Module 7

Internet And Internet Protocol Suite

Lesson

24

TCP

Version 1 ECE , IIT Kharagpur

LESSON OBJECTIVE

General

The lesson will discuss in depth a very popular transport layer protocol, i.e. the TC Protocol

Specific

The focus areas of this lesson are:

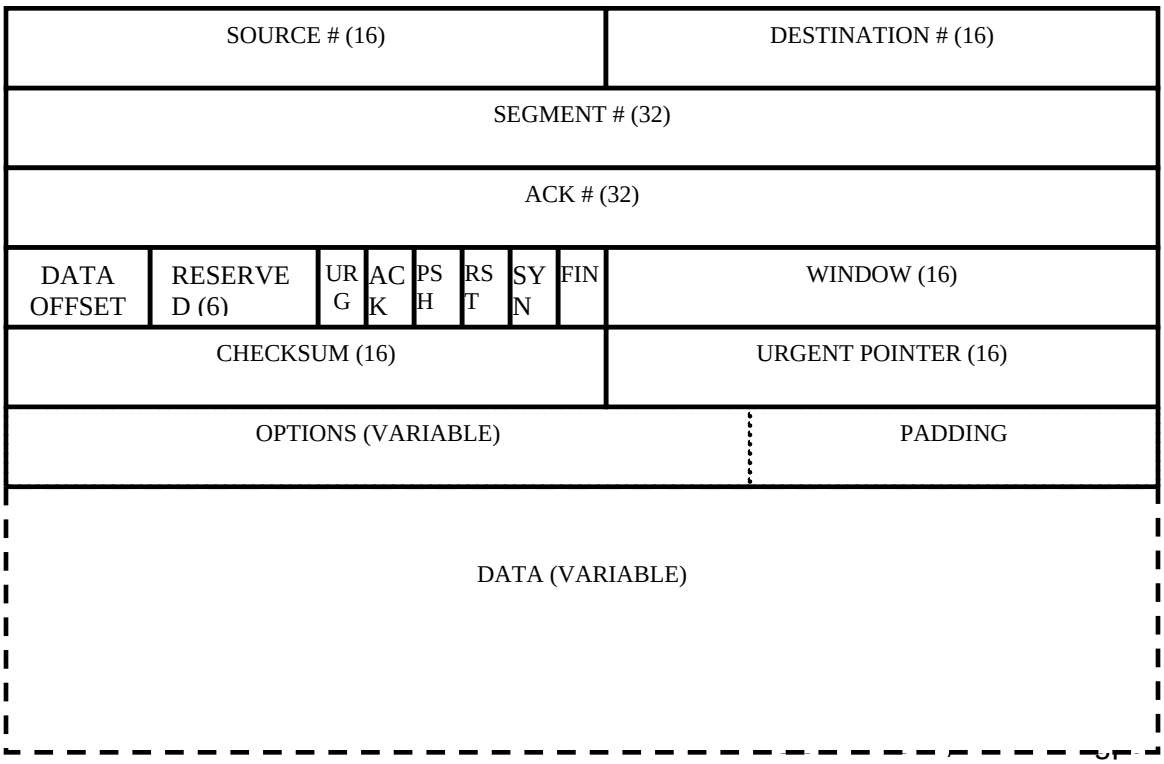
1. idea of TCP
2. the TCP header
3. the scope of TCP

7.4.1 INTRODUCTION

The sending and receiving TCP entities exchange data in the form of segments. A segment consists of a fixed 20 byte header plus an optional part followed by 0 or more data bytes. The TCP software decides how big segments should be. To limits restrict the segment size. First, two segments, including the TCP header, must fit in the 65536 byte IP payload. Segment each network has a maximum transfer unit or MTU and each segment must fit into the MTU. A segment that is too large for the network can be broken into multiple segments by a router. Each new segment gets its own IP header, so fragmentation by routers increases the total overhead. The basic protocol used by TCP entities is the sliding window protocol.

7.4.2 TCP HEADER

Presented below is the TCP header. TCP normally works in full duplex mode.



Source port number and Destination port number	These correspond to different application layer services like email, file transfer, etc. The data will start at a particular port at the transmitter and go to a particular port at the receiver. It is at the machine level, which basically works at service access points. Port addresses are not movable. The IP addresses are also not movable. Though they are normally called user address but they are actually the address of the points to which the machines are connected.
Sequence number	This is for connection-oriented service to check whether each segment is transmitted correctly.
Acknowledgement number	These are for piggybacking
Data offset	This field specifies where in the PDU the user data resides,
Flags	
URG	Is used to specify if the PDU contains any urgent data
ACK	This flag indicates whether the segment has any acknowledgement to be considered.
PSH	The PUSH flag is used to request the TCP to transmit all segments up to the current one.
RST	It is used for resetting the connection
SYN	It is used for synchronization
FIN	It is used for indicating end of data.
Window	This field is for flow control with credit allocation. It specifies the number of data octets beginning with the one indicated in ACK field, which the sender is willing to receive
Check sum field	This is the one’s complement of modulo 16 additions on all 16 bit words in the header.
Urgent pointer	This points to the next segment after the urgent data. It indicates to the receiver the length of the urgent data coming in.
Options and Padding	Some options may be available which are specified in the options field

The pseudo header contains the 32 bit IP addresses of the source and destination machines, the protocol number for TCP and the byte count for the TCP segment including the header. Including the pseudo header in the TCP checksum computation helps detect undelivered packets, but doing so violates the protocol hierarchy, since the IP addresses in it belong to the IP layer and not to the TCP layer. PAGE NUMBER 528 FIGURE 6.25 TANNENBAUM.

7.4.3 FEATURES OF TCP

TCP Connection establishment and Release

Connections are established in TCP is by means of a three-way handshake protocol. The servers passively wait for an incoming connection by executing the LISTEN and ACCEPT primitives. The other side say the client, executes a CONNECT primitive, specifying the IP address and the port to which it wants to connect, the maximum TCP segment size it is willing to accept, and some other user data. At the destination the TCP entity checks if there is any process that is doing a LISTEN on the specified port in the destination port of the incoming packet. If there is no such process the TCP entity rejects the connection request. In case there is any process that is listening, then it can either accept or reject the connection.

Although TCP connections are full duplex, we can consider them to be a pair of simplex connections. Each simplex connection is released independently of the other. To release a connection, either party can send a disconnect request TCP segment. The connection will be finally closed when this TCP segment is acknowledged by the other end. However to avoid the two army problem the TCP protocol implements a timer. If a response to the disconnect request is not received within a specified interval the connection is dropped. The other side will eventually notice that nobody seems to be listening to it anymore and will time out as well.

TCP transmission policy

Window management in TCP is not directly related to acknowledgement as in most data link protocols. For example, suppose that the receiver has a 4096-byte buffer. If the sender transmits a 2048-byte segment that is correctly received, the receiver will acknowledge the segment, however, since it has only 2048 bytes of buffer space (until the application removes some data from the buffer), it will advertise a window of 2048 starting at

the next byte expected. Now the sender transmits another 2048 bytes, which are acknowledged, but the advertised window is 0. The sender must stop transmitting, until the application process on the receiving host has removed some data from the buffer, at which TCP can advertise a larger window.

TCP congestion control

TCP tries to avoid congestion by choosing a suitable window size. The receiver can specify a window based on its buffer size. However congestion can still occur in the network. Thus we have to solve two potential problems -network capacity and receiver capacity. To do so each user maintains two separate windows: the window the receiver has granted and a second window, the congestion window. The number of bytes that may be sent is the minimum of the two windows. Thus the effective window is the minimum of what the sender thinks is all right and what the receiver thinks is all right.

7.4.4 CAPABILITIES OF TCP

Since segments can be fragmented, it is possible that a part of the transmitted segment arrives but the rest never arrives. Segments can also arrive out of order and they cannot be acknowledged because the previous have not turned up yet. Segments can also be so long in transit that the sender times out and retransmits. If a retransmitted takes a different route than the original, and is fragmented differently, bits and pieces of the original and the duplicate can arrive sporadically, requiring careful administration to achieve a reliable byte stream, and with so many networks making up the internet, it is possible that a segment may occasionally hit a congested or broken network, along its path.

TCP has to deal with these problems and solve them in an efficient way.

Objective Questions

24.01 A segment consists of a fixed __ byte header plus an optional part followed by ____ bytes.

24.02 The source and destination addresses are ____ bits long.

24.03 Segment number is for connection-oriented service to check whether each segment is transmitted correctly. (*True/ False*)

24.04 There are ___ Flags in TCP header.

24.05 The servers passively wait for an incoming connection by executing the _____ and _____ primitives.

24.06 TCP connections are full-duplex. (*True/False*)

Subjective Questions

24.11 Describe the TCP header.

24.12 Discuss TCP connection establishment and release.

24.13 Give and idea of the TCP transmission policy

24.14 How can TCP be used to deal with network or internet congestion?

Level 2 Questions

24.21