**Networking Introduction.**

**Computer Networks** are Collections of Computing Devices logically connected to Communicate & Share Resources. They’re made up of the following:

* Nodes are Computers/Routers/Switches/Modems/Printers connected through links (i.e. - Cables) that follow rules to send/receive data.
* Hosts are Nodes that provide a Unique Function (i.e. - *Servers*).

**Switches** connect Multiple Devices on a Network (only using MAC), whereas **Routers** connect Multiple Switches on a Network and **Modems** connect your house to the Internet through the ISP with a Coaxial Cable.

**Data** are bits & bytes which equal 0/1 (i.e. – *Characters, Text, Numbers, Media*). The flow of data is demonstrated through the following model:

The **Open Systems Interconnection (OSI)** Model defines the standard on how Computers share Information over a Network and follows these steps.:

1. Source Computer sends Data to the Target Computer
2. The Application Layer (7) provides Interface enabling Applications to access Network Services.
3. The Presentation Layer (6) then ensures data is in Usable Format & encrypts/decrypts.
4. The Session Layer (5) maintains distinction between Separate Application Data using AAA protection.
5. The Transport Layer (4) establishes Logical Connection between Source & Destination, specifying which Transmission Protocol to use with firewalls.
6. The Network Layer (3) decides which Physical Path the Data takes using route tables.
7. The Data Link Layer (2) defines the Format of the Data on the Network using MAC Address Filtering.
8. The Physical Layer (1) Transmits the Raw Bitstream over the Physical Network.
9. Once the Target Computer receives the Data Package, it does this series of events in reverse to unpack it.

**Clients** are a Computer Hardware Device/Software that access Data/Services managed by a Server over a Network.

**Servers** provide responses to client requests with the requested content (i.e. – Web/Database/File/Mail/Print Servers)

The **Network Interface Card (NIC)** connects A Computer to a Network using a cable connected to a hub/switch. Each have their own Unique MAC Address.

**Network Cables** are used to physically connect network cables, there are three types:

* Fibre-Optic: Most common, Transmits Light instead of Electricity.
* Coaxial: Replaced by the above, mainly used to connect Cable TVs to ISPs.
* Twisted-Pair: Most Common Type, also known as Ethernet.

**Network Concepts**

**Local-Area Networks (LAN)** connect Devices in a Limited Geographical Area (i.e. – *Floor/Building/Campus*).

**Wide-Area Networks (WAN)** connect Devices in a Large Geographical Area (i.e. – *Cities/Countries/LANs*)

**Network Topologies** are Patterns/Diagrams that show how Nodes connect to each other.

1. Physical Topology: The Physical Layout of Wires on the Network.
2. Logical Topology: How Data Moves through the Network.

|  |  |
| --- | --- |
| **Topology & Management Forms** | |
| **Bus** | All Devices Connected along a Single Cable, Only one PC sends signals at time. |
| **Star** | All Nodes connected to Central Switch using wired connection. |
| **Mesh** | Nodes are Interconnected, like p2p. |
| **Hybrid** | Combines two or more of the above. |
| **Client-Server** | Data Management & Application Hosting Centralized at the Server & Distributed to Clients. |
| **Peer-to-Peer** | Each Node has Individual Data & Applications, responsible for their own Management & Security. |

|  |  |
| --- | --- |
| **Topology** | **AWS Service/Capability** |
| **Isolated Network** | **Amazon VPC** |
| **Network Segment** | **Security Group & Network Access Control List (NACL)** |
| **Server** | **Elastic Compute Cloud (EC2) Instance** |

**Internet Protocol**

**IP** is a Network Protocol which Establishes Rules for Relaying/Routing Data, with Port Numbers to Identify Endpoints & Subnetting to Subdivide a Network.

**IP Addresses** Uniquely Identify Network Devices and can be used for Location Addressing.

* Private IP Addresses are only accessed within a Logically Isolated Private Network.
* Public IP Addresses can be Publicly Accessed over the Internet by Anyone.

**IPv4** is 32-bit Number Address, with a Network Portion (Numbers Assigned to Network) and Host Portion (Numbers Assigned to Host)

**IPv6** has x1028 More Range, with Hexadecimal Numbers, which Increase Security, Handles Packages More Efficiently and Improve Performance.

**Dynamic IP Addresses** can change, they’re useful for leaving a network and returning.

**Static IP Addresses** can’t change, they’re useful for connecting often like printers.

**IP Subnetting**

**Subnetting** is a Technique for Logically Partitioning a Single Physical Network into Multiple Smaller Networks/Subnets. They are used by Organizations to: *Conceal Network Complexity, Reduce Network Traffic and Split into More Efficient Subnetworks.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Subnet Class Allowances** | | | |
| 0-126 | Class A | i.e. – 34.126.35.125 | 8, provides for the most hosts. |
| 128-191 | Class B | i.e. – 134.23.45.123 | 16 |
| 192-223 | Class C | i.e. – 212.11.123.3 | 24 |
| 224-239 | Class D | i.e. – 225.2.3.40 | Multicast & Not for Regular Internet Traffic. |
| 240-255 | Class E | i.e. – 245.192.1.123 | Reserved & Not Used on Public Internet. |

**Parts of a Subnet**

* Network ID: Identifies Network & Makes it Unique.
* Subnet Mask: All IP Addresses that can be used within a Network/Subnet, divides IP Address into Network Bits & Host Bits.
* Host ID Range: All IP Address between Subnet Address & Broadcast Address. (Usable Host IP Addresses in Subnet – First & Last)
* Usable Host ID Number: Depending on Class & Subnet Prefix, can run between 30-254. Always Minus 2.
* Broadcast ID: Target All Systems on and Permits Traffic to be sent Across a Subnet instead of Single Host.

**Networking in the AWS Cloud**

A screenshot of a computer

Description automatically generated **Amazon VPC** is a service used to logically isolate sections of the AWS Cloud creating a private cloud with a user-defined virtual network. Each VPC is *logically isolated* and has a *dedicated AWS account* within a Single AWS Region, potentially spanning across *Multiple Availability Zones*.

|  |  |  |
| --- | --- | --- |
| Cost-Effective (PAYG) | Customize Network Configuration | Migrate & Use AWS easily |
| Secure, Scalable & Reliable | Multiple Security Layers | Works with AWS & 3rd Parties |
| Can Create Multiple VPC | Test Environments before going live/ | Can Create Subnets |

|  |  |
| --- | --- |
| **CIDR Block** are Private IP Range given from /16 (65,536 Addresses) to /28 (16 Addresses). | |
| **RFC 1918 Range** | **Example Amazon VPC CIDR Block** |
| 10.0.0.0-10.255.255.255 | 10.0.0.0/16 |
| 172.16.00.0-172.31.255.255 | 172.3.0.0/16 |
| 192.168.0.0-192.268.255.255 | 192.168.0.0/16 |

**A screenshot of a computer

Description automatically generated**

**Internet Gateways** permit Communication from VPC to Internet using a *Public Subnet* (Route 0.0.0.0/0) and *Public IP Address*.

**NAT Gateways** permit Private Subnet Instances to connect to Outside the VPC, but nothing outside the VPC can initiate a connection therefore it *has no Public IP Address*. The Public Subnet is an elastic IP address, the Private Subnet has the route 0.0.0.0/0.

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

**Route Tables** hold Routes/Targets that direct network traffic within the VPC. There are Destinations (*IP Address & CIDR Range*) and Targets (*Gateway/Network Interface*).

**A close-up of a logo

Description automatically generatedSecurity Groups** are *EC2 Instance Level Firewalls controlling incoming traffic*. They are Stateful (If your instance send requests response traffic can flow back regardless of inbound rules) and Block All Traffic by default.

**Network ACLs** are *Stateless Subnet Level Firewalls*, can be customized to deny all (inbound/outbound) traffic with rules.

VPC Endpoint: Private Connection between AWS Services (internet not required).

**Networking Protocols**

**Network Protocols** define Rules for Formatting/Transmitting Data between Network Devices (*across OSI layers 3 & 4*). There are Two Categories:

* Connection-Orientated: Establishes Connection, Awaits Response & Creates Session of Synchronous Communication between Sender & Receiver.
* Connectionless: Sends Messages from Endpoint to Endpoint, not requiring Session using Asynchronous Connection (*like sending mail*).

|  |  |
| --- | --- |
| **Network Protocols** | |
| **Transport Protocols** | |
| **Internet Protocol (IP)** | Establishes Rules for Relaying/Routing Data Online. |
| **Transmission Control Protocol (TCP)** | Provides Reliable, Connection-Orientated & Ordered Delivery across an IP Network. Has a handshake with the messages SYN, SYN/ACK and ACK. |
| **User Datagram Protocol (UDP)** | Simple Connectionless Communication Model to Deliver Data across IP Network, Unreliable as it doesn’t guarantee Delivery/Ordering of Data. Faster than TCP.DW |
| **Application Protocols** | |
| **(Secure) Hypertext Transmission Protocol (HTTP/HTTPS)** | Used to reach Web Pages, with a full address expressed as a Uniform Resource Locator (URL). *Port 80/443.* |
| **SSL/TLS** | Standard for Securing/Safeguarding Communications for systems using Encryption, TLS is a more secure SSL. |
| **Mail Protocols** |  |
| **SMTP** | Transfers Email Messages between Mail Servers. |
| **POP/IMAP** | Used by Email Clients to Retrieve Messages from Servers. |
| **Remote Desktop Protocols** |  |
| **RDP** | Remotely Access Machines/Servers, essential for securely accessing cloud-based servers & help remote employees use infrastructure on-premises. |
| **SSH** | Opens a Secure CLI on a Remote Linux/Unix PC, *Port 22*. |
| **Management/Support Protocols** | |
| **Domain Name System (DNS)** | Domain Name Database like a contact list, translates URLs into IP Addresses. Port Number |
| **Internet Control Message Protocol (ICMP)** | Diagnoses Network Communication Issues & Generate Responses to Errors in IP Networks. |
| **Dynamic Host Configuration Protocol (DHCP)** | Automatically Assigns IP Addresses, Subnet Masks, Gateways and other IP Parameters to Network Devices. |
| **File Transfer Protocol (FTP)** | Permits Transfer of Files between Computers. *Port 22* |

**Network Technologies**

|  |  |
| --- | --- |
| **Network Utilities** | |
| **HPing3** | Command tests Connectivity, whether the Remote Device is on the Network. The format is *hping3 -s -c 0 -V <Public IP of EC2 Instance/On-Premises Host* |
| **Traceroute** | identifies the Client-Destination Node Path, recording time (mms) for each router to respond. The format is *sudo traceroute -n -T -p 22 <Public IP of EC2 Instance/On-Premises Host>* |
| **NSLookup** | performs a DNS lookup for domain URLs and retrieves the server IP Address, or the inverse. The format is *nslookup -type=NS <domain>.com* |
| **MTR** | provides Continual Updated Output used to Analyse Network Performance. The format is *mtr -n -T -c 200 <Public IP of EC2 Instance/On-Premises Host>* |
| **Telnet** | Command tests whether Individual Ports are open or not, if open it’ll say connected and if not, it’ll say could not connect. The format is *telnet [domain name/Ip] [port]* |

|  |  |
| --- | --- |
| **Wireless Technologies** | |
| **Wired Equivalent Privacy (WEP)** | Wireless Protection & Added Security by Encrypting Data. |
| **Wi-Fi Protected Access (WPA)** | Replacing WEP, Better Security Key & User Authorization Handling |
| **Bluetooth Low Energy (BLE)** | Optimizes Energy Consumption across domains i.e. – Healthcare, Fitness, Security, etc. |
| **5G Cellular Systems** | Eventual Download Speeds of 10GB/s |

**Internet of Things** connects Physical Devices to the Internet to Share/Collect Data while Self-Reporting in Real-Time through Device Gateways using HTTPS.

* **AWS IoT core** connects billions of devices & routes trillions of messages within user-preferred communication protocols

**Enterprise Mobility** support Remote-Working Options allowing workers to Connect & Access Data through Cloud Technology i.e. – *Bring Your Own Device (BYOD), Mobile Device Management (MDM).*

* **Amazon Workspaces** provision Virtual Cloud-Based Windows/Linux Desktop using AWS Directory Services, deploying & managing apps through Amazon Workspaces Applications Manager (Amazon WAM).