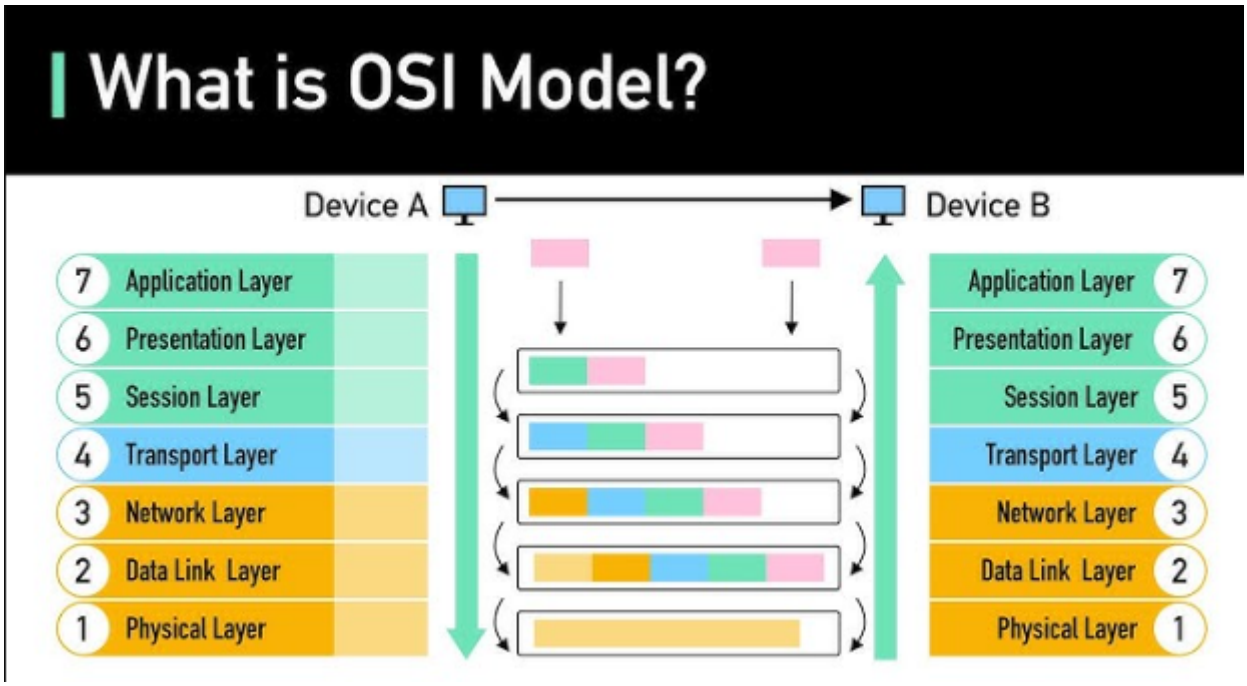
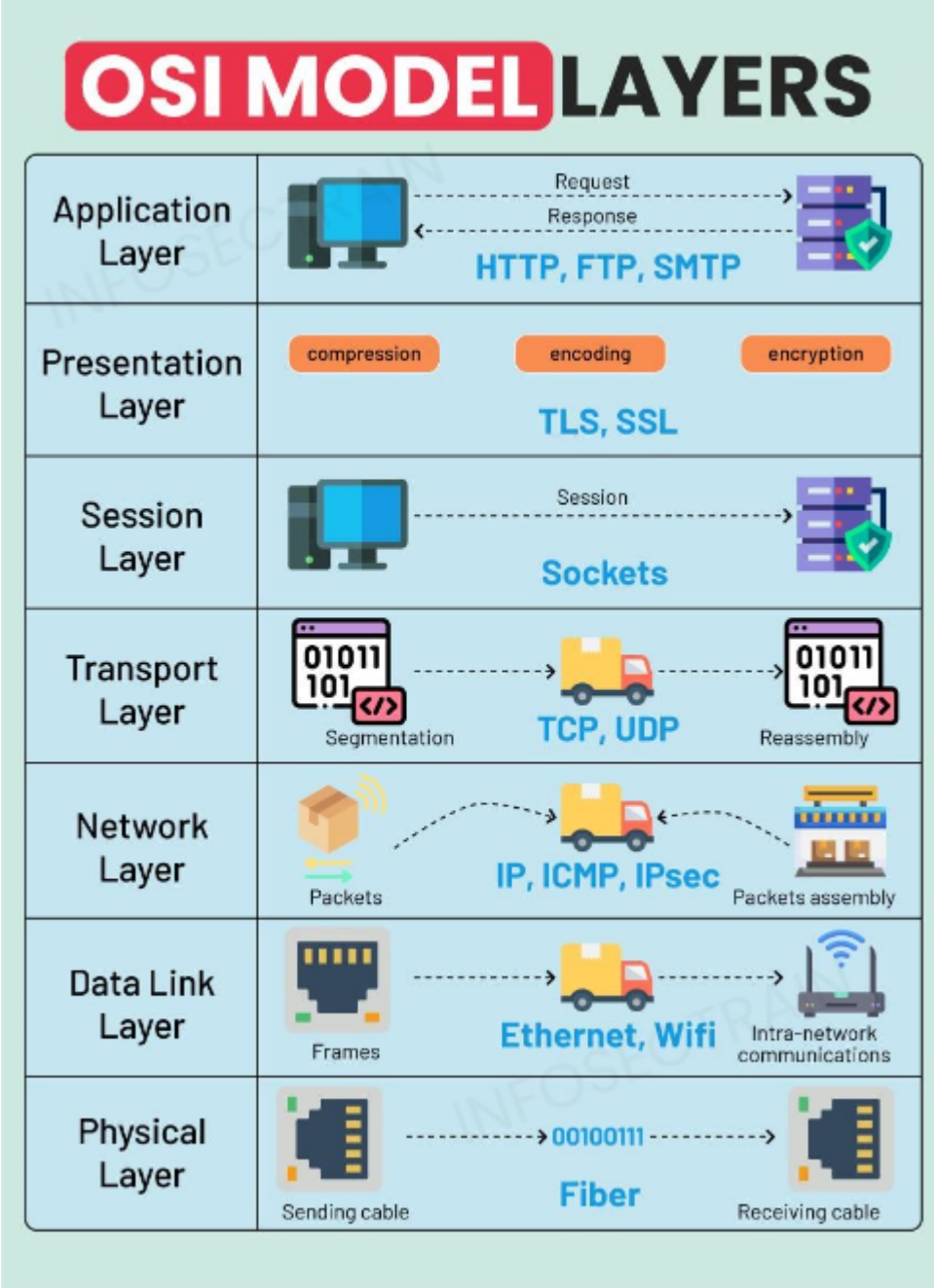
OSI Model in Networking

- Refer Here for osi model





- Each layer adds some protocol additions as well as some networking concept additions
    - Layer 2: Ethernet and Wifi

Reference

The OSI (Open Systems Interconnection) model is a conceptual framework used to understand and standardize the functions of a telecommunication or computing system into seven distinct layers. Each layer has specific protocols associated with it, which facilitate various aspects of network communication. Below is an overview of the protocols commonly associated with each layer of the OSI model.

1. Physical Layer (Layer 1)

- **Function**: Deals with the physical connection between devices, including the transmission of raw bitstreams over a physical medium.
- **Protocols/Technologies**:

    - Ethernet (IEEE 802.3)
    - RS-232
    - Token Ring
    - USB

2. Data Link Layer (Layer 2)

- **Function**: Responsible for node-to-node data transfer and error detection and correction from the Physical layer.
- **Protocols**:
    - Ethernet (IEEE 802.3)
    - Point-to-Point Protocol (PPP)
    - Frame Relay
    - HDLC (High-Level Data Link Control)

3. Network Layer (Layer 3)

- **Function**: Manages the routing of data packets between devices across different networks.
- **Protocols**:
    - Internet Protocol (IP)
    - Internet Control Message Protocol (ICMP)
    - Address Resolution Protocol (ARP)
    - IPv6

4. Transport Layer (Layer 4)

- **Function**: Ensures reliable data transfer between end systems, providing error recovery and flow control.
- **Protocols**:
    - Transmission Control Protocol (TCP)
    - User Datagram Protocol (UDP)
    - Stream Control Transmission Protocol (SCTP)

5. Session Layer (Layer 5)

- **Function**: Manages sessions between applications, handling the opening, closing, and management of connections.
- **Protocols**:
    - NetBIOS
    - RPC (Remote Procedure Call)
    - PPTP (Point-to-Point Tunneling Protocol)

6. Presentation Layer (Layer 6)

- **Function**: Translates data between the application layer and the network, including data formatting, encryption, and compression.
- **Protocols**:
    - Secure Sockets Layer (SSL)

    - Transport Layer Security (TLS)
    - JPEG, GIF (for image formats)

7. Application Layer (Layer 7)

- **Function**: Provides network services directly to end-user applications, facilitating user interaction with the network.
- **Protocols**:
    - Hypertext Transfer Protocol (HTTP)
    - File Transfer Protocol (FTP)
    - Simple Mail Transfer Protocol (SMTP)
    - Domain Name System (DNS)

Each layer of the OSI model plays a crucial role in the overall network communication process, and the protocols associated with each layer ensure that data is transmitted efficiently and reliably across diverse network environments

HTTP

HTTP, or Hypertext Transfer Protocol, is the foundational protocol used for transmitting data over the web. It defines how messages are formatted and transmitted, and how web servers and browsers should respond to various commands.

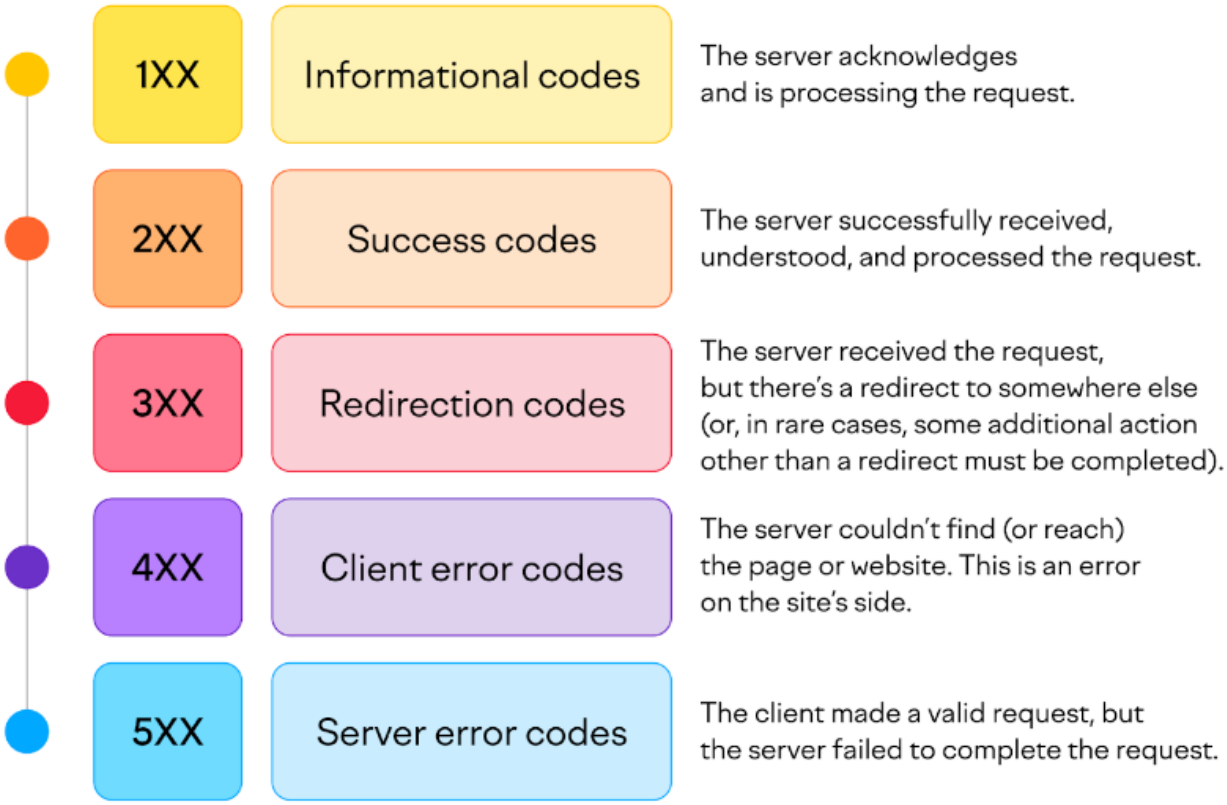
Key Features of HTTP

- **Protocol Type**: HTTP is an application layer protocol in the OSI model, facilitating communication between clients (like web browsers) and servers.
- **Request-Response Model**: HTTP operates on a request-response model. A client sends an HTTP request to the server, which then responds with the requested resource, typically in the form of HTML pages, images, or other data types.
- **Statelessness**: HTTP is stateless, meaning each request from a client to a server is treated as an independent transaction, unrelated to previous requests. This simplifies server design but requires additional mechanisms (like cookies) for maintaining state across multiple requests.
- **Methods**: HTTP supports several request methods, including:
    - **GET**: Retrieve data from the server.
    - **POST**: Send data to the server, often used for form submissions.
    - **PUT**: Update existing resources.
    - **DELETE**: Remove resources from the server.

Security Concerns

While HTTP is widely used, it is inherently insecure because it transmits data in plain text, making it vulnerable to interception and attacks such as man-in-the-middle (MITM) attacks. This is where HTTPS (Hypertext Transfer Protocol Secure) comes into play, which adds a layer of security through encryption using SSL/TLS protocols, ensuring that data exchanged between the client and server is encrypted and secure from eavesdropping and tampering

Status Codes



URI/URLs

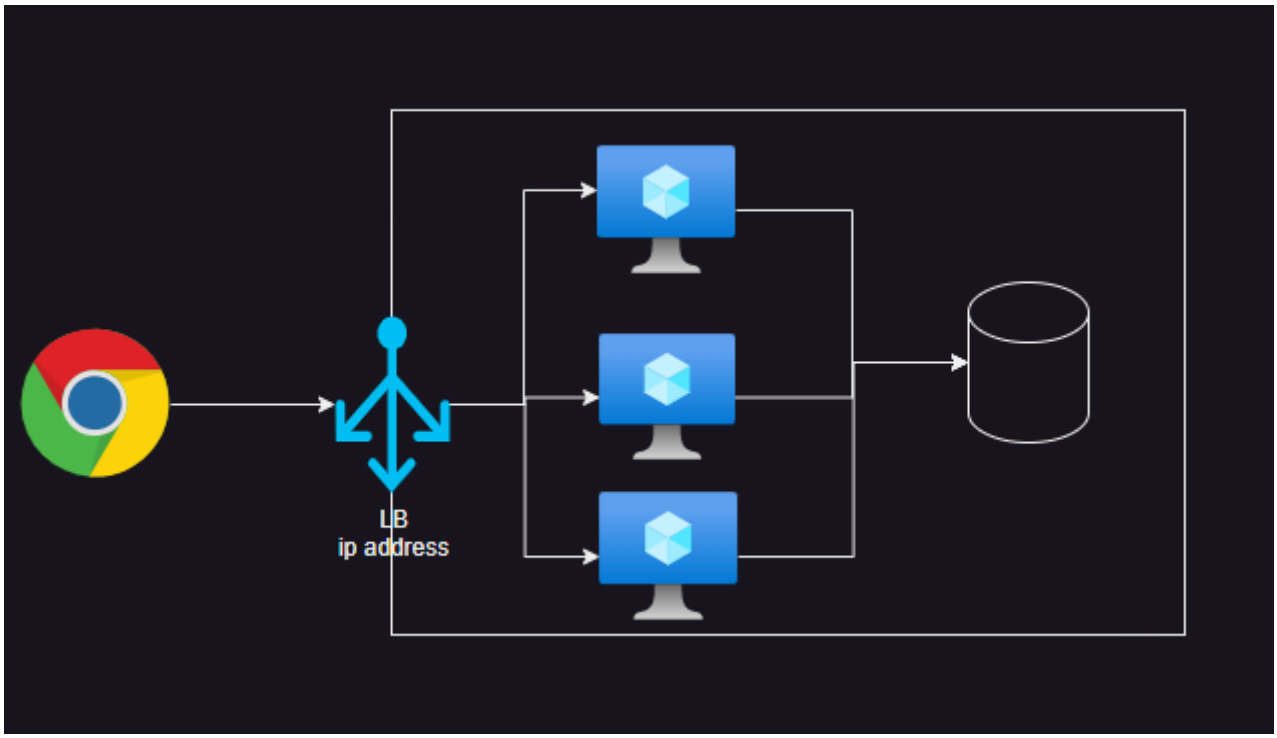
- Refer Here

Conclusion

In summary, HTTP is a crucial protocol for web communication, enabling the transfer of hypertext documents and other resources. However, due to its lack of security features, HTTPS is recommended for any website that handles sensitive information, as it provides encryption and ensures data integrity and confidentiality during transmission.

Load Balancers

- When your application is Highly available (multiple servers running the same application), we would have a load balancer which forwards the traffic to one of the server



- Refer Here for types of load balancing
- We have two types of Load Balancers
    - Layer 4 load Balancer
    - Layer 7 Load Balancer
- Layer 4: layer4 is aware of
    - ip address
    - tcp/udp
    - port
- Layer 7: Layer 7 is aware of
    - ip address
    - tcp/udp
    - port
    - http(s)