Table of Contents

AWS Connectivity

1 AWS Connectivity

Amazon Web Services (AWS) provides a comprehensive set of services and features to establish connectivity within your cloud infrastructure and between your on premises data centers and the AWS Cloud.

2 Core Components

The core components that play a crucial role in AWS connectivity (Connectivity, n.d.) Include Virtual Private Cloud (VPC), Internet Gateways, Direct Connect, Virtual Private Network (VPN), Security Groups, and more.

2.1 Virtual Private Cloud (VPC):

Definition: A logically isolated section of the AWS Cloud where you can launch AWS resources.

Purpose: Enables you to define your own virtual network topology, including IP address range, subnets, and route tables.

2.2 Internet Gateways (IGW):

Definition: A horizontally scaled, redundant component that allows communication between instances in your VPC and the internet.

Purpose: Facilitates internet connectivity for resources in public subnets.

2.3 Direct Connect:

Definition: A dedicated network connection from your on premises data center to AWS.

Purpose: Provides more reliable and consistent connectivity compared to internet based connections, suitable for large data transfers and sensitive workloads.

2.4 Virtual Private Network (VPN):

AWS VPN Services: AWS offers managed VPN solutions, including Site-to-Site VPN, allowing you to securely connect your on premises network to AWS.

Purpose: Extends your corporate network into the cloud securely.

2.5 Security Groups:

Definition: Acts as a virtual firewall for your instances, controlling inbound and outbound traffic.

Purpose: Manages access to instances, enhancing security by specifying allowed traffic.

2.6 Network Access Control Lists (Network ACLs):

Definition: An optional layer of security for your VPC, operating at the subnet level.

Purpose: Allows or denies inbound and outbound traffic at the subnet level, providing an additional layer of control.

2.7 Elastic Load Balancing (ELB):

Definition: Automatically distributes incoming application traffic across multiple targets.

Purpose: Enhances availability, fault tolerance, and scalability of applications by distributing traffic among healthy instances.

3 Advanced networking & Load Balancing

Advanced networking (Advanced Networking, n.d.) And load balancing (Load Balancing, n.d.) Are crucial aspects of designing and managing scalable, resilient, and high performance applications on cloud platforms like Amazon Web Services (AWS). Here is an overview of advanced networking concepts and load balancing in the AWS context:

3.1 Advanced Networking in AWS:

3.1.1 Virtual Private Cloud (VPC) Peering:

Definition: VPC peering allows direct networking connections between two VPCs.

Purpose: Enables resources in separate VPCs to communicate securely as if they are on the same network.

3.1.2 Transit Gateway:

Definition: AWS Transit Gateway is a service that simplifies the network architecture by connecting multiple VPCs and on premises networks.

Purpose: Centralizes connectivity, simplifies route management, and improves scalability for larger network architectures.

3.1.3 Elastic Network Interfaces (ENIs):

Definition: ENIs provide networking capabilities for EC2 instances in a VPC.

Purpose: Allows instances to have multiple network interfaces with different IP addresses and MAC addresses, facilitating advanced networking configurations.

3.1.4 VPC Endpoints:

Definition: VPC endpoints enable private connectivity between your VPC and supported AWS services without traversing the internet.

Purpose: Enhances security and performance by avoiding internet data transfer AWS services.

3.1.5 AWS Direct Connect Gateway:

Definition: Connects multiple Virtual Interfaces (VIFs) from the same or different AWS Direct Connect locations to a Direct Connect Gateway.

Purpose: Simplifies connectivity between on premises networks and multiple VPCs.

3.2 Load Balancing in AWS:

3.2.1 Elastic Load Balancing (ELB):

Application Load Balancer (ALB): Distributes incoming application traffic across multiple targets, such as EC2 instances, at the application layer (HTTP/HTTPS).

Network Load Balancer (NLB): Routes traffic at the transport layer (TCP/UDP), offering ultralow latency and high throughput.

Classic Load Balancer (CLB): Legacy load balancer supporting both application and network level traffic.

3.2.2 Auto Scaling:

Definition: Dynamically adjusts the number of EC2 instances in a group based on specified policies or conditions.

Purpose: Maintains application availability and performance by automatically scaling capacity up or down.

3.2.3 Target Groups:

Definition: Used with ALB and NLB to route requests to register targets.

Purpose: Allows efficient routing of requests based on different criteria, such as host, path, or IP address.

3.2.4 Network Load Balancer TCP/UDP Listener Rules:

Definition: NLB supports flexible rules for routing TCP/UDP traffic.

Purpose: Enables complex routing scenarios for applications with diverse requirements.

3.2.5 Cross-Zone Load Balancing:

Definition: Distributes traffic evenly across instances in all enabled Availability Zones.

Purpose: Enhances fault tolerance and ensures even utilization of resources.

4 Conclusion

In conclusion, Amazon Web Services (AWS) offers a robust set of connectivity solutions and core components that empower users to build secure and scalable cloud infrastructures. The Virtual Private Cloud (VPC) serves as the foundation, allowing users to create isolated environments with customizable network configurations. Internet Gateways facilitate internet connectivity, while Direct Connect and VPN services establish secure connections between on premises data centers and the AWS Cloud.

Security is enhanced through the use of Security Groups and Network Access Control Lists (Network ACLs), providing fine grained control over inbound and outbound traffic at both the instance and subnet levels. Elastic Load Balancing (ELB) ensures high availability and optimal resource utilization by distributing incoming application traffic across multiple targets.

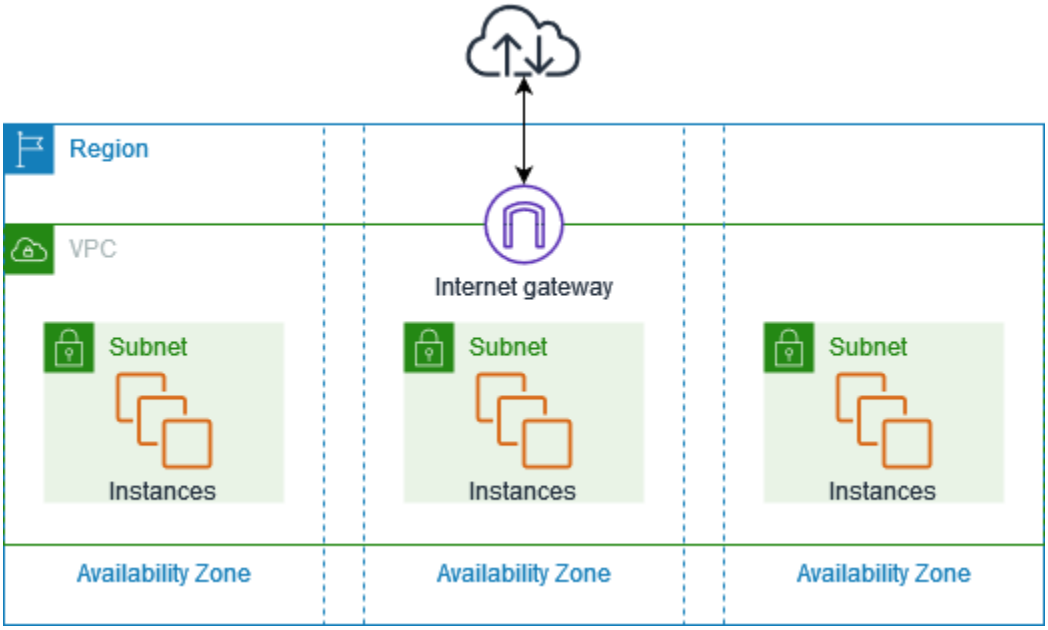
This comprehensive suite of connectivity features empowers users to design and implement network architectures tailored to their specific needs, whether for hosting public facing applications, establishing secure connections with on premises infrastructure, or ensuring the integrity and confidentiality of data within the AWS environment.

Amazon Virtual Private Cloud (VPC)

1 Amazon Virtual Private Cloud (VPC)

With Amazon Virtual Private Cloud (Amazon (VPC, n.d.)), you can launch AWS resources in a logically isolated virtual network that you have defined. This virtual network closely resembles a traditional network that you would operate in your own data center, with the benefits of using the scalable infrastructure of AWS.

The following diagram shows an example VPC. The VPC has one subnet in each of the Availability Zones in the Region, EC2 instances in each subnet, and an internet gateway to allow communication between the resources in your VPC and the internet.



2 Features

The following features help you configure a VPC to provide the connectivity that your applications need:

2.1 Virtual private clouds (VPC)

A VPC is a virtual network that closely resembles a traditional network that you would operate in your own data center. After you create a VPC, you can add subnets.

2.2 Subnets

A subnet is a range of IP addresses in your VPC. A subnet must reside in a single Availability Zone. After you add subnets, you can deploy AWS resources in your VPC.

2.3 IP addressing

You can assign IP addresses, both IPv4 and IPv6, to your VPCs and subnets. You can also bring your public IPv4 addresses and IPv6 GUA addresses to AWS and allocate them to resources in your VPC, such as EC2 instances, NAT gateways, and Network Load Balancers.

2.4 Routing

Use route tables to determine where network traffic from your subnet or gateway is directed.

2.5 Gateways and endpoints

A gateway connects your VPC to another network. For example, use an internet gateway to connect your VPC to the internet. Use a VPC endpoint to connect to AWS services privately, without the use of an internet gateway or NAT device.

2.6 Peering connections

Use a VPC peering connection to route traffic between the resources in two VPCs.

2.7 Traffic Mirroring

Copy network traffic from network interfaces and send it to security and monitoring appliances for deep packet inspection.

2.8 Transit gateways

Use a transit gateway, which acts as a central hub, to route traffic between your VPCs, VPN connections, and AWS Direct Connect connections.

2.9 VPC Flow Logs

A flow log captures information about the IP traffic going to and from network interfaces in your VPC.

2.10 VPN connections

Connect your VPCs to your on premises networks using AWS Virtual Private Network (AWS VPN).

3 Where VPCs live

VPCs (Virtual Private Clouds) in AWS are confined to a specific AWS region. Here are some key points about the location and characteristics of VPCs in AWS:

3.1 Regional Scope:

Each VPC is associated with and confined to a single AWS region. AWS regions are distinct geographic locations globally where Amazon has data centers, and they serve as the foundation for AWS services.

3.2 Advantages of Regionalization:

Regional VPCs provide network services that originate from a specific geographical area. This enables users to optimize performance and reduce latency for applications by locating resources closer to the intended users.

3.3 Scalability and Closer Access:

If users need to provide closer access for customers or resources in another AWS region, they can set up an additional VPC in that specific region. This aligns with the scalable and distributed nature of AWS services.

3.4 Cloud Computing Principles:

The theory of AWS cloud computing involves delivering IT applications and resources over the internet on demand, with a pay as you go pricing model. This allows users to scale resources as needed and aligns with the core principles of cloud computing.

3.5 Selective Network Service Provisioning:

By limiting VPC configurations to specific regions, users can selectively provide network service where they are needed and when they are needed. This flexibility supports efficient resource allocation and optimization.

3.6 Multiple VPCs in an Account:

Each Amazon account can host multiple VPCs. This allows users to create and manage separate VPCs to isolate resources, applications, and network configurations.

3.7 Isolation and Duplicity:

VPCs are isolated from each other, meaning they operate independently. Users can duplicate private subnets among VPCs, similar to using the same subnet in different physical data centers. This provides flexibility in managing and replicating network configurations.

3.8 Public IP Addresses:

Users can add public IP addresses to resources within a VPC, allowing instances launched in the VPC to be reachable from the internet. This is particularly useful for services or applications that require internet connectivity.

The regional nature of VPCs in AWS enables users to strategically deploy and manage resources, providing the flexibility to scale and optimize performance based on geographical considerations and user requirements.

4 Virtual Private Cloud (VPC) for each AWS account

Amazon Web Services (AWS) automatically creates one default Virtual Private Cloud (VPC) for each AWS account. This default VPC comes preconfigured with several components to facilitate a basic cloud environment. Here are the details of what the default VPC includes:

4.1 Default Subnets:

The default VPC is set up with one subnet in each Availability Zone within the AWS region. These subnets are created with specific CIDR (Classless Inter-Domain Routing) blocks, and each subnet is associated with a particular Availability Zone.

4.2 Routing Tables:

Routing tables define the rules for routing network traffic within the VPC. The default VPC comes with preconfigured routing tables that allow communication between the subnets within the VPC. It also includes a default route to the internet through an Internet Gateway (IGW).

4.3 Security Groups:

Security group's act as virtual firewalls for instances in the VPC. The default VPC includes a default security group, and instances launched into the default VPC are automatically associated with this group. Users can modify the rules of the default security group or create additional security groups as needed.

4.4 Network Access Control List (NACL):

NACLs are stateless packet filters that control inbound and outbound traffic at the subnet level. The default VPC comes with a default NACL that allows all inbound and outbound traffic by default. Users can customize NACL rules to meet their specific security requirements.

4.5 User Options:

Users have the flexibility to choose how they want to use the default VPC. They can:

- Modify Default VPC: Users can make modifications to the default VPC, such as adding additional subnets, changing routing tables, or adjusting security group settings to meet their specific needs.
- Use Default VPC for Cloud Configurations: Users can leverage the default VPC for their cloud configurations and deploy resources directly into it.
- Build a New VPC: Alternatively, users can choose to create a new VPC from scratch. This allows them to have complete control over the VPC's configuration, including choosing CIDR blocks, subnets, routing tables, security groups, and more.

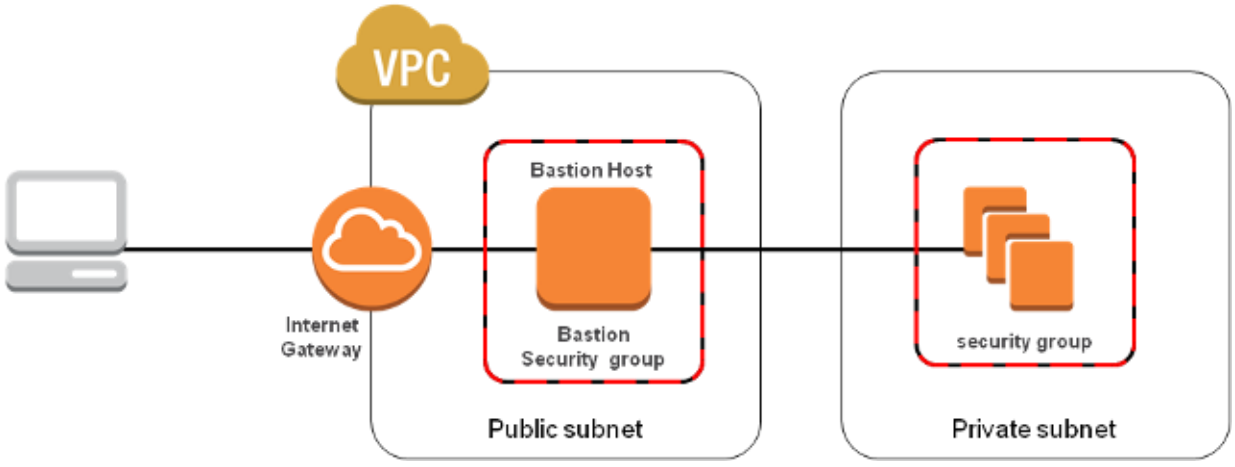
AWS provides users with the default VPC as a convenient starting point, but users have the flexibility to modify the default VPC or create entirely new VPCs to suit their specific requirements and preferences.

Subnet

1 Subnet

A subnet (Subnets, n.d.) is a range of IP addresses in your VPC. You need to provide at least two subnets to create a VPC connection. Each subnet must belong a different availability zone. You can attach AWS resources, such as Amazon EC2 instances and Amazon RDS DB instances, to subnets. You can create subnets to group instances together according to your security and operational needs.

For Amazon QuickSight to connect to your database, the network needs to route traffic to the data sources that you want to reach from one of the subnets used by the QuickSight network interface. QuickSight determines which subnet to route traffic through on the backend. If the availability zone that the subnet is attached to experiences an outage, QuickSight reroutes the traffic to one of the other subnets that are configured in the VPC connection. If the data sources are on different subnets make sure that, there is a route from the QuickSight network interface to your database instance. By default, each subnet in a VPC is associated with one main route table and can reach the other subnets.



2 Types of Subnets

The three common types of subnets within an Amazon Virtual Private Cloud (VPC): public subnets, private subnets, and VPN only subnets are described below:

2.1 Public Subnet:

- Routing: Has a direct route to an internet gateway.
- Internet Access: Resources in a public subnet can access the public internet.
- Use Case: Typically contains resources that need to be directly accessible from the internet, such as a web server hosting a website or an application.

2.2 Private Subnet:

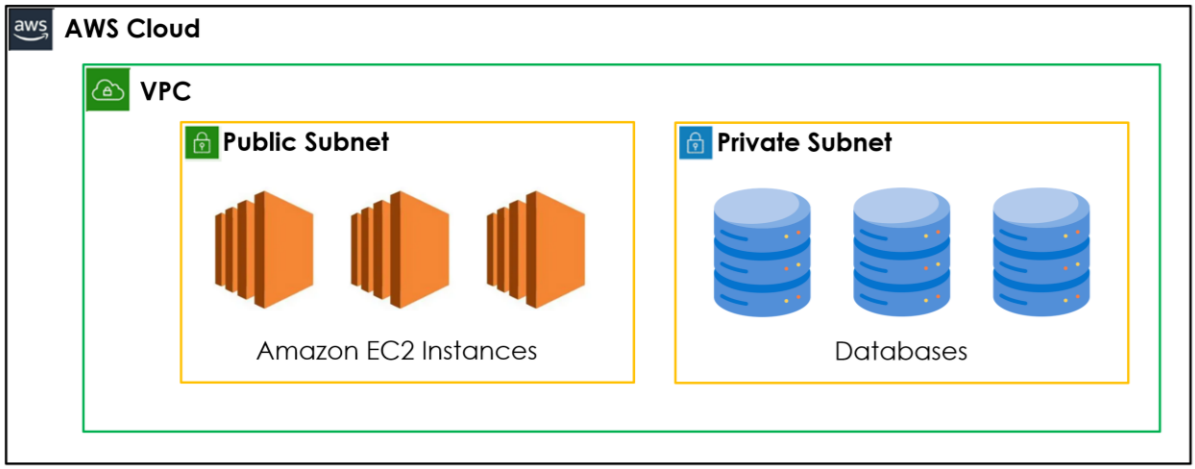
- Routing: Does not have a direct route to an internet gateway.
- Internet Access: Resources in a private subnet require a Network Address Translation (NAT) device to access the public internet. NAT allows resources in the private subnet to initiate outbound connections but prevents incoming traffic from directly reaching them.
- Use Case: Contains resources that should not be directly accessed from the internet, such as databases or application servers that store sensitive information.

2.3 VPN only Subnet:

- Routing: Has a route to a Site-to-Site VPN connection through a virtual private gateway.
- Internet Access: Does not have a direct route to an internet gateway.
- Use Case: Designed for scenarios where communication between the VPC and an on premises network is necessary. The VPN only subnet facilitates secure communication over a VPN connection.

It is worth noting that in a VPC, subnets within the same VPC can communicate with each other by default. This internal communication is useful for scenarios where different components of

an application, such as web servers in a public subnet and databases in a private subnet, need to interact.



3 Subnets Security

In Amazon Web Services (AWS), there are several features that users can leverage to enhance the security of resources within their Virtual Private Cloud (VPC). Two key components for managing network security are Security Groups and Network Access Control Lists (ACLs):

3.1 Security Groups:

- Functionality: Security Groups act as virtual firewalls for your Amazon EC2 instances and other resources in your VPC.
- Scope: Operate at the instance level, controlling inbound and outbound traffic for each instance.
- Rules: Users can define rules that allow traffic based on protocols, ports, and IP addresses. Security groups are stateful, meaning if you allow outbound traffic, the corresponding inbound response traffic is automatically allowed.
- Default Deny: All inbound traffic is denied by default, and users must explicitly allow the desired traffic.

3.2 Network Access Control Lists (Network ACLs):

- Functionality: Network ACLs operate at the subnet level, providing an additional layer of security for controlling traffic in and out of subnets.
- Scope: Apply to all resources within a subnet, affecting inbound and outbound traffic.
- Rules: Users can define rules to allow or deny traffic based on IP addresses, protocols, and ports. Network ACLs are stateless, meaning if you allow outbound traffic, you must also explicitly allow the corresponding inbound response traffic.
- Order of Rules: Network ACLs have numbered rules, and the rules are processed based on their order. The rule order is significant.

In most cases, users can achieve the necessary security configurations using Security Groups alone. Security Groups are more straightforward to configure and are typically the primary mechanism for controlling traffic at the instance level. However, Network ACLs offer an additional layer of security at the subnet level, allowing users to define broader rules that affect all resources in a subnet.

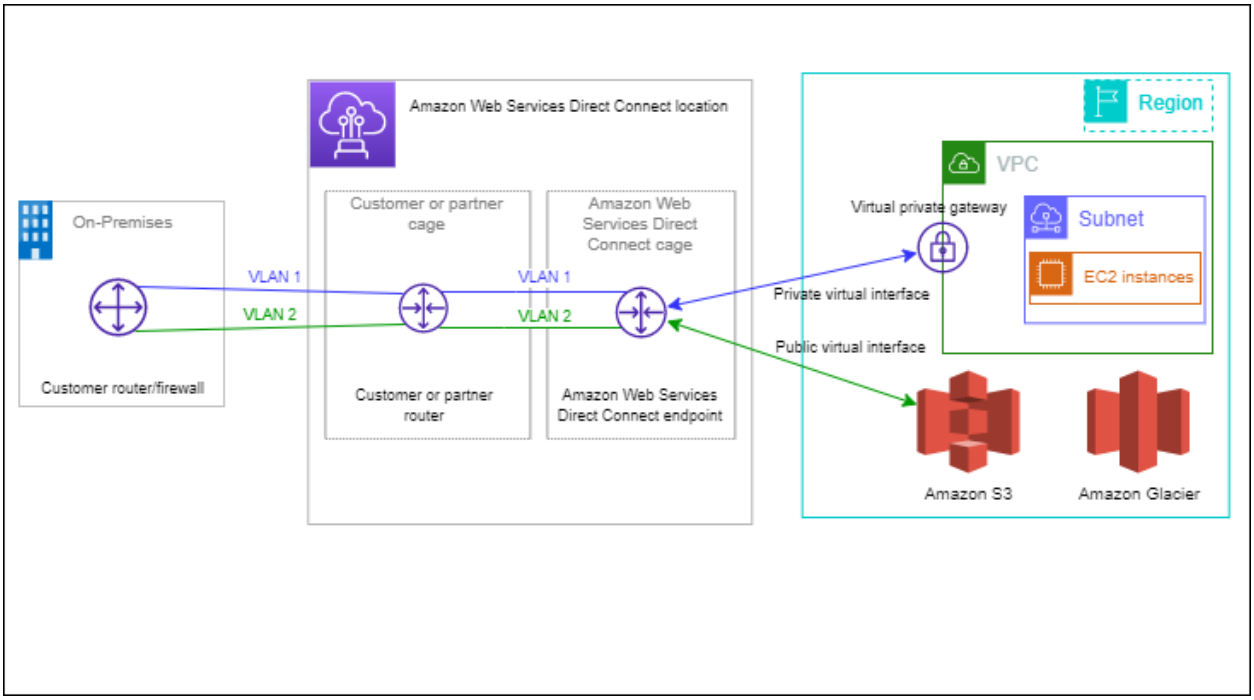
It is essential to design security groups and network ACLs carefully to balance security requirements with the necessary communication between resources in the VPC. Users often use a combination of both security groups and network ACLs to implement a comprehensive security strategy for their AWS environments.

AWS Direct Connect

1 AWS Direct Connect

AWS Direct Connect (Direct Connect, n.d.) Links your internal network to an AWS Direct Connect location over a standard Ethernet fiber optic cable. One end of the cable is connected to your router, the other to an AWS Direct Connect router. With this connection, you can create virtual interfaces directly to public AWS services (for example, to Amazon S3) or to Amazon VPC, bypassing internet service providers in your network path. An AWS Direct Connect location provides access to AWS in the Region with which it is associated. You can use a single connection in a public Region or AWS Gov-Cloud (US) to access public AWS services in all other public Regions.

The following diagram shows a high-level overview of how AWS Direct Connect interfaces with your network.



2 Features

AWS Direct Connect is a service that facilitates private network connections from an on premises data center, office, or colocation environment to AWS (Amazon Web Services) in a secure and scalable manner. Here are some key details about AWS Direct Connect:

2.1 Global Availability:

AWS Direct Connect is available at various locations worldwide, ensuring that users can establish connections close to their geographical locations for optimal performance and reduced latency.

2.2 Connection Speeds:

Users can choose from a range of connection speeds, starting at 50 Mbps and scaling up to 100 Gbps. This flexibility allows users to select the appropriate connection speed based on their specific application requirements.

2.3 Security Features:

AWS Direct Connect provides encryption options to enhance the security of communications between the user's data centers, branch offices, or colocation facilities.

MACsec (Media Access Control Security) is available for secure point-to-point encryption at select locations, offering native IEEE 802.1AE encryption for 10 Gbps and 100 Gbps connections.

IPsec encryption is available through AWS Site to Site VPN, providing an additional layer of security for connections.

2.4 SiteLink Feature:

The SiteLink feature enables users to create private, end-to-end network connections between their offices, data centers, and colocation facilities within their global network.

SiteLink can be easily configured or disabled using the AWS Management Console, AWS Command Line Interface (CLI), or APIs. This flexibility allows users to manage their network connections efficiently.

2.5 Multiple Deployment Options:

Dedicated Connections: Users can establish dedicated links to AWS using Ethernet ports with speeds ranging from 1 Gbps to 100 Gbps.

AWS Direct Connect Partners: These partners provide hosted connections with pre-established network links between themselves and AWS. Hosted connections are available in speeds ranging from 50 Mbps to 10 Gbps.

Overall, AWS Direct Connect offers a robust and versatile solution for users looking to establish secure, high-speed connections between their on premises environments and AWS services, with options for encryption, global network linking, and various deployment choices.

Access Control List (ACL)

1 Access Control List (ACL)

A Network Access Control List (ACL, n.d.) in Amazon Web Services (AWS) is a virtual firewall that operates at the subnet level, controlling both inbound and outbound traffic.

Here are the key details about AWS Network ACLs:

1.1 Default and Custom Network ACLs:

- Each AWS account comes with a default network ACL.
- When configuring a Virtual Private Cloud (VPC), users have the option to use the default network ACL or create custom network ACLs according to their specific requirements.

1.2 Default Network ACL:

- The default network ACL for a user's AWS account initially allows all inbound and outbound traffic.
- Users have the flexibility to modify the default network ACL by adding their own rules to control traffic more precisely.

1.3 Custom Network ACLs:

- Custom network ACLs, if chosen, start with a default rule that denies all inbound and outbound traffic.
- Users need to explicitly add rules to specify which traffic is allowed.

1.4 Rule Configuration:

- For both default and custom network ACLs, users define rules to control traffic.
- Rules can be configured based on IP addresses, protocols, ports, and other criteria.

1.5 Explicit Deny Rule:

- All network ACLs include an explicit deny rule by default.

- If a packet does not match any of the other rules in the list, the explicit deny rule ensures that the packet is denied.

1.6 Rule Evaluation Order:

- Rules in a network ACL are evaluated based on their order.
- The order of rules is crucial, as the first rule that matches the traffic is applied.

1.7 Inbound and Outbound Traffic Control:

- Network ACLs control both inbound and outbound traffic at the subnet level.
- Inbound rules are applied to traffic entering the subnet, and outbound rules are applied to traffic leaving the subnet.

1.8 Security Enhancement:

- Network ACLs provide an additional layer of security beyond Security Groups, operating at a different level of the networking stack.

2 Explanation with an Airport Analogy:

In the realm of computer networks, Access Control Lists (ACLs) act as virtual gatekeepers, regulating the flow of network traffic based on predefined rules.

Let us delve into an analogy to simplify the concept:

2.1 Airport Scenario:

Imagine you are at an airport, a bustling hub with travelers moving in and out, much like data packets in a network.

2.2 Travelers as Packets:

Travelers represent data packets. These packets, like people, need to go through a controlled entry and exit process.

2.3 Passport Control Officer as Network ACL:

The passport control officer symbolizes the network ACL. This officer (ACL) is responsible for examining the credentials (attributes) of each traveler (packet) entering or leaving the country (network).

2.4 Credential Check for Entry and Exit:

Just as travelers go through passport control both when entering and exiting the country, packets are subject to ACL rules for both inbound and outbound traffic.

2.5 Approved List Analogy:

If a traveler is on an approved list, meaning their credentials align with the predefined rules, they are allowed to proceed. Similarly, if a packet matches the criteria specified in the ACL rules, it is permitted to enter or exit the network.

2.6 Denied Entry for Unauthorized Travelers:

Conversely, if a traveler is not on the approved list or is explicitly on a list of banned travelers, the passport control officer will deny entry. In the networking context, packets that do not meet the ACL criteria are blocked from entering or leaving the network.

2.7 Granular Control:

ACLs provide granular control, allowing administrators to define specific conditions under which traffic is allowed or denied.

2.8 Security Analogy:

This process mirrors the security measures implemented by ACLs to safeguard a network by permitting or restricting the flow of data based on specified rules.

In summary, Access Control Lists act as digital passport control officers, scrutinizing the credentials (attributes) of data packets entering and exiting a network, determining their eligibility to traverse the network's borders based on predefined rules.

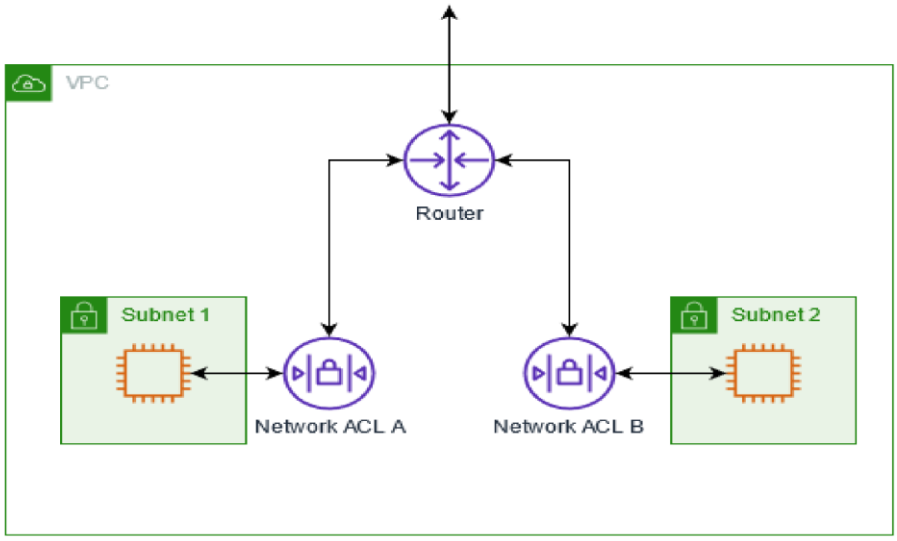
This analogy simplifies the abstract concept of ACLs, making it more relatable by drawing parallels with a familiar real world scenario.

3 Virtual Private Cloud (VPC) scenario

Let us break down the details of a Virtual Private Cloud (VPC) scenario with two subnets, each having its own Network Access Control List (ACL):

3.1 VPC Overview:

A VPC is a logically isolated section of the AWS Cloud where you can launch AWS resources in a virtual network.



3.2 Two Subnets:

The VPC is configured with two subnets, let us call them Subnet 1 and Subnet 2.

3.3 Network ACLs for Each Subnet:

Each subnet is associated with its own Network Access Control List (ACL).

Network ACLs act as virtual firewalls at the subnet level, controlling inbound and outbound traffic.

3.4 Traffic Entry into the VPC:

Traffic can enter the VPC through various sources such as a peered VPC, VPN connection, or the internet.

3.5 Router Handling Incoming Traffic:

Upon entering the VPC, the traffic is directed by the VPC's router to its destination based on the destination's subnet.

3.6 Network ACL A for Subnet 1:

Network ACL A is responsible for determining which traffic is allowed to enter Subnet 1 and which traffic is allowed to leave Subnet 1.

Ingress Rules for Subnet 1: Define which incoming traffic from external sources (peered VPC, VPN, or internet) is allowed into Subnet 1.

Egress Rules for Subnet 1: Specify which traffic from Subnet 1 is allowed to leave and reach external destinations.

3.7 Network ACL B for Subnet 2:

Similarly, Network ACL B is responsible for determining traffic rules for Subnet 2.

Ingress Rules for Subnet 2: Define which incoming traffic from external sources is allowed into Subnet 2.

Egress Rules for Subnet 2: Specify which traffic from Subnet 2 is allowed to leave and reach external destinations.

3.8 Traffic Flow Control:

Network ACLs provide granular control over traffic, allowing or denying based on defined rules.

Rules can be configured based on source and destination IP addresses, protocols, and ports.

3.9 Security Measures:

Network ACLs enhance the security of the VPC by controlling the flow of traffic at the subnet level.

In summary, in this VPC setup, each subnet has its own Network ACL that governs the ingress and egress traffic, allowing administrators to define specific rules for controlling the flow of data. These Network ACLs act as a security layer at the subnet level, ensuring that only authorized traffic is allowed into and out of each subnet.

Stateless packet filtering

1 Stateless packet filtering

Network Access Control Lists (ACLs) in the context of stateless packet filtering refer to a mechanism that examines packets independently without considering the state of the connection. Stateless packet filtering is a fundamental aspect of ACL functionality in many networking devices and firewall systems.

Stateless packet filtering involves inspecting each packet in isolation without considering the state of the connection. Each packet is evaluated independently based on predetermined rules.

1.1 ACLs and Stateless Filtering:

ACLs are commonly used for stateless packet filtering. They operate at the network layer (Layer 3) of the OSI model and make decisions about whether to permit or deny traffic based on criteria such as source and destination IP addresses, protocols, and port numbers.

1.2 Key Characteristics:

Stateless ACLs do not maintain any information about the state or context of a communication session.

Each packet is evaluated based on its individual characteristics, and decisions are made without reference to previous packets in the same connection.

1.3 Packet Inspection Criteria:

Stateless ACLs examine various attributes of a packet, including:

- Source and destination IP addresses.
- Protocol type (e.g., TCP, UDP, ICMP).
- Source and destination port numbers.
- Other relevant header information.

1.4 Rule Configuration:

- Administrators configure ACL rules specifying the criteria for allowing or denying packets.
- Rules are typically defined based on a combination of source and destination information, protocol types, and port numbers.

1.5 Packet Flow Decision:

- When a packet arrives at a network device (router, firewall, etc.) that is configured with a stateless ACL, the device examines the packet and compares it against the predefined rules.
- The ACL makes a decision to either permit or deny the packet based on the matching criteria.

1.6 Stateless Limitations:

- Stateless packet filtering is effective for simple access control scenarios but has limitations in handling complex network protocols and dynamic connections.
- It may not be suitable for tracking the state of connections, which is essential for more sophisticated security measures.

1.7 Use Cases:

- Stateless packet filtering is commonly employed in scenarios where basic packet filtering based on static criteria is sufficient.
- It is often used in conjunction with stateful inspection and other security measures for a layered approach to network security.

Understanding stateless packet filtering with ACLs is fundamental to implementing basic access control policies in network infrastructure.

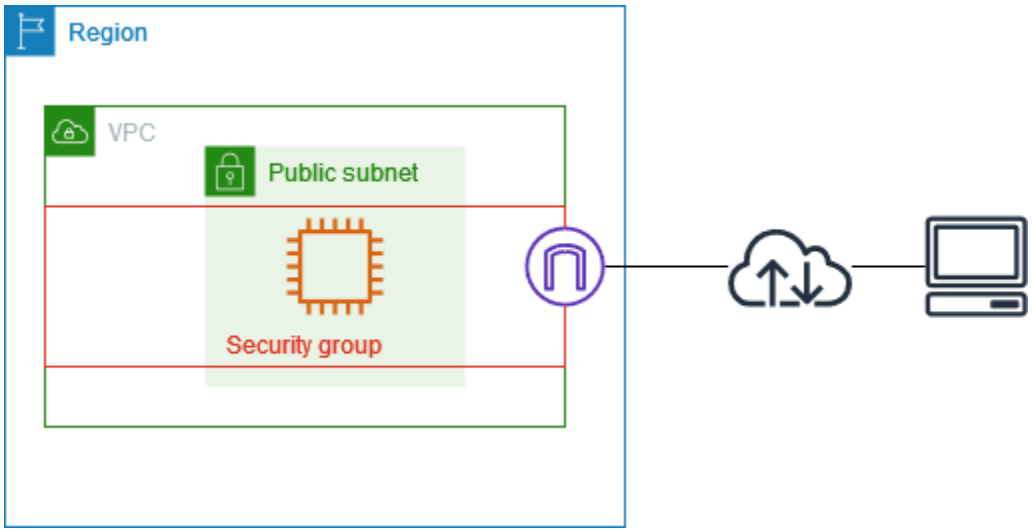
Security Groups and Stateful packet filtering

1 Security Groups

A security group (Security Groups, n.d.) Controls the traffic that is allowed to reach and leave the resources that it is associated with. For example, after you associate a security group with an EC2 instance, it controls the inbound and outbound traffic for the instance.

When you create a VPC, it comes with a default security group. You can create additional security groups for a VPC, each with their own inbound and outbound rules. You can specify the source, port range, and protocol for each inbound rule. You can specify the destination, port range, and protocol for each outbound rule.

The following diagram shows a VPC with a subnet, an internet gateway, and a security group. The subnet contains an EC2 instance. The security group is assigned to the instance. The security group acts as a virtual firewall. The only traffic that reaches the instance is the traffic allowed by the security group rules. For example, if the security group contains a rule that allows ICMP traffic to the instance from your network, then you could ping the instance from your computer. If the security group does not contain a rule that allows SSH traffic, then you could not connect to your instance using SSH.



2 Security group rules

The rules of a security group control the inbound traffic that is allowed to reach the resources that are associated with the security group. The rules also control the outbound traffic that's allowed to leave them.

You can add or remove rules for a security group (also referred to as authorizing or revoking inbound or outbound access). A rule applies either to inbound traffic (ingress) or outbound traffic (egress). You can grant access to a specific source or destination.

- You can specify allow rules, but not deny rules.
- When you first create a security group, it has no inbound rules. Therefore, no inbound traffic is allowed until you add inbound rules to the security group.
- When you first create a security group, it has an outbound rule that allows all outbound traffic from the resource. You can remove the rule and add outbound rules that allow specific outbound traffic only. If your security group has no outbound rules, no outbound traffic is allowed.
- When you associate multiple security groups with a resource, the rules from each security group are aggregated to form a single set of rules that are used to determine whether to allow access.
- When you add, update, or remove rules, your changes are automatically applied to all resources associated with the security group. The effect of some rule changes can depend on how the traffic is tracked. For more information, see Connection tracking in the Amazon EC2 User Guide for Linux Instances.
- When you create a security group rule, AWS assigns a unique ID to the rule. You can use the ID of a rule when you use the API or CLI to modify or delete the rule.

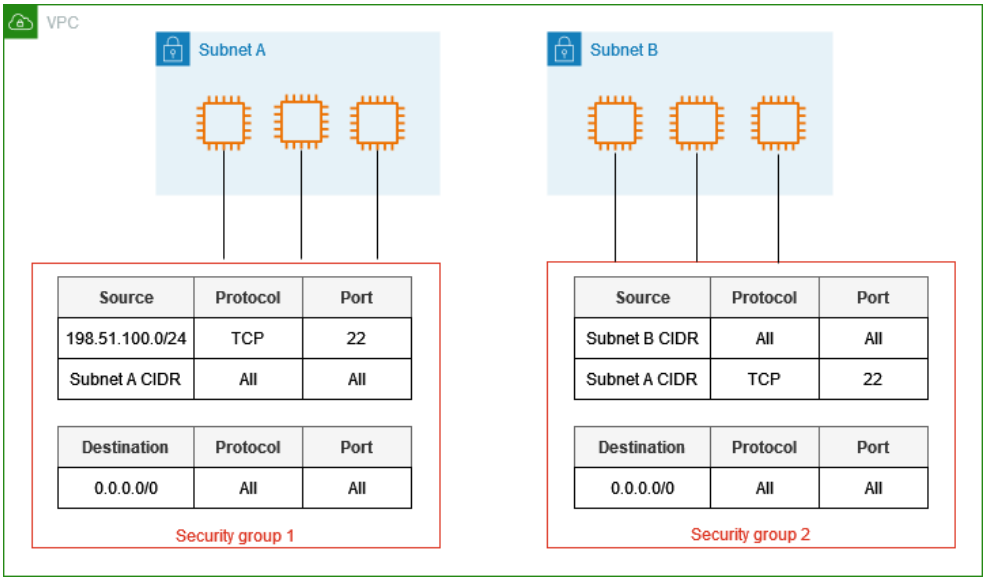
3 Security Group Examples

The following diagram shows a VPC with two security groups and two subnets. The instances in subnet a have the same connectivity requirements, so they are associated with security group,

the instances in subnet B have the same connectivity requirements, so they are associated with security group

The security group rules allow traffic as follows:

- The first inbound rule in security group 1 allows SSH traffic to the instances in subnet A from the specified address range (for example, a range in your own network).
- The second inbound rule in security group 1 allows the instances in subnet A to communicate with each other using any protocol and port.
- The first inbound rule in security group 2 allows the instances in subnet B to communicate with each other using any protocol and port.
- The second inbound rule in security group 2 allows the instances in subnet A to communicate with the instances in subnet B using SSH.
- Both security groups use the default outbound rule, which allows all traffic.



4 Stateful packet filtering

Security Groups in AWS perform stateful packet filtering, which means they keep track of the state of active connections. They remember decisions made for incoming packets, facilitating a higher level of security by maintaining context.

4.1 Example Scenario EC2 Instance Request:

- Use Case: Consider a scenario where an Amazon EC2 instance sends a request to the internet.
- Stateful Memory: The Security Group remembers the user's previous outbound request.

4.2 Handling Packet Responses:

- Packet Return: When a packet response corresponding to the user's request returns to the EC2 instance, the Security Group utilizes its stateful memory.
- Response Handling: The Security Group allows the response to proceed, regardless of inbound Security Group rules. This is because it recognizes the response as part of an established, permitted connection.

4.3 Comparison with Network ACLs:

- Custom Rule Configuration: Both Network Access Control Lists (ACLs) and Security Groups allow users to configure custom rules for traffic within their Virtual Private Cloud (VPC).
- Understanding Differences: Users must understand the differences between Network ACLs and Security Groups, as they serve different purposes and operate at different levels of the networking stack.

This summary effectively captures the essence of stateful packet filtering and the role of Security Groups in AWS. It emphasizes the importance of comprehending the distinctions between Security Groups and Network ACLs for effective network configuration and security in AWS. As users delve deeper into AWS security and networking, this understanding becomes crucial for designing robust and secure architectures.

AWS Global Networking and DNS

1 AWS Global Networking and DNS

The interaction between a user's AWS infrastructure and their customers, with a focus on AWS's global networking (AWS Global Networking, n.d.) Services, particularly Route 53:

1.1 Hosting on AWS:

A user is hosting their website on AWS infrastructure, utilizing services such as Amazon EC2 instances or Amazon S3 for static content.

1.2 User's Customers:

Customers, without necessarily being aware of the hosting infrastructure, access the website by entering its URL into their browsers.

1.3 AWS DNS Service Route 53:

- AWS provides a Domain Name Service (DNS) called Route 53.
- DNS Functionality: Route 53 acts as a translation service, akin to translating website names into Internet Protocol (IP) addresses that computers can understand.

1.4 Translation Process:

When a user's customer enters the website's address into their browser, Route 53 translates the human readable domain name (e.g., www.example.com) into the corresponding IP address of the hosting infrastructure.

1.5 Highly Available and Scalable:

Route 53 is designed to be highly available and scalable, ensuring reliable and efficient DNS resolution globally.

This global infrastructure helps reduce latency and enhances the overall performance of DNS queries.

1.6 Customer Browser Interaction:

As the customer's browser requests the IP address from Route 53, the DNS resolution occurs.

Once the IP address is obtained, the browser is directed to that specific address, effectively routing the customer's computer to the hosting infrastructure.

1.7 Transparent Hosting:

The customer, in their browser interaction, might not be aware that the website is hosted on AWS. AWS's infrastructure works seamlessly in the background to provide a fast and reliable user experience.

1.8 Behind the Scenes Routing:

Route 53's role is behind the scenes, ensuring that customers are efficiently directed to the correct IP address of the hosted website without any noticeable delay.

1.9 Global Accessibility:

Due to the global nature of AWS's infrastructure and Route 53, customers worldwide experience fast and reliable DNS resolution, contributing to a positive user experience.

In summary, AWS's Route 53 plays a crucial role in translating user friendly domain names into IP addresses, making websites hosted on AWS globally accessible and ensuring a smooth and reliable browsing experience for customers. The integration of scalable and highly available DNS services contributes to the overall efficiency and performance of the user's AWS hosted infrastructure.

2 Accelerating Content Delivery:

2.1 Amazon Cloud Front Service:

- Purpose: Amazon Cloud Front is a scalable and highly secure CDN service provided by AWS.
- Acceleration Focus: It focuses on accelerating the delivery of various web assets, including static and dynamic content, videos, and APIs.

2.2 Edge Locations:

- Key Concept: Edge locations, as discussed, are strategically placed data centers that serve content as close to customers as possible.
- Improving Latency: This geographic distribution of edge locations contributes to reducing latency and enhancing the overall performance of content delivery.

2.3 Content Delivery Network (CDN):

- Definition: A CDN is a network designed to deliver content, including web pages, images, videos, and other assets, to users based on their geographic location.
- Objective: The primary objective is to optimize the delivery speed and efficiency of web content.

2.4 Latency Improvement through CDN:

Geographic Proximity: CDNs advantage the geographic proximity of edge locations to users, ensuring that content is served from a location that is physically closer to the end-user.

2.5 Cloud Front's Role:

Deployment of Static Assets: Cloud Front allows users to deploy static assets such as images, scripts, and stylesheets across its network of edge locations.

2.6 Content Replication:

- In Multiple Regions: Users can replicate the same static assets in Cloud Front across multiple AWS Regions.

- Advantage: This strategy ensures that customers can access the same content, but from a location that is closest to them, improving response times and reducing latency.

2.7 Dynamic Content Delivery:

Not Limited to Static Content: While commonly associated with static content, Cloud Front can also accelerate the delivery of dynamic content and APIs.

2.8 Improved User Experience:

Reduced Latency: By deploying content in Cloud Front and different AWS Regions, users experience reduced latency and faster loading times for website assets.

2.9 Scalable and Secure Delivery:

- Scalability: Cloud Front automatically scales to handle varying levels of user traffic.
- Security Features: It provides security features such as DDoS protection and encryption for secure content delivery.

In summary, Amazon Cloud Front, as a globally distributed CDN, significantly contributes to optimizing the delivery of website assets to users. By strategically deploying content in Cloud Front and different AWS Regions, customers can access the same content from locations that are closer to them, thereby enhancing the overall speed and efficiency of content delivery.

3 Domain Name System

DNS, or the Domain Name System (DNS, n.d.), translates human readable domain names (for example, www.amazon.com) to machine-readable IP addresses (for example, 192.0.2.44).

4 DNS Basics

All computers on the Internet, from your smart phone or laptop to the servers that serve content for massive retail websites, find and communicate with one another by using numbers.

These numbers are known as IP addresses. When you open a web browser and go to a website, you do not have to remember and enter a long number. Instead, you can enter a domain name like example.com and still end up in the right place.

A DNS service such as Amazon Route 53 is a globally distributed service that translates human readable names like www.example.com into the numeric IP addresses like 192.0.2.1 that computers use to connect to each other. The Internet's DNS system works much like a phone book by managing the mapping between names and numbers. DNS servers translate requests for names into IP addresses, controlling which server an end user will reach when they type a domain name into their web browser. These requests are called queries.

5 DNS Resolution Overview:

5.1 Web Address Access:

Customer Interaction: When customers enter a web address into their browser, they initiate the process of accessing a website.

5.2 Role of DNS:

Critical Function: DNS plays a crucial role in this process by facilitating the translation of human readable domain names into IP addresses that computers can understand.

5.3 DNS Comparison to Phone Book:

- Analogy: DNS can be likened to the phone book of the internet.
- Translation Function: Similar to how a phone book translates a person's name to their phone number, DNS translates a domain name to an IP address.

5.4 DNS Resolution Definition:

Process Explanation: DNS resolution is the process of converting a domain name (e.g., www.example.com) into the corresponding IP address.

5.5 Phone Book Analogy Continued:

Customer DNS Resolver: In the analogy, the customer's DNS resolver can be considered as the individual looking up information in the phone book.

5.6 Communication with DNS Server:

Communication Steps: The customer's DNS resolver communicates with a DNS server belonging to the company hosting the website.

5.7 Phone Book Lookup Parallel:

Lookup Similarity: The DNS resolver looking up the IP address is akin to someone looking up a phone number in the phone book.

5.8 IP Address Retrieval:

Result of Communication: The DNS server responds with the IP address associated with the provided domain name.

5.9 IP Address Usage:

Routing Significance: the browser to route the customer’s request to the correct web server hosting the website then uses the obtained IP address.

5.10 Efficient Content Retrieval:

Optimization Aspect: DNS resolution plays a key role in optimizing content retrieval by ensuring accurate and efficient routing.

In summary, the DNS resolution process is essential for customers to access websites by translating human readable domain names into machine-readable IP addresses. The analogy of DNS as the phone book of the internet provides a simple yet effective way to understand its role in facilitating communication between users and web servers.

6 Example: DNS Resolution for example.com

6.1 Customer Action:

A customer wants to visit the website "example.com."

6.2 Browser Request:

The customer enters "example.com" into their web browser's address bar.

6.3 Customer DNS Resolver:

The browser's request is sent to the customer's DNS resolver.

6.4 Resolver Inquiry:

The customer DNS resolver, which is typically provided by their internet service provider (ISP) or another DNS service, is responsible for translating the domain name to an IP address.

6.5 Request to Company DNS Server:

The customer DNS resolver communicates with the company's DNS server, asking for the IP address associated with "example.com."

6.6 Company DNS Server Lookup:

The company's DNS server looks up the IP address associated with "example.com" in its records.

6.7 IP Address Retrieval:

The company's DNS server responds to the customer DNS resolver with the IP address corresponding to "example.com," let us say X.X.X.X.

6.8 Customer Browser Routing:

Armed with the IP address, the customer's browser can now route the request to the correct location on the internet to retrieve the content from the "example.com" website.

6.9 Efficient Content Retrieval:

The IP address obtained through DNS resolution ensures efficient and accurate routing, enabling the browser to retrieve the website's content.

In summary, this example highlights the interaction between the customer's DNS resolver and the company's DNS server, illustrating how the DNS resolution process translates a human readable domain name ("example.com") into a machine-readable IP address (X.X.X.X), allowing the customer's browser to access the desired website.

Amazon Route 53

1 Amazon Route 53

Amazon Route 53 (Route 53, n.d.) Is a highly available and scalable Domain Name System (DNS) web service. You can use Route 53 to perform three main functions in any combination: domain registration, DNS routing, and health checking.

If you choose to use Route 53 for all three functions, be sure to follow the order below:

1.1 Register domain names

Your website needs a name, such as example.com. Route 53 lets you register a name for your website or web application, known as a domain name.

1.2 Route internet traffic to the resources for your domain

When a user opens a web browser and enters your domain name (example.com) or subdomain name (acme.example.com) in the address bar, Route 53 helps connect the browser with your website or web application.

1.3 Check the health of your resources

Route 53 sends automated requests over the internet to a resource, such as a web server, to verify that it is reachable, available, and functional. You also can choose to receive notifications when a resource becomes unavailable and choose to route internet traffic away from unhealthy resources.

2 How domain registration works

If you want to create a website or a web application, you start by registering the name of your website, known as a domain name. Your domain name is the name, such as example.com, that your users enter in a browser to display your website.

Here is an overview of how you register a domain name with Amazon Route 53:

1) You choose a domain name and confirm that it is available, meaning that no one else has registered the domain name that you want.

   If the domain name you want is already in use, you can try other names or try changing only the top-level domain, such as .com, to another top-level domain, such as .ninja or .hockey. For a list of the top-level domains that Route 53 supports, see Domains that you can register with Amazon Route 53.
2) You register the domain name with Route 53. When you register a domain, you provide names and contact information for the domain owner and other contacts. When you register a domain with Route 53, the service automatically makes itself the DNS service for the domain by doing the following
   - Creates a hosted zone that has the same name as your domain.
   - Assigns a set of four name servers to the hosted zone. When someone uses a browser to access your website, such as www.example.com, these name servers tell the browser where to find your resources, such as a web server or an Amazon S3 bucket. (Amazon S3 is object storage for storing and retrieving any amount of data from anywhere on the web. A bucket is a container for objects that you store in S3.)
   - Gets the name servers from the hosted zone and adds them to the domain.
3) At the end of the registration process, we send your information to the registrar for the domain. The domain registrar is either Amazon Registrar, Inc. or our registrar associate, Gandi.
4) The registrar sends your information to the registry for the domain. A registry is a company that sells domain registrations for one or more top-level domains, such as .com.
5) The registry stores the information about your domain in their own database and stores some of the information in the public WHOIS database.

If you already registered a domain name with another registrar, you can choose to transfer the domain registration to Route 53. This is not required to use other Route 53 features. For more information, see Transferring registration for a domain to Amazon Route 53.

Amazon API Gateway

1 Amazon API Gateway

Amazon API Gateway (API Gateway, n.d.) Is an AWS service for creating, publishing, maintaining, monitoring, and securing REST, HTTP, and Web Socket APIs at any scale. API developers can create APIs that access AWS or other web services, as well as data stored in the AWS Cloud. As an API Gateway API developer, you can create APIs for use in your own client applications. Alternatively, you can make your APIs available to third-party app developers.

API Gateway creates RESTful APIs that:

- Are HTTP-based.
- Enable stateless client-server communication.
- Implement standard HTTP methods such as GET, POST, PUT, PATCH, and DELETE.

API Gateway creates Web Socket APIs that:

- Adhere to the Web Socket protocol, which enables stateful, full-duplex communication between client and server.
- Route incoming messages based on message content.

2 Key Benefits

2.1 Seamless Integration:

- Direct Connectivity: Provides direct connectivity to various AWS services such as Lambda for server less functions, EC2 for virtual servers, and Dynamo DB for NoSQL database services.
- Effortless Interaction: Enables seamless integration and interaction with diverse AWS resources within the architecture.

2.2 Robust Traffic Handling:

- Throttling offers throttling mechanisms to control the rate at which API Gateway accepts requests, preventing abuse and ensuring optimal usage.
- Caching Implements caching for frequently requested content, reducing the load on backend services and improving response times.
- Burst Management: Handles bursts of traffic efficiently, ensuring the system remains responsive and performs optimally even during peak loads.

2.3 Enhanced Security:

- Authentication Mechanisms: Supports multiple authentication mechanisms to secure API access, including Identity and Access Management (IAM) roles for AWS security management.
- Authorization Controls: Provides authorization controls through various mechanisms, such as Lambda authorizers and Amazon Cognito, to manage access to resources based on defined policies.

2.4 Cost-Efficient:

- Pay-as-You-Go Model: Adopts a pay-as-you-go pricing model where users are billed based on the number of API calls and data transfers.
- No Upfront Costs: Minimizes upfront costs, allowing users to scale resources based on demand and pay only for the actual usage.

These benefits collectively contribute to building a scalable, efficient, and secure architecture for API-based applications. Whether you are designing server less applications using Lambda, managing traffic efficiently, enhancing security through IAM roles, or aiming for cost optimization, the mentioned features align with best practices for API Gateway services, especially within the AWS ecosystem.

3 Features of API Gateway

Amazon API Gateway offers features such as the following:

- Support for stateful (Web Socket) and stateless (HTTP and REST) APIs.
- Powerful, flexible authentication mechanisms, such as AWS Identity and Access Management policies, Lambda authorizer functions, and Amazon Cognito user pools.
- Canary release deployments for safely rolling out changes.
- Cloud Trail logging and monitoring of API usage and API changes.
- Cloud Watch access logging and execution logging, including the ability to set alarms.
- Ability to use AWS CloudFormation templates to enable API creation.
- Support for custom domain names.
- Integration with AWS WAF for protecting your APIs against common web exploits.
- Integration with AWS X-Ray for understanding and triaging performance latencies.

Amazon Cloud Front

1 Amazon Cloud Front

Amazon Cloud Front (CloudFront, n.d.) Is a content delivery network (CDN) service built for high performance, security, and developer convenience.

- If the content is already in the edge location with the lowest latency, Cloud Front delivers it immediately.
- If the content is not in that edge location, Cloud Front retrieves it from an origin that you've defined—such as an Amazon S3 bucket, a Media Package channel, or an HTTP server (for example, a web server) that you have identified as the source for the definitive version of your content.

2 Key Benefits

The key benefits of Amazon Cloud Front in discuss in detail below:

2.1 Global Reach:

- Content Caching: Cloud Front caches and delivers content from the nearest edge location to end-users, minimizing latency and providing optimal performance.
- Geographic Distribution: With a vast network of edge locations worldwide, Cloud Front ensures global reach, serving content to users from locations close to them.

2.2 Integrated AWS Experience:

- Seamless Connection: Integrates seamlessly with other AWS services such as Amazon S3, EC2, and Elastic Load Balancing.
- Ease of Use: Users can easily configure and manage Cloud Front distributions through the AWS Management Console or programmatically via the AWS CLI.

2.3 Enhanced Performance:

- Low Latency: Cloud Front is designed for low-latency content delivery, ensuring faster loading times for websites and applications.
- High-Speed Delivery: Utilizes a global network infrastructure to deliver content at high speeds, enhancing overall user experiences.

2.4 Built for Scale:

- Traffic Handling: Capable of handling sudden traffic surges and high request volumes efficiently.
- Scalability: Scales resources dynamically to accommodate varying levels of user traffic, making it suitable for both small and large-scale applications.

2.5 Advanced Security:

- AWS WAF Integration: Integrates with AWS Web Application Firewall (WAF) for protection against common web exploits and attacks.
- AWS Shield Integration: Works seamlessly with AWS Shield, a managed Distributed Denial of Service (DDoS) protection service.
- HTTPS Support: Supports secure communication by allowing users to configure SSL/TLS encryption (HTTPS) for content delivery.

In summary, Amazon Cloud Front is a powerful Content Delivery Network (CDN) service that brings several benefits to the table, including global reach, seamless integration with AWS services, enhanced performance, scalability, and advanced security features. Whether you are aiming for improved user experiences, efficient content delivery, or robust security measures, Cloud Front is designed to meet these requirements in a comprehensive and integrated manner within the AWS ecosystem.

3 Core Features

3.1 Content Customization with Lambda@Edge:

- Lambda Integration: Amazon Cloud Front integrates with Lambda@Edge, allowing users to run server less functions in response to Cloud Front events.
- Tailored Content Delivery: Lambda@Edge enables customized content delivery and response manipulations based on specific requirements.

3.2 Cost-Efficiency:

- Pay-as-You-Go Model: Adopts a pay-as-you-go pricing model where users are charged based on actual usage, making it cost-efficient.
- No Upfront Fees: Users are not required to pay any upfront fees, contributing to a flexible and scalable cost structure.
- Customizable Price Classes: Allows users to choose different price classes to control the number of edge locations and associated costs.

3.3 Streaming Support:

- Live and On-Demand Streaming: Cloud Front supports both live and on-demand video streaming, providing a scalable solution for delivering streaming content globally.
- Low Latency: Ensures low-latency streaming experiences for end-users.

3.4 Dynamic Invalidation:

- Efficient Cache Management: Offers dynamic content invalidation, allowing users to efficiently manage and update cached content.
- Real-Time Updates: Enables real-time updates to the content cache, ensuring that the latest version of the content is delivered to users.

3.5 Robust Access Controls:

- Signed URLs and Cookies: Provides robust access controls through features like signed URLs and cookies.

- Granular Control: Users can enforce granular control over content access, ensuring that only authorized users can access specific content.

4 Additional Considerations:

- Global Edge Locations: Leverages a vast network of global edge locations for content delivery, reducing latency and improving performance.
- Secure Communication: Supports HTTPS (SSL/TLS) for secure communication between end-users and Cloud Front, ensuring data integrity and privacy.
- Real-Time Analytics: Integrates with AWS Cloud Watch for real-time monitoring and analytics, providing insights into usage patterns and performance metrics.

In summary, Amazon Cloud Front's core features encompass a range of capabilities, including content customization, cost-efficiency, streaming support, dynamic cache management, and robust access controls. These features collectively contribute to a highly scalable, performant, and secure content delivery solution within the AWS ecosystem.

AWS Global Accelerator

1 AWS Global Accelerator

AWS Global Accelerator (Global Accelerator, n.d.) Is a service in which you create accelerators to improve the performance of your applications for local and global users.

Depending on the type of accelerator you choose, you can gain additional benefits:

- With a standard accelerator, you can improve availability of your internet applications that are used by a global audience. With a standard accelerator, Global Accelerator directs traffic over the AWS global network to endpoints in the nearest Region to the client.
- With a custom routing accelerator, you can map one or more users to a specific destination among many destinations.

Global Accelerator is a global service that supports endpoints in multiple AWS Regions. To determine if Global Accelerator or other services are currently supported in a specific AWS Region, see the AWS Regional Services List.

By default, Global Accelerator provides you with static IP addresses that you associate with your accelerator. The static IP addresses are any cast from the AWS edge network. For IPv4, Global Accelerator provides two static IPv4 addresses. For dual-stack, Global Accelerator provides a total of four addresses: two static IPv4 addresses and two static IPv6 addresses. For IPv4, instead of using the addresses that Global Accelerator provides, you can configure these entry points to be IPv4 addresses from your own IP address ranges that you bring to Global Accelerator (BYOIP).

2 Key Benefits

2.1 Consistent Performance:

- Global Network Infrastructure: Utilizes AWS's global network infrastructure to minimize the number of hops between users and applications.
- Optimized Data Transfer Speeds: Improves data transfer speeds by ensuring efficient and optimized routing across the global network.

2.2 Enhanced Availability:

- Static any cast IP Addresses: Utilizes static any cast IP addresses that provide consistent and optimal routing based on the health and geographical proximity of endpoints.
- Global Availability: Ensures that users are directed to the nearest healthy endpoint, contributing to enhanced availability and reduced latency.

2.3 Fault Tolerance:

- Real-Time Rerouting: In the event of regional failures or disruptions, traffic is rerouted in real-time to healthy endpoints.
- Continuous User Experiences: Ensures fault tolerance, preventing service interruptions and providing users with uninterrupted and reliable experiences.

2.4 DDoS Protection:

- Automatic Integration with AWS Shield Standard: AWS Global Accelerator integrates automatically with AWS Shield Standard, which offers robust protection against Distributed Denial of Service (DDoS) attacks.
- Enhanced Security: Provides an additional layer of security to defend against malicious traffic and ensure the availability of applications.

In summary, AWS Global Accelerator provides consistent performance, enhanced availability, fault tolerance, and DDoS protection for global applications. By leveraging AWS's global network infrastructure and any cast IP addresses, it ensures that users are directed to the

optimal endpoint, offering a reliable and efficient solution for delivering applications worldwide.

3 Core Features

3.1 Traffic Dials:

- Definition: Traffic Dials allow users to manage the volume of traffic directed to specific endpoint groups.
- Usage: Users can dynamically adjust the traffic volume to different endpoint groups based on factors such as application version, user segment, or other custom criteria.
- Flexibility: Provides flexibility in controlling and optimizing traffic distribution for specific use cases or scenarios.

3.2 Zone-Independent Mapping:

- Definition: Zone-Independent Mapping enables intelligent routing irrespective of specific availability zones or regions.
- Optimized Routing: Allows AWS Global Accelerator to route traffic to the healthiest endpoints without being bound to specific zones, optimizing for availability and performance.
- Enhanced Resilience: Reduces dependency on individual zones, enhancing application resilience and availability.

3.3 Health Checks:

- Definition: Health Checks are continuous monitoring mechanisms that assess the health of an application's endpoints.
- Endpoint Monitoring: Regularly checks the health of endpoints, ensuring that only healthy instances receive traffic.
- Avoiding Unhealthy Instances: Prevents routing traffic to instances that are experiencing issues, maintaining a high level of application reliability.

- Customizable Health Checks: Users can customize health check settings to align with specific application requirements and thresholds.

In summary, AWS Global Accelerator's core features, including Traffic Dials, Zone-Independent Mapping, and Health Checks, provide users with granular control over traffic management, resilient routing, and continuous monitoring of endpoint health. These features collectively contribute to a reliable, high-performance solution for globally distributed applications.

4 Integrations

4.1 Multiple AWS Services:

4.1.1 Application Load Balancers (ALB):

- Integration: AWS Global Accelerator seamlessly integrates with Application Load Balancers, enabling efficient load balancing for HTTP and HTTPS traffic.
- Use Case: ALBs are often used for routing traffic to different services or applications based on content-based routing rules.

4.1.2 Network Load Balancers (NLB):

- Integration: Supports integration with Network Load Balancers, providing high-performance, low-latency load balancing for TCP and UDP traffic.
- Use Case: NLBs are suitable for scenarios that require ultra-low latency and high-throughput load balancing.

4.1.3 EC2 Instances:

- Integration: Directly integrates with Amazon EC2 instances, allowing users to route traffic to specific instances based on their health and other criteria.
- Use Case: Ideal for scenarios where users need to distribute traffic across a fleet of EC2 instances.

4.1.4 Elastic IPs (EIPs):

- Integration: Supports Elastic IP addresses, providing users with the flexibility to route traffic to specific Elastic IP addresses associated with resources.
- Use Case: Useful when applications or services need a static IP address for outbound communication.

These integrations highlight the flexibility and versatility of AWS Global Accelerator, allowing users to route traffic to a variety of AWS services based on their specific requirements. Whether it is load balancers, EC2 instances, or Elastic IPs, Global Accelerator ensures efficient and optimized traffic distribution across various AWS resources.

AWS VPN

1 AWS VPN

AWS VPN (VPN, n.d.) Enables secure encrypted connections between on-premises networks and the AWS global network, comprising two main offerings: Site-to-Site VPN and Client VPN.

2 Primary Services:

AWS VPN is comprised of two services: AWS Site-to-Site VPN and AWS Client VPN. AWS Site-to-Site VPN enables you to securely connect your on-premises network or branch office site to your Amazon Virtual Private Cloud (Amazon VPC). AWS Client VPN enables you to securely connect users to AWS or on-premises networks.

2.1 Site-to-Site VPN:

AWS Client VPN is a fully managed remote access VPN solution used by your remote workforce to securely access resources within both AWS and your on-premises network. Fully elastic, it automatically scales up, or down, based on demand. When migrating applications to AWS, your users access them the same way before, during, and after the move. AWS Client VPN, including the software client, supports the Open VPN protocol.

2.2 Client VPN:

AWS Site-to-Site VPN is a fully managed service that creates a secure connection between your data center or branch office and your AWS resources using IP Security (IPsec) tunnels. When using Site-to-Site VPN, you can connect to both your Amazon Virtual Private Clouds (VPC) as well as AWS Transit Gateway, and two tunnels per connection are used for increased redundancy.

For globally distributed applications, the Accelerated Site-to-Site VPN option provides even greater performance by working with AWS Global Accelerator to intelligently route your traffic to the nearest AWS network endpoint with the best performance.

3 Core Features

The core features of AWS VPN (Virtual Private Network) are discussed below:

3.1 Security:

- Industry-Standard Encryption:
- Encryption Protocols: Employs industry-standard encryption protocols, ensuring secure communication between on-premises networks and AWS resources.
- Data Protection: Ensures the confidentiality and integrity of data during transmission over the VPN connection.

3.2 High Availability:

- Redundancy and Failover:
- Built-In Redundancy: Includes built-in redundancy mechanisms to enhance availability.
- Failover Support: Facilitates failover between VPN connections or VPN gateways, ensuring continuous and uninterrupted connectivity.

3.3 Scalability:

- Multiple VPN Connections:
- Diverse VPC Connectivity: Allows the establishment of multiple VPN connections to different Virtual Private Clouds (VPCs).
- Scalable Architecture: Adapts to varying networking requirements, supporting scalability in terms of VPC connections.

3.4 Monitoring:

- Seamless Integration with Cloud Watch and Cloud Trail:

- Cloud Watch Integration: Integrates seamlessly with AWS Cloud Watch for real-time monitoring of VPN connection metrics.
- Cloud Trail Integration: Logs and monitors VPN-related events and activities through AWS Cloud Trail for auditing and compliance purposes.

3.5 Elastic IP:

- Consistent Public IP for AWS End in Site-to-Site VPN:
- Public IP Stability: Provides an Elastic IP address for the AWS end of the Site-to-Site VPN connection.
- Stable Identifier: Ensures a consistent and stable public IP for the AWS end of the VPN, facilitating configuration and management.

3.6 Device Flexibility:

- Compatibility with Numerous On-Premises Devices:
- Broad Device Support: Compatible with a variety of on-premises devices, routers, and VPN appliances.
- Vendor Agnostic: Allows flexibility in choosing on-premises networking equipment, supporting a vendor-agnostic approach.

These core features collectively contribute to the robustness, security, and flexibility of AWS VPN services. Whether focusing on encryption standards, ensuring high availability, supporting multiple connections, monitoring network health, providing stable IP addresses, or accommodating diverse on-premises devices, AWS VPN is designed to meet various connectivity needs while maintaining a strong emphasis on security and reliability.

4 Benefits

4.1 Peace of Mind:

- Encrypted Connections:
- Data Privacy and Security: AWS VPN ensures the privacy and security of data through encrypted connections, providing peace of mind regarding the confidentiality and integrity of transmitted information.

4.2 Consistent Uptime:

- High Availability:
- Redundancy and Failover: The built-in high availability features minimize potential downtimes by incorporating redundancy and failover mechanisms, ensuring a consistent and reliable network.

4.3 Future Proofing:

- Scalability:
- Adaptable to Growth: AWS VPN is easily scalable, allowing businesses to expand and adapt to changing demands seamlessly.
- Future Proofing: Ensures that the network infrastructure can evolve alongside the business, providing a future-proof solution.

4.4 Enhanced Oversight:

- Comprehensive Monitoring:
- Cloud Watch and Cloud Trail Integration: Integration with AWS Cloud Watch and Cloud Trail offers comprehensive monitoring, providing a clear view of the network's health, activities, and events.
- Real-Time Insights: Enables real-time insights into VPN connection metrics and logs, enhancing oversight and facilitating proactive management.

4.5 Flexibility in Access:

- Secure Access from Diverse Locations and Devices:
- Geographical Flexibility: Users can securely access AWS resources from diverse geographical locations.
- Device Agnosticism: Supports access from various devices, offering flexibility in how users connect to and interact with AWS resources securely.

These benefits collectively contribute to creating a secure, reliable, and flexible network environment with AWS VPN. Whether it's ensuring the confidentiality of data, maintaining consistent uptime, accommodating future growth, enabling comprehensive monitoring, or providing flexible access options, AWS VPN addresses critical aspects for businesses seeking a robust and adaptable networking solution.

References

- *ACL*. (n.d.). Retrieved from www.fortinet.com: https://www.fortinet.com/resources/cyberglossary/network-access-control-list
- *Advanced Networking*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/AmazonVPC/latest/UserGuide/what-is-amazon-vpc.html
- *API Gateway*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/apigateway/latest/developerguide/welcome.html
- *AWS Global Networking*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/network-manager/latest/tgwnm/what-are-global-networks.html
- *CloudFront*. (n.d.). Retrieved from aws.amazon.com: https://aws.amazon.com/cloudfront/
- *Connectivity*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/vpc/
- *Direct Connect*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html
- *DNS*. (n.d.). Retrieved from aws.amazon.com: https://aws.amazon.com/route53/what-is-dns/
- *Global Accelerator*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/global-accelerator/latest/dg/what-is-global-accelerator.html
- *Load Balancing*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/elasticloadbalancing/
- *Route 53*. (n.d.). Retrieved from awws.amazon.com: https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/Welcome.html

- *Security Groups*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security-groups.html
- *Subnets*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/quicksight/latest/user/vpc-subnets.html
- *VPC*. (n.d.). Retrieved from aws.amazon.com: https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html
- *VPN*. (n.d.). Retrieved from aws.amazon.com: https://aws.amazon.com/vpn/