Topics: Vpn,cookies,IP,ping,traceroute,ARP,TCP(layers: application ,transport,network,datalink),ip addressing and subnetting

Natting,SSL TLS HTTPS,Asymmetric and Symmetric encryption,Monolith,MIcroservices

1. VPN (Virtual Private Network)

- **Definition**: A VPN is a secure connection over the internet between a user and a network, typically to protect privacy and securely access resources.
- **Purpose**:
  - Protect data from being intercepted.
  - Hide user's IP address and encrypt internet traffic.
  - Provide secure access to a corporate or private network remotely.
- **Types**:
  - **Remote Access VPN**: Used for individual users connecting to a network remotely.
  - **Site-to-Site VPN**: Connects entire networks (e.g., branch offices to headquarters).
- **Protocols**:
  - **IPSec**: Internet Protocol Security.
  - **PPTP**: Point-to-Point Tunneling Protocol.
  - **L2TP**: Layer 2 Tunneling Protocol.
  - **OpenVPN**: Open-source, secure.
- **Encryption**: Typically uses strong encryption algorithms (e.g., AES) to ensure security.

2. Cookies

- **Definition**: Small pieces of data stored by a web browser that track and remember user actions on a website.
- **Types**:
  - **Session Cookies**: Temporary cookies that are deleted once the browser is closed.
  - **Persistent Cookies**: Stored on the device for a set period or until deleted.
- **Usage**:
  - **Authentication**: Remember user login credentials.
  - **Preferences**: Store user preferences for websites.
  - **Tracking**: Track user activity for analytics or targeted advertising.
- **Privacy**: Can be a concern as they may track users across websites.
- Setting a cookie
- Cookies are set using the `Set-Cookie` HTTP header sent in an HTTP response from the webserver. This header instructs the web browser to store the cookie and send it back in future requests to the server

(the browser will ignore this header if it does not support cookies or has disabled cookies).

- As an example, the browser sends its first request for the homepage of the `www.networkencyclopedia.com` website:

- `GET /index.html HTTP/1.1`
- `Host: www.networkencyclopedia.com`
- `...`
- The server responds with two `Set-Cookie` headers:
- `HTTP/1.0 200 OK`
- `Content-type: text/html`
- `Set-Cookie: theme=bluesky`
- `Set-Cookie: sessionToken=xyz003; Expires=Wed, 02 Jun 2021 12:15:28 GMT`
- `...`
- The server's HTTP response contains the contents of the website's homepage. But it also instructs the browser to set two cookies. The first, "theme", is considered to be a *session cookie* since it does not have an `Expires` or `Max-Age` attribute. Session cookies are intended to be deleted by the browser when the browser closes. The second, "sessionToken", is considered to be a *persistent cookie* since it contains an `Expires` attribute, which instructs the browser to delete the cookie at a specific date and time.

- Next, the browser sends another request to visit the `spec.html` page on the website. This request contains a `Cookie` HTTP header, which contains the two cookies that the server instructed the browser to set:

- `GET /spec.html HTTP/1.1`
- `Host: www.networkencyclopedia.com`
- `Cookie: theme=bluesky; sessionToken= xyz003`
- …
- This way, the server knows that this request is related to the previous one. The server would answer by sending the requested page, possibly including more `Set-Cookie` headers in the response in order to add new cookies, modify existing cookies, or delete cookies.
-

3. IP (Internet Protocol)

- **Definition**: A set of rules that govern how data is sent and received over the internet, identifying devices on a network.

- **Versions**:
  - **IPv4**: 32-bit address space, written as four octets (e.g., 192.168.0.1).
  - **IPv6**: 128-bit address space, written in hexadecimal (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334).
- **Responsibilities**:
  - Addressing: Ensures each device has a unique identifier (IP address).
  - Routing: Determines the path that data packets should take.
  - Fragmentation: Breaks data into smaller packets for transmission.

4. Ping

- **Definition**: A network diagnostic tool used to test the reachability of a host and measure the round-trip time for data packets.
- **How It Works**:
  - Sends an ICMP Echo Request to the destination.
  - The destination replies with an ICMP Echo Reply.
- **Purpose**:
  - Check network connectivity.
  - Measure latency (response time).
- **Common Uses**:
  - Troubleshoot network issues.
  - Test if a website or server is online.

5. Traceroute

- **Definition**: A tool used to track the path data takes to reach its destination and identify where delays or issues occur.
- **How It Works**:
  - Sends packets with incrementing TTL (Time-to-Live) values to determine each hop (router) along the path.
- **Purpose**:
  - Identify bottlenecks and network congestion.
  - Trace the route taken by data to its destination.

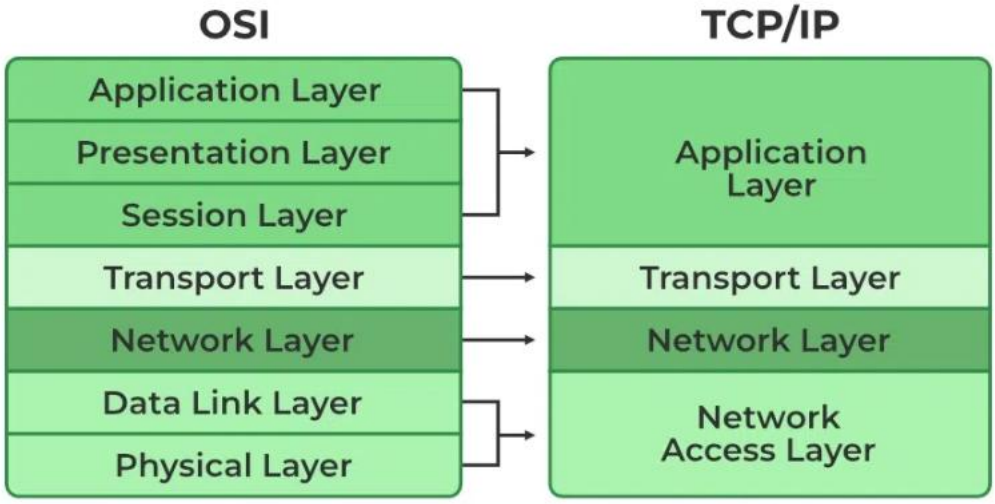
6. ARP (Address Resolution Protocol)

- **Definition**: A protocol used to map a known IP address to its corresponding MAC address in a local network.
- **How It Works**:
  - When a device needs to communicate with another device in the same network, it broadcasts an ARP request.
  - The device with the matching IP responds with its MAC address.
- **Purpose**: Essential for communication within a local network, as Ethernet requires MAC addresses for data transfer.

7. TCP (Transmission Control Protocol)

TCP is a connection-oriented protocol that ensures reliable data transfer. It operates across multiple layers in the OSI model:

- **Layers in the OSI Model**:
    1. **Application Layer**:
        - Deals with user interfaces and high-level protocols (e.g., HTTP, FTP).
        - It is the layer where TCP connections are initiated for communication.
    2. **Transport Layer**:
        - **TCP**: Manages end-to-end communication, ensures reliable transmission (by sequencing packets and requesting retransmission if necessary).
        - Ensures data integrity using checksums.
    3. **Network Layer**:
        - Responsible for logical addressing and routing (e.g., IP).
    4. **Data Link Layer**:
        - Responsible for physical addressing and frame delivery over local networks (e.g., Ethernet, Wi-Fi).



8. IP Addressing & Subnetting

- **IP Addressing**:
    - Assigns unique identifiers to devices on a network.
    - **IPv4** is commonly used, consisting of four 8-bit octets.
- **Subnetting**:
    - The process of dividing a network into smaller sub-networks to improve security and efficiency.
    - **Subnet Mask**: Determines which portion of the IP address represents the network and which represents the host.
    - Example: In a `255.255.255.0` subnet mask, the first three octets are the network part, and the last octet is the host part.

9. NAT (Network Address Translation)

- **Definition**: A method used in routing that allows private IP addresses within a local network to be mapped to a single public IP address when accessing external networks like the internet.
- **Types**:

    - **Static NAT**: One-to-one mapping between private and public IPs.
    - **Dynamic NAT**: A pool of public IPs is used for a pool of private IPs.
    - **PAT (Port Address Translation)**: Many private IPs map to a single public IP, but each connection uses a unique port number.

10. SSL, TLS, and HTTPS

- **SSL (Secure Sockets Layer)** and **TLS (Transport Layer Security)** are cryptographic protocols designed to secure communications over a computer network.
- **TLS** is the successor to **SSL**, offering better security features.
- **HTTPS (HyperText Transfer Protocol Secure)**:
    - A protocol for secure communication over the internet.
    - Uses **SSL/TLS** to encrypt data between the client (browser) and the server.

11. Asymmetric and Symmetric Encryption

- **Symmetric Encryption**:
    - Uses the same key for both encryption and decryption (e.g., AES, DES).
    - Faster but less secure for key distribution.
- **Asymmetric Encryption**:
    - Uses a pair of keys: a **public key** (for encryption) and a **private key** (for decryption).
    - More secure but slower than symmetric encryption.
    - Examples: **RSA**, **ECC**.

12. Monolithic Architecture

- **Definition**: A traditional software architecture where the entire application is built as a single unit.
- **Characteristics**:
    - Tightly coupled components.
    - Hard to scale and maintain as the application grows.
    - Typically difficult to deploy and update in parts.

13. Microservices Architecture

- **Definition**: A modern software architecture that breaks an application into small, independently deployable services.
- **Characteristics**:
    - Each service is focused on a specific business function and can be developed, deployed, and scaled independently.
    - Promotes flexibility, scalability, and resilience.
- **Advantages**:
    - Better fault isolation.
    - Easier scaling and deployment.
- **Challenges**:
    - Complexity in managing multiple services.
    - Communication overhead between services.

Topics for presentation:Firewall