One IGW per VPC
- Enables inbound and outbound internet traffic (for instances with public IP)
- High availability & horizontally scaled by AWS
- Subnet becomes public when route table points 0.0.0.0/0 → IGW

```
Internet
   ↓
Internet Gateway
   ↓
Public Subnet → EC2 with
Public IP
```



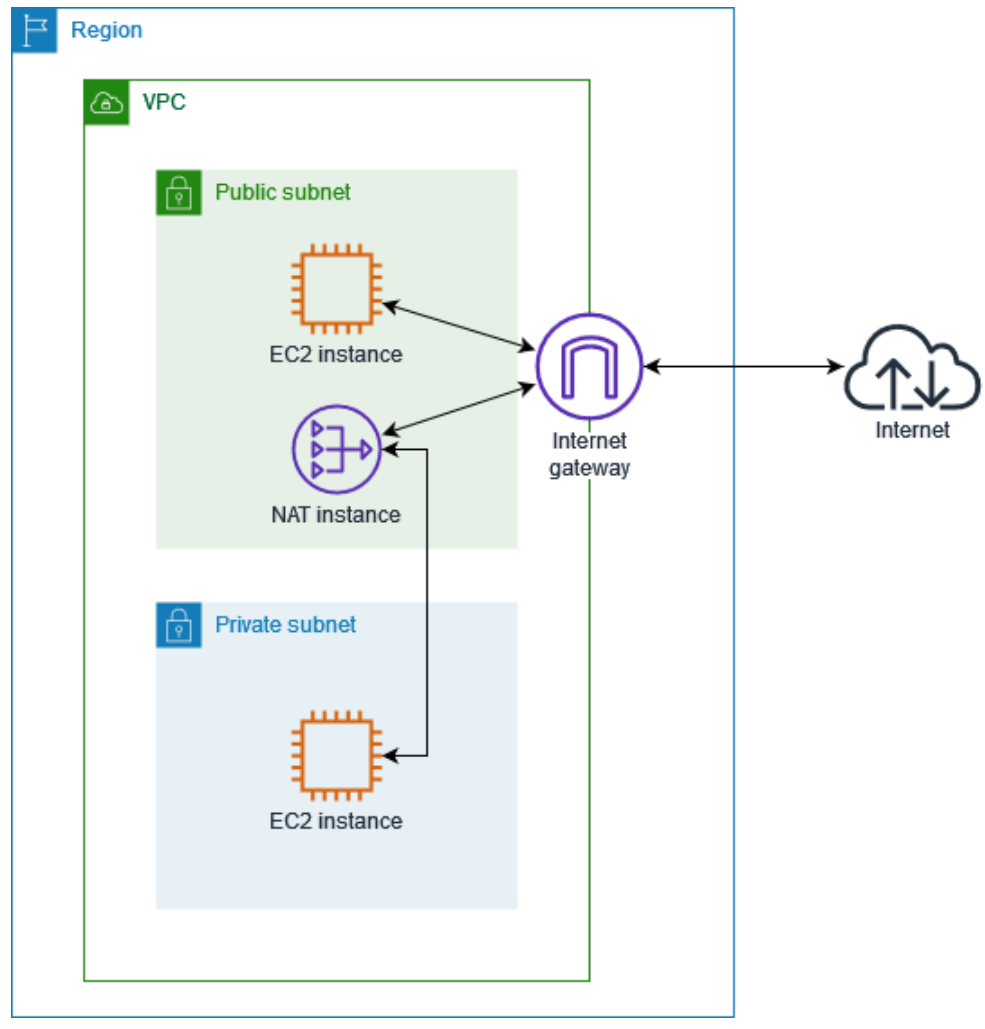
From Aws Docs

NAT GATEWAY (OR NAT INSTANCE)

Allows instances in private subnets to access the internet for updates, downloads, or API calls without exposing them publicly.

- Placed in a public subnet
- Provides outbound internet access for private subnets
- Private instances remain unreachable from the internet
- Managed service → NAT Gateway is recommended over NAT Instance

```
Private Subnet → Route Table → NAT Gateway → IGW → Internet
```

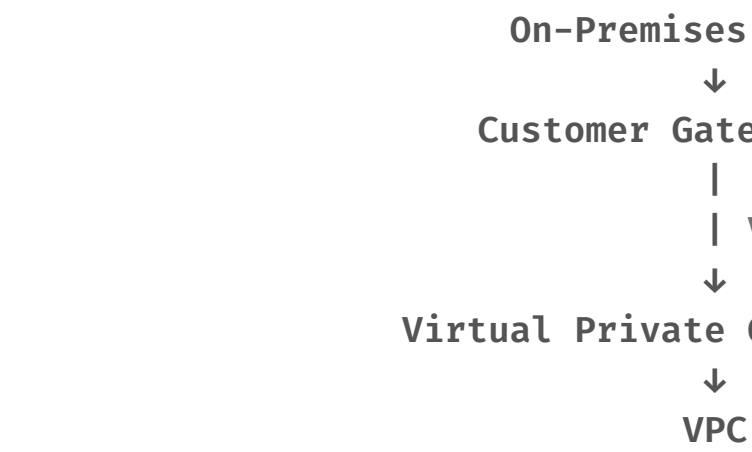


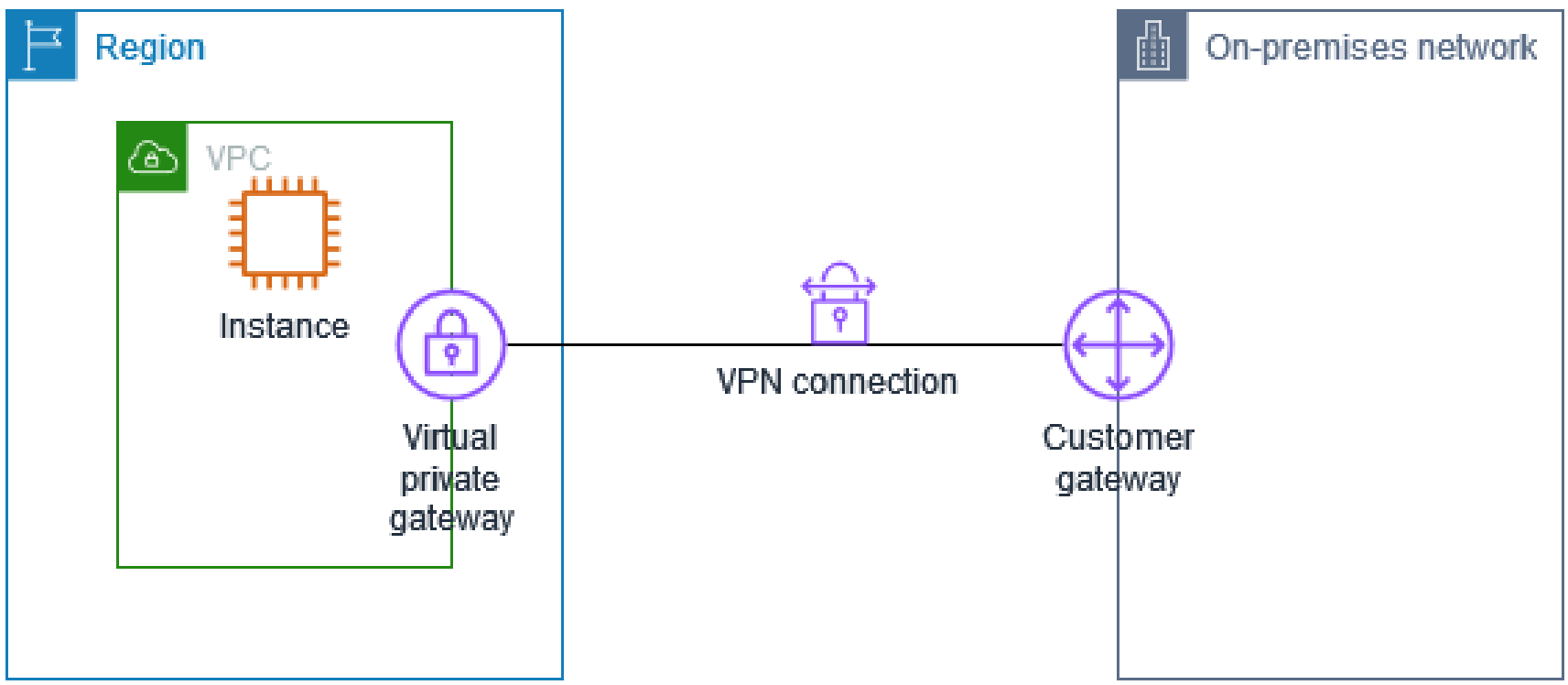
From Aws Docs

VIRTUAL PRIVATE GATEWAY (VGW)

A VGW is a gateway on the AWS side that allows your VPC to connect securely to on-premises networks or remote networks via VPN.

- Attached to a single VPC
- Works with Site-to-Site VPN connections
- Enables hybrid cloud architectures
- Provides secure, encrypted connectivity



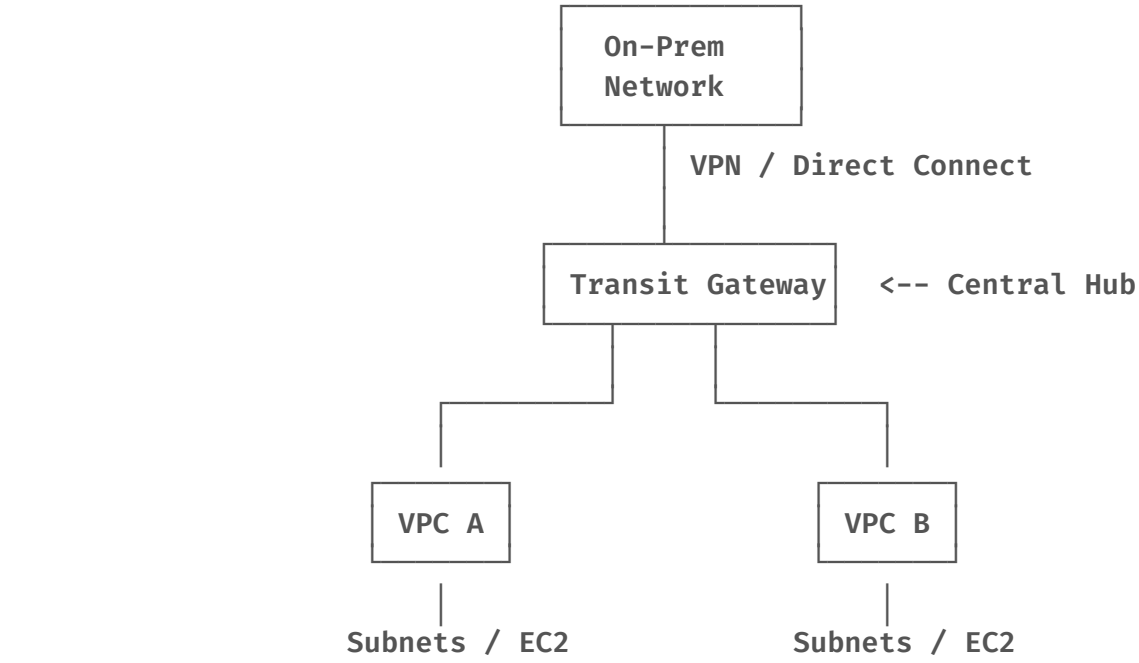


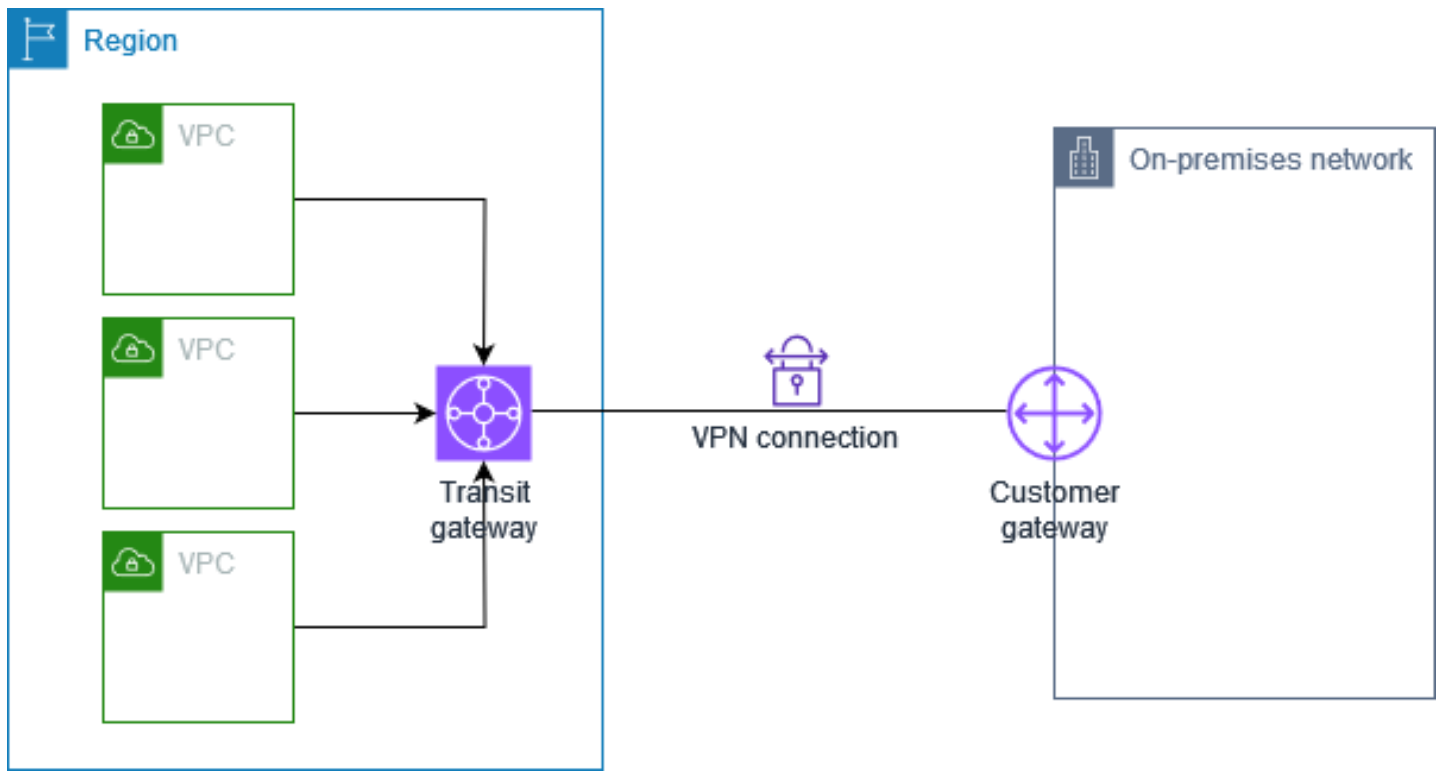
From Aws Docs

TRANSIT GATEWAY (TGW)

A central hub that connects multiple VPCs and on-premises networks, replacing complex point-to-point VPC peering connections.

- Supports hundreds of VPCs and VPNs
- Simplifies network routing at scale
- Can connect VPCs in same or different regions
- Works with VGW, Direct Connect, and VPC attachments





From Aws Docs

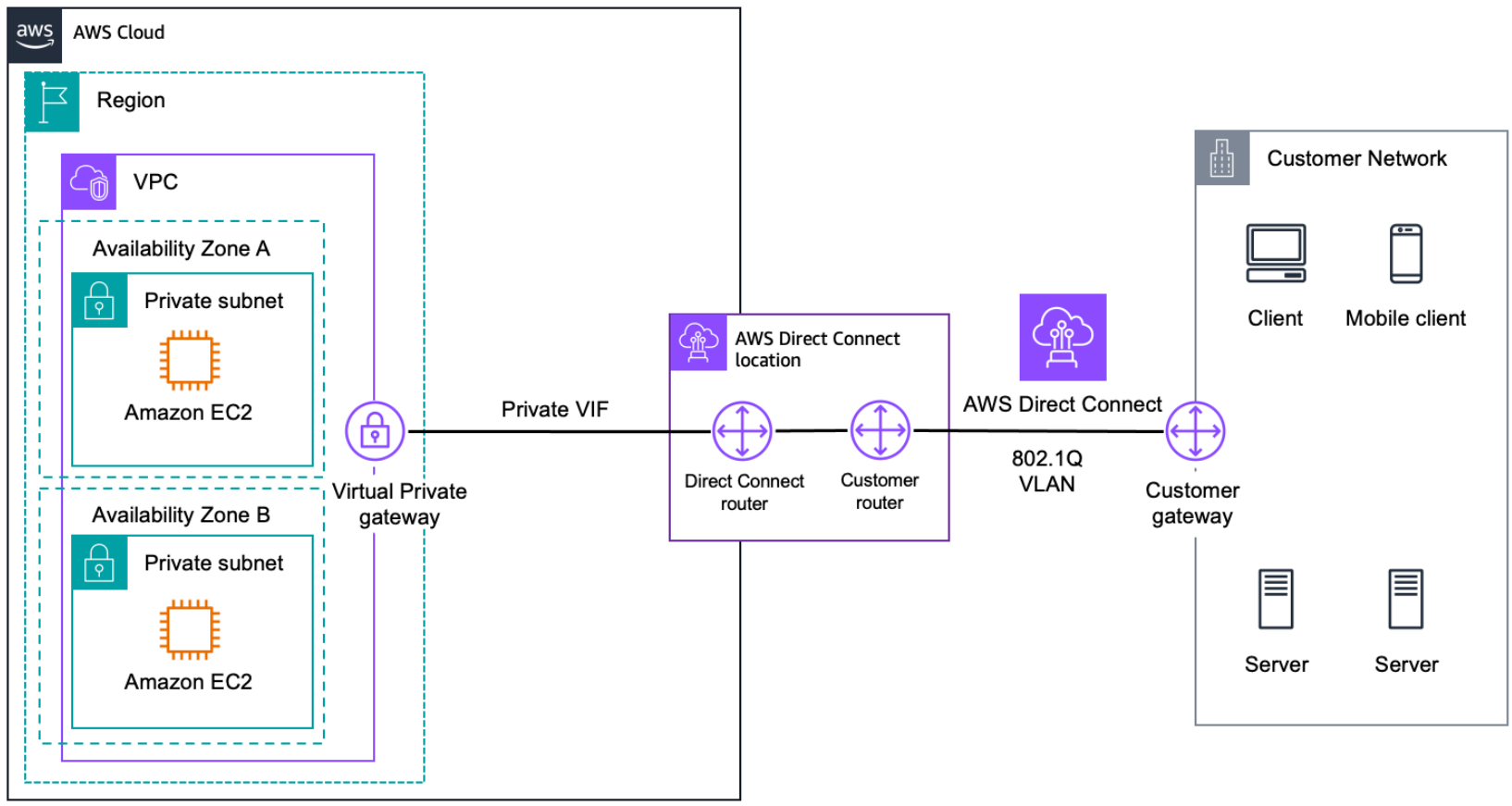
Other Connections Types

DIRECT CONNECT (DX)

AWS Direct Connect is a dedicated private network connection from your on-premises data center to AWS.

- Provides low-latency, high-bandwidth connectivity
- Bypasses the public internet → more secure and stable
- Can be used with VGW or Transit Gateway
- Useful for hybrid cloud setups or large data transfers

```
On-Premises Network
        |
        | Dedicated Fiber / DX
        ↓
   Direct Connect
        ↓
Transit Gateway or VGW
        ↓
       VPC
```

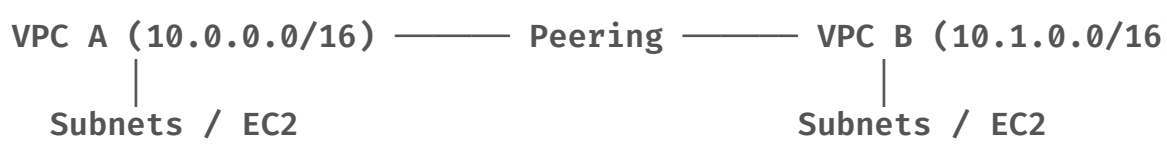


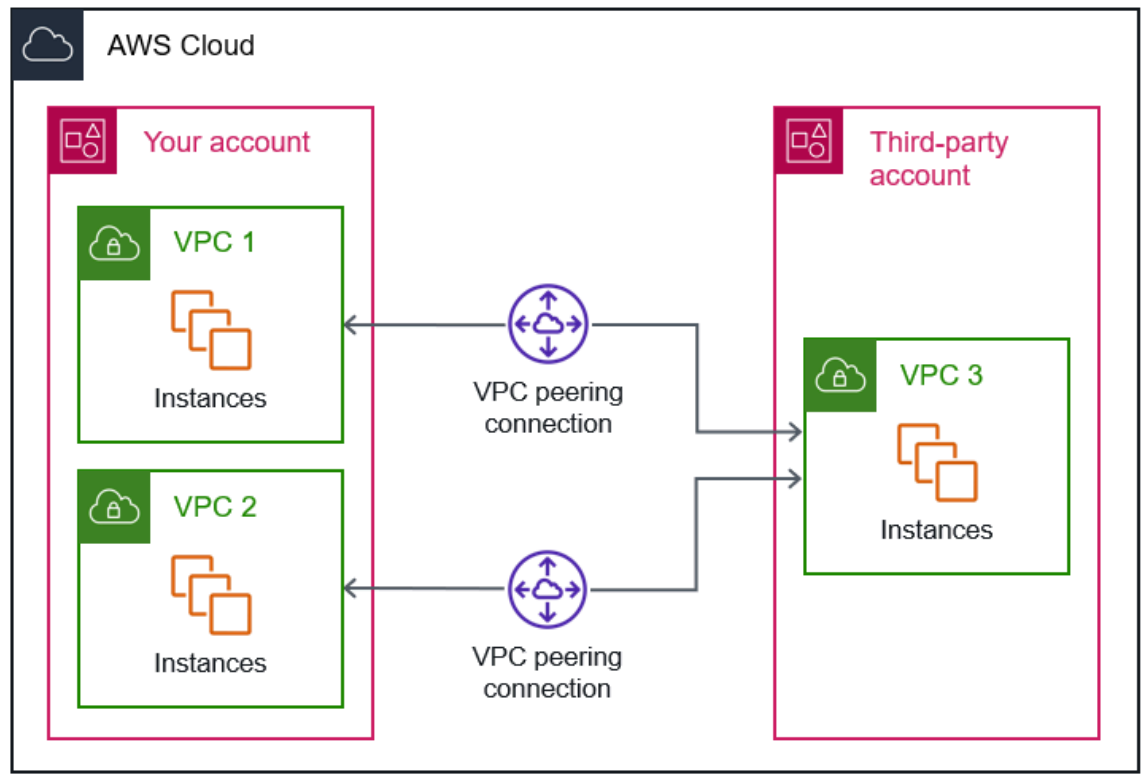
From Aws Docs

VPC PEERING

A private network connection between two VPCs that enables them to communicate as if they are in the same network.

- VPCs can be in the same or different AWS regions
- Uses private IP addresses for communication
- Non-transitive: VPC A → B, B → C does not mean A → C automatically
- Route tables must be updated in both VPCs for traffic





From Aws Docs

Security Groups vs NACLs

SECURITY GROUPS (SGS)

A virtual firewall at the resource level controlling inbound and outbound traffic.

- A security group can be attached to multiple instances.
- Stateful → if inbound is allowed, the response is automatically allowed outbound
- Controls ports and protocols
- Default SG allows all outbound, no inbound

A security group rule has 4 main components:

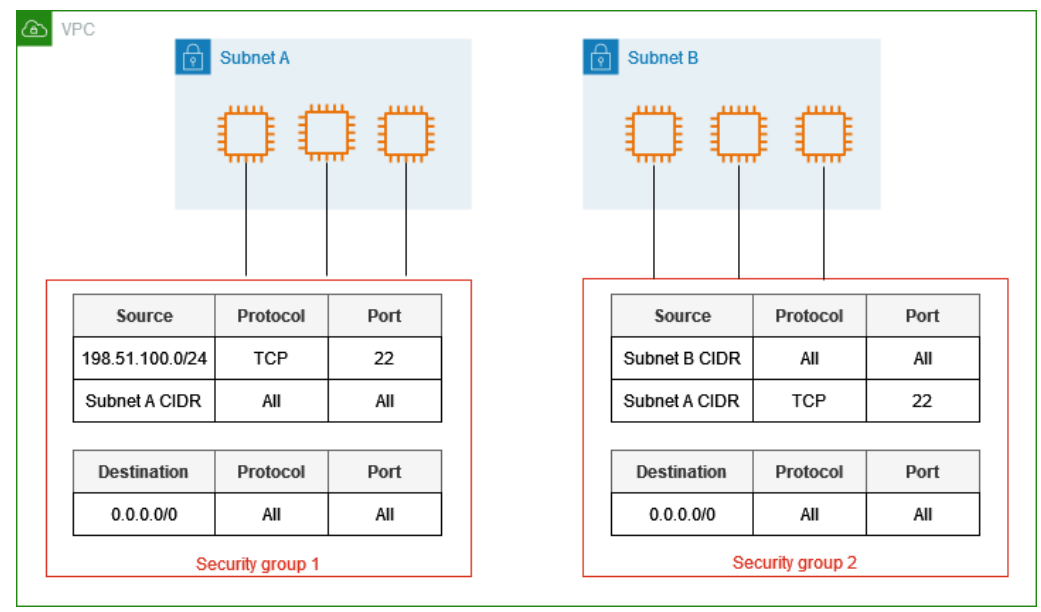
Component	Description
Type / Protocol	The protocol allowed, e.g., TCP, UDP, ICMP, or ALL
Port Range	Ports that are allowed, e.g., 80 for HTTP, 443 for HTTPS
Source / Destination	The IPs, CIDR, or security group that is allowed
Direction	Inbound (incoming) or Outbound (outgoing)

Example: Web Server SG

Direction	Protocol	Port	Source / Destination	Purpose
Inbound	TCP	80	0.0.0.0/0	HTTP traffic from internet
Inbound	TCP	443	0.0.0.0/0	HTTPS traffic from internet
Outbound	ALL	ALL	0.0.0.0/0	Allow all outbound traffic

Inbound
`Internet ──> [Security Group] ──> EC2 Instance`

Outbound
`EC2 Instance ──> [Security Group] ──> Internet or other resources`



From Aws Docs

NETWORK ACLS (NACLS) — AWS SUBNET-LEVEL FIREWALL

A stateless firewall at the subnet level controlling inbound and outbound traffic.

- Associated with subnets, not individual instances
- Stateless: return traffic must have its own explicit rule.
- Evaluated in numerical order of rules (lowest rule number first)
- Default NACL allows all traffic in and out
- Can block unwanted traffic at subnet level for extra security

A NACL rule has 6 main components:

Component	Description
Rule Number	Determines the **evaluation order** (lowest number first).
Protocol	The network protocol (TCP, UDP, ICMP, or ALL).
Port Range	The specific ports affected by the rule (e.g., 22 for SSH, 80 for HTTP).
Source / Destination	The IP range (CIDR block) the rule applies to. For inbound, it's **source**; for outbound, it's **destination**.
Allow / Deny	Whether traffic matching this rule is **allowed** or **blocked**.
Direction	Indicates whether the rule applies to **inbound** or **outbound** traffic.

Example: NACL Rule Structure

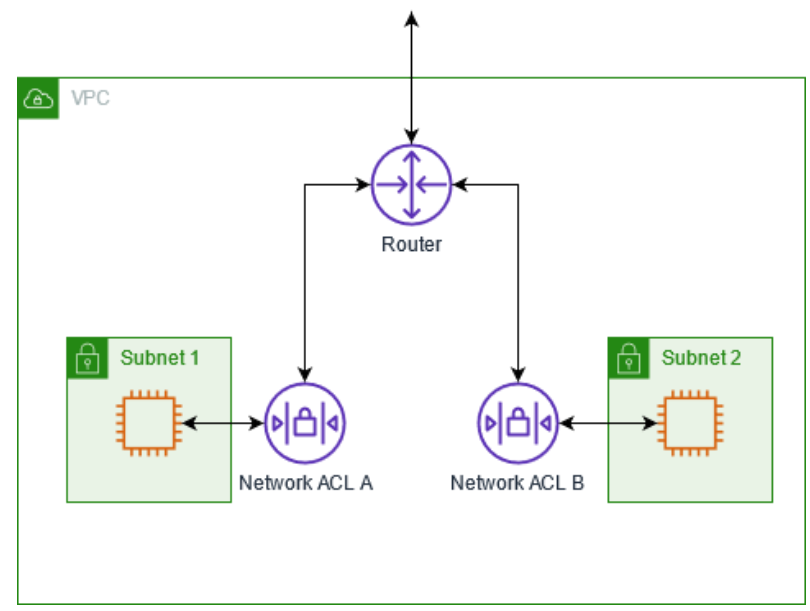
Direction	Rule #	Protocol	Port Range	Source / Destination	Allow/Deny
Inbound	100	TCP	80	0.0.0.0/0	Allow
Inbound	110	TCP	22	203.0.113.0/24	Allow
Inbound	*	ALL	ALL	0.0.0.0/0	Deny
Outbound	100	ALL	ALL	0.0.0.0/0	Allow

Inbound Traffic

```
Internet ──> NACL (subnet-level firewall) ──> Subnet ──> EC2 Instances
```

Outbound Traffic

```
EC2 Instances ──> NACL (subnet-level firewall) ──> Internet
```



From Aws Docs

THANK YOU !

FOR YOUR TIME

@CODEBYHP