2. Communication and Internet Technologies

14.1 Protocols

Purpose of Protocols

- Provide a standard set of rules to enable successful data transfer
- Enables communication/compatibility between devices from different manufacturers or platforms
  - ➔ makes communications independent of software and hardware
- If two devices with different protocols were sending messages between each other, they would not be able communicate properly

Purpose of SSL and TSL protocols

- Provide communications security over the internet/network (by providing encryption)
- Enable two parties to identify and authenticate each other – allows them to communicate with confidentiality and integrity

Protocol Suites

- The protocols in a stack determine the interconnectivity rules for a layered network model such as the TCP/IP model

TCP/IP Protocol Suite

- *Maintains connection between two hosts and ensures successful delivery between them*
- *During data transfer, the self-contained modules (layers) are applied in order using the device's software*
- *Separate layers make hardware and software compatibility easier to implement*
- Can be represented as a stack, with each layer having a separate functionality

Application
Transport
Internet
Link

1. Application Layer (Protocols Used)

- HTTPS – for sending/receiving/transfer of web pages and hypertext documents
- FTP – for sending and receiving files over a network/between devices (used to transfer data from server to client on the network)
- SMTP – push protocol for sending/uploading emails
- POP3 or IMAP – pull protocols for retrieving/receiving emails
    - ➔ Keeps the server and client in sync by not deleting the original email
- BitTorrent – provides peer-to-peer file sharing over a network
    - ➔ Allows sharing of files between many users connected together over the internet
    - ➔ Allows more users to share files than a normal peer-to-peer network would
    - ➔ Users share files directly with each other – there is no web server (all users are of equal status)

Purpose of Application Layer

- Provides access to all programs that exchange data - interacts directly with the user
- Used by web browsers or server software
- Enables data transfer to/from the Transport Layer - allows applications to access services in other TCP/IP layers
- Defines the protocols that an application uses to allow the exchange of data

2. Transport Layer – handles packets

- Responsible for delivery of data from source host to destination host
- Breaks data into manageable packets (performs segmentation) and sends them to the internet layer
- Adds a packet header and the sequence number to the header (sequences packets)
- Controls flow of packets
- Handles packet loss/corruption – ensures data arrives error free

3. Internet Layer – handles transmission of packets

- Identifies the intended network and host
- Transmits packets to the Data Link
- Routes packets independently – through optimum route
- Addresses packets with their source and destination IP addresses
- Uses an IP address and port number to form a socket

4. Link Layer (Network Access Interface) – Handles how data is physically sent

- Ensures the correct network protocols are followed
- Enables the upper layers to access the physical medium (allows communication with the network layer)
- Responsible for transporting data within the network – formats data into frames for transmission
- Maps IP addresses to MAC addresses

TCP/IP Layer Interactions

- Each layer can only accept input from adjacent layer (next higher or next lower layer)
- There is an interface between the adjacent layers which the only interaction between them
- Interactions are carried out by installed software
- User interaction takes place at the application layer of stack through protocols
- Direct access to hardware takes place at the Link layer of the stack

14.2 Circuit Switching, Packet Switching

Circuit Switching

- Data is transferred using a dedicated circuit/channel and implemented at the physical layer
- Circuit is established before transmission starts
- Circuit lasts for for entire duration of transmission and closes after it ends
- Data is transferred using the whole bandwidth
- All data is transferred over the same route
- Transmission is generally bidirectional

Use

- When a dedicated path needs to be sustained throughout a call
- Where the whole bandwidth is required or real time communication is used
- E.g. standard voice communications, video streaming, private data networks
- Suitable for long continuous communication

Advantages

- No need to reassemble data - frames arrive in same order in which they are sent
- Entire bandwidth is available
- Simpler and fast method of data transfer – data is transmitted with a fixed data rate and follows the same path (means no data is lost or disordered)
- Fast data transfer rate - suitable for real time transmission

Disadvantages

- No other transmission can occur when circuit is in use - wasted bandwidth (cannot be shared)
- Not very secure - can be intercepted more easily as all data travels along same route
- Significant cost and time required to establish dedicated connection between stations
- Only one route is available - if an error occurs, transmission ends
- Can take time to set up circuit before start of transmission
- Circuit is always there, even it is not being used

Packet Switching

- Implemented at the network layer

Use

- On digital data networks such as the internet – for sending large files that don’t need to be live streamed
- When it is necessary to overcome faulty lines through rerouting
- For secure communication and high volume data transmission
- When the entire bandwidth isn’t required
- E.g. emails, text messages, documents etc.

Transferring Messages Across Internet

- Data/message to be transmitted is divided into equal-sized packets (consist of a header, payload and trailer)
- A packet header is attached to each packet containing key information (source/destination IP addresses, packet number etc.)
- Each packet is dispatched independently and given its own route - the routing for a packet depends on the network traffic
- Routes are determined using a routing table - optimum route is always taken
- Packets usually arrive out of order - are reassembled in the correct order at destination (using sequence number in header)
- If packets are missing/corrupted a resend request is sent

Advantages

- Packets are more likely to arrive as they can be rerouted or retransmitted if an error occurs
- Bandwidth can be shared - packets from multiple messages can share paths
- Secure - packets travel via different routes
- High data transmission rate is possible

Disadvantages

- Time delay occurs - packets need to be reassembled at destination, channel bandwidth has to be shared with other packets, packets might need to be re-sent
- Requires a complex algorithm to function
- Needs lots of RAM to handle large amounts of data

Function of a Router in Packet Switching

- Router examines the packet’s header – reads the IP address of destination
- Has access to a routing table – contains info about the netmask/gateway used, available hops and the status of the routes along route
- It decides on the next hop/best route and sends the packet on its next hop

Benefits

- Accuracy – Ensures accurate delivery of message
- Completeness – Missing packets can be easily detected and a resend request sent to message arrives complete
- Router can detect changes in networks and send data another way, ensuring it arrives
- Allows simultaneous use of channel by multiple users
- Better security – packets are hashed and send by different routes

Drawbacks

- Time delays – due to correcting potential errors in packets caused by network problems
- Requires complex protocols for delivery
- Unsuitable for real time transmission applications