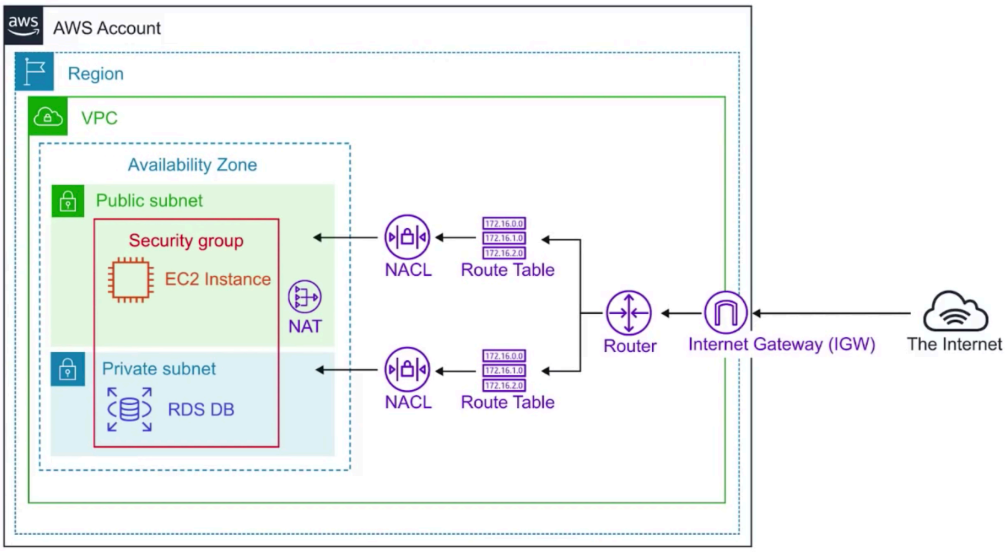
Virtual Private Cloud (VPC)

Provision a **logically isolated section of the AWS Cloud** where you can launch AWS resources in a **virtual network** that you define

Introduction to VPC

Think of a AWS VPC as your own **personal data centre.**

Gives you complete control over your virtual networking environment



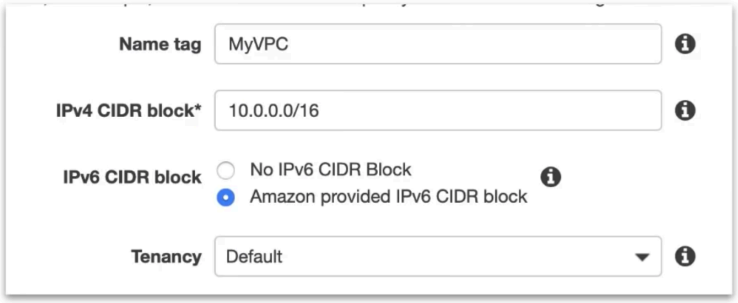
Core Components

Combining these components and services is what makes up your VPC.

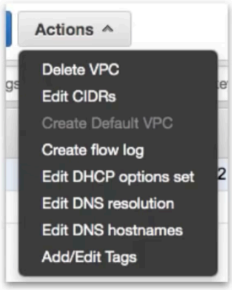
- Internet Gateway (IGW)
- Virtual Private Gateway (VPN Gateway)
- Routing Tables
- Network Access Control Lists (NACLs) - Stateless
- Security Groups (SG) Stateful
- Public Subnets
- Private Subnets
- Nat Gateway
- Customer Gateway
- VPC Endpoints
- VPC Peering

Key Features

- VPCs are **Region Specific** they do not span regions
- You can create upto **5 VPC** per region.
- Every region comes with a default VPC
- You can have **200 subnets** per VPC
- You can use **IPv4 Cidr Block** and in addition to a **IPv6 Cidr Blocks** (the address of the VPC)
- **Cost nothing:** VPC's, Route Tables, Nacls, Internet Gateways, Security Groups and Subnets, VPC Peering
- **Some things cost money:** eg. NAT Gateway, VPC Endpoints, VPN Gateway, Customer Gateway
- **DNS hostnames** (should your instance have domain name addresses)



IPv6 Cidr Block 2600:1f16:9e0:8d00::/56



Disabled by default, turn on **hostnames**

Default VPC

AWS has a default VPC in every region so you can **immediately** deploy instances.

- Create a VPC with a size /16 IPv4 CIDR block (172.31.0.0/16).
- Create a size /20 **default subnet in each Availability Zone**.
- Create an **Internet Gateway** and connect it to your default VPC.
- Create a **default security group** and associate it with your default VPC.
- Create a **default network access control list (NACL)** and associate it with your default VPC.
- Associate the **default DHCP** options set for your AWS account with your default VPC.
- *When you create a VPC, it automatically has a main route table

0.0.0.0/0

0.0.0.0/0 is also know as **default**

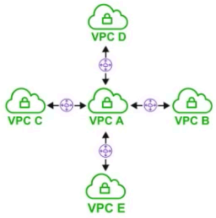
It represents **all possible IP addresses**

When we specify **0.0.0.0/0** in our route table for IGW we are allow internet access
When we specific **0.0.0.0/0** in our security groups inbound rules we are allowing all traffic from the internet access our public resources

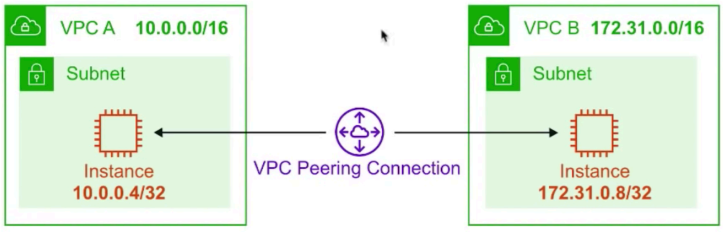
When you see **0.0.0.0/0**, just *think* of giving access from anywhere or the internet.

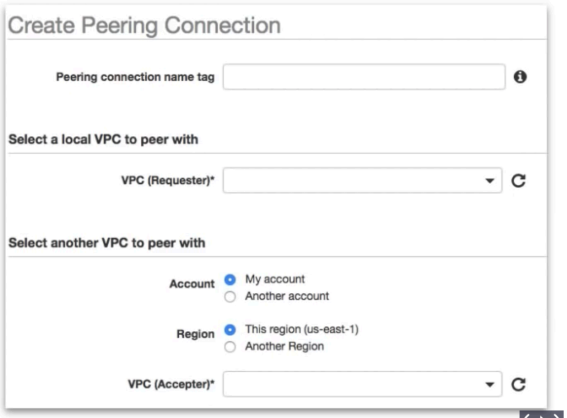
VPC Peering

VPC Peering allows you to connect one VPC with another over a **direct network route** using **private IP addresses.**



- Instances on peered VPCs **behave** just like they are on the **same network**
- Connect VPCs across **same** or **different AWS accounts** and **regions**
- Peering uses a **Star Configuration: 1 Central VPC - 4 other VPCs**
- **No Transitive Peering** (peering must take place directly between VPCs)
  - Needs a one to one connect to immediate VPC
- **No Overlapping CIDR Blocks**





Route Tables

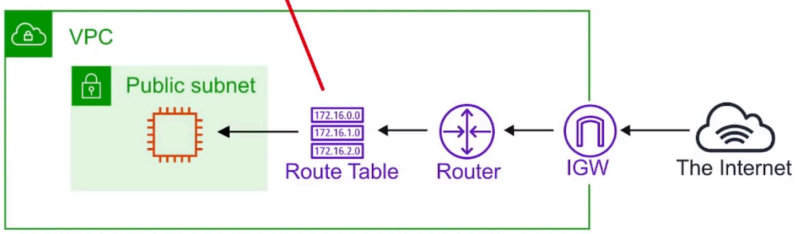
Route tables are used to determine where **network traffic is directed**

Each **subnet** in your VPC **must be associated** with a route table

Each record is called a “route”

A subnet can only be associated **with one route table at a time**, but you can associate multiple subnets with the same route table.

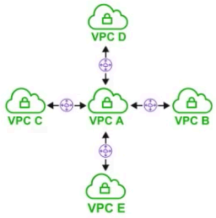
Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
0.0.0.0/0	igw-19e3a2e134fe086e2	active	No

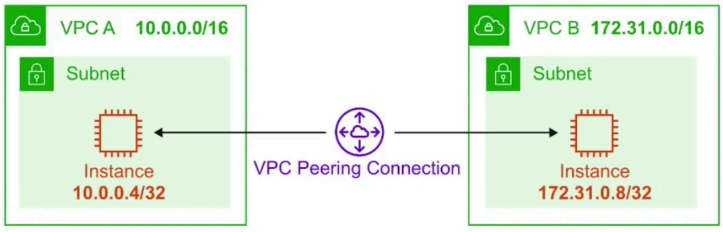


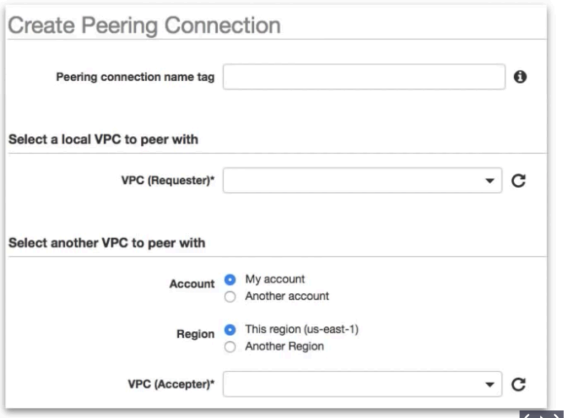
VPC Peering

VPC Peering allows you to connect one VPC with another over a **direct network route** using **private IP addresses.**

- Instances on peered VPCs **behave** just like they are on the **same network**
- Connect VPCs across **same** or **different AWS accounts** and **regions**
- Peering uses a **Star Configuration: 1 Central VPC - 4 other VPCs**
- **No Transitive Peering** (peering must take place directly between VPCs)
  - Needs a one to one connect to immediate VPC
- **No Overlapping CIDR Blocks**







Route Tables

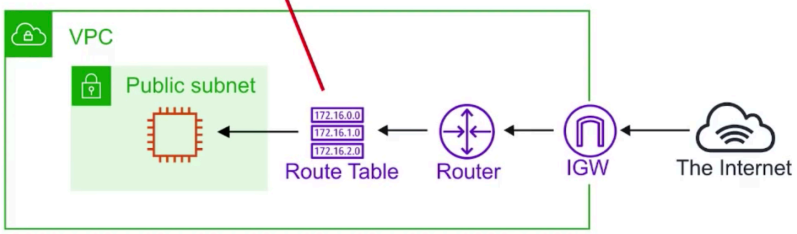
Route tables are used to determine where **network traffic is directed**

Each **subnet** in your VPC **must be associated** with a route table

Each record is called a “route”

A subnet can only be associated **with one route table at a time**, but you can associate multiple subnets with the same route table.

Destination	Target	Status	Propagated
10.0.0.0/16	local	active	No
0.0.0.0/0	igw-19e3a2e134fe086e2	active	No



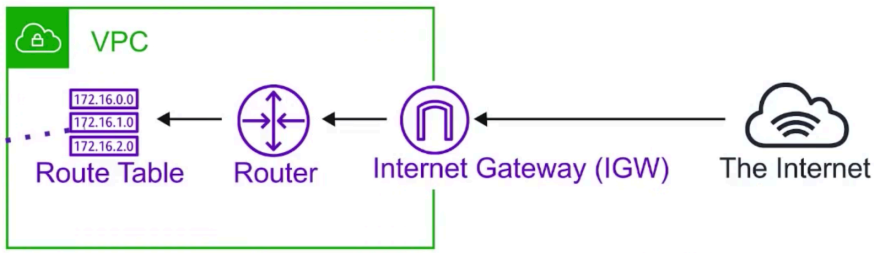
Internet Gateway (IGW)

The Internet Gateway allows **your VPC access to the internet**.

IGW does ✌️ two things:

1. provide a target in your VPC route tables for internet-routable traffic
2. perform network address translation (NAT) for instances that have been assigned **public IPv4 addresses**.

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	**igw-id**



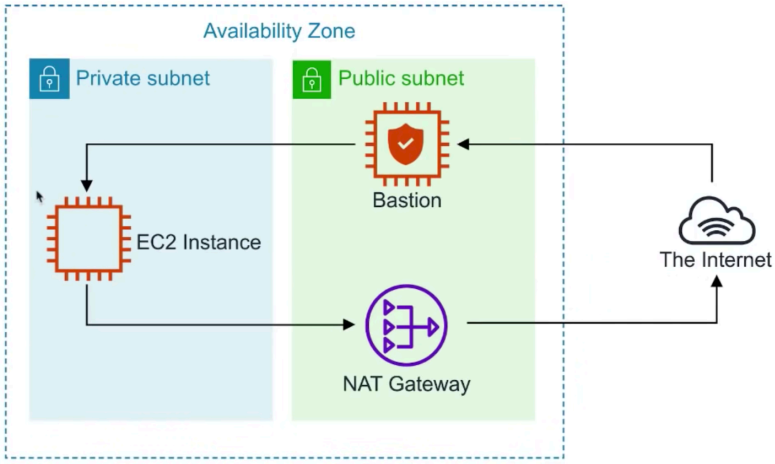
To route out to the internet you need to add in your route tables you need to add a route
To the internet gateway and set the Destination to be **0.0.0.0/0**

Bastion / Jumpbox

Bastions are EC2 instances which are security harden. They are designed to help you gain access to your EC2 Instances via SSH or RCP That are in a **private subnet**.

They are also known as Jump boxes because you are jumping from one box to access another.

NAT Gateways/Instances are only intended for EC2 instances to gain outbound access to the internet for things such as security updates. NATs cannot/should not be used as Bastions

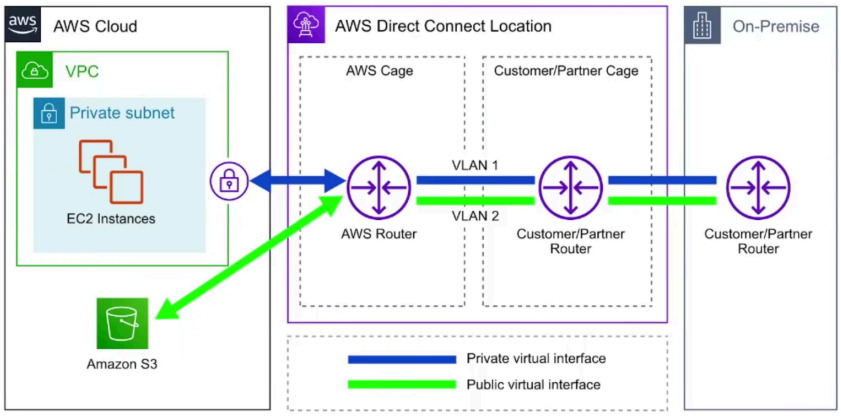


System Manager's **Sessions Manager** replaces the need for Bastions

Direct Connect

AWS Direct Connect is the AWS solution for establishing
dedicated network connections from on-premises locations to AWS.

Very fast network Lower Bandwidth **50M-500M** or Higher Bandwidth **1GB** or **10GB**



- Helps **reduce network costs** and **increase bandwidth throughput**. (great for high traffic networks)
- Provides a **more consistent network experience** than a typical internet-based connection. (reliable and secure)

VPC Endpoints

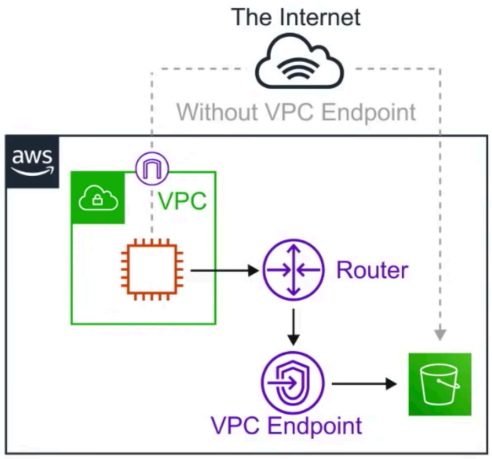
Think of a secret tunnel where you don't have to leave the AWS network

VPC Endpoints allow you to **privately connect your VPC to other AWS services**, and VPC endpoint services.

- **Eliminates** the need for an **Internet Gateway**, **NAT device**, **VPN connection**, or **AWS Direct Connect** connections.
- **Instances** in the VPC **do not require a public IP address** to communicate with service resources.
- **Traffic** between your VPC and other services **does not leave the AWS network**.
- **Horizontally scaled**, **redundant**, and **highly available** VPC component.
- Allows secure communication between instances and services - **without adding availability risks or bandwidth constraints** on your traffic.

There are ✌️ **2 Types** of VPC Endpoints

1. Interface Endpoints
2. Gateway Endpoints



Interface Endpoints

Interface Endpoints are **Elastic Network Interfaces (ENI)** with a **private IP address**.
They serve as an entry point for traffic going to a supported service.

Interface Endpoints are powered by **AWS PrivateLink**

Access services hosted on AWS easily and securely by keeping your network traffic within the AWS network

Pricing per VPC endpoint per AZ ($/hour)	**0.01**	~$7.5 / mo
Pricing per GB data processed ($)	**0.01**	

Interface Endpoints support the following AWS Services...

- API Gateway
- CloudFormation
- CloudWatch
- Kinesis
- SageMaker
- Codebuild
- AWS Config
- EC2 API
- ELB API
- AWS KMS
- Secrets Manager
- Security Token Service
- Service Catalog
- SNS
- SQS
- Systems Manager
- Marketplace Partner Services
- Endpoint Services in other AWS accounts

VPC Gateway Endpoints

VPC Gateway Endpoints are **Free!**

A **Gateway Endpoint** is a gateway that is a target **for a specific route** in your **route table**, used for traffic destined for a supported AWS service.

To create a Gateway Endpoint, you must specify the VPC in which you want to create the endpoint, and the service to which you want to establish the connection.

AWS Gateway Endpoint currently only supports 2 services...

Amazon S3

DynamoDB

VPC Endpoint *CheatSheet*

- VPC Endpoints help keep traffic between AWS services **within the AWS Network**
- There are two kinds of VPC Endpoints. Interface Endpoints and Gateway Endpoints
- Interface Endpoints **cost money**, Gateway Endpoints **are free**
- Interface Endpoints uses an Elastic Network Interface (ENI) with Private IP (powered by AWS PrivateLink)
- Gateway Endpoints is a target for a specific route in your route table
- Interface Endpoints support many AWS services
- Gateway Endpoint only support DynamoDB and S3