**AWS CSA-Pro Notes ACloudGuru Section 3 Networking**

**Exam Tips:**

**VPCs in General:**

**Know the Pros and Cons of each On-prem to AWS connection mode**

* Talked about in the Network to Network Connectivity section: Page 5 of these notes

**Know the functions of the different VPC components**

* With a VPN you set up a Virtual Private Gateway on the AWS side
* This connects to a Customer Gateway on the on-prem side
* Other VPC components can be:
  + Route tables
  + IGW, Nat Gateways, Egress-only-gateways, Peering connections, endpoints
  + Basic stuff like subnets etc..

**Know that Direct Connect is not inherently redundant, so know how to architect a network with Direct Connect and redundancy**

* You can either setup a secondary Direct Connect preferably with a different ISP so if one Internet Service Provider goes down the second is fine
* The other option would be to setup an AWS VPN as your backup. When your Direct Connect goes down you would lose the benefits of having a dedicated line

**Multicast and Broadcast aren’t supported in VPCs**

**Know what is meant by ‘stateless’, ‘stateful’, ‘connectionless’, and ‘connection-based’**

* Stateful from a port standpoint means firewalls understand the state of traffic. In SG once specific traffic is allowed in it is auto allowed back out because the SG understands that it was previously approved.
* Stateless is when a firewall does not understand the state of traffic. With NACLs you need to allow both Inbound and Outbound rules for traffic to successfully escape.
* Connection-based refers to TCP. There needs to be an ack between the sender and the destination.
* Connectionless refers to UDP. There is no need for an ack because traffic will be sent no matter if someone receives it or not. Think of streaming on twitch. The stream will transmit packets if you are watching or not.

**Know what the ephemeral ports are and why they might need to be in NACLs or SGs**

* More important for NACLs. The uncommon ports of 1024 or higher.
* If you have a website and you are using NACLs you would allow Inbound on 80 and 443. The website port, but because NACLs are stateless and there is no way of knowing which port the traffic is sent back on you would on the Oubound rule allow these ephemeral ports.

**Routing:**

**Understand BGP and how to use weighting to shift network traffic**

* To shift network traffic you would need to assign a higher number to the connection you want your traffic to use.

**Know how routes in route tables are prioritized**

* Routes are prioritized by most specific first

**What other routing protocols does AWS support**

* AWS only supports BGP for routing

**VPC Peering:**

**CIDR ranges cannot overlap**

**After VPC owner accepts a peering request, routes must be added to respective route tables**

* Not adding the route would be like creating an internet gateway and not adding the route and expecting you to have internet access on your EC2 instances

**Transitive peering is not supported, but mesh or hub-and-spoke architectures are**

**A transit VPC is supported**

**Route 53:**

**Understand the different types of routing policies and use cases**

* Page 13

**Know the Weighted Routing formula**

* Specific weight on type and the sum of all weights on the bottom
* 50 50 50 you add all of these up to get 150 and dive 50/150 which gets you 33

**Route 53 is a global service**

**CloudFront:**

**Understand what must happen to use a custom domain with CloudFront**

* You need to use ACM to get a SSL cert on a custom domain name to use with CloudFront

**Understand what SNI enables and the necessary alternative**

**Elastic Load Balancer:**

**Know the 3 different types of Load Balancers and at which OSI layer they work**

* Classic (both 7 and 4)
* Application (Layer 7)
* Network (Layer 4)

**Understand which major features each deliver**

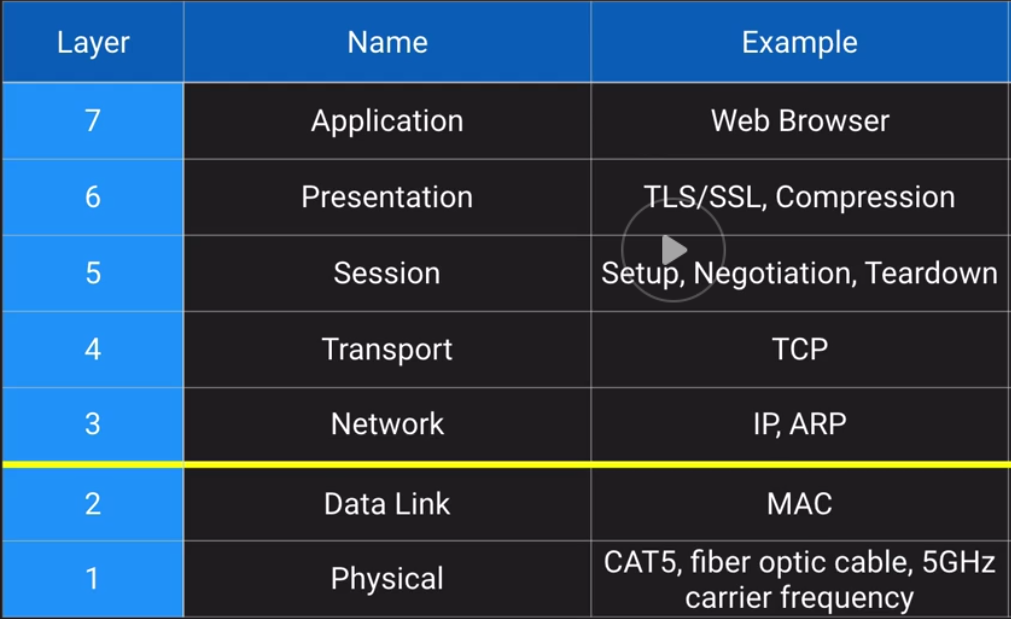
* Page 16

**Know what sticky session are and when they come into play**

* Sticky sessions are used when you want to make sure that clients stay to a single web server to preserve session data
* Just going to point this out, but this is a bad architecture for websites for a couple of reasons
  + If one instance fails then all of the users with session data on those webserver lose it
  + It decreases the impact of autoscaling as usually when an ASG is triggered then users on other web servers being overloaded get rerouted whenever they would navigate to a new page of your website. Obviously if session data is stuck to instances this cannot happen
  + The best practice would be to store session data in a Redis cache and have Redis be setup for High Availability so if one fails it can failover and you do not lose session data.

**Concepts:**

**OSI Model:**

****

Layer 1 and 2 are AWS full responsibility. Once you get to layer 3 AWS takes care some of it while you take care of other parts of that layer. Then from layer 4 and up it is fully your responsibility.

**Unicast vs Multicast**

**Unicast:**

* It is like a phone call to someone
* You dial a phone number and when someone answers you can have a conversation that will not bother anyone else on the network
* A direct conversation between 2 people

**Multicast:**

* Is like someone shouting over a load speaker to everyone
* For everyone to hear

AWS and other cloud providers do not allow multicast. A Network card will send messages to everyone on the network. Often times you are not the only person on the network. This would negatively impact other customers network activities.

**TCP vs UDP vs ICMP:**

**TCP:**

* Connection-based, stateful, acknowledges receipt
* After everything I say, I want you to confirm that you receive it
* Examples: Web, Email, File Transfer

**UDP:**

* Connectionless, stateless, simple, no retransmission delays
* I am going to start talking and it is ok if you miss some words
* Streaming media, DNS

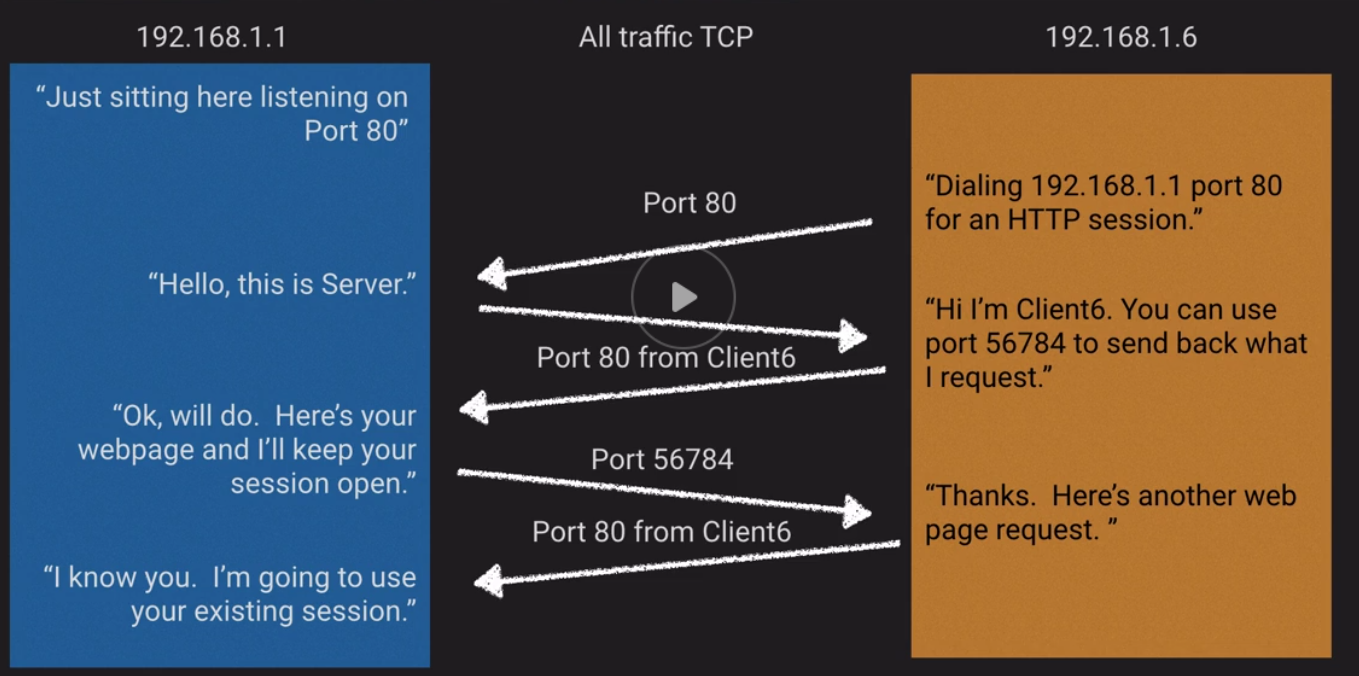
**ICMP:**

* Used by network devices to exchange info
* We routers can keep in touch about the health of the network using our own language
* Traceroute, ping

**Ephemeral Ports:**

* Short-lived transport protocol ports using in IP communications
* Above the “well-known” IP ports (above 1024)
* “Dynamic Ports”
* Suggested range is 49152 to 65535
  + Linux kernels generally use 32568 to 610004
  + Windows platforms default form 1025

**How Ephemeral Ports Work:**



**AWS Availability Zones:**

* The physical to logical assignment of AZ’s is done at the account level
* When your account gets created your physical AZ gets assigned to it’s logical AZ

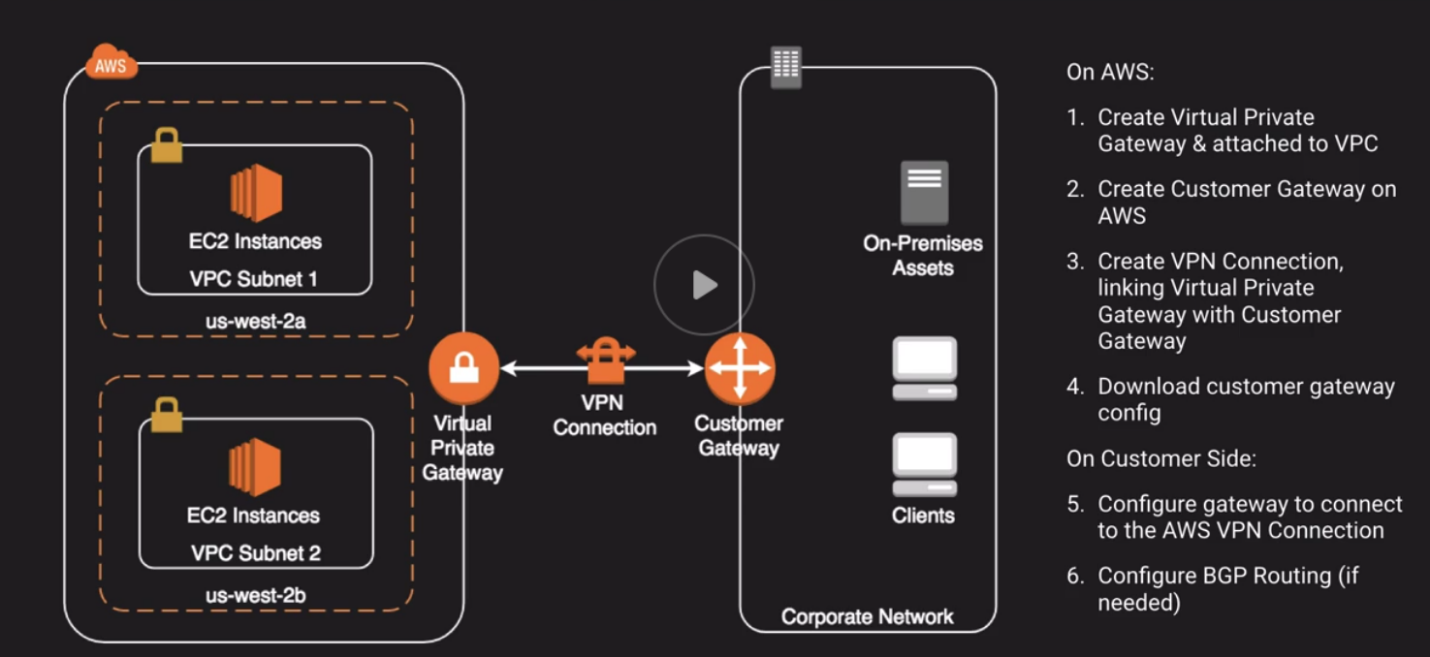
**Network to VPC Connectivity:**

* AWS managed VPN
* AWS Direct Connect
* AWS Direct Connect + VPN
* AWS VPN CloudHub
* Software VPN
* Transit VPC

**Managed VPN:**

* What: AWS managed IPsec VPN connection over your existing internet
* When: Quick and usually simple way to establish a secure tunneled connection to a VPC
* Pros: Supports static routes or BGP peering and routing
* Cons: Dependent on your Internet connection

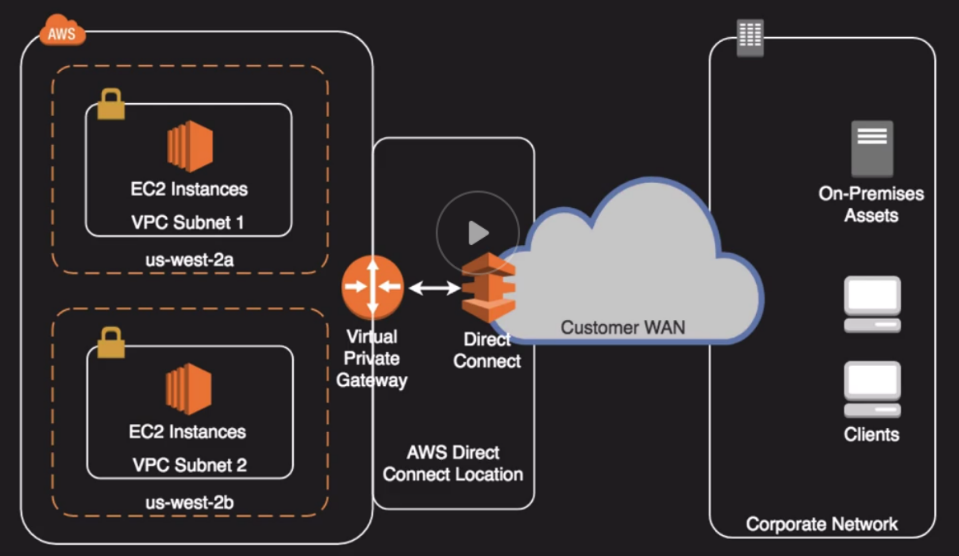
**How to setup Managed VPN:**



In order to setup a redundant connection you would create 2 Customer Gateways on the on-prem side. You would need to manage the failover.

**Direct Connect:**

* What: Dedicated network connection over private lines straight into AWS backbone
* When: Require a “big pipe” into AWS; lots of resources and services being provided on AWS to your corporate users
* Pros: More predictable network performance; potential bandwidth cost reduction; up to 10 Gbps provisioned connection; supports BGP peering and routing
* Cons: May require additional telecom and hosting provider relationships and/or new network circuits
* How: Work with your existing Data Networking Provider; Create Virtual Interfaces (VIF) to connect to VPCs (private VIF) or other AWS services like S3 or Glacier (public VIF)



Direct Connect is not redundant by default.

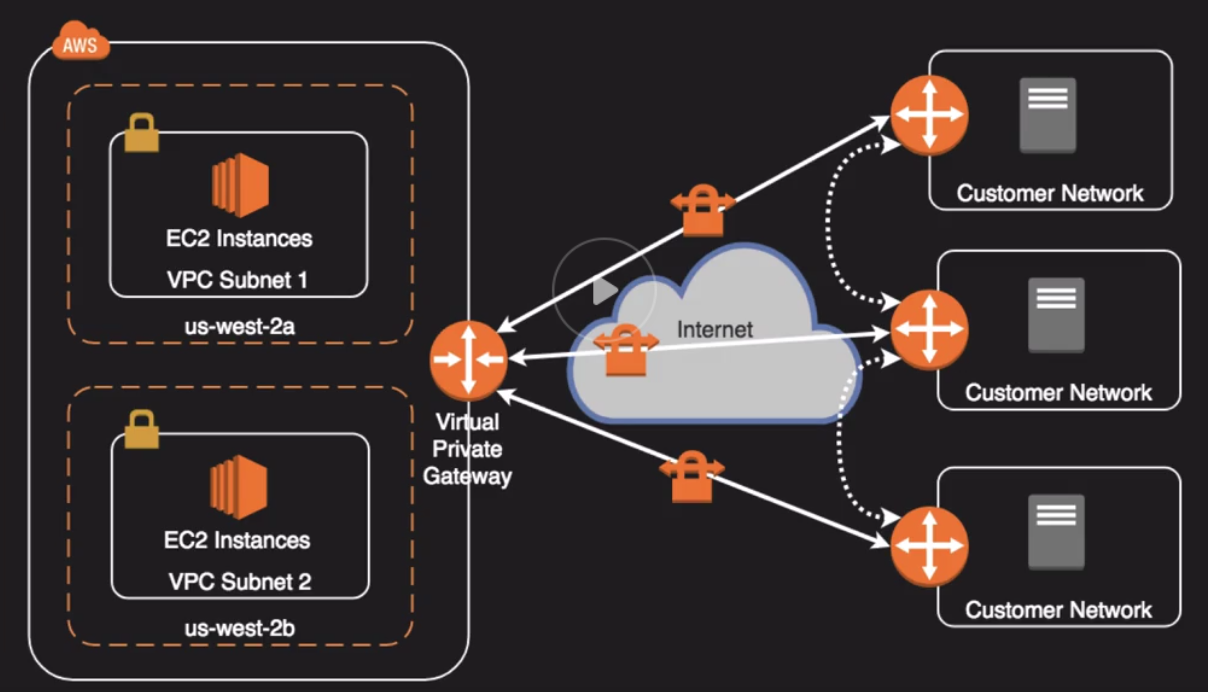
AWS recommend you setup a second connection like a VPN or a second Direct Connect.

**AWS Direct Connect Plus VPN:**

* What: IPsec VPN connection over private lines
* When: Want added security of encrypted tunnel over Direct Connect
* Pros: More secure (in theory) than direct connect alone
* Cons: More complexity introduced by VPN layer

**AWS VPN CloudHub:**

* What: Connect locations in a Hub and Spoke manner using AWS resources and each other
* When: Link remote offices for backup or primary WAN access to AWS resources and each other
* Pros: Reuses existing internet connection; supports BGP routes to direct traffic
* Cons: Dependent on internet connection; no inherent redundancy
* How: Assign multiple Customer Gateways to a Virtual Private Gateway, each with their own BGP ASN and unique IP ranges

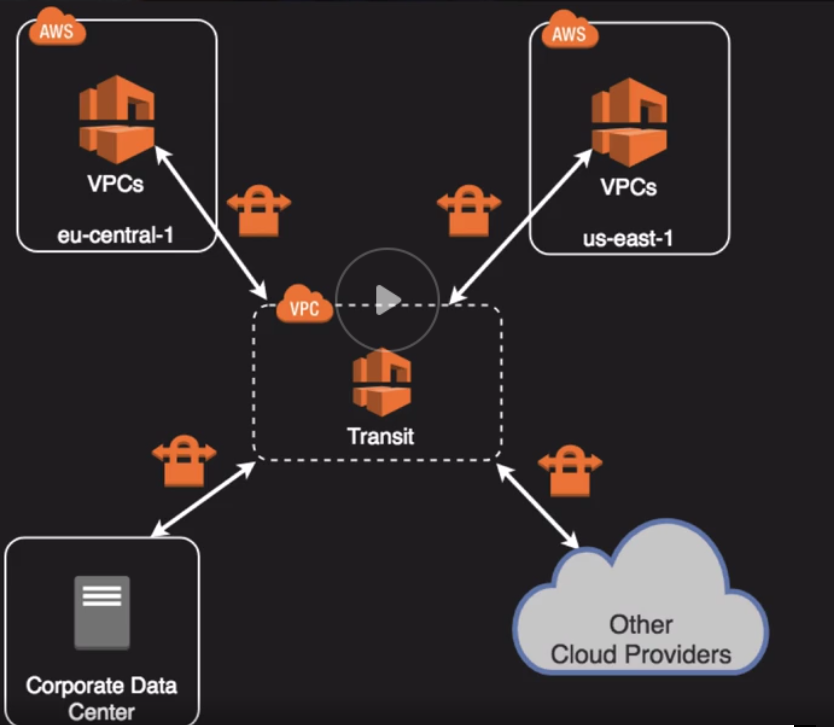


**Software VPN:**

* What: You provide your own VPN endpoint and software
* When: You must manage both ends of the VPN connection for compliance reasons or you want to use a VPN option not supported by AWS
* Pros: Ultimate flexibility and manageability
* Cons: You must design for any needed redundancy across the whole chain
* How: Install VPN software via Marketplace appliance or an EC2 instance

**Transit VPC:**

* What: Common strategy for connection geographically disperse VPCs and locations in order to create a global network transit center
* When: Locations and VPC-deployed assets across multiple regions that need to communicate with on another
* Pros: Ultimate flexibility and manageability but also AWS-managed VPN hub-and-spoke between VPCs
* Cons: You must design for any needed redundancy

****

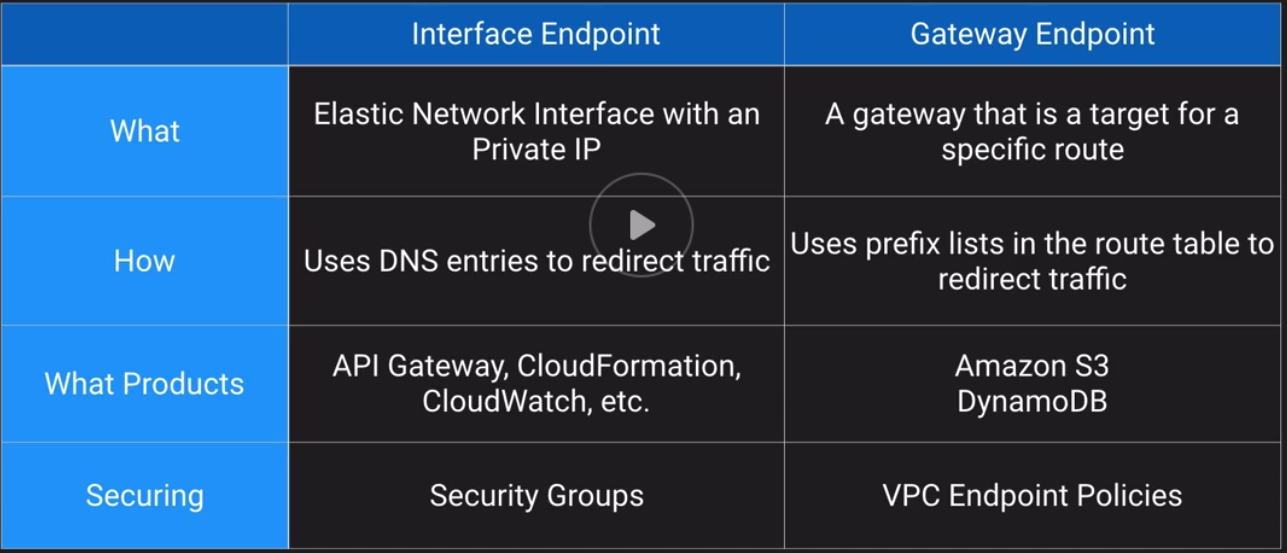
**VPC to VPC Connectivity:**

* VPC Peering
* AWS PrivateLink

**VPC Peering:**

* What: AWS-Provided network connectivity between 2 VPCs
* When: Multiple VPCs need to communicate or access each others resources
* Pros: Uses AWS backbone without touching the Internet
* Cons: No transitive peering
* How: VPC Peering request is made; accepter accepts request; Can be cross account

**AWS PrivateLink:**

* What: AWS-provided network connectivity between VPCs and/or AWS services using interface endpoints
* When: Keep private subnets truly private by using the AWS backbone to reach other services rather than the public internet
* Pros: Uses AWS backbone, redundant
* Only 2 services use the Gateway Endpoint: S3, and DynamoDB

**Internet Gateways:**

* Internet Gateway
* Egress-Only Internet Gateway
* NAT Instance
* NAT Gateway

**Internet Gateway:**

* Horizontally scaled, redundant, and highly available component that allows communication between your VPC and the internet
* No availability risk or bandwidth constraints
* If your subnet is associated with a route to the internet, then it is a public subnet
* Supports IPv4 and IPv6

Purpose 1: Provide route table target for Internet-bound traffic

Purpose 2: Perform NAT for instances with public IP addresses

Does not perform NAT for instances with Private IPs only.

**Egress-Only Internet Gateways:**

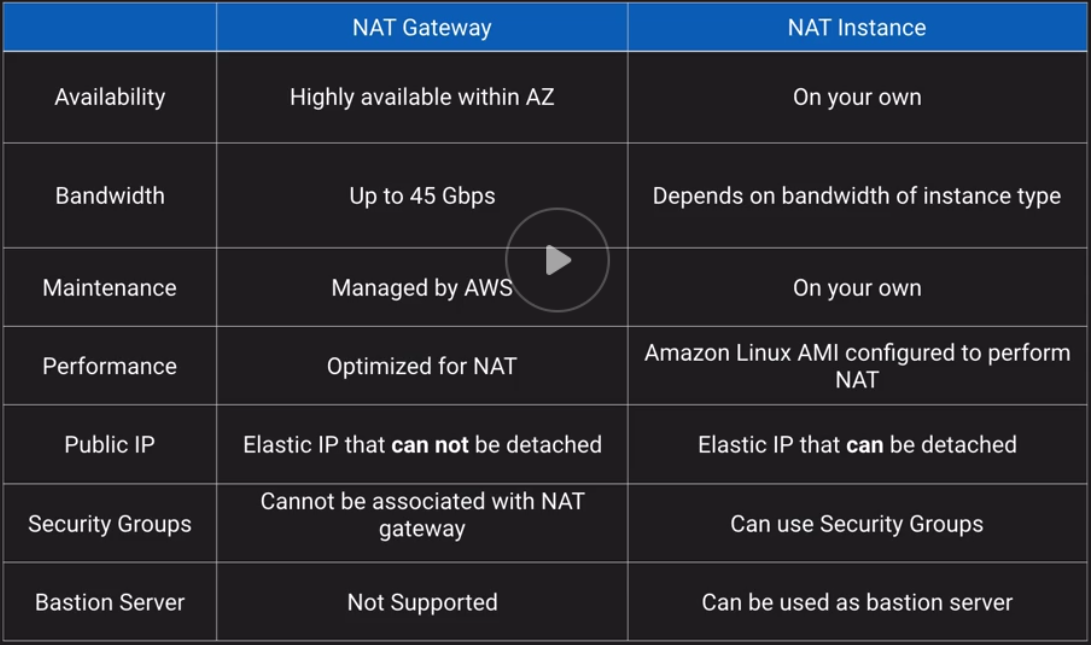
* IPv6 addresses are globally unique and are therefore public by default
* Provides outbound internet access for IPv6 addressed instances
* Prevents inbound access to those IPv6 instances
* Stateful-forwards traffic form instance to internet and then sends back the response
* Must create a custom route for ::/0 to the Egress-Only Internet Gateway
* Use Egress-Only Internet Gateway instead of NAT for IPv6

**NAT Instance:**

* EC2 instance from a special AWS-provided AMI
* Translate traffic form many private IP instances to a single public IP and back
* Does not allow public Internet initiated connections into private instances
* Not supported for IPv6
* NAT Instance must live on a public subnet with route to Internet Gateway
* Private instances in private subnet must have route to the NAT instance, usually the default route destination of 0.0.0.0/0

**NAT Gateway:**

* Fully-managed NAT service that replaced need for NAT instance on EC2
* Must be created in a public subnet
* Uses an Elastic IP for public IP for the life of the Gateway
* Private instances in private subnet must have route to the NAT instance, usually the default route destination of 0.0.0.0/0
* Created in specified AZ with redundancy in that zone
* For multi-AZ redundancy, create NAT Gateways in each AZ with routes for private subnets to use the local Gateway
* Up to 5Gbps bandwidth that can scale up to 46 Gbps
* Can not use a Nat Gateway to access VPC peering, VPN, or Direct connect, so be sure to include specific routes to those in your route table

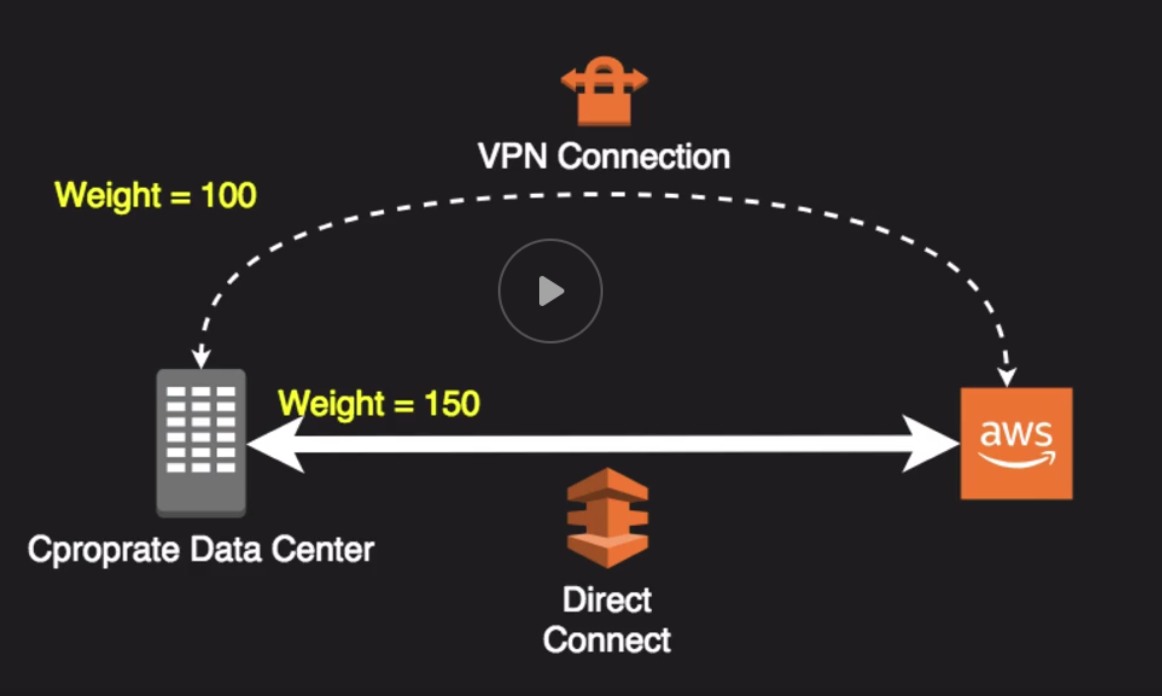


**Routing:**

* VPCs have an implicit router and main routing table
* You can modify the main routing table or create new tables
* Each route table contains a local route for the CIDR block
* Most specific route for an address wins

**Border Gateway Protocol:**

* Popular routing protocol for the internet
* “Propagates” information about the network to allow for dynamic routing
* Required for Direct Connect and optional for VPN
* Alternative of not using BGP with AWS VPC is static routes
* AWS supports BGP community tagging as a way to control traffic scope and route preference
* Required TCP port 179 + ephemeral ports
* Autonomous System Number (ASN) = Unique endpoint identifier
* Weighting is local to the router and higher weight is preferred path for outbound traffic



If you want to transition traffic you would assign a higher number to your preferred path.

If Direct Connect path goes down then the traffic auto routes over the VPN connection.

**Enhanced Networking:**

* Generally used for High Performance Computing use-cases
* Uses single root I/O virtualization to deliver higher performance than traditional virtualized network interfaces
* Might have to install driver if other than Amazon Linux HVM AMI
* Intel 82599 VF Interface 10 Gbps
* Elastic Network Adapter 25 Gbps

**Placement Groups:**

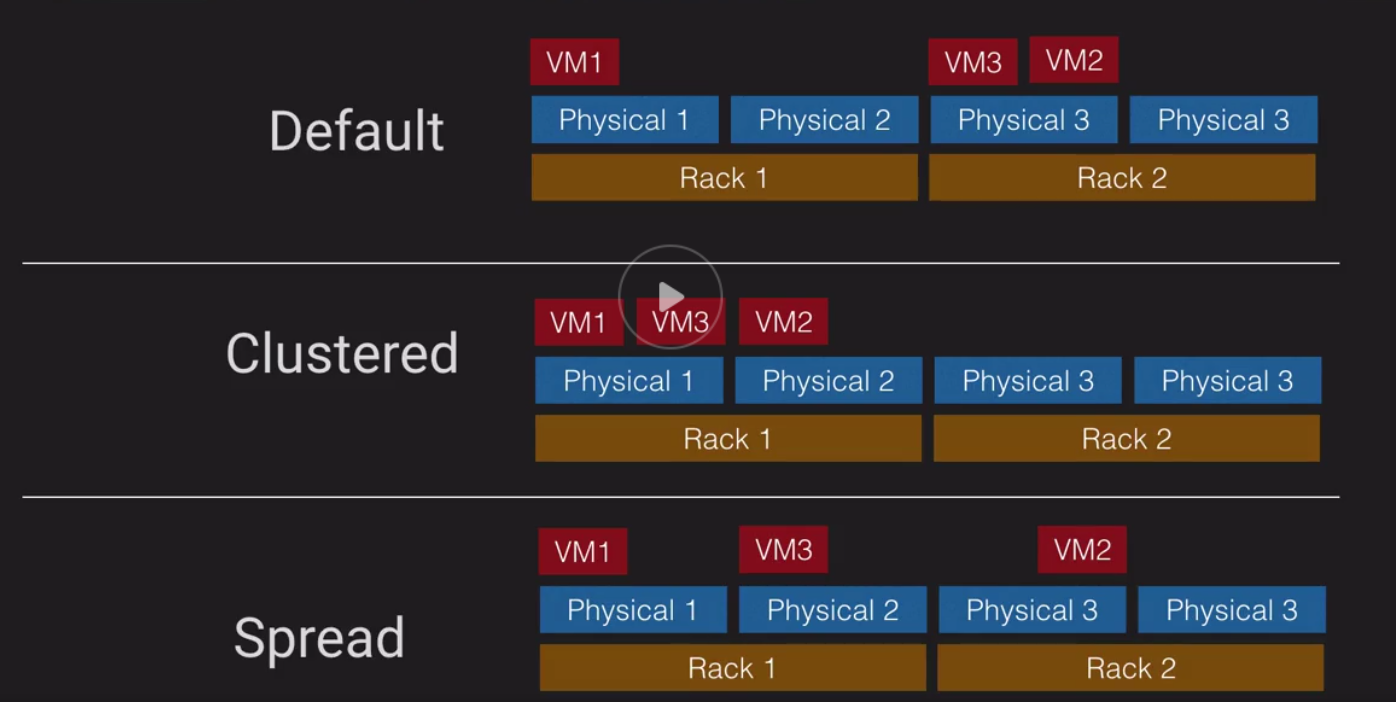
**Clustered:**

* What: Instances are placed into a low-latency group within a single Az
* When: Need low network latency and/or high network throughput
* Pros: Get the most out of Enhanced Networking Instances
* Cons: Finite capacity: recommend launching all you might need up front

**Spread:**

* Instances spread across underlying hardware
* Reduce risk of simultaneous failure if underlying hardware fails
* Can span multiple AZ’s
* Max of 7 instances running per group per AZ

Placement group architectures:



Default Instances can be placed anywhere.

Clustered Instances are placed as close together as possible.

Spread Instances are placed on distinct underlying hardware.

**Route 53:**

* Use it to register domain names
* Check health of your domain resources
* Route internet traffic for your domain

**Route 53 Routing Policies:**

**Simple:**

* Simple: 1 to 1 relationship between a host and the domain

**Failover:**

* Normally, I’d route you to <primary>, but it appears down based on my health checks so I will failover to <Backup>

**Geolocation:**

* Looks like you are in Europe so I will route you to a resource closer to you in that region
* It is important to have a default route for Geolocation just in case Route 53 can’t figure out where a request is coming from

**Geoproximity:**

* You are closer to the US-EAST-1 regions than US-WEST-2 region so I will route you to the east
* Set positive and negative bias to increase or decrease the proximity
* Anywhere from 99 to -99

**Latency:**

* Let me see which resources has lower latency from you, then I will direct you that way

**Multi Value Answer:**

* I will return several IP addresses, as sort of a basic load balancer

**Weighted:**

* You can setup multiple resource and I will route according to the percentage of weight you assign each
* Weight numbers can be from 0 to 255
* You add up all the numbers in the weighted and then you divide a specific record from the sum of all weights

**CloudFront:**

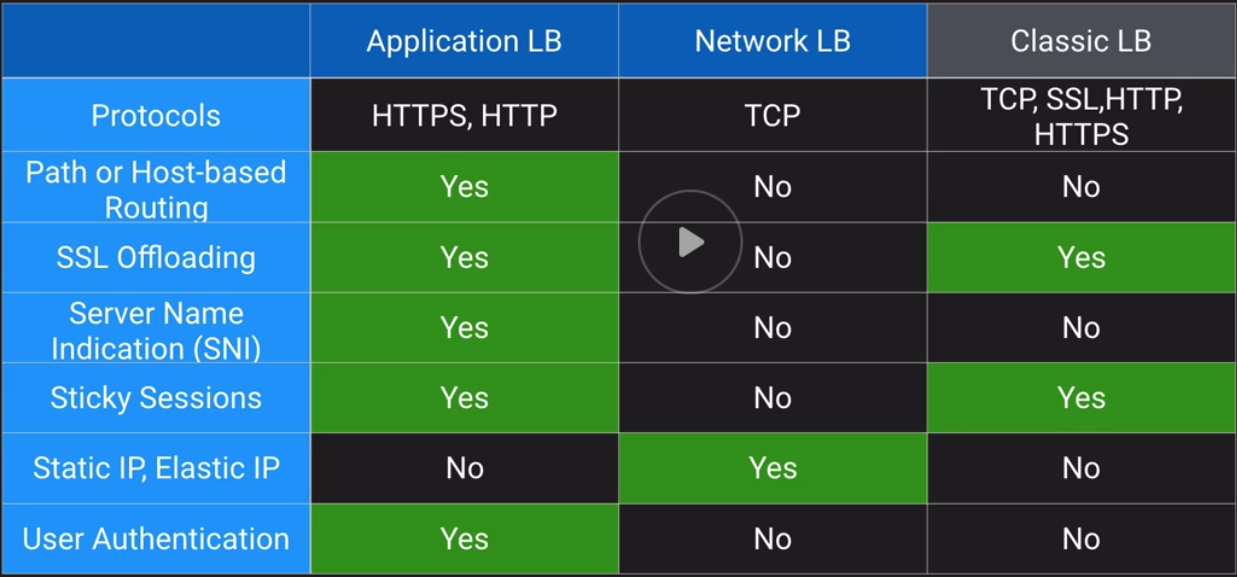
* Distributed content delivery service for simple static asset caching up to 4k live and on-demand video streaming
* Integrated with Amazon Certificate Manager and supports SNI

You can use a default CloudFront certificate for encryption in Transit.

If you want to server this traffic with your own domain you have to use your own SSL cert with a 3rd party like godaddy or you can generate it with ACM. Amazon Certificate Manager

**Elastic Load Balancers:**

* Application Load Balancer: Layer 7
* Network Load Balancer: Layer 4
* Classic Load Balancer: Layer 4 or 7



**Extra:**

What are Jumbo Frames?

These are frames with more than 1500 bytes of payload.

How can you setup your VPC instances to resolve using on-prem DNS?

You need to configure a DHCP option set to issue your on-prem DNS to VPC clients.

Can you access your IGW from a direct connect connection?

You cannot use a Direct Connect or for that matter any traffic coming from on-prem are restricted from accessing the internet.

What are public and private VIFs and when would you use them?

AWS Direct Connect provides 2 types of Virtual Interfaces (VIFs):

* Public- Use when you want to connect to AWS public endpoints such as S3
* Private- Use to connect to your VPC to access EC2, ELBs, and RDS