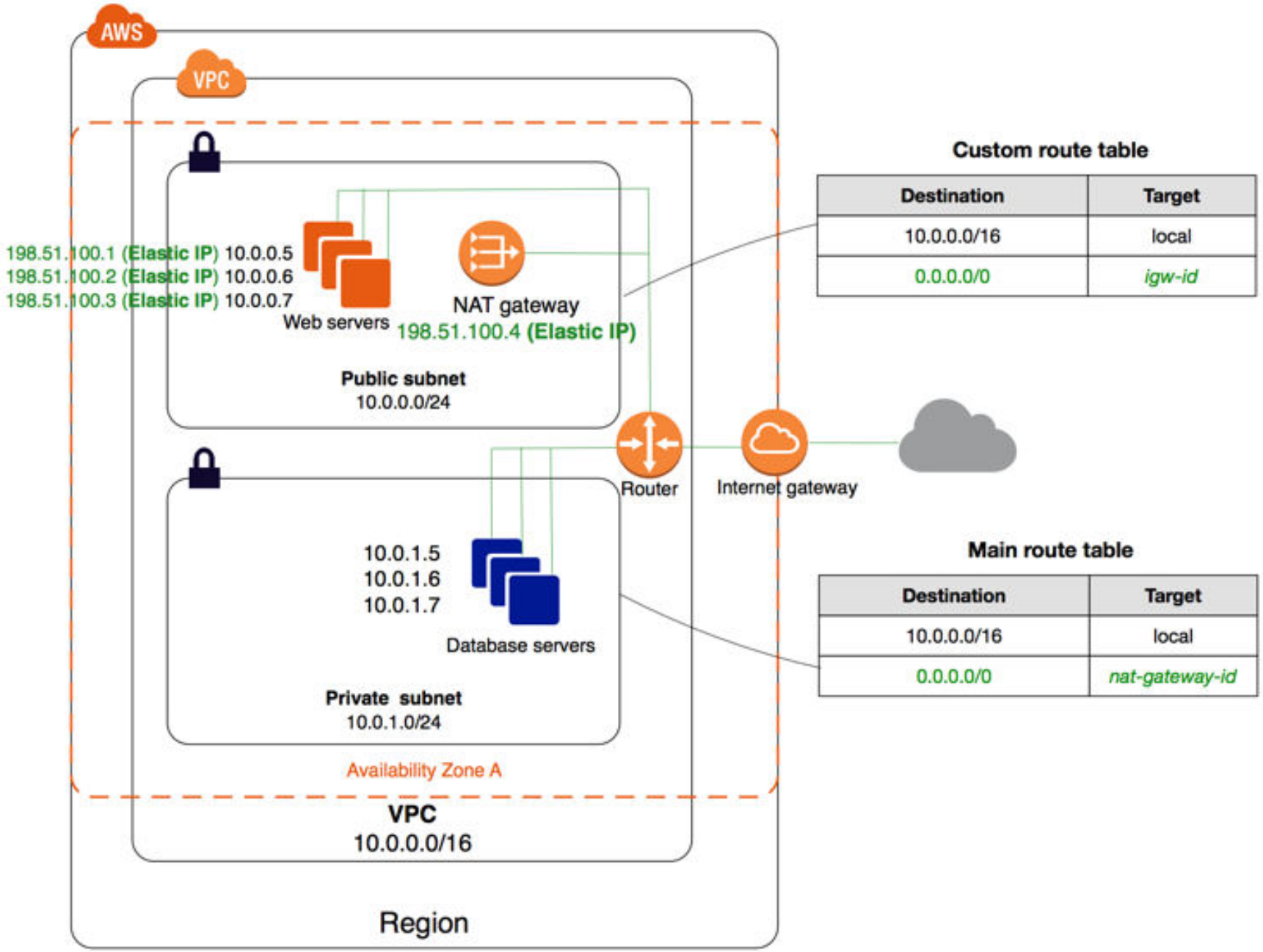
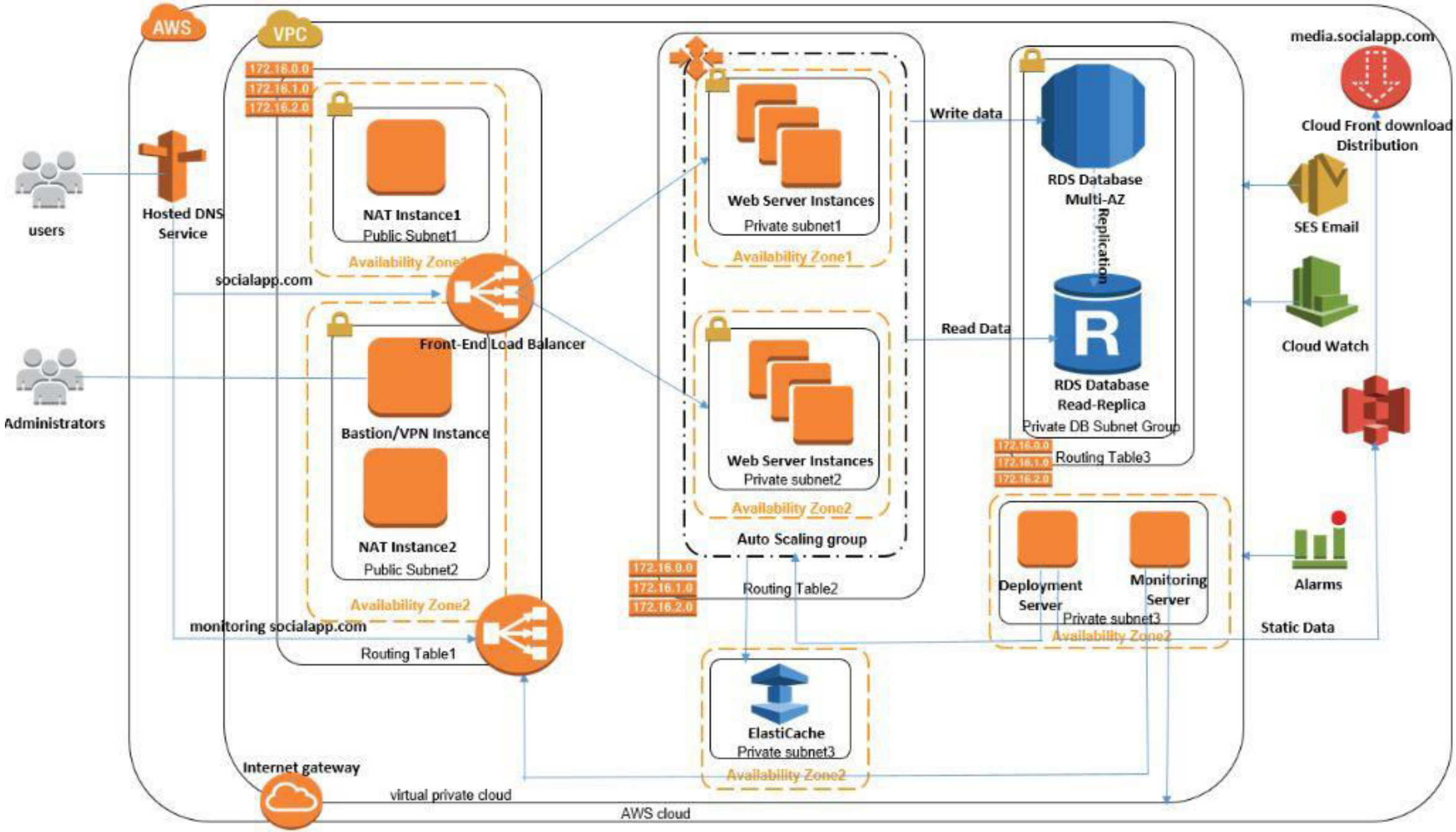
AWS Virtual Private Cloud (VPC)

What is VPC?

- Amazon VPC is your own private network inside Amazon’s cloud infrastructure.
- It is an alternative to maintaining your own data center and is cheaper since it creates resources on demand.
- It is also more secure since Amazon takes care of the infrastructure security for you.


AWS
VPC
198.51.100.1 (Elastic IP) 10.0.0.5
198.51.100.2 (Elastic IP) 10.0.0.6
198.51.100.3 (Elastic IP) 10.0.0.7
Web servers
NAT gateway
198.51.100.4 (Elastic IP)
Public subnet
10.0.0.0/24
Router
Internet gateway
10.0.1.5
10.0.1.6
10.0.1.7
Database servers
Private subnet
10.0.1.0/24
Availability Zone A
VPC
10.0.0.0/16
Region
Custom route table
Destination
Target
10.0.0.0/16
local
0.0.0.0/0
igw-id
Main route table
Destination
Target
10.0.0.0/16
local
0.0.0.0/0
nat-gateway-id


AWS
VPC
172.16.0.0
172.16.1.0
172.16.2.0
users
Hosted DNS Service
NAT Instance1
Public Subnet1
Availability Zone1
socialapp.com
Front-End Load Balancer
Administrators
Bastion/VPN Instance
NAT Instance2
Public Subnet2
Availability Zone2
monitoring socialapp.com
Routing Table1
Internet gateway
virtual private cloud
AWS cloud
Web Server Instances
Private subnet1
Availability Zone1
Write data
Read Data
Web Server Instances
Private subnet2
Availability Zone2
Auto Scaling group
172.16.0.0
172.16.1.0
172.16.2.0
Routing Table2
ElastiCache
Private subnet3
Availability Zone2
RDS Database
Multi-AZ
Replication
RDS Database
Read-Replica
Private DB Subnet Group
172.16.0.0
172.16.1.0
172.16.2.0
Routing Table3
Deployment Server
Monitoring Server
Private subnet3
Availability Zone2
media.socialapp.com
Cloud Front download Distribution
SES Email
Cloud Watch
Alarms
Static Data

CIDR

- CIDR or Classless Inter-Domain Routing is used to allocate IP address within a network.
- We will use CIDR blocks to mark a range of IP addresses for each subnet within a VPC.
- The VPC itself would have a CIDR block that lists all the IP addresses available with it.

CIDR

- The CIDR has two components
  - The base IP (xx.xx.xx.xx)
  - The Subnet Mask (/26)
- The base IP represents an IP contained in the range
- The subnet masks defines how many bits can change in the IP

CIDRs : Subnet Masks

- The subnet masks basically allows part of the underlying IP to get additional next values from the base IP: Formula 2^32-n

- /32 allows for 1 IP = 2^0
- /31 allows for 2 IP = 2^1
- /30 allows for 4 IP = 2^2
- /29 allows for 8 IP = 2^3
- /28 allows for 16 IP = 2^4
- /27 allows for 32 IP = 2^5
- /26 allows for 64 IP = 2^6
- /25 allows for 128 IP = 2^7
- /24 allows for 256 IP = 2^8
- /16 allows for 65,536 IP = 2^16
- /0 allows for all IPs = 2^32

- Quick memo:
- /32 – no IP number can change
- /24 - last IP number can change
- /16 – last IP two numbers can change
- /8 – last IP three numbers can change
- /0 – all IP numbers can change

Private IP and Public IP

- The Internet Assigned Numbers Authority (IANA) established certain blocks of IPV4 addresses for the use of private (LAN) and public (internet) addresses.

Private IP can only allow certain values

- 10.0.0.0 – 10.255.255.255 (10.0.0.0/8) **<= in big networks**
- 172.16.0.0 – 172.31.255.255 (**172.16**.0.0/12) **<= default AWS one**
- 192.168.0.0 – 192.168.255.255 (192.168.0.0/16) **<= home networks**

- All the rest of the IP on the internet are public IP

VPC in AWS

You can have multiple VPCs in a region (max 5 per region – soft limit).

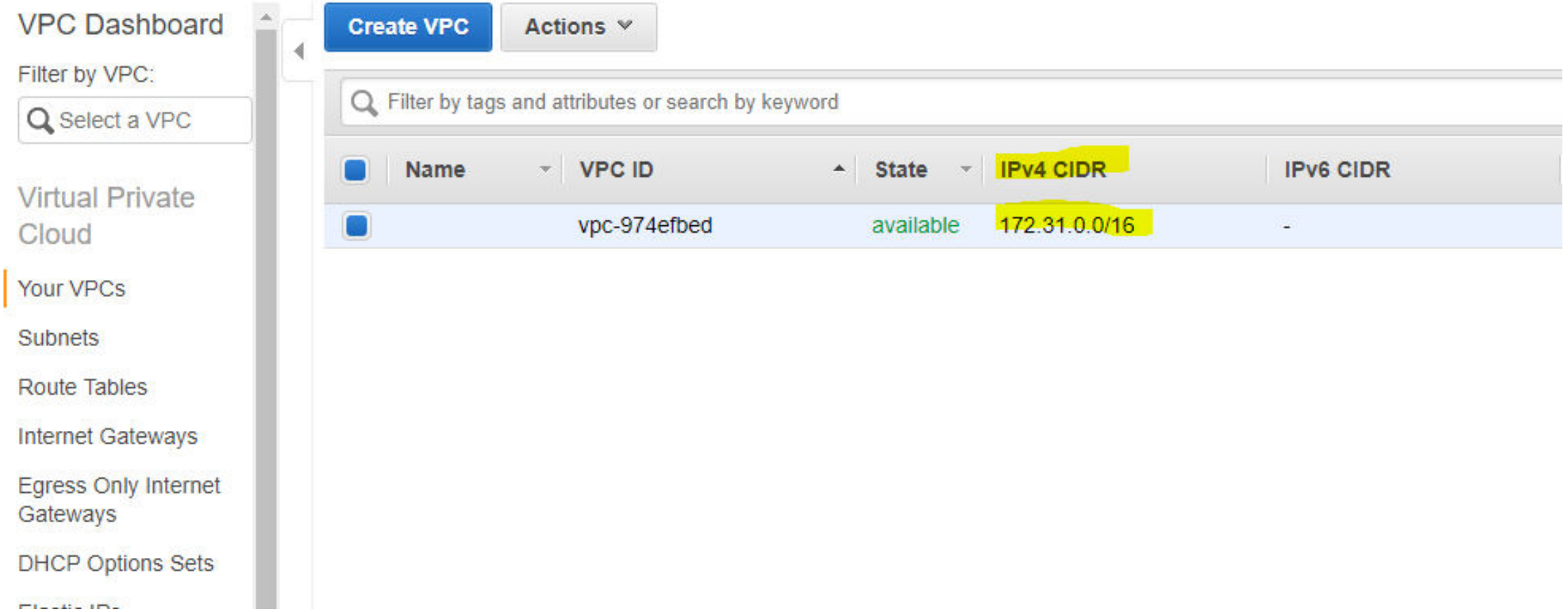
- Max CIDRs per VPC is 5. For each CIDR,
- Min size is /28 = 16 IP Addresses
- Max size is /16 = 65536 IP Addresses

Because VPC is private, only the Private IP ranges are allowed

- 10.0.0.0 – 10.255.255.255 (10.0.0.0/8)
- 172.16.0.0 – 172.31.255.255 (172.16.0.0/12)
- 192.168.0.0 – 192.168.255.255(192.168.0.0/16)

Your VPC CIDR should not overlap with your other networks (ex: corporate)

CIDR


VPC Dashboard
Filter by VPC:
Select a VPC
Virtual Private Cloud
Your VPCs
Subnets
Route Tables
Internet Gateways
Egress Only Internet Gateways
DHCP Options Sets
Create VPC
Actions
Filter by tags and attributes or search by keyword
Name
VPC ID
State
IPv4 CIDR
IPv6 CIDR
vpc-974efbed
available
172.31.0.0/16
-

VPC concepts and terminologies

- VPC is nothing more than a network service provided by AWS using which you can create logically isolated environments for your EC2 instances.
  - **CIDR Block**
  - **Subnets**
  - **Security groups and network ACLs**
  - **Routing tables**
  - **Internet Gateways**
  - **NAT instances/Gateways**

Subnets

- The subnets are nothing more than a range of valid IP addresses that you specify.
- VPC provides you with two different subnet creation options: a publically or Internet routed subnet called as a **Public subnet** and an isolated subnet called as a **private subnet.**

Security Groups

- Security groups are nothing but simple firewall rules that you can configure to safeguard your instances.
- You can create a maximum of 100 security groups for a single VPC, with each Security Group containing up to 50 firewall rules in them.
- Also, it is very important to remember that a Security Group does not permit inbound traffic by default.
- You have to explicitly set inbound traffic rules to allow traffic to flow to your instance.
- However, all outbound traffic from the instance is allowed by default.

Network ACLs

- These provide an added security measure over security groups as they are instance specific, whereas Network ACLs are subnet specific.
- Unlike your security groups, however, you can both allow and restrict inbound and outbound traffic using ACL rules.
- Each ACL rule is evaluated by AWS based on a number. The number can be anything from 100 all the way up to 32,766.
- When you create a new NACL, both inbound and outbound rules are deny by default
- But default NACL, allows everything inbound and outbound as well
- One subnet is associated with one NACL, however NACL can have multiple subnets associated
- Deny will apply first and then allow

Network ACLs

View: All rules

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW
400	HTTP (80)	TCP (6)	80	107.16.110.187/32	DENY
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY
*	ALL Traffic	ALL	ALL	::/0	DENY

Rule #	Type	Protocol	Port Range	Source	Allow / Deny
99	HTTP (80)	TCP (6)	80	107.16.110.187/32	DENY
100	HTTP (80)	TCP (6)	80	0.0.0.0/0	ALLOW
200	HTTPS (443)	TCP (6)	443	0.0.0.0/0	ALLOW
300	SSH (22)	TCP (6)	22	0.0.0.0/0	ALLOW
*	ALL Traffic	ALL	ALL	0.0.0.0/0	DENY
*	ALL Traffic	ALL	ALL	::/0	DENY

Security Groups Vs NACL

Security Group	Network ACL
Operates at the instance level	Operates at the subnet level
Supports allow rules only	Supports allow rules and deny rules
Is stateful: Return traffic is automatically allowed, regardless of any rules	Is stateless: Return traffic must be explicitly allowed by rules
We evaluate all rules before deciding whether to allow traffic	We process rules in number order when deciding whether to allow traffic
Applies to an instance only if someone specifies the security group when launching the instance, or associates the security group with the instance later on	Automatically applies to all instances in the subnets it's associated with (therefore, you don't have to rely on users to specify the security group)

Exam Tips: Network ACL's

- Your VPC automatically comes a default network ACL, and by default it allows all outbound and inbound traffic.
- You can create custom network ACLs. By default, each custom NACL denies all inbound and outbound traffic until you add rules.
- Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a NACL, the subnet is automatically associate with default NACL.
- Block IP addresses using NACL not security groups
- You can associate a network ACL with multiple subnets; however a subnet can be associated with only one NACL at a time. When you associate a NACL with a subnet, previous association will be removed.
- NACL are stateless and it can deny traffic, where in SG it only allows.

Routing Table

- A route table contains rules for routing traffic within a subnet and from the subnet to outside world.
-
- Amongst other things, we use routing tables to add internet gateways and NAT gateways to the subnet.

Internet Gateways

- Internet Gateways, as the name suggest, are primarily used to provide Internet connectivity to your VPC instances.
- An Internet Gateway allows you to make a subnet public by providing a route to the internet.
- All instances within the subnet can access the internet only through this gateway. Also, resources from the internet can access the instances in your subnet using this gateway.

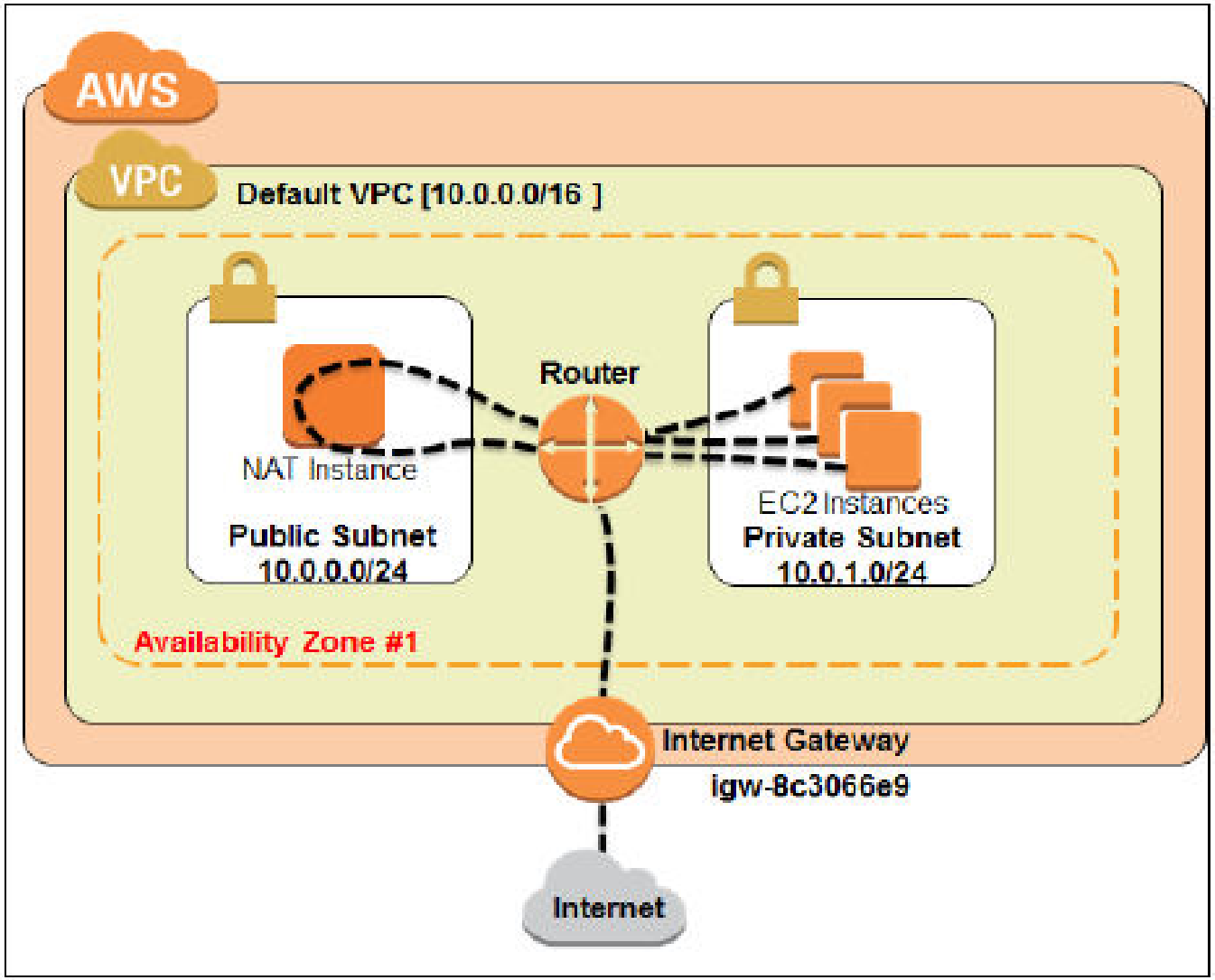
NAT Instances – Outdated

- Allows instances in the private subnets to connect to the internet.
- Must be launched in a public subnet
- Must disable EC2 flag: Source / Destination Check
- Must have Elastic IP attached to it.
- Route table must be configured to route traffic from private subnets to NAT instance.

NAT Gateway

- You can allow instances from your private subnet to connect to the internet using a NAT gateway.
- The instances in the private subnet do not have an public IP address, so the NAT gateway translates the private IP to a public IP before routing the traffic out to the internet.
- NAT stands for Network Address Translation and it does just that – translates private IPs to public IP.

NAT Gateway


AWS
VPC
Default VPC [10.0.0.0/16]
Router
NAT Instance
Public Subnet
10.0.0.0/24
EC2 Instances
Private Subnet
10.0.1.0/24
Availability Zone #1
Internet Gateway
igw-8c3066e9
Internet

Exam Tips: NAT instance & gateway summary

- NAT gateways are generally preferred against NAT instances.
- For NAT instance, you must remember to disable source/destination check for each NAT instance.
- NAT gateways/instances must be placed in public subnets.
- You'll need to configure the routing table of the private subnet to receive traffic via the NAT gateway/instance.
- You bandwidth for NAT instances is limited by the size of the instance it's running on.
- You can achieve high availability from a NAT instances via auto scaling groups and you can achieve high availability from NAT gateways via multiple availability zones.

VPC deployment scenarios

- VPC provides a simple, easy-to-use wizard that can spin up a fully functional VPC within a couple of minutes. All you need to do is select a particular deployment scenario out of the four scenarios provided and configure a few basic parameters such as subnet information, availability zones in which you want to launch your subnets, and so on, and the rest is all taken care of by AWS itself.

- **VPC with a single public subnet**
- **VPC with public and private subnets (NAT)**
- **VPC with public and private subnets and hardware VPN access**
- **VPC with a private subnet only and hardware VPN access**

VPC with a Single Public Subnet

Step 1: Select a VPC Configuration

VPC with a Single Public Subnet

VPC with Public and Private Subnets

VPC with Public and Private Subnets and Hardware VPN Access

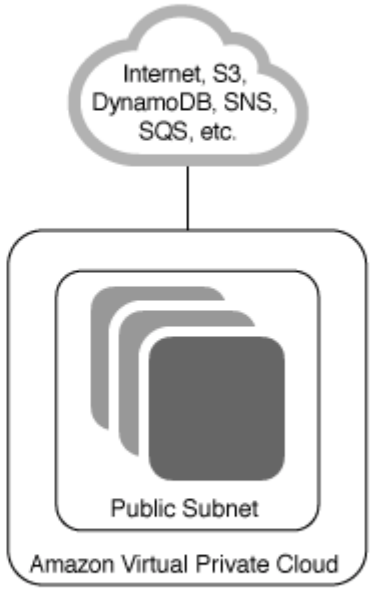
VPC with a Private Subnet Only and Hardware VPN Access

Your instances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

Creates:

A /16 network with a /24 subnet. Public subnet instances use Elastic IPs or Public IPs to access the Internet.

Select



VPC with Public and Private Subnets

Step 1: Select a VPC Configuration

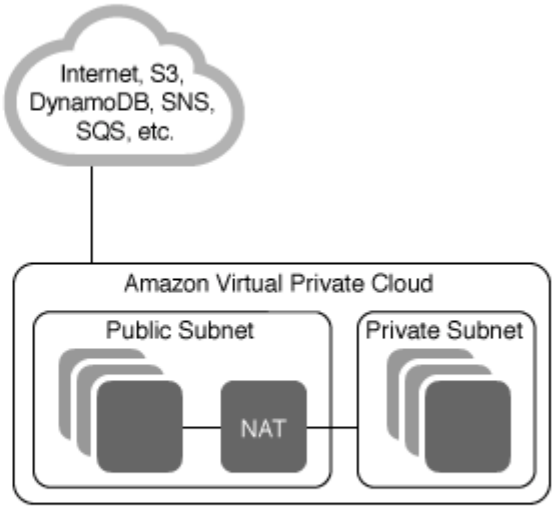
- VPC with a Single Public Subnet
- **VPC with Public and Private Subnets**
- VPC with Public and Private Subnets and Hardware VPN Access
- VPC with a Private Subnet Only and Hardware VPN Access

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.)

Select



VPC with Public and Private Subnets and Hardware VPN Access

Step 1: Select a VPC Configuration

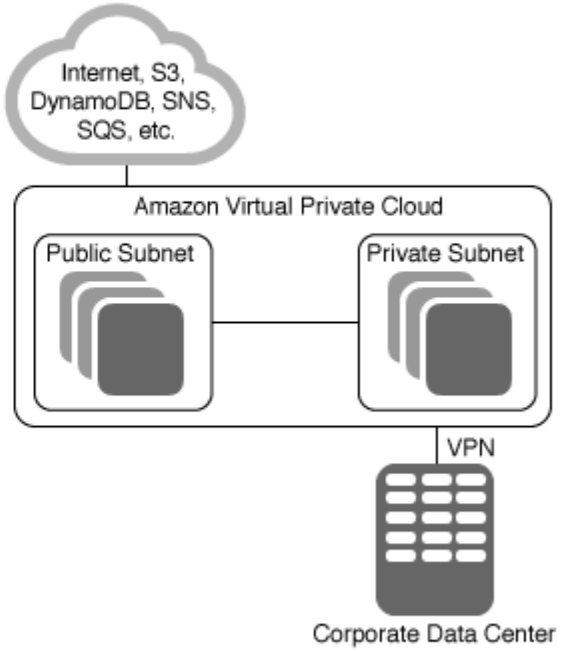
- VPC with a Single Public Subnet
- VPC with Public and Private Subnets
- **VPC with Public and Private Subnets and Hardware VPN Access**
- VPC with a Private Subnet Only and Hardware VPN Access

This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your data center - effectively extending your data center to the cloud while also providing direct access to the Internet for public subnet instances in your Amazon VPC.

Creates:

A /16 network with two /24 subnets. One subnet is directly connected to the Internet while the other subnet is connected to your corporate network via IPsec VPN tunnel. (VPN charges apply.)

Select



VPC with a Private Subnet Only and Hardware VPN Access

Step 1: Select a VPC Configuration

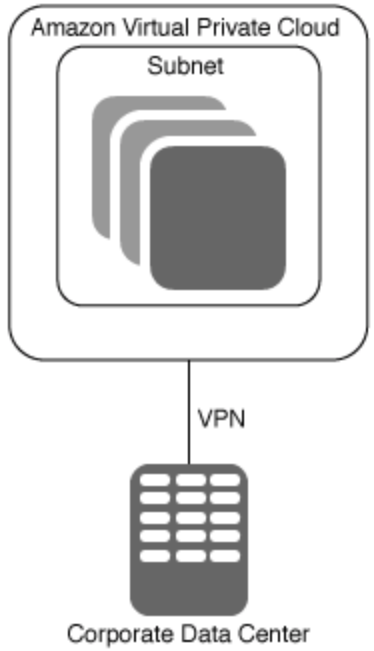
- VPC with a Single Public Subnet
- VPC with Public and Private Subnets
- VPC with Public and Private Subnets and Hardware VPN Access
- **VPC with a Private Subnet Only and Hardware VPN Access**

Your instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate data center via an IPsec Virtual Private Network (VPN) tunnel.

Creates:

A /16 network with a /24 subnet and provisions an IPsec VPN tunnel between your Amazon VPC and your corporate network. (VPN charges apply.)

Select



The Default VPC

The default VPC comes preconfigured with the following set of configurations:

The default VPC is always created with a CIDR block of /16, which means it supports 65,536 IP addresses in it.

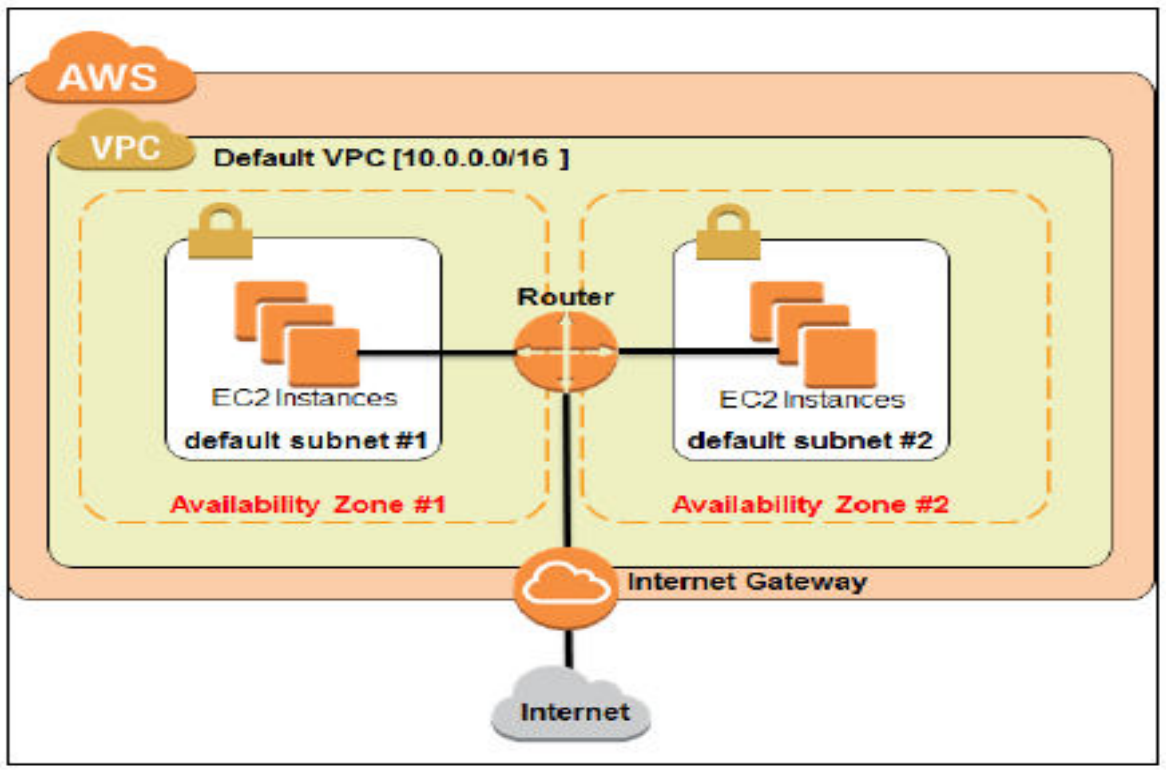
A default subnet is created in each AZ of your selected region. Instances launched in these default

subnets have both a public and a private IP address by default as well.

An Internet Gateway is provided to the default VPC for instances to have Internet connectivity.

A few necessary route tables, security groups, and ACLs are also created by default that enable the instance traffic to pass through to the Internet. Refer to the following figure:

The Default VPC



You can use this default VPC just as any other VPC by creating additional subnets in it, provisioning route tables, security groups, and so on.

Note: Any other VPC that you create besides the default VPC is called as the non-default VPC. Each non-default VPC in turn contains non-default subnets, and so on and so forth.

VPCs and flow logs

VPC flow logs are a way of capturing and recording IP traffic data that enters your VPC from its network interfaces.

As the data is collected it is stored in Amazon CloudWatch Logs and S3. When a flow log is up and running you can review and retrieve data in Amazon CloudWatch Logs.

Flow logs can be created at 3 levels:

- VPC
- Subnet
- Network Interface Level

VPCs and flow logs

Flow logs helps to monitor & troubleshoot connectivity issues

Flow logs Syntax:

<version><account-id><interface-id>
<srcaddr><dsaddr>
<srcport><dsport>
<protocol>
<packets><bytes>
<start><end><action><logstatus>

- Srcaddr, dstaddr help identify problematic IP
- Srcport, dstport help identify problematic ports
- Action: success or failure of the request due to the SG / NACL

Query VPC flow logs using Athena on S3 or Cloudwatch Logs.

Flow log Exam Tips

- Tags are currently not supported by flow logs.

- After a flow log is created, you can NOT change its configuration

- You cannot enable flow logs for VPCs that are peered with your VPC, unless you enable VPC peering for your account.
- After you have created a flow log, you cannot change the configuration like you cant associate IAM roles to flow logs etc

Not all traffic is monitored by flow logs, notable exceptions include:

- DNS server traffic from your VPC instances.

- Traffic generated by windows instances searching for windows license activation.

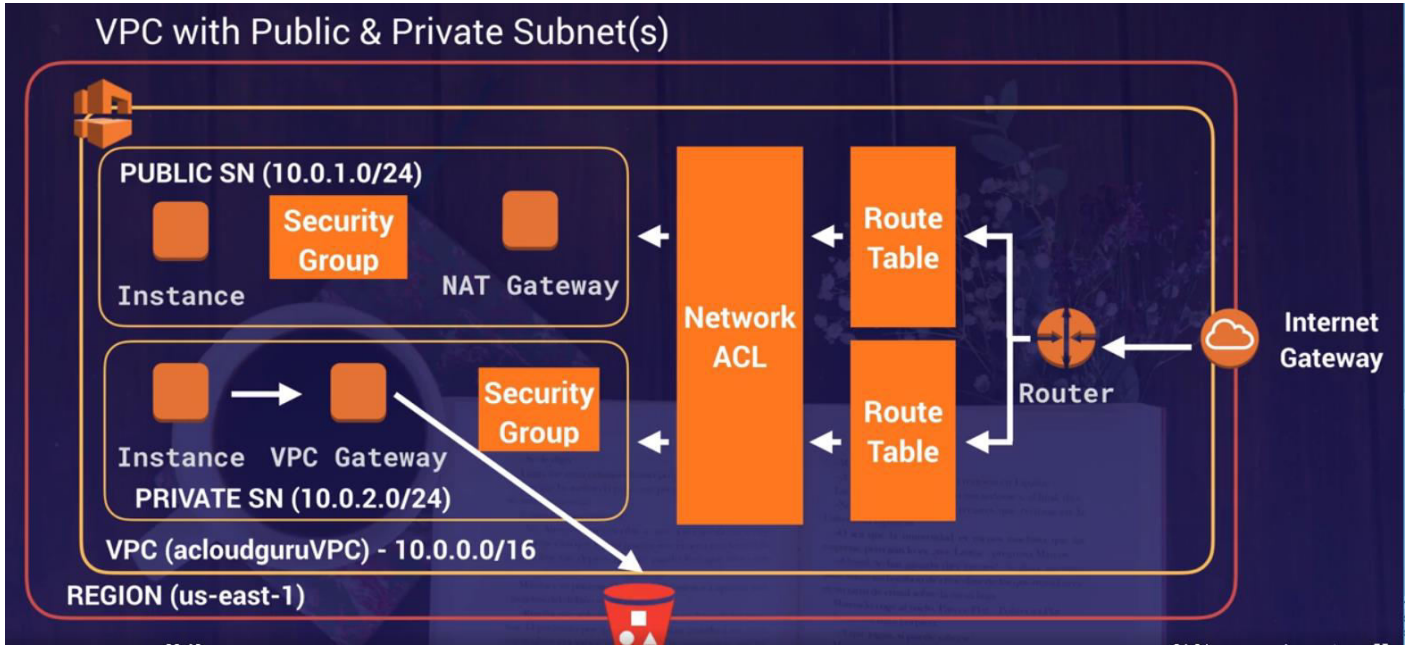
- DHCP traffic.

- Traffic from the reserved IP address for the default VPC router.

VPCs Endpoints

VPC endpoints simply allow for you private instances to access AWS resources such as S3 without the need to gain public internet access(IG), NAT etc. Which of course instances that reside in a private subnet won't have.

By configuring a VPC endpoint those private instances will now be able to access services such as S3 without the next to go via a NAT gateway.



VPCs Endpoints

- A VPC endpoint allows you to securely connect your VPC to another service.
- Endpoints allow you to connect to AWS services using a private network instead of the public network.
- They scale horizontally and are redundant
- They remove the need of IGW, NAT, etc.. To access AWS Services

There are two types of ENDPOINTS

- An interface endpoint is powered by PrivateLink, and uses an elastic network interface (ENI) as an entry point for traffic destined to the service.
- A gateway endpoint serves as a target for a route in your route table for traffic destined for the service.

Currently Gateway Endpoints supports Amazon S3 and DynamoDB

Aws s3 ls –region ap-south-1

VPCs Endpoints

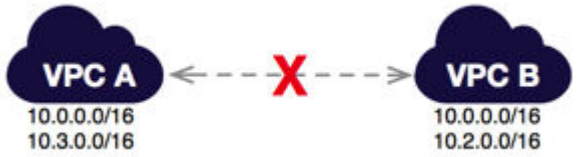
VPC Peering

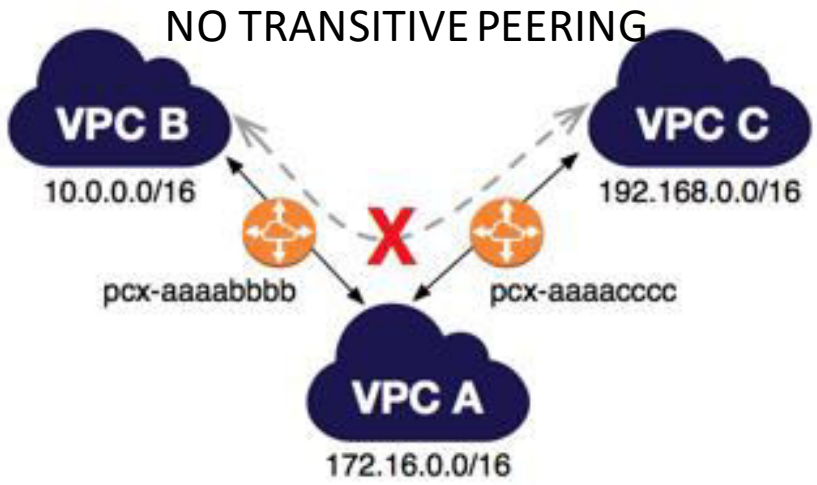
VPC peering is essentially a strong link between 2 separate VPCs.

It allows for communication between the 2 VPCs over private IP addresses as if they were in the same network.

You create VPC peering between VPCs within the same account or even VPCs across multiple accounts provided they are in the same region.

Things to note about VPC peering is that you can't peer 2 VPCs if they share overlapping IP address ranges.





What do I need to know about VPCs?

Internet Gateway: Each VPC has only one internet gateway. It allows the instances inside your VPC access to the internet and allows vise versa.

Virtual Private Gateway: Again, very similar to the internet gateway, allowing an entry point to access the VPC from a remote session.

Router: Dictates where traffic entering the VPC should be sent to.

Router Table: contains a subset of rules determining which subnet traffic is sent to. All subnets must be associated with a route table.

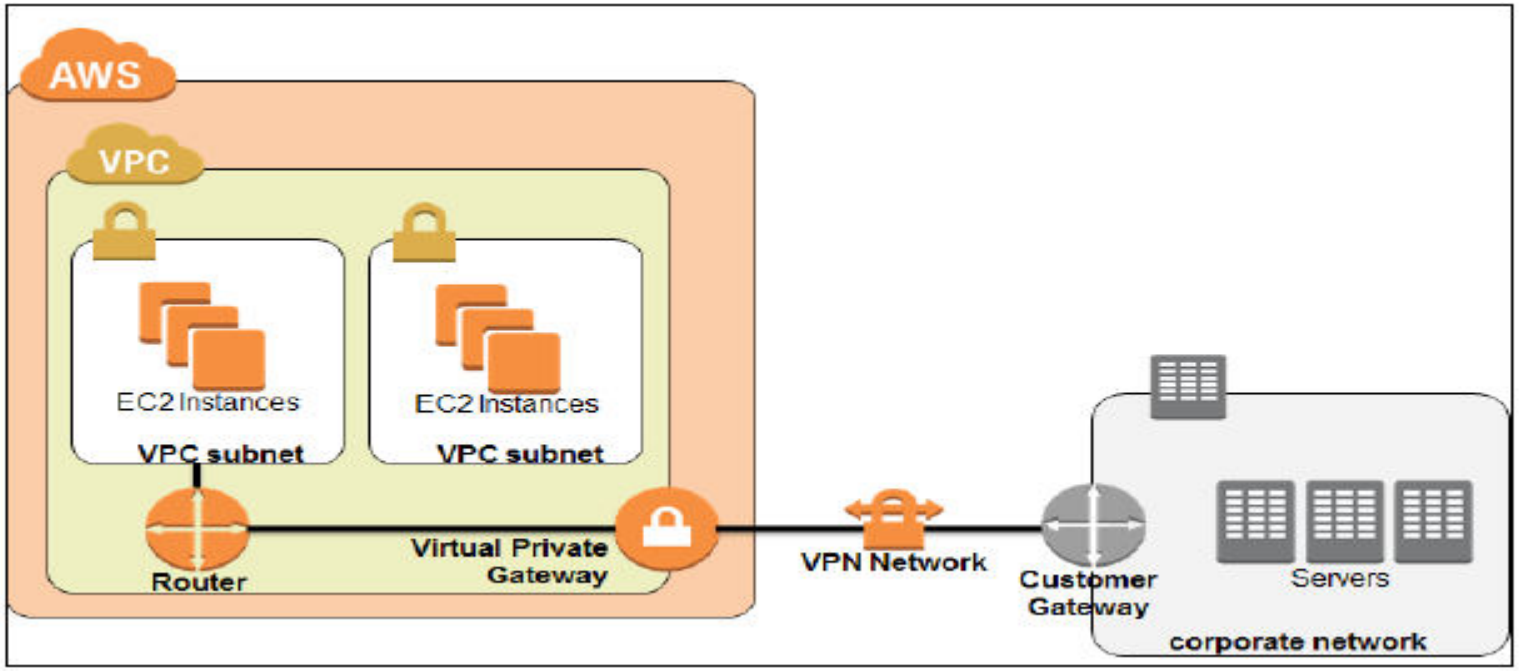
Network Access Control List (NACL): acts as an optional firewall that sits outside of your subnet. All the rules defined in the NACL apply to all of the instances in the subnet.

Subnet: a range of IP addresses that house instances. Each subnet must be allocated a availability zone.

NAT gateway: a means for instances that reside in private subnets to access the internet for tasks such as software updates.

VPC and VPN

- VPCs also provide an added functionality using which you can connect and extend your on premise datacenters to the AWS cloud.
- This is achieved using an IPsec VPN tunnel that connects from your on premise datacenter's gateway device to the VPC's **Virtual Private Gateway**



Site to Site VPN

Virtual Private Gateway

- VPN concentrator on the AWS side of the VPN connection
- VGW is created and attached to the VPC fgrom which you want to crate the site to site VPN connection

Customer Gateway

- Software application or physical device on customer side of the VPN connection
- **IP address:**
- use static, internet-routable IP address for your customer gateway device.
- If behind a CGW behind NAT, use NAT public IP address

Egress only Internet Gateways

- Egress only Internet Gateway is for IPv6 only
- Similar function as NAT, but a NAT is for IPv4
- Good to know: IPv6 are all public address, there are no private IPs in IPV6. In that case you should use IPv4 as private.
- Therefore all our instances with IPv6 are publicly accessible
- Egress only Internet Gateway gives our IPv6 instances access to the internet, but they wont be directly reachable by the internet.
- After creating an Egress Only Internet Gateway, edit the route tables.

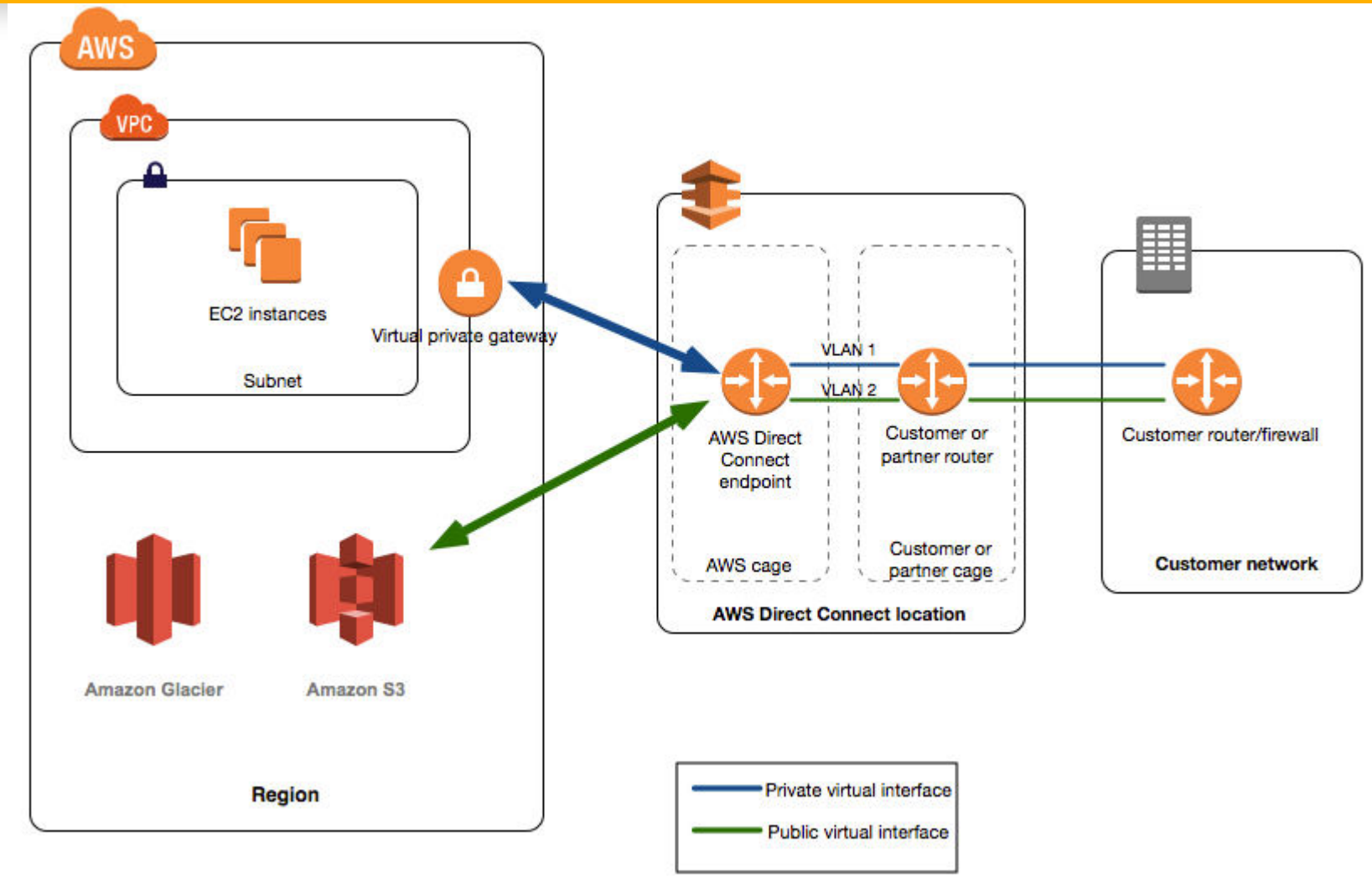
AWS Direct Connect

- Provides a dedicated private connection from a remote network to your VPC.
- Dedicated connection must be setup between your DC and AWS Direct Connect locations.
- You need to setup a Virtual Private Gateway on your VPC
- Access public resources (S3) and private (Ec2) on the same connection

Use cases

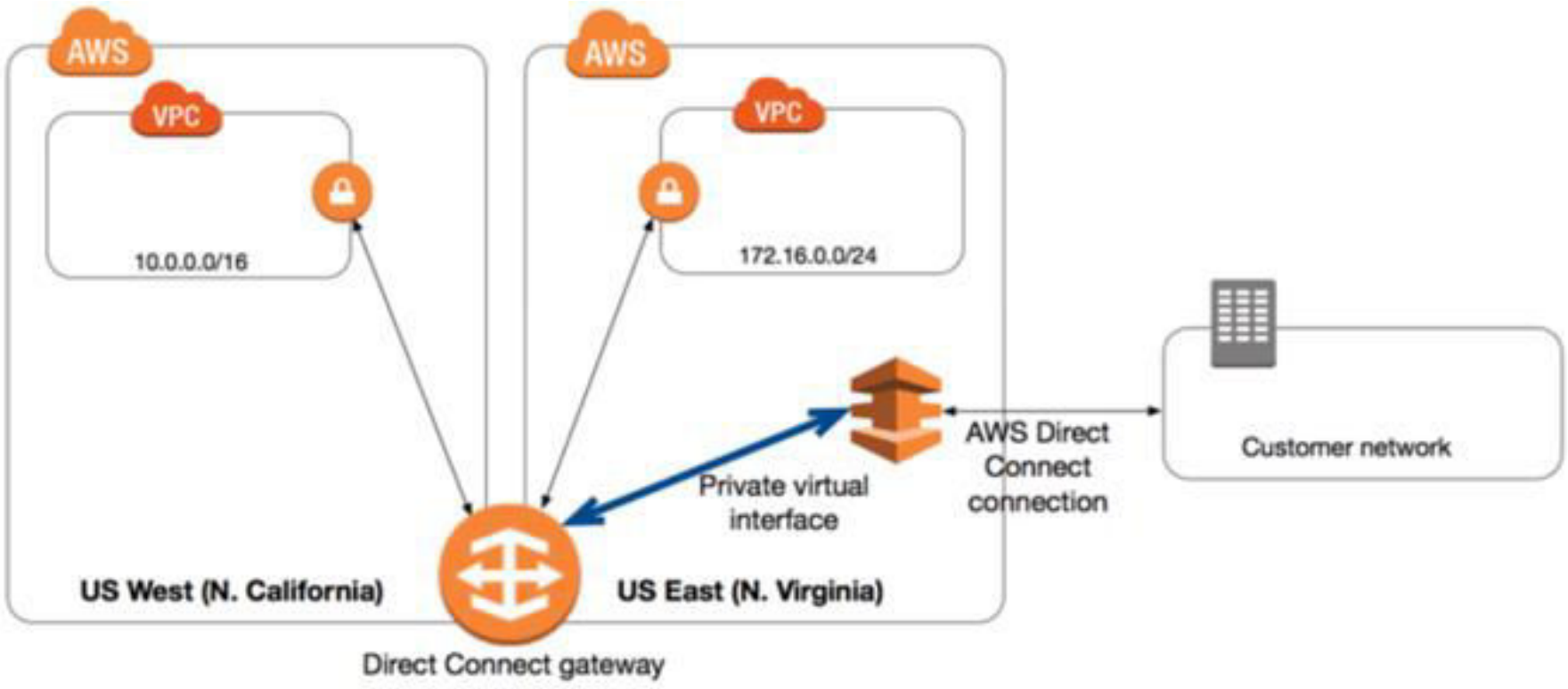
- Increase bandwidth throughput.
- More consistent network experience
- Hybrid Environments (on prem + cloud)

AWS Direct Connect



AWS Direct Connect

If you want to setup a Direct Connect to one or more VPC in many different regions (same account), You must use Direct Connect Gateway



VPC Overview

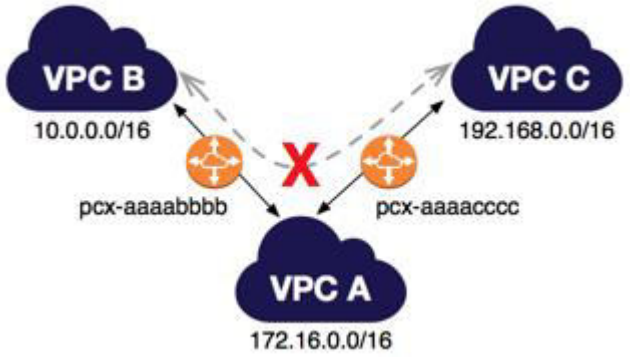
Think of a VPC as a logical datacenter in AWS

Consists of IGWs (or Virtual private Gateways), Route tables, NACLs, subnet and security groups

1 subnet = 1 AZ

Security groups are stateful , NACLs are stateless

NO TRANSITIVE PEERING



VPC Overview

When you create a VPC a default Route table, NACL and a default security group will be created

It wont create any subnets, nor it will create a default internet gateway

Amazon always reserve 5 IP addresses within your subnets (Example: 10.0.0.0/24)

- 10.0.0.0: Network address.
- 10.0.0.1: Reserved by AWS for the VPC router.
- 10.0.0.2: Reserved by AWS for DNS
- 10.0.0.3: Reserved by AWS for future use.
- 10.0.0.255: Network broadcast address

You can have only 1 Internet Gateway per VPC

Security Groups cant span VPC's

VPC Exam Tips

- VPCs are region specific and there is a “limit” on the number of VPC’s you can have per region. That limit is 5. However you can easily increase that limit simply by contacting AWS support and requesting the limit be increased.

- You can not specify multiple internet gateways per VPC. You get only 1 per VPC. This shouldn’t be an issue as they are highly available and if one were to fail then another availability zone would take over.

- A concept that is important to get your head around early on is that a subnet does not span multiple availability zones.

- The default VPC that gets created when you account is setup has subnets associated with it. However they all have internet exposure. None of them are private. This is to make setting up AWS resources simpler initially.

- Lastly, Network Area Control Lists or NACLs for short let you block specific IP address and allow others for your VPC and subnets. This gives a much greater level of control.

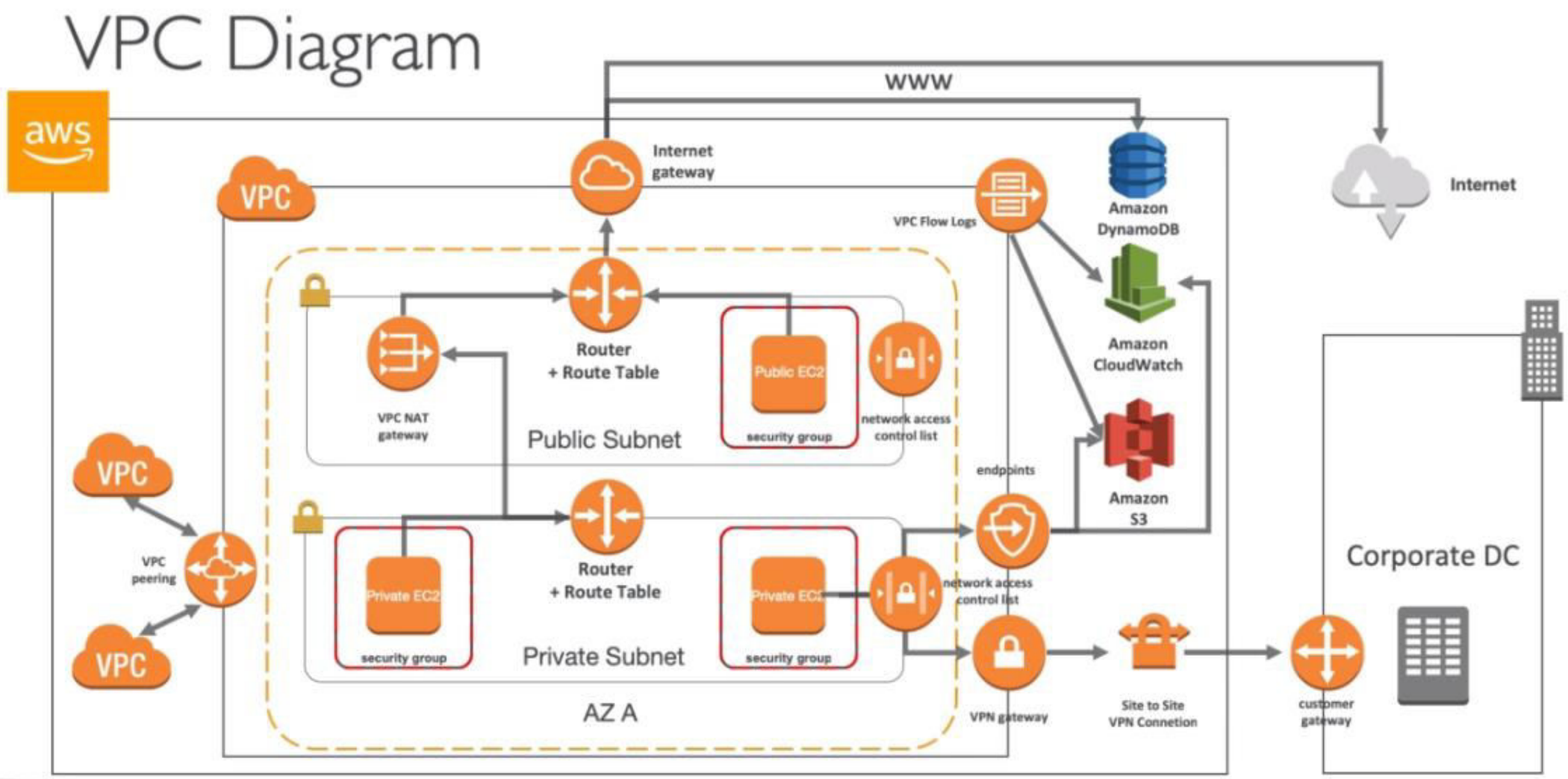
VPC FASK Mode

- **CIDR**: IP Range
- **VPC**: Virtual Private Cloud ➔ we define a list of IPv4 and IPv6 CIDR
- **Subnets**: Tied to an AZ, we define a CIDR
- **Internet Gateway**: at the VPC level, provide IPv4 and IPv6 Internet access
- **Route Tables**: must be edited to add routes from subnets to the IGW, VPC peering, Endpoint, etc…
- **NAT Instances**: Gives internet access to instances in private subnets. Old NAT should setup in public subnet, disable Source / Destination check flag
- **NAT Gateway**: managed by AWS, provides scalable internet access to private instances, IPv4 only
- **Egress Gateway**: just like NAT, NAT is for IPv4 and EG is for IPv6.
- **Private DNS | Route 53**: Enable DNS Resolution + DNS hostnames (VPC)
- **NACL**: Stateless, subnet rules of inbound and outbound.
- **Security Groups**: Stateful, operate at the EC2 instance level

VPC FASK Mode

- **VPC Peering**: Connect two VPC with non overlapping CIDR, non trasitive
- **VPC Endpoints**: Provide private access to AWS services (S3, DynamoDB, Cloudformation, SSM) with in VPC.
- **VPC Flow Logs**: Can be setup at the VPC / Subnet / ENI level for ACCEPT and REJECT traffic, helps identifying attacks, analyze using Athena or Cloudwatch logs.
- **Bastion Host**: Public instance to SSH into, that has SSH connectivity to instances in private subnets.
- **Site to Site VPN:** setup a Customer Gateway on DC, a Virtual Private Gateway on VPC, and site to site VPN over public internet
- **Direct Connect**: Setup a Virtual Private Gateway on VPC, and establish a direct private connection to an AWS Direct Connection Location
- **Direct Connect Gateway:** setup a Direct Connect to many VPC in different regions.

AWS VPC Diagram



Tooooo Much ?