



Module **5**: Networking and Content Delivery

2081/04/25

Networking Basics

Network: A computer network is when two or more machines are connected to communicate with each other. Networks can be divided into smaller sections called subnets. To connect these machines and allow them to communicate, we use devices like routers or switches.

IP Address: Every machine on a network has a unique identifier called an IP address, which is like a phone number for computers. An IP address is written as four numbers separated by dots (e.g., 192.0.2.0). Each number can range from 0 to 255, and together they make up 32 bits, known as an IPv4 address. However, there's also an IPv6 address, which uses 128 bits and can support more devices because IPv4 addresses are running out.

IPv6 Address:

An IPv6 address is designed to support more devices on the internet, as IPv4 addresses are running out. Unlike IPv4, which uses 32 bits, an IPv6 address uses 128 bits. This allows for a much larger number of unique addresses.

An IPv6 address is composed of eight groups of four characters (which can be letters or numbers) separated by colons. For example:

`2001:0db8:85a3:0000:0000:8a2e:0370:7334`. This format allows IPv6 to accommodate a vast number of devices, making it future-proof as the number of internet-connected devices continues to grow.

Classless Inter-Domain Routing (CIDR):

Classless Inter-Domain Routing (CIDR) is a method for allocating IP addresses more efficiently. Instead of using the old system of class-based IP addresses (Class A, B, C, etc.), CIDR allows for more flexible and precise IP address assignments.

In CIDR notation, an IP address is followed by a slash (/) and a number, which indicates how many bits of the address are used for the network portion. For example, `192.168.1.0/24` means that the first 24 bits of the IP address are for the network, leaving the remaining bits for individual devices on that network. This system helps in reducing the wastage of IP addresses.

OSI Model:

The OSI (Open Systems Interconnection) model is a framework used to understand how different networking protocols interact and work together. It is divided into seven layers, each with a specific role:

1. **Physical Layer:** Deals with the physical connection between devices (e.g., cables, switches).
2. **Data Link Layer:** Manages the data transfer between devices on the same network (e.g., Ethernet).
3. **Network Layer:** Handles the routing of data between different networks (e.g., IP addresses).
4. **Transport Layer:** Ensures that data is transferred reliably and without errors (e.g., TCP).
5. **Session Layer:** Manages sessions or connections between applications.
6. **Presentation Layer:** Translates data between the network and application layers (e.g., encryption, data format translation).
7. **Application Layer:** Provides the interface for the end-user to interact with the network (e.g., web browsers, email clients).

Each layer of the OSI model interacts with the layers directly above and below it, ensuring that data is transmitted from one device to another in a structured and reliable way.

Layer	Number	Function	Protocol/Address
Application	7	Means for an application to access a computer network	HTTP(S), FTP, DHCP, LDAP
Presentation	6	<ul style="list-style-type: none"> Ensures that the application layer can read the data Encryption 	ASCII, ICA
Session	5	Enables orderly exchange of data	NetBIOS, RPC
Transport	4	Provides protocols to support host-to-host communication	TCP, UDP
Network	3	Routing and packet forwarding (routers)	IP
Data link	2	Transfer data in the same LAN network (hubs and switches)	MAC
Physical	1	Transmission and reception of raw bitstreams over a physical medium	Signals (1s and 0s)

Amazon Virtual Private Cloud (Amazon VPC):

Amazon VPC is a service that lets you create a private section of the AWS cloud, known as a VPC. This is like having your own private space in the AWS cloud where you can launch virtual machines and other resources.

With Amazon VPC, you have control over your virtual network. You can choose your own IP address range, create subnets, configure route tables, and set up network gateways. You can use both IPv4 and IPv6 addresses in your VPC to securely access resources and applications. You also have the flexibility to customize the network configuration of your VPC.

Example:

You can divide your VPC into subnets and create:

- A **public subnet** for your web servers that need access to the internet.
- A **private subnet** for your backend systems, like databases or application servers, which don't need internet access.

Additionally, you can apply multiple layers of security, such as security groups and network access control lists (ACLs), to control access to Amazon EC2 instances in each subnet. This helps ensure that only authorized users can access your resources.

Amazon VPC:

- **Isolated Network:** A VPC is a virtual network in AWS that is isolated from other networks.
- **Dedicated to Your Account:** Each VPC is specific to your AWS account and belongs to a single AWS region.
- **Multi-AZ Support:** A VPC can span multiple Availability Zones (AZs) for high availability.
- **Subnet Creation:** After creating a VPC, you can divide it into one or more subnets.

Subnets:

- **Isolated Segments:** Subnets are segments of your VPC with their own range of IP addresses.
- **Availability Zone Bound:** Each subnet is tied to a single Availability Zone.
- **Public vs. Private:** Public subnets have internet access; private subnets do not.
- **High Availability:** You can create subnets in different AZs for redundancy.

IP Addresses and CIDR Blocks:

- **Communication:** IP addresses enable communication within the VPC and with the internet.
- **CIDR Block:** When you create a VPC, you assign it a CIDR block (range of IP addresses).
- **IPv4 Limits:**
 - Largest block: `/16` (65,536 addresses).
 - Smallest block: `/28` (16 addresses).
- **IPv6 Support:** You can also assign an IPv6 CIDR block to your VPC and subnets.
- **No Overlap:** Subnet CIDR blocks cannot overlap; each must be unique within the VPC.

Reserved IP Addresses:

- **Reserved by AWS:** When you create a subnet, AWS reserves five IP addresses in the subnet's CIDR block for specific purposes:
 - Network address
 - Internal communication
 - DNS resolution
 - Future use
 - Network broadcast address

Public IP Address Types:

1. Public IPv4 Address:

- **Automatic Assignment:** A public IPv4 address is automatically assigned to an AWS resource, like an EC2 instance, when it is launched in a public subnet.
- **Direct Internet Access:** It allows the resource to communicate directly with the internet.
- **Dynamic:** The address is released when the instance is stopped or terminated, meaning you lose it if the instance is stopped and then restarted.

2. Elastic IP Address:

- **Static Public IP:** An Elastic IP address is a static public IPv4 address that you can allocate to your AWS account.
- **Reassignable:** You can attach it to any instance in your account, and it remains yours until you release it. This means you can move it between instances, keeping the same IP address even if you stop and start instances.
- **Use Case:** Elastic IP addresses are useful when you need a consistent public IP address for long-running instances or to recover from instance failures by quickly remapping the IP address to a different instance.

Elastic Network Interface (ENI):

- **Virtual Network Card:** An Elastic Network Interface (ENI) is like a virtual network card for your EC2 instances.
- **Multiple ENIs:** You can attach multiple ENIs to a single EC2 instance, allowing it to connect to different networks or subnets.
- **Flexible Networking:** ENIs are useful for managing network traffic, such as isolating traffic for security or load balancing.
- **Detachable:** You can detach an ENI from one instance and attach it to another, making it flexible for network configurations and migrations.

Route Tables and Routes:

1. Route Tables:

- **Traffic Rules:** A route table is like a set of rules that controls where network traffic is directed within your VPC.
- **Associated with Subnets:** Each subnet in your VPC is associated with a route table that determines how the traffic in that subnet is routed.
- **Default Route Table:** AWS automatically creates a default route table for your VPC, which you can customize.

2. Routes:

- **Path Definitions:** A route is an individual rule within a route table that specifies the destination of network traffic.
- **Targets:** Routes direct traffic to specific targets, such as an internet gateway, a NAT gateway, or another subnet within the VPC.
- **Custom Routes:** You can add or modify routes in a route table to control how traffic flows between different parts of your VPC or out to the internet.



Section 2 key take aways

A VPC is a logically isolated section of the AWS Cloud.

A VPC belongs to one Region and requires a CIDR block.

A VPC is subdivided into subnets

A subnet belongs to one Availability Zone and requires a CIDR block.

Route tables control traffic for a subnet.

Route tables have a built in local route.

You add additional routes to the table.

The local route cannot be deleted.

VPC Networking

Internet Gateway

An internet gateway is a scalable, redundant, and highly available VPC component that enables communication between instances in your VPC and the internet. It serves two primary purposes: providing a target in your VPC route tables for internet-routable traffic and performing network address translation for instances assigned public IPv4 addresses. To make a subnet public, you attach an internet gateway to your VPC and add a route in the route table to direct non-local traffic through the internet gateway to the internet (0.0.0.0/0). A network address translation (NAT) gateway allows instances in a private subnet to connect to the internet or other AWS services while preventing the internet from initiating a connection with those instances. When creating a NAT gateway, you must specify the public subnet and associate an Elastic IP address with the NAT gateway. Once created, the route table associated with one or more private subnets must be updated to point internet-bound traffic to the NAT gateway, enabling instances in private subnets to communicate with the internet. AWS recommends using a NAT gateway over a NAT instance for better availability, higher bandwidth, and reduced administrative effort.

VPC sharing allows customers to share subnets with other AWS accounts within the same organization, enabling multiple accounts to create application resources in shared, centrally managed VPCs. The account that owns the VPC shares subnets with participant accounts, allowing them to view, create, modify, and delete resources within the shared subnets, while ensuring security and separation of duties. VPC sharing offers benefits like efficient use of resources, optimized costs, and simplified network architecture.

VPC peering connects VPCs within the same AWS account, between accounts, or even across AWS regions, allowing instances in either VPC to communicate as if they are within the same network. Peering connections require specific route table configurations and come with restrictions, such as non-overlapping IP spaces and the need for explicit connections between VPCs. For more advanced connectivity, an AWS Site-to-Site VPN connection can be established, linking a VPC with a remote network. This setup involves creating a VPN gateway, configuring the customer gateway, and updating route tables and security groups to facilitate communication between the VPC and the remote network.

VPC Peering

- **Definition:** VPC peering is a networking connection between two VPCs, allowing them to route traffic privately as if they were in the same network.
- **Setup:** You create a peering connection and then configure route tables to direct traffic between the VPCs.
- **Restrictions:**
 - IP ranges of the VPCs cannot overlap.
 - Transitive peering is not supported (VPC B cannot connect to VPC C through VPC A).
 - Only one peering connection is allowed between two VPCs.

AWS Site-to-Site VPN

- **Purpose:** Allows instances in a VPC to communicate with a remote network through a VPN connection.
- **Setup:**

1. Create a virtual private gateway and attach it to your VPC.
2. Define the customer gateway configuration to provide VPN device information to AWS.
3. Create a custom route table to direct traffic to the VPN gateway.
4. Establish a Site-to-Site VPN connection.
5. Configure routing to pass traffic through the VPN connection.

AWS Direct Connect

- **Purpose:** Provides a dedicated, private network connection between your data center and AWS to improve network performance.
- **Benefits:** Reduces network costs, increases bandwidth throughput, and provides a more consistent network experience compared to internet-based connections.
- **Technology:** Uses 802.1q VLANs for the connection.

VPC Endpoints

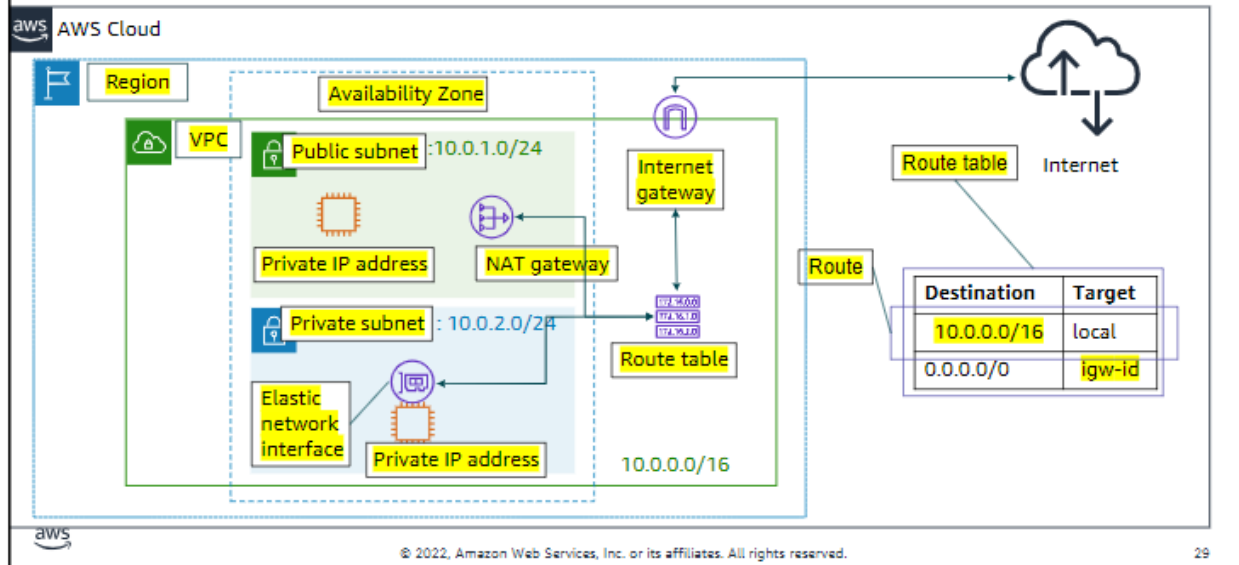
- **Definition:** A VPC endpoint is a virtual device that allows private connections from your VPC to supported AWS services without needing an internet gateway or other external connections.
- **Types:**
 - **Interface Endpoints:** Powered by AWS PrivateLink, these connect to services within AWS or hosted by other customers/APN Partners.
 - **Gateway Endpoints:** Specifically for Amazon S3 and DynamoDB, allowing private access with no extra charge.

AWS Transit Gateway

- **Purpose:** Simplifies network management by acting as a central hub for connecting multiple VPCs, on-premises networks, and remote offices.
- **Functionality:** Centralizes traffic routing, reducing the need for point-to-point connections between VPCs, which simplifies scaling and management.

- **Benefit:** Reduces operational costs and complexity by requiring only one connection to the transit gateway for each network.

Activity: Solution



Source: AWS academy

VPC Networking Options



VPC Networking Options:

- **Internet Gateway:** Connects your VPC to the internet, allowing public access to instances within the VPC.
- **NAT Gateway:** Enables instances in a private subnet to access the internet while blocking inbound connections.
- **VPC Endpoint:** Provides a private connection between your VPC and supported AWS services without using the internet.
- **VPC Peering:** Allows you to connect your VPC to other VPCs, enabling communication between them as if they were in the same network.
- **VPC Sharing:** Enables multiple AWS accounts to share and manage application resources within a centrally managed Amazon VPC.
- **AWS Site-to-Site VPN:** Establishes a secure connection between your VPC and remote networks.
- **AWS Direct Connect:** Provides a dedicated network connection between your VPC and a remote network, improving performance.
- **AWS Transit Gateway:** Acts as a hub-and-spoke connection model, simplifying the management of multiple VPCs and networks, and offering an alternative to VPC peering.

VPC Wizard:

- **Purpose:** The VPC Wizard is a tool that helps implement your VPC design by simplifying the setup of these networking options.

These points provide a summary of the essential VPC networking options and tools, which are crucial to understand for AWS certification exams.

VPC Security

Two amazon VPC firewall options that you can use to secure your VPC. The first, security groups. The second, network access control lists.

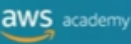
Security groups

A security group acts as a virtual firewall that controls inbound and outbound traffic, to and from your instance. Security groups acts at the instance level. Particularly, the network interface card and you can assign each instance in your VPC to different set of security groups. Think of security group as way to filter traffic, to and from your instances. Security groups are the equivalent of firewalls for your EC2 instances. They contain rules to allow inbound traffic. By default security groups are sealed shut. You then proceed to define the type of traffic that will be allowed. Security groups are stateful. We only concern ourselves with defining the inbound traffic rules. the outbound traffic is always allowed.

Network Access Control Lists

Network access control lists, work at the subnet level and control traffic in and out of the subnet. You can set up network ACLs (Access Control Lists) with rules that allow or deny. you can also specify ports and protocols. Each subnet in your VPC must be associated with a network ACL. If you don't explicitly associate a subnet with a network ACL then the default network ACL is used. You can associate a network ACL with multiple subnets. However, a subnet can only be associated with one network ACL.

A network ACL and subnet relationship is one-to-one relationship. A network ACL is stateless means that no information about a request is maintained after a request is processed.. It has separate inbound and outbound rules that require configuration.

Security groups versus network ACLs 		
Attribute	Security Groups	Network ACLs
Scope	Instance level	Subnet level
Supported Rules	Allow rules only	Allow and deny rules
State	Stateful (return traffic is automatically allowed, regardless of rules)	Stateless (return traffic must be explicitly allowed by rules)
Order of Rules	All rules are evaluated before decision to allow traffic	Rules are evaluated in number order before decision to allow traffic

Amazon Route 53 service

Amazon Route 53 Overview:

Amazon Route 53 is a highly available and scalable cloud Domain Name System (DNS) web service. It is designed to route users to internet applications by translating domain names (like www.example.com) into the numeric IP addresses (like 192.0.2.1) that computers use to connect to each other. It is fully compliant with both IPv4 and IPv6, and it connects user requests to infrastructure running in AWS, such as Amazon EC2 instances, Elastic Load Balancing load balancers, or Amazon S3 buckets. Additionally, Route 53 can be used to route users to infrastructure outside of AWS.

DNS Resolution and Traffic Flow:

Amazon Route 53 allows users to configure DNS health checks to route traffic to healthy endpoints, ensuring high availability. It supports various routing types like latency-based routing, geolocation routing, and failover routing, which can be managed using a simple visual editor. This flexibility enables the creation of low-latency, fault-tolerant architectures, whether in a single AWS Region or globally distributed.

Routing Policies:

Amazon Route 53 supports several routing policies, each catering to different needs:

- **Simple routing:** Used in single-server environments.
- **Weighted round-robin routing:** Allows traffic distribution to multiple resources in specified proportions, useful for A/B testing.
- **Latency routing:** Routes traffic to the AWS Region providing the best latency, improving the user experience.
- **Geolocation routing:** Routes traffic based on the user's location, enabling localized content delivery.
- **Failover routing:** Provides active-passive failover, where traffic is redirected to a backup site if the primary site becomes unreachable.

Multi-Region Deployment:

An example use case for Route 53 is multi-region deployment, where the service automatically directs users to the Elastic Load Balancing load balancer that is geographically closest. This approach offers latency-based routing to ensure that users experience the lowest possible latency.

DNS Failover

Amazon Route 53 improves application availability by enabling DNS failover. Users can configure backup and failover scenarios to ensure that if the primary site becomes unavailable, traffic is redirected to a backup site, such as a static Amazon S3 website. This setup is critical for maintaining high availability in multi-region architectures.

Amazon CloudFront

One of the challenges of network communication is network performance. When you browse to a website your request is routed through different networks. The origin server stores the original version of the data which is commonly high density, like images, songs, or even videos. The distance between customer and original data server significantly affects performance in the playback and user experience. Also, network latency happens to be different depending on the geographic location of your users.

For this reason, a content delivery network is an essential part of smooth user experience. Amazon CloudFront is a fast content delivery service that securely delivers data to customers at high transfer speeds. It also provides a developer friendly environment.

CloudFront delivers files to users over a global network of edge locations. It is different from traditional content delivery solutions because you can take advantage of high-performance content delivery without negotiated contracts, high prices or minimum fees. Like other AWS services, Amazon CloudFront is self-service offering with pay-as-you-go pricing.



Edge Locations: Network of data centers that CloudFront uses to serve popular content quickly to customers.

Regional edge cache: CloudFront location that caches content that is not popular enough to stay at an edge location. It is located between the origin server and the global edge location.

Amazon CloudFront relies on Route 53's geolocation routing. Basically, a customer makes a request. Route 53 finds out where the customer is located in the world, and it responds with the IP address of edge location closest to that customer. CloudFront then obtains the data from where it normally lives, copies it to edge location. Then the customer's user experience begins. As data becomes stale, it is removed from the cache at the edge location in order to make room for new content. You can define the expiration of data in the cache using a time-to-live number. This defines the amount of time in which the data cache will remain valid.

2081/04/25

Thank you 😊