Chapter 3

TCP/UDP

LEARNING OBJECTIVES

- *Transport layer*
- *User Diagram Protocol (UDP)*
- *TCP/IP*
- *TCP/IP vs OSI reference model*
- *TCP state transition diagram*
- *TCP flow control*
- *Application layer*
- *ICMP, SMTP, POP3, IMAP 4, HTTP, FTP*
- *DNS*
- *Network devices*

Transport Layer

Real communication takes place between two applications programs i.e., processes. For this, process-to-process delivery is needed. A mechanism is required in order to deliver data from one of these processes running on the source host to the corresponding process running on the destination host.

The transport layer is responsible for process-to-process delivery.

Addressing in Transport Layer

Port addresses

- A transport layer address is a port number.
- The destination port number is needed for delivery and the source port number is needed for reply.
- The port numbers are 16-bit integers ranging from 0 to 65535.

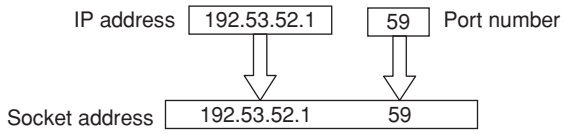
The IANA (Internet Assigned Number Authority) has divided the port numbers as:

- Well-known ports (0 to 1023)
- Registered ports (1024 to 49,151)
- Dynamic or private or ephemeral ports (49,152 to 65,535)

Socket address

Process to process delivery needs two identifiers, IP address and port address at each end to make a connection.

The combination of an IP address and a port number is socket address.



Protocols at transport layer

1. UDP
2. TCP
3. SCTP

User Datagram Protocol (UDP)

- UDP is connectionless protocol.
  - There is no mechanism for connection establishment or connection termination.
  - The packets may be delayed or lost or may arrive out of sequence, i.e., there is no acknowledgement.
  - Each user datagram sent by UDP is an independent program. Even if the user datagram's are coming from the same source program and going to the same destination process, there is no relationship between the different datagrams.

  Thus, user datagrams can travel on a different path.
- Multicasting capability is embedded in UDP.
- It is a simple, unreliable transport protocol.
  - There is no flow control, no window mechanism.
  - There is no error control as well except for the checksum. The sender does not know if a message has been lost or duplicated. When the receiver detects an error through the checksum, the datagram is discarded silently.
- It is used in real-time applications.
  - The header length is fixed, of 8 bytes. Real time applications require a constant flow of data. Moreover, the unreliability (fast and less complex service) of UDP aids in real-time applications like voice over IP, online games etc.
- It encapsulates and decapsulates messages in an IP datagram.

User Datagram

UDP packets have other name called user datagrams. They have a fixed size header of 8 bytes. The datagram is divided into 4 fields.

Source port number (16-bits)	Destination port number (16-bits)
Total length (16-bits)	Checksum (16-bits)

Figure 1 User datagram header format

1. ***Source Port Number*** It is a 16-bit number used by the process running on the source host.
2. ***Destination Port Number*** It is also a 16 bit number used by the process running on the destination host.
3. ***Total length*** It is a 16-bit field, it defines the total length of the user datagram header and data. It can define a total length of 0 to 65535 bytes. A UDP packet is encapsulated in an IP packet.

UDP length = IP length – IP header's length

4. ***Checksum:*** It is optional field, if not available the field is filled with 1's. It is used to detect errors in user datagram (header plus data).

Protocols That Take UDP Services

Following are a few protocols that take the services of UDP:

1. Domain Name Service (port – 53): UDP is used to send small data. If the data is less than 512 bytes, then DNS uses UDP else it goes for TCP.
2. Trivial File Transfer Protocol (port – 69): TFTP is used to transfer simple and small files, it uses UDP service.
3. Routing Information protocol: It uses UDP service on port number 520 to update routers.
4. Simple Network Management Protocol (SNMP): The SNMP agent receives requests on UDP port 161 for management process.
5. Bootstrap protocol (BOOTP): For client (port 68) and for server (port – 67).

UDP Checksum Calculation

- The checksum includes a pseudo header, the UDP header and the data coming from the application layer.

32-bit source IP address		
32-bit destination IP address		
All 0's	8-bit protocol (17)	16-bit UDP total length

Figure 2 Pseudo header of UDP for checksum calculation

- The value of protocol field is 17. If this value is changed during transmission, the checksum calculation at the receiver will detect it and UDP drops the packet.
- If the checksum is not calculated, the field is filled with 0's. This means checksum calculation is optional.
- The calculated checksum can never be all 1's as this implies that the sum is all 0's. But this is impossible because for this the value of fields have to be 0's.

TCP/IP

TCP/IP is a network model which is used for the internet architecture, its main objectives are

- Connecting the multiple networks.
- Maintaining the intact connection between two machines, which are functioning.

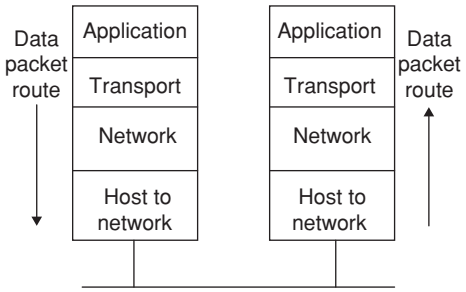


Figure 3 TCP/IP network protocol

TCP/IP vs OSI Reference Model

	OSI		TCP/IP
(1)	There are 7 layers	(1)	There are 5 layers
(2)	There is no definition for multicasting	(2)	Multicasting is clearly defined
(3)	Less flexibility	(3)	Lot of flexibility
(4)	Practically it is not suggestible as it is based on theoretical rules	(4)	It is based on practical rules

- TCP stands for Transmission Control Protocol.
- It is connection-oriented protocol.
  - It creates a virtual connection between two TCPs to send data then data is transferred and at the end the connection is released.
  - There is acknowledgement mechanism for safe and sound arrival of data.
- It is a reliable transport protocol.
  - Uses flow and error control.
  - Slower and more complex service.
  - Duplicate segments are detected, lost segments are resent, the bytes are delivered to the end process in order.
- It is a stream-oriented protocol.

- Allows the sending process to deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes.
- TCP offers full-duplex service.
  - Data can flow in both directions at the same time.
  - Each TCP has a sending and receiving buffer.
  - It cannot be used in real time applications as the header length varies from 20-to-60 bytes, moreover it needs reliability.

TCP Header Format

- A packet in TCP is called a segment. The segment consists of a 20-to-60 bytes header.
- If there are no options, the header is of 20 bytes.

Source port address (16-bits)					Destination port address (16-bits)			
Sequence number (32-bits)								
Acknowledgement number (32-bits)								
HLEN (4-bits)	Reserved (6-bits)	URG	ACK	PSH	RST	SYN	FIN	Window size (16-bits)
Checksum (16-bits)					Urgent Pointer (16-bits)			
Options and Padding								

Figure 4 TCP header format

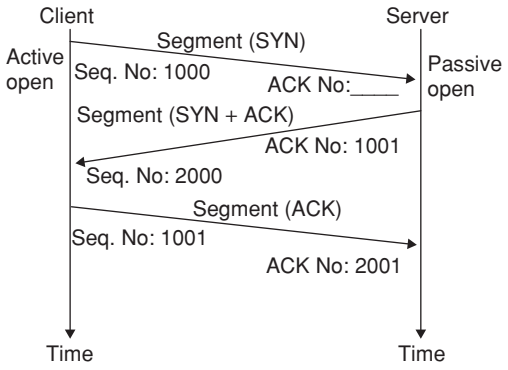
- If there are options, the header goes upto 60 bytes.
- ***Source Port addresses*** A 16-bit field that defines the port number of the application program in the host that is sending the segment.
- ***Destination Port address*** A 16-bit field that defines the port number of the application program in the host is receiving the segment.
- ***Sequence number*** A 32-bit field whose value defines the number of the first data byte contained in that segment. During connection establishment, a random number is generated to create an initial sequence number (ISN) which is usually different in each direction.
- ***Acknowledgement Number*** A 32-bit field whose value defines the number of the next byte, a party expects to receive. If the receiver of the segment has successfully received byte number x from the other party, it defines $x + 1$ as the acknowledgement number. The acknowledgement number is cumulative.
- ***HLEN(Header Length)*** This field is of 4-bit. The header length can be between 20 and 60 bytes. The value of this field can be between $5(5 \times 4 = 20)$ and $15(15 \times 4 = 60)$.
- ***Reserved*** This is a 6-bit field which is reserved for future use.
- ***Control*** This field contains 6 control flags. These are as follows.
  - URG: Urgent pointer. This flag is set when the value of urgent pointer field is valid.
  - ACK: Acknowledgement pointer. This flag is set when the value of acknowledgement field is valid. It is not set at the start of connection during 3-way handshake.
  - RST: Reset pointer. Used to reset the connection, reject an invalid segment or refuse an attempt to open a connection.
  - PSH: Push pointer. When a data is pushed the flag is set.
  - SYN: Synchronization pointer, used to synchronize sequence numbers during connection. If it is set to 1, then it is ISN. If set to 0, then it is the accumulated sequence number of the first data byte of the segment for the current session.
  - FIN: Finish Pointer. It is used to terminate a connection. It indicates that the sender is not interested in sending any more data.
- ***Window size*** The field size is of 16-bits and thus the maximum size of the window is 65,535 bytes. This field is determined by the receiver and thus referred to as the receiving window. The window size is variable.
- ***Checksum*** The inclusion of this 16-bits field is mandatory in TCP. The calculation of the checksum for TCP follows the same procedure as in UDP, only the value of protocol field in TCP is 6.
- ***Urgent pointer*** This 16-bit field, is valid only if the urgent flag is set. This field is used when the segment contains urgent data.
- ***Options and padding*** When the header length is greater than 5, option field is used to make the segment into the multiples of 32. Padding is used to ensure the ending of TCP header, it is composed to 32 zeros.

TCP Connection

- TCP is connection-oriented and the connection is virtual not physical.
- TCP operates at a higher level. TCP uses the services of IP to deliver individual segments to the receiver, but it controls the connection itself. Lost or corrupted segments are retransmitted.
- In TCP, connection-oriented transmission requires three phases:
  1. Connection establishment
  2. Data transfer
  3. Connection termination

Connection establishment

- The connection establishment in TCP is called three-way handshaking.
- The process starts with the server. The server program tells its TCP that it is ready to accept a connection. This is a request for a passive open.
- The client program issues a request for an active open. A client that wishes to connect to an open server tells its TCP that it needs to be connected to a particular server. Hence the TCP can start the three-way handshaking process as shown in the figure.



1. The first segment which is a SYN segment is identified by the randomly generated number and is assigned to a 1 byte dummy data indicating the sequence number.
2. Again from the server side a randomly generated number is assigned for the dummy data indicating the first byte.
3. A SYN segment cannot carry data, but it consumes one sequence number.

   A (SYN + ACK) segment cannot carry data, but consumes one sequence number.

   An ACK segment, if carrying no data, consumes no sequence number.
4. Initial Sequence Number (ISN) 1000 is sent from the client to server. Server receives the segment 1000 and is expecting segment 1001 as the next one.

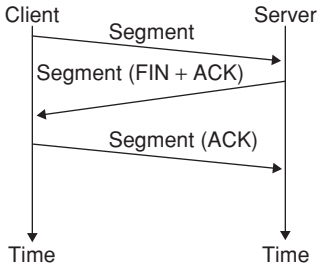
Data transfer

- After the connection is established, bidirectional data transfer can take place. Both the client and server can send data and acknowledgements.
- The data segments sent by the client have the PSH (push) flag set so that the server TCP knows to deliver data to the server process as soon as they are received.
- Sometimes the sending application program wants a piece of data to be read out of order by the receiving application program that means an application program needs to send urgent bytes then in this case the URG bit is set and the segment is send. The sending TCP creates a segment and inserts the urgent data at the beginning of the segment.

Connection termination

There are two options for connection termination.

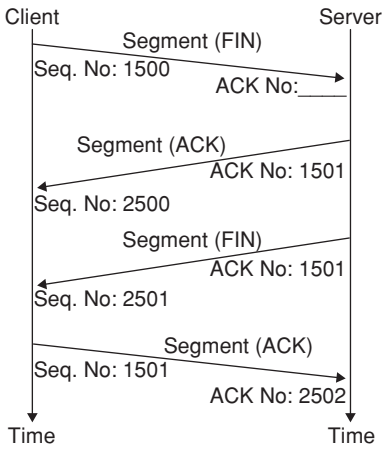
Three-way handshaking



- The client process sends the first segment, a FIN segment in which the FIN flag is set. The FIN segment consumes one sequence number if it does not carry data.
- The server TCP sends the second segment, a FIN + ACK segment, to confirm the receipt of the FIN segment from the client and at the same time announce the closing of the connection in the other direction.
  The FIN + ACK segment consumes one sequence number if it does not carry data.
- The client sends the last ACK segment to the server. This segment contains acknowledgement number which is 1 plus, the sequence number received in the FIN segment from the server.

Four-way handshaking

- ***Half-close**:* In TCP, one end can stop sending data while still receiving data. This is half close.
- The client half-closes the connection by sending a FIN segment.
- The server accepts the half-close by sending the ACK segment. The data transfer from the client to the server stops.
- When the server has sent all the processed data, it sends a FIN segment, which is acknowledged by an ACK from the client.



TCP State Transition Diagram

The functionality of TCP connection setup, communication phase and termination phase can be easily depicted by the state transition diagram where the TCP will be only at one state at a time with respect to server or client.

A change in the state is only observed after receiving a request for change like ACK (acknowledgement).

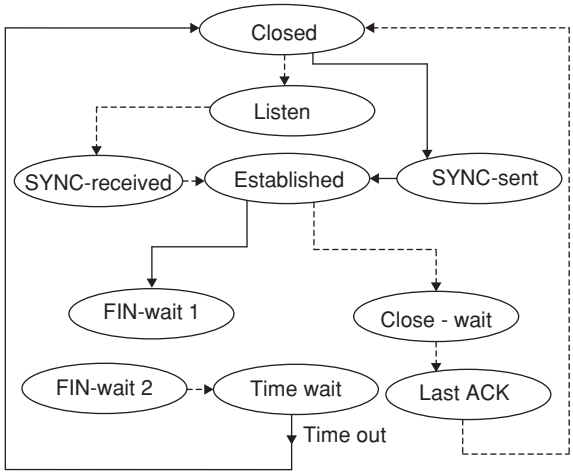


Figure 5 State transition diagram

Here, Solid line '–' is for client states, Break line '---' is for server states

State 'Closed' is common for both client and server. Initially the client and the server are in the closed state where no TCP connection is set. When an application request for a TCP connection then the client changes its state from closed to SYNC-sent state.

Client states

1. SYNC-sent — After the client sends a SYNC-sent and receives an ACK for the sent SYNC segment, it changes its state to ESTABLISHED STATE.
2. Established — In this state the client and the server exchange user data. After the requested application is completed, it sends a FIN segment and changes its state to FIN-wait 1.
3. FIN-wait 1 — FIN-wait 1 changes to FIN-wait 2 after receiving an ACK for sent FIN segment.
4. FIN-wait 2 — The client will remain in this state until it receives a FIN segment from the server. When the last ACK is sent by the client, the client changes its state to Time-wait.
5. Time-wait — A timer is set at this state for any delayed segment from the server which are removed or discarded at the client and after the timeout is reached, the client changes its state from present state to the closed state again.

Server states

1. Listen — This is a passive state where the server always listens for the SYNC request segment on different TCP ports.
2. SYN-received — After receiving the SYNC request from the client, the server acknowledges its state to the Established state.
3. Closed-wait — The server changes its state from Established to close-wait after receiving the finish segment from the client. In this state the server sends an ACK and finish segments. Afterwards it changes the state to last-ACK.
4. Last-ACK — In this state the server expects the last ACK segment from the client, as and when it receives the ACK segment it changes its state to again closed state.

TCP Congestion Control

- Deals with end-to-end delivery.
- Congestion handling in TCP is based on three phases:
  - Slow start
  - Congestion avoidance
  - Congestion detection

Slow start (exponential increase)

1. By default the receiver window size is initially set to 1.
2. In the first instance the transmitter receives an ACK for the window size indicating the receiver window size as 2 segments.
3. After 2 segments are sent it is acknowledged with 4 segments.
4. After 4 segments are sent it it acknowledged with 8 segments.
5. This is exponential growth and this growth continue until the window size reaches the threshold value.
6. If there is delayed ACKs, the increase in the size of the window is less than power of 2.

Congestion avoidance (additive increase)

1. To avoid the congestion before it happens, the exponential growth of slow start algorithm must be slowed down.
2. When the threshold is reached, then the additive phase begins. Here each time the whole window of segments is acknowledged, the size of congestion window is increased by 1.

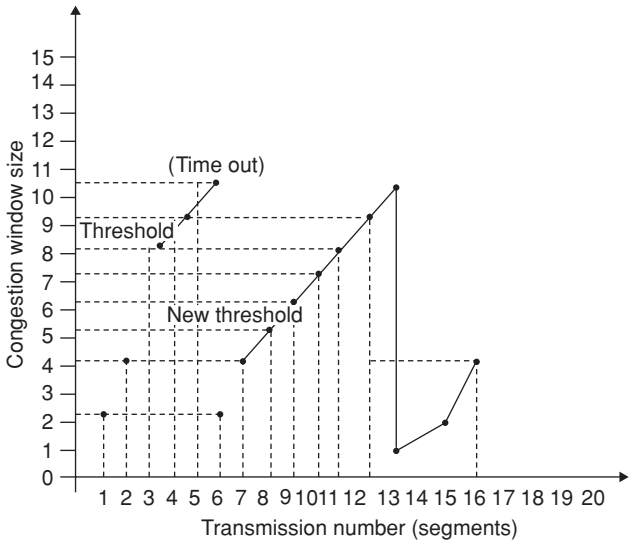
Congestion detection (multiplicative decrease)

1. If congestion occurs, the congestion window size must be decreased. The only way the sender can guess the congestion has occurred is by the need to retransmit a segment.
2. Retransmission can occur in two cases:
   (i) When a timer times out.
   (ii) When 3 ACKs are received.

3. In both the cases the size of threshold is dropped to one-half of the current window size and the window size is decreased to initial window size "1".
   This is multiplicative decrease.

Example: Let us take an example to explain the TCP congestion control.

Consider an instance of TCP additive increase, multiplicative decrease algorithm where the window size at the start of slow-start phase is 2 MSS (Maximum Segment Size) and threshold value is 8 MSS. The timeout occurs at the fifth transmission. Then what is the congestion window size at the end of the tenth transmission?



Window size is 2 MSS initially.

8 MSS is threshold value, after this there is only increase of 1-1 window size till timeout value which is 10.

The new threshold value becomes half of the value of current congestion window i.e., 5.

Timeout remains the same i.e., 10.

At 10th transmission the window size is 7.

After time-out, at 13th transmission window size = 1 and at 14th transmission window size = 2.

TCP Flow Control

- For flow control sliding window protocol is used.
- The window size is set by the receiver and is controlled by the receiver. The window size is not fixed (variable).
- The sliding window protocol in TCP looks like the Go-Back-N protocol because it does not use NAKs; it looks like Selective Repeat because the receiver holds the out-of-order segments until the missing ones arrive.
- A sliding window is used to make transmission more efficient as well as to control the flow of data so that the destination does not become overwhelmed with data. TCP sliding windows are byte-oriented.

TCP Error Control

- TCP provides reliability using error control.
- Error control includes mechanism for detecting corrupted segments, lost segments, out-of-order segments and duplicated segments.
- Error detection and correction in TCP is achieved through the use of three tools:
  - Checksum
  - Acknowledgment
  - Time-out

Checksum

Each segment includes a checksum field which is used to check for a corrupted segment. A 16-bit checksum is mandatory in every segment.

Acknowledgement

- There is no negative ACK in TCP.
- There is no ACK for the received ACK.
- Only the correctly received segments are acknowledged, if any segment if found to be corrupted through checksum such segments are not acknowledged.

Time-out

Different timers are deployed for error control.

1. ***Time-awaited timer:*** This timer is used to handle TCP termination process specially to handle duplicate finish segments.
   Its value is set to twice the life time of a segment.
2. ***Keep-Alive Timer:*** This timer is used to handle long idle TCP connections.
   By default its value is 2 hours, beyond which a probe (1 byte dummy data) is used for 10 consecutive times with a separation of 75 milliseconds. If there is no response beyond this, then the connection is terminated.
3. ***Persistence Timer:*** This timer is used to handle Zero(0) window size scenario.
   The sender sends 1 probe every 60 seconds until it receives a non-zero window size from where the communication resumes.
4. ***Retransmission Timer:*** This timer is used for handling any lost segments. Its value is twice the Round trip time, i.e., $2 \times RTT$.
   RTT is time needed for a segment to reach a destination and for an acknowledgement to be received.

Application Layer

An interface between the networks is called application. This section introduces two important concepts:

- Application Layer: The application layer of the OSI model provides the first step of getting data onto the network.

- Application Software: Applications are the software programs used by people to communicate over the network. Examples of application software, includes HTTP, FTP, e-mail, and others, used to explain the differences between these two concepts.

In the OSI model, information is passed from one layer to the next, starting at the application layer on the transmitting host and proceeding down the hierarchy to the physical layer, then passing over the communications channel to the destination host, where the information proceeds back up the hierarchy, ending at the application layer.

The following six steps explain the procedure:

1. People create the communication.
2. The application layer prepares human communication for transmission over the data network.
3. Software and hardware converts communication to digital format.
4. Application layer services initiate the data transfer.
5. Each layer plays its role. The OSI layers encapsulate data down the stack. Encapsulated data travels across the media to the destination. OSI layers at the destination unencapsulate the data.
6. The application layer receives data from the network and prepares it for human use.

The application layer, layer 7, is the top layer of both the OSI and TCP/IP models. Layer 7 provides the interface between the application you use to communicate and the underlying network over which your messages are transmitted. Application layer protocols are used to exchange data between programs, running on the source and destination hosts.

TCP/IP Application Layer Protocol

The most widely known TCP/IP application layer protocols are those that provide the exchange of user information. These protocols specify the format and control information necessary for many of the common internet communication functions. Among these, TCP/IP protocols are the following.

- Domain name system (DNS) is used to resolve internet names to IP addresses.
- Hypertext transfer protocol (HTTP) is used to transfer files that make up the web pages of the world wide web.
- Simple mail transfer protocol (SMTP) is used for the transfer of mail messages and attachments.
- Telnet, a terminal emulation protocol, is used to provide remote access to servers and networking devices.
- File transfer protocol (FTP) is used for interactive file transfers between systems.

Application Layer Services

Programs such as file transfer or network print spooling, might need the assistance of application layer services to use network resources. Although transparent to the user, these services have interface with the network and prepares the data for transfer. Different types of data whether it is text, graphics or video require different network services to ensure that it is properly prepared for processing by the functions occurring at the lower layers of OSI model. Application layer services establish an interface to the network and protocols provide the rules and formats that govern how data is treated, a single executable program can use all three components. For example, while discussing "Telnet", you could be referring to the Telnet application, the Telnet service, or the Telnet protocol.

Application Layer Protocol Functions

Both the source and destination devices use application layer protocols during a communication session. For the communications to be successful, the application layer protocols implemented on the source and destination host must match.

Protocols perform the following tasks

- Establish consistent rules for exchanging data between applications and services loaded on the participating devices.
- Specifies how data inside the messages is structured and the types of messages that are sent between source and destination. These messages can be requests for services, acknowledgements, data messages, status messages, or error messages.
- Defines message dialogues, ensuring that a message being sent is met by the expected response and that the correct services are invoked when data transfer occurs.

Applications and services can also use multiple protocols in the course of a single conversation. One protocol might specify how to establish the network connection and another might describe the process for the data transfer when the message is passed to the next lower layer.

A single application can employ many different supporting application layer services. Thus, what appears to the user as one request for a web page might, in fact, amount to dozens of individual requests. For each request, multiple processes can be executed. For example, the FTP requires a client to initiate a control process and a data stream process to a server. Additionally, servers typically have multiple clients requesting information at the same time, as shown in the figure below. For example, a Telnet server can have many clients requesting connections to it. These individual client requests must be handled simultaneously and separately for the network to succeed. The application layer processes and services rely on support from lower layer functions to successfully manage the multiple conversations.

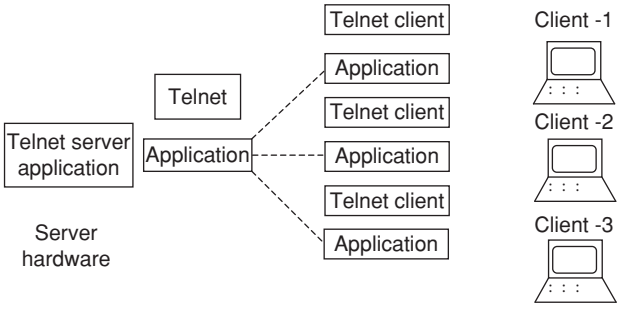


Figure 6 Multiple client's service Requests

Application Layer Protocols

The transport Layer uses an addressing scheme called a port number. Port numbers identify application layer services that are source and destination of data. Server programs generally use predefined port numbers that are commonly known by clients.

Some of these services are

- Domain Name System (DNS): TCP/UDP Port 53
- HTTP: TCP Port 80
- Simple Mail Transfer Protocol (SMTP): TCP Port 25
- Post office Protocol (POP): UDP Port 110
- Telnet: TCP Port 23
- DHCP: UDP Port 67
- FTP: TCP Ports 20 and 21

Internet Control Message Protocol (ICMP)

- Used by hosts and gateways to send notification of datagram problems back to the sender.
- Used for error reporting and query messages.
- Helpful in network debugging.
- Uses the services of TCP and UDP with the port number 7 as the ping command which is used for testing, this testing is done from a source which starts at the application layer and reaches network through transport layer.
- ICMP is encapsulated into an IP datagram and then transmitted into the network, if the protocol filed in the IP datagram is 1 then the IP datagram is said to be carrying ICMP message.

Types of messages

Error reporting

- **Destination Unreachable:** The packet is discarded due to the host not present in the network or the host is not responding to the request.
- **Source Quench:** The packet is discarded due to the congestion in the network.
- **Parameter Problem:** The packet is discarded due to the processing problem observing a change in the header format of the I/P datagram.
- **Time Exceeded:** The packet is discarded because the TTL value is decremented to zero(0).
- Redirection: Here the packet is not discarded but redirected to a network as the host doesn't belong to this network.

Query message

Router solicitation and router advertisement request and reply: Router solicitation is a request generated by the source requesting the router's presence in the network.

The response is a router advertisement generated by the router broadcasting its network id and its presence in the network.

Address mask request and reply: If by any means the node is unable to identify the network bits in its I/P address then this request is used by the source to a router requesting for the network id, the reply is also unicast in this scenario.

Time stamp echo request and reply: This is used to calculate the round trip time of a packet for network diagnose or debugging.

Echo request and reply: This is used to see the presence of a host or a router in the network. For example PING.

SMTP

- SMTP stands for simple mail transfer protocol.
- It uses the services of TCP on port number 25.
- It is a push protocol. Even when the destination is not interested to receive the message this push approach of the SMTP makes the receiver receive the message.
- Components of SMTP:

1. User Agent (UA) :
   (i) It provides Graphical User Interface access to the user.

   Example: Netscape navigation, Mozilla Firefox. It also provides command-driven access in early days.

   (ii) It handles the inbox transactions:
   (a) Composing messages: Helps the user compose the e-mail message to be sent out.
   (b) Reading messages: Helps to read incoming messages by checking the mail in the incoming mail box.
   (c) Replying to messages: Sends the message to the sender or recipients of the copy.
   (d) Forwarding messages: Sends the message to a third party.
   (e) Handling mailboxes: Two mailboxes, an inbox and an outbox are created by the user agent. The inbox keeps all the received e-mails until they are deleted by the user. The outbox keeps all the sent e-mails until the user deletes them.

2. Mail transfer agent (MTA): The actual mail is transferred using MTA.
3. Multipurpose Internet mail extension (MIME): By default SMTP uses ASCII format for transaction. But few languages like Japanese, German etc do not support ASCII format. Hence for carrying non-ASCII form of transactions MIME is used in congestion with SMTP. Thus, MIME is a set of software functions that transforms non-ASCII data (stream of bits) to ASCII data and vice-versa.
4. Mail access protocol (MAP): MAP is a pull approach where the emails of a client are retrieved from the mail server i.e., it is used to retrieve the clients emails from the mail server.

   Two protocol of MAP are

   (i) POP 3 (Post Office Protocol)
   (ii) IMAP4 (Internet MAP)

POP3

1. It is a pull protocol.
2. It uses the services of TCP on port number 110.
3. POP3 has several drawbacks and hence it is currently not in use.
   - A user cannot have different folders on the server.
   - A user cannot partially check the contents of the mail before downloading.
   - A user cannot search a mail with a keyword.
   - The user is not allowed to organize the mail on the server.

(4) Modes of POP3

(i) Copy mode: The mails are copied from the mail server onto the client.

(ii) Delete mode: The mails are transferred from the mail server to the client and deleted at the mail server. By default POP3 uses delete mode.

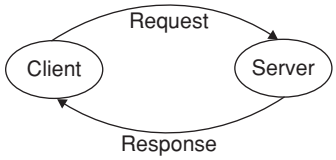
IMAP 4

To overcome the drawbacks of POP3, IMAP4 is in current use. It provides the following functions:

1. A user can create, delete or rename mail boxes on the mail server.
2. A user can create a hierarchy of mailboxes in a folder.
3. A user can partially download e-mail.
4. A user can check the e-mail header before downloading and can search the contents of the e-mail for any specific character prior to downloading.

HTTP

- HTTP stands for Hyper Text Transfer Protocol.
- It uses the services of TCP on well known port 80.
- It is a protocol mainly used to access data on the World Wide Web (www).
- HTTP functions as a combination of FTP and SMTP.
- It uses only one TCP connection, there is no separate control connection, only data is transferred between the client and the server.
- HTTP messages are read and interpreted by the HTTP server and HTTP client (browser).
- It works on two commands request and reply.
- It is a stateless protocol as it does not have any mapping from one transaction onto the other and treats a request and reply as a pair every time.



HTTP1.1 has several request types called methods:

1. GET: Requests a document from the server.
2. HEAD: Requests information about a document but not the document itself.
3. POST: Sends some information from the client to the server.
4. PUT: Sends a document from the server to the client.
5. TRACE: Echoes the incoming request.
6. CONNECT: Reserved.
7. OPTION: Inquires about available options.

- HTTP supports proxy servers. A proxy server is a computer that keeps copies of responses to recent requests. This reduces the load on the original server, decreases traffic and improves latency.
- HTTP Connections:

  (i) *Non-persistence:* In this connection approach for every request and reply (response) as a pair, a separate TCP connection is established every time. It suffers from slow start process. This was present in http version 1.0. Two RTTS are required to fetch each object.

  (ii) *Persistence:* Here a single TCP connection is set on which multiple request and response can be made. This is observed from http version 2.0 onwards (apache http server). For http/1.1 is default. Hence we have reduced network congestion and faster content delivery.

File Transfer Protocol (FTP)

- FTP uses the services of TCP.
- It needs two TCP connections:
  - Uses well-known port 21 for the control connection.
  - Uses well-known port 20 for the data connection.

- Mode of access:

  FTP(TCP) – requires username and password.
  TFTP(UDP) – requires no username and password.
- Types of files supported by FTP:

  i. ASCII: By default FTP follows ASCII mode for file transfer. It is composed of 7-bit + 1 parity bit.
  ii. EBCDIC: If any node supports EBCDIC then this type of technique is used for file transfer. BCDIC supports 8 bits data format and is used in IBM. There is no error control i.e., there is no parity bit.
  iii. Image file: If the file to be sent is very large then continuous streams of 0s and 1s are sent to the transport layer. This is image file. Here FTP does not care of code, it is done by the lower layers.
- **Transmission mode of FTP:** FTP can transfer a file across the data connection by using one of the following three transmission modes:

  i. Stream mode: This is the default mode. Data are delivered from FTP to TCP as a continuous stream of bytes.
  ii. Block mode: Data is delivered from FTP to TCP in blocks. Each block is preceded by a 3-byte header. The first byte is called the block descriptor, the next two bytes define the size of the block in bytes.
  iii. Compressed mode: If the file is big then the data is compressed. The compression method which is mostly used is run-length encoding. Consecutive appearances of a data unit are replaced by one occurrence and the number of repetitions. In a binary file, null characters are compressed.

DNS

- Stands for Domain Name System.
- The DNS is a client/server application that identifies each host on the Internet with a unique user-friendly name i.e., it is used to map an Uniform Resource Locator (URL) to an IP address.
- DNS can use the services of UDP or TCP using the well-known port 53.
- If the size of the response message is more than 512 bytes, it uses the TCP connection.
- When the size of the response message is less than 512 bytes, UDP connection is used. Even though the size of message is not known then also UDP can be used. The UDP server will truncate the message if the message size is more than 512 bytes.
- DNS organizes the namespace in a hierarchical structure to decentralize the responsibilities involved in naming.
- DNS can be pictured as an inverted hierarchical tree structure with one root node at the top and a maximum of 128 levels. Each node in the tree has a domain name.

For example, on the Internet, the domain names, such as http//www. cisco.com, are much easier for people to remember than 198.132.219.25. Also if, cisco decides to change the numeric address, it is transparent to the user, because the domain name will remain http//www.cisco.com. The new address will simply linked to the existing domain name and connectivity is maintained as shown in the figure.

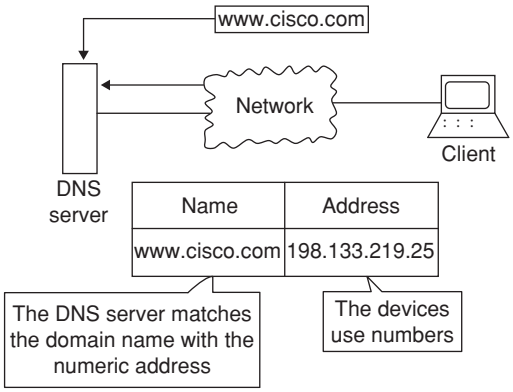


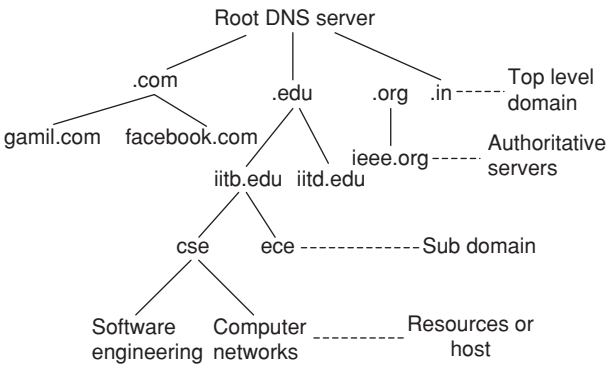
Figure 7 DNS addresses

When networks were small, it was a simple task to maintain the mapping between domain names and the addresses they represent. However, as networks began to grow and the number of devices increased, this manual system became unworkable. DNS was created for domain name to address resolution for these networks. DNS uses a distributed set of servers to resolve the names associated with these numbered addresses.

The DNS protocol defines an automated service that matches resource names with the required numeric network address. It includes the format for queries, responses, and data formats. DNS protocol communications use a single format called a message. This message format is used for all types of client queries and server responses, error messages, and the transfer of resource record information between servers. DNS is a client/server service, however, it differs from the other client/server services. Where as other services use a client that is an application (Web browser, e – mail, client, and so on) the DNS client runs as a service itself. The DNS client, sometimes called the DNS resolver, supports name resolution for the other network applications and other services that need it.

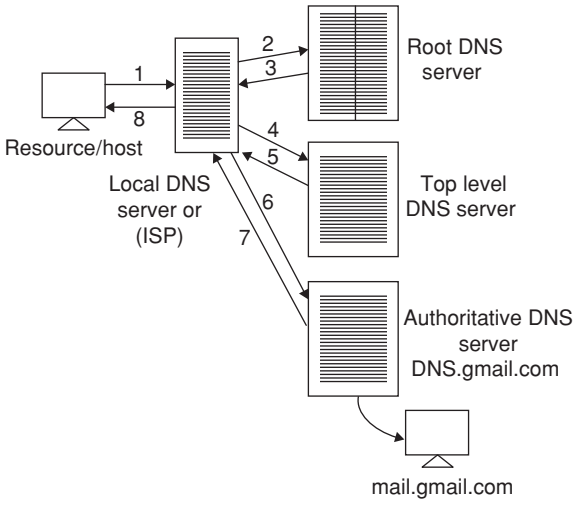
When configuring a network device, you generally provide one or more DNS server addresses that the DNS client can use for name resolution. Usually the Internet Service Provider (ISP) gives you the address to use for the DNS servers. When a user's application requests to connect to a remote device by name, the requesting DNS client queries one of these DNS servers to resolve the name to a numeric address.

- The domain name space consists of a tree of domain names. Each node or leaf in the tree has zero or more resource records, which holds information associated with the domain name. The tree sub-divides into zones beginning at the root zone. A DNS zone consists of a collection of connected nodes authoritatively served by an authoritative name server.



Components of DNS

1. Root DNS Server : Root name servers keep track of all the authoritative name servers of each of the top level domain (TLD) name servers.
2. Top Level Domain: It provides the information regarding the presence of different zone files like
   (i) based on geographical location (country domain): us—for United Statesm, in—for India
   (ii) based on general attributes (generic domain):
   com—used by commercial organization
   Example, gmail.com
   .edu—used by educational institutes
   .org—used by non-profit organizations
   Example, ieee.org
   .gov—used by government institutions
   Example, nasa.gov
   .mil—used by military organizations
   Example, army.mil
3. Zones: The TLD and the domains under TLD are divided into smaller units with the help of delegation. The domain is divided into small units, so that it can be managed easily. These small units are zones.
4. Authoritative DNS servers checks whether authoritative name servers are located in the DNS hierarchy.



Dns Resource Records (RR)

- Every domain, whether it is a TLD, subdomain or single host have a set of resource records associated with it in the DNS distributed data base.
- Resource Records provide the mapping of host name to IP address. When a query is made to the DNS server, the host or server. who sends that query receives a response which is nothing but the resource record associated with it.
- A Resource Record (RR) is a 5 tuple that contains (Name, Time to live, class, Type, Value)
  (i) Name: It is the domain name to which this RR belongs to. More than one resource records may exists for the same domain.
  (ii) Time to live: The TTL is measured in seconds and it is a 32-bit integer.
  (iii) Class: This field contains the value 'IN' which tells whether this record is used by internet or not.
  (iv) Type: Defines type of RR address, name service, canonical name.
  (v) Value: This field can be a number, ASCII strings or any domain.

Networking Devices

Repeater

In digital communication systems, a repeater is a device that receives a digital signal on an electromagnetic or optical transmission medium and regenerates the signal. Repeaters remove the unwanted noise in an incoming signal. Unlike an analog signal, the original digital signal, even if weak or distorted, can be clearly perceived and restored. With analog transmission, signals are re strengthened with *amplifiers* which unfortunately also amplify noise as well as information.

Hub

A hub is the central part of a wheel where the spokes come together. The term is familiar to frequent fliers who travel through airport "hubs" to make connecting flights from one point to another. In data communications, a hub is a place of convergence where data arrives from one or more directions and is forwarded out in one or more other directions. A hub usually includes a switch of some kind. (And a product that is called a "switch" could usually be considered a hub as well.)

Switch

In a telecommunications network, a switch is a device that channels incoming data from any of multiple input ports to the specific output port that will take the data towards its intended destination. In the traditional circuit-switched telephone network, one or more switches are used to set up a dedicated though temporary connection or circuit for an exchange between two or more parties.

In the open systems Interconnection (OSI) communications model, a switch performs the Layer 2 or Data-link layer

function. That is, it simply looks at each packet or data unit and determines from a physical address (the "MAC address") which device a data unit is intended for and switches it out towards that device. However, in wide area networks such as the Internet, the destination address requires a look-up in a routing table by a device known as a router. Some newer switches also perform routing functions (Layer 3 or the Network layer functions in OSI) and are sometimes called IP switches. On larger networks, the trip from one switch point to another in the network is called a hop. The time a switch takes to figure out where to forward a data unit is called its latency. The price paid for having the flexibility that switches provide in a network is this latency. In the simplest networks, a switch is not required for messages that are sent and received within the network. For example, a local area network may be organized in a token ring or bus arrangement in which each possible destination inspects each message and reads any message with its address.

Bridge

A bridge is a product that connects a local area network (LAN) to another local area network that uses the same protocol (for example, Ethernet or token ring). You can envision a bridge as being a device that decides whether a message from you to someone else is going to the local area network in your building or to someone on the local area network in the building across the street. A bridge examines each message on a LAN, passing those to be within the same LAN and forwarding those known to be on the other interconnected LAN (or LANs).

In bridging networks, computer or node addresses have no specific relationship to location. For this reason, messages are sent out to every address on the network and accepted only by the intended destination node. Bridges learn which addresses are on which network and develops a *learning table* so that subsequent messages can be forwarded to the right network.

Bridging networks are generally always interconnected local area networks since broadcasting every message to all possible destinations would flood a larger network with unnecessary traffic. For this reason, router networks such as the Internet uses a scheme that assigns addresses to nodes so that a message or packet can be forwarded only in one general direction rather than forwarded in all directions. A bridge works at the data-link (physical network) level of a network, copying a data frame from one network to the next network along the communications path. A bridge is sometimes combined with a router in a product called a brouter.

Routers

Routers operate on the Network layer, which is a higher level in the OSI conceptual model. Routers use a combination of software and hardware, but it is used to route data from its source to its destination. Routers actually have a sophisticated OS that allows them to configure various connection ports. You can setup a router to route data packets from different network protocol stacks, which include TCP/IP, IPX/SPX and AppleTalk.

Routers are also used to connect remote LANs together using different WAN technologies. But, when a router has become large, the large network is divided into logical segments called subnets. This division of the network is based on the addressing scheme related to a particular subnet is kept local. The router only forwards data that is meant for the subnets on the extended network.

Routers also help to decide how to forward data packets to their destination based on the routing table. The protocols built into the router's operating system is used to identify neighboring routers and their network addresses. This allows routers to build a routing table.

Brouter

A brouter is a network bridge and a router combined in a single product. A bridge is a device that connects one local area network (LAN) to another local area network that uses the same protocol (for example, Ethernet or token ring). If a data unit on one LAN is intended for a destination on an interconnected LAN, the bridge forwards the data unit to that LAN; otherwise, it passes it along the same LAN. A bridge usually offers only one path to a given interconnected LAN. A router connects a network to one or more other networks that are usually part of a wide area network and may offer a number of paths out to destinations on those networks. A router therefore needs to have more information than a bridge about the interconnected networks. It consults a routing table for this information. Since a given outgoing data unit from a computer may be intended for an address on the local network, on an interconnected LAN, or the wide area network, it makes sense to have a single unit that examines all data units and forwards them appropriately.

Gateway

A gateway is a network point that acts as an entrance to another network. On the Internet, a node or stopping point can be either a gateway node or a host (end-point) node.

In the network for an enterprise, a computer server acting as a gateway node is often also acting as a proxy server and a firewall server. A gateway is often associated with a router, which knows where to direct a given packet of data that arrives at the gateway, and a switch, which furnishes the actual path in and out of the gateway for a given packet.

EXERCISES

Practice Problems I

Directions for questions 1 to 15: Select the correct alternative from the given choices.

1. If TCP RTT is currently 40 m/sec and the following acknowledgements come in after 26, 32 and 24 m/sec respectively. What is the new RTT estimate? $\alpha = 0.9$.
(A) 32.69 (B) 24.31 (C) 36.55 (D) 42.23

2. If a TCP connection is transferring a file of 5000 bytes. The first byte is numbered 1001. What are the sequence numbers for each segment if data is sent in five segments, each carrying 1000 bytes?
(A) 1001, 2001, 3001, 4001, 5001
(B) 1000, 2000, 3000, 4000, 5000
(C) 5000, 6000, 7000, 8000, 9000
(D) 5001, 6001, 7001, 8001, 9001

3. Which of the below statements hold good with respect to routing done by a bridge?
(i) they can route packets using IP addresses
(ii) they use data link layer addresses to do routing
(iii) the LAN route IPv4,IPv6,Apple Talk, ATM, OSI packets
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

4. Match the following:

i	Repeaters	p	connects different nodes of a LAN
ii	Hub	q	amplifies the signal between segments
iii	Switch	r	connects different LANs
iv	Bridge		

(A) i – q ii – r iii – p iv – r
(B) i – r ii – p iii – q iv – q
(C) i – q ii – p iii – p iv – r
(D) i – p ii – p iii – q iv – r

5. Match the following.

i	Retransmission timer	p	goes off when a TCP connection is idle for a long time
ii	Keep-alive timer	q	goes off if sender and receiver are waiting for each other
iii	Persistence timer	r	goes off to trigger the delivery of a segment in case acknowledgement is not received for first attempt

(A) i – r ii – q iii – p
(B) i – q ii – r iii – p
(C) i – p ii – r iii – q
(D) i – p ii – q iii – r

6. In T/TCP (Transactional TCP) what does the packet that is sent by client, consist of?
(i) SYN (ii) REQUEST (iii) FIN
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

7. Assume TCP uses 32-bit sequence numbers and sequence numbers are given to each byte that gets transmitted. If data is transmitted at 1 Gbps. What is the wraparound time for sequence numbers?
(A) 14.4 sec (B) 24.24 sec
(C) 34.36 sec (D) 44.45 sec

8. What are the disadvantages of NAT?
(i) NAT forms link between sender and receiver and then link can be broken irreparably during a connection.
(ii) NAT violates architectural model of IP.
(iii) NAT hacks source port field of TCP header which is of limited size.
(iv) NAT alleviates IP shortage.
(A) (i), (ii), (iii) (B) (ii), (iii), (iv)
(C) (i), (iii), (iv) (D) (i), (ii), (iv)

9. What is the main protocol in the transport layer?
(A) TCP (B) UDP
(C) FTP (D) Both (A) and (B)

10. Number of bytes for header in UDP segment and TCP segment are
(A) 8 bytes, 20 bytes (B) 16 bytes, 16 bytes
(C) 32-bits, 20-bits (D) None of these

11. TCP maintains a variable RTT (Round trip time), for determining the time to reach destination and receiving acknowledgement, the formula for RTT is
(A) RTT = RTT + D
(B) RTT = 4RTT
(C) RTT = αRTT + $(1 - \alpha)$ M ($\alpha = 7/8$)
(D) None of these.

12. Maximum segment size is
(A) The size of the segment without header.
(B) The size of the segment with limit.
(C) The transmission link capacity.
(D) Less than maximum transfer unit.

13. What is meant by silly window syndrome that ruins TCP performance?
(A) This occurs when sender sends data in large blocks and receiver receives in large blocks.
(B) This occurs when sender sends data in large blocks and receiver receives in or reads one byte at a time.
(C) Both (A) and (B)
(D) None of these

Common data for questions 14 and 15: A TCP segment begins with a fixed-format, 20-byte header. The header is followed by reader options. After the options, upto 65,495 bytes of data may follow.

14. Number of one bit flags available in the TCP header are
(A) 5 (B) 6
(C) 2 (D) None of these

15. Which of the flags is used for establishing connections?
(A) PSH (B) ACK (C) URG (D) SYN

Practice Problems 2

Directions for questions 1 to 15: Select the correct alternative from the given choices.

1. Which of the below TCP primitives block a port?
(i) LISTEN (ii) CONNECT (iii) RECEIVE
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

2. In the context of TCP sockets how is a symmetric DISCONNECT different from that of an asymmetric one?
(i) In symmetric DISCONNECT each direction is closed separately.
(ii) In asymmetric DISCONNECT each direction is closed separately.
(iii) In asymmetric DISCONNECT transport user can release the connection
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

3. When does RPC/UDP does not make a good combination?
(i) When the caller and callee machines are separated by small network distance.
(ii) When the parameters of the procedures are too huge in size.
(iii) When the procedure requested cannot be repeated safely as needed.
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

4. Which of the following statements below are true with reference to RTP (Real Time Transport Protocol)?
(i) It multiplexes server real time data stream into a single stream of UDP packets.
(ii) RTP has flow control, error control mechanism.
(iii) RTP has no mechanism for retransmission.
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

5. What does RTCP (real time transport control protocol) accomplish?
(i) Provides feedback on delay, jitter etc to sources.
(ii) Handles introstream synchronization.
(iii) Provides a way to name the sources.
(A) (i), (ii) (B) (ii), (iii)
(C) (i), (iii) (D) (i), (ii), (iii)

6. Which of the following are applicable to TCP?
(i) Breaks the data coming from upper layers into 64 kbyte size packets and transmits them.
(ii) Manages the time out and re-uses them.
(iii) Should reassemble the packets in correct order at receiving end.
(iv) TCP supports multicasting.
(A) (i), (ii), (iii) (B) (ii), (iii), (iv)
(C) (i), (iii), (iv) (D) (i), (ii), (iv)

7. Which of the below statements about sockets is/are true?
(i) For sender and receiver to avail TCP service sockets have to be created.
(ii) Each socket is a 16 bit number local to that host.
(iii) Sockets can involve themselves in one connection at a time.
(iv) Ports below 1024 are reserved.
(A) (i), (ii), (iii) (B) (ii), (iii), (iv)
(C) (iii), (iv), (i) (D) (i), (ii), (iv)

8. What are the functions of application layer?
(A) Mail service provides a basis for electronic mails forwarding and storage
(B) File access transfer and management
(C) Creates virtual terminal that allows us to log onto remote host
(D) All the above

9. Which of the following application uses UDP?
(A) Streaming a multimedia
(B) Client–server interaction
(C) Internet telephony
(D) All the above

10. What are the reasons for choosing an UDP by an application?
(A) No connection establishment
(B) No connection state
(C) Small packet header
(D) All the above

11. TCP uses multiple timers to do its work, the timers are
(A) Retransmission timer
(B) Persistence timer
(C) Keep alive timer
(D) All the above

12. Which of the following is supported by TCP connections?
(A) Full-duplex (B) Point-to-point
(C) Multicasting (D) Both (A) and (B)

13. TCP connection is _____ stream.
(A) Byte (B) Message
(C) Packet (D) None of these.

14. If a sender wants to indicate that, it has no data for the receiver, one of the following bits is set.
(A) PSH (B) RST
(C) FIN (D) ACK

15. If the receiver host is responding by sending a primitive SYN (SEQ = y, ACK = $x + 1$) means
(A) The receiver data sequence number is y.
(B) It has received up to $x + 1$ bytes of data.
(C) Both (A) and (B)
(D) None of these

Previous Years' Questions

1. The transport layer protocols used for real time multimedia, file transfer, DNS and email respectively are **[2013]**

(A) TCP, UDP, UDP and TCP
(B) UDP, TCP, TCP and UDP
(C) UDP, TCP, UDP and TCP
(D) TCP, UDP, TCP and UDP

2. Which one of the following socket API functions converts an unconnected active TCP socket into a passive socket? **[2014]**

(A) Connect (B) Bind
(C) Listen (D) Accept

3. Suppose two hosts use a TCP connection to transfer a large file. Which of the following statements is/are FALSE with respect to the TCP connection? **[2015]**

I. If the sequence number of a segment is m, then the sequence number of the sub sequent segment is always $m + 1$.
II. If the estimated round trip time at any given point of time is t sec, the value of the retransmission timeout is always set to greater than or equal to t sec.
III. The size of the advertised window never changes during the course of the TCP connection.
IV. The number of unacknowledged bytes at the sender is always less than or equal to the advertised window.

(A) III only (B) I and III only
(C) I and IV only (D) II and IV only

4. In one of the pairs of protocols given below, both the protocols can use multiple TCP connections between the same client and the server. Which one is that? **[2015]**

(A) HTTP, FTP (B) HTTP, TELNET
(C) FTP, SMTP (D) HTTP, SMTP

5. Assume that the bandwidth for a TCP connection is 1048560 bits/sec. Let α be the value of RTT in milliseconds (rounded off to the nearest integer) after which the TCP window scale option is needed. Let β be the maximum possible window size with window scale option. Then the values of α and β are **[2015]**

(A) 63 milliseconds, 65535×2^{14}
(B) 63 milliseconds, 65535×2^{16}
(C) 500 milliseconds, 65535×2^{14}
(D) 500 milliseconds, 65535×2^{16}

6. Consider the following statements

1. TCP connections are full duplex
2. TCP has no option for selective acknowledgement
3. TCP connections are message streams

(A) Only 1 is correct
(B) Only 1 and 3 are correct
(C) Only 2 and 3 are correct
(D) All of 1, 2 and 3 are correct

7. Which one of the following protocols is **NOT** used to resolve one form of address to another one? **[2016]**

(A) DNS (B) ARP
(C) DHCP (D) RARP

8. Which of the following is/are example(s) of stateful application layer protocols? **[2016]**

(i) HTTP (ii) FTP
(iii) TCP (iv) POP3

(A) (i) and (ii) only
(B) (ii) and (iii) only
(C) (ii) and (iv) only
(D) (iv) only

9. Identify the correct sequence in which the following packets are transmitted on the network by a host when a browser requests a webpage from a remote server, assuming that the host has just been restarted. **[2016]**

(A) **HTTP GET** request, **DNS** query, **TCP SYN**
(B) **DNS** query, **HTTP GET** request, **TCP SYN**
(C) **DNS** query, **TCP SYN, HTTP GET** request
(D) **TCP SYN**, **DNS** query, **HTTP GET** request

10. Consider a TCP client and a TCP server running on two different machines. After completing data transfer, the TCP client calls **close** to terminate the connection and a FIN segment is sent to the TCP server. Server-side TCP responds by sending an ACK, which is received by the client-side TCP. As per the TCP connection state diagram (RFC 793), in which state does the client-side TCP connection wait for the FIN from the server-side TCP? **[2017]**

(A) LAST-ACK
(B) TIME-WAIT
(C) FIN-WAIT-1
(D) FIN-WAIT-2

11. Consider socket API on a Linux machine that supports connected UDP sockets. A connected UDP socket is a UDP socket on which **connect** function has already been called. Winch of the following statements is/are CORRECT? **[2017]**

I. A connected UDP socket can be used to communicate with multiple peers simultaneously.
II. A process can successfully call **connect** function again for an already connected UDP socket.

(A) I only
(B) II only
(C) Both I and II
(D) Neither I nor II

12. Consider the following statements regarding the slow start phase of the TCP congestion control algorithm. Note that cwnd stands for the TCP congestion window

and MSS denotes the Maximum Segment Size.

(i) The cwnd increases by 2 MSS on every successful acknowledgment.

(ii) The cwnd approximately doubles on every successful acknowledgement.

(iii) The cwnd increases by 1 MSS every round trip time.

(iv) The cwnd approximately doubles every round trip time.

Which one of the following is correct? **[2018]**

(A) Only (ii) and (iii) are true

(B) Only (i) and (iii) are true

(C) Only (iv) is true

(D) Only (i) and (iv) are true

13. Consider a long-lived TCP session with an end-to-end bandwidth of 1 Gbps (= 10^9 bits-per-second). The session starts with a sequence number of 1234. The minimum time (in seconds, rounded to the closest integer) before this sequence number can be used again is _____. **[2018]**

Answer Keys

Exercises

Practice Problems 1

1. C	**2.** A	**3.** B	**4.** D	**5.** A	**6.** D	**7.** C	**8.** A	**9.** D	**10.** A
11. C	**12.** D	**13.** B	**14.** B	**15.** D					

Practice Problems 2

1. C	**2.** C	**3.** B	**4.** C	**5.** D	**6.** A	**7.** D	**8.** D	**9.** D	**10.** D
11. D	**12.** D	**13.** A	**14.** C	**15.** C					

Previous Years' Questions

1. C	**2.** C	**3.** B	**4.** A	**5.** C	**6.** A	**7.** C	**8.** C	**9.** C	**10.** D
11. B	**12.** C	**13.** 34							