Lecture 4 (Application Layer pt 2)

1   HTTP (continued)

1.1   HTTP/2

In *HTTP1.1*, the responses happen in FCFS manner. This leads to head-of-line (HOL) blocking and loss recovery time is huge. HTTP/2 overcomes these issues in the following ways:

1. Transmission happens on a priority order and not FCFS
2. Objects are divided into frames and scheduling is done to mitigate HOL blocking

This was proposed by **Google** and was called **SPDY** earlier. This later went into **Request for Comments (RFC)** and then became HTTP/2.

2   Email

This is made up of three main components:

1. User agents
2. Mail servers
3. Simple Mail Transfer Protocol (SMTP)

2.1   User Agent

1. User is the *mail reader*
2. Performing actions of composing, editing, reading mail messages
3. Outgoing, incoming messages are stored on the server

2.2   Mail Server

1. *Mailbox* contains incoming messages for user
2. *Message queue* of outgoing messages

2.3   SMTP Protocol

1. Mail server functions as both client and server in different scenarios

2. Sender is client and receiver is server
3. Uses TCP protocol on port 25
4. Phases of transfer:
    i. Handshaking - 220 sent and 250 received
    ii. Transfer of messages
    iii. Closure
5. Interaction similar to HTTP, differences are:
    i. SMTP has client *push* unlike client *pull* in HTTP
    ii. SMTP has multple objects in multipart message unlike HTTP which has a single response message for each response
    iii. SMTP requires message to be in 7-bit ASCII (other data is encoded)
    iv. "\r\n.\r\n" is used to identify end of message in SMTP

2.4 (Internet Mail Access Protocol) IMAP

This protocol is used to manage and access messages on user's email server

3 Domain Name System (DNS)

1. Distributed database is implemented in the hierarchy of many name servers
2. Application Layer Protocol: host and DNS server communicate to *resolve* names

3.1 Services Provided by DNS

1. Hostname-to-IP-address translation
2. Host aliasing
3. Mail server aliasing
4. Load distribution (same URL has multiple web servers)

3.2 Why is it not Centralised?

1. Can lead to single point of failure
2. Traffic volume
3. Distant centralised database
4. Maintenance

3.3 DNS Architecture

1. A lot more reads than writes
2. Almost every internet transaction interacts with a DNS server
3. Each organisation needs to ensure updation of its record

3.3.1   Implementation

Hierarchical database:

1. Root
2. Top Level Domain (.com, .org, .edu)
3. Authorative

3.3.1.1   Root Name Server

1. Contact-of-last-resort for name servers that cannot resolve names
2. DNSSEC provides security and management
3. ICANN (International Corporation for Assigned Names and Numbers) manages root DNS domains
4. 13 logical root name servers exist which are replicated world-wide

3.3.2   Local DNS Name Servers

1. To prevent the situation of high load on the few root name servers, the query is first sent to the ISP's local DNS server.
2. If record is found in the local DNS's cache, it is retured
3. Else, the query is forwarded to the DNS hierarchy for resolution
4. Caching can lead to delayed updation of IP address of a URL worldwide

3.3.3   Iterated Query

Local DNS server makes multiple queries based on the response from the DNS server, decreasing the level in the hierarchy each time

3.3.4   Recursive Query

The contacted name server makes further queries before sending back the result, hence the local DNS server only makes a single query. The drawback this faces is the excessive load on the root DNS sever

3.4   DNS Record

Resource Records (RRs) have the format of **(name, value, type, ttl)**. The different values type can have are:

1. A - name is hostname, value is IP address
2. NS - name is domain, value is hostname of authorative name server for domain
3. CNAME - name is alias name, value is canonical name
4. MX - value is name of SMTP mail server associated with name

3.5 DNS Protocol Messages

1. Header has a 16 bit unique identifier for the query, reply contains the same ID
2. Next come the flags
3. This is followed by the number of questions, answer RRs, authority RRs and additional RRs
4. The details on each of the above follows in the same order

3.6 How to add new DNS Entry

1. Register URL at *DNS Registrar*
2. Names, IP addresses of authorative name servers are to be provided
3. A, NS Resource Records are entered into corresponding TLD server