Solved Question Paper

December 2015

3.b)Write short notes on the following Application Layer protocols : (i) FTP (ii) TFTP (iii) TELNET [10 marks]

i. FTP

It is one of the oldest application layer protocols. It was created before the TCP/IP protocol suite. It is used for copying files between computer systems. It is hardware dependent, so its services can function anywhere.

It is a client-server protocol where an FTP client accesses and FTP server. FTP client authenticates itself with a username and a password to establish an FTP session. Successful authentication results in the establishment of a FTP session, where an FTP client can download ("get") and upload ("put") files and file lists. While transferring files, FTP respects the ownership and access privileges of files.

Most hosts have a utility program, generally also called FTP, which provides an interactive command line interface to run an FTP session. FTP clients can also be integrated in other applications, such as web browsers. Anonymous FTP is a special form of file transfer service that allows public access to files at an FTP server. An FTP client can establish an anonymous FTP session by supplying the user name "anonymous" and an arbitrary password.

FTP uses TCP as its transport protocol. This ensures a reliable transfer of transmitted data. Two connections are established for each FTP session, namely control connection and data connection. The control connection is used for commands from the client and messages from the server. The data connection is used for the transport of files. FTP uses TCP port 21 for control connection, and TCP port 20 for data connection, and an FTP client selects available ephemeral port numbers. The control connection is established at the beginning of the FTP session and stays up for the entire lifetime of the session. Control connection is used by the FTP client for authentication, for setting various session parameters, and for commands to download or upload files. Data connection is opened and closed for each transfer of a file or file list. As soon as a file or a file list has been transferred, the data connection is closed. If there is another file transfer, the data connection is re-opened.

By default, the data connection is established upon request by the FTP server. The FTP client starts a TCP server that waits for a connection on an ephemeral port, and sends the port number of this port on the control connection to the FTP server. After the FTP server receives the port, it can request a data connection to the FTP client.

A security concern with FTP is that the username and password provided in the beginning of an FTP session are not encrypted. Therefore, anyone with the ability to capture traffic from the FTP client can obtain the username and password used by an FTP client.

FTP client sends commands to the FTP server, and the FTP server responds with a three-digit response code and an explaining text message. The commands from the client as well as the responses from the FTP server are transmitted as ASCII characters 13.

The end of a client command and the end of a server response is represented by an end of line sequence, which consists of the ASCII special characters Carriage Return (ASCII 10) followed by line feed (ASCII 13). When the TCP connection to TCP port 21 of the FTP server is established, the FTP server sends a message that it is ready to interact on a new FTP session. Then, the user supplies a user name and a password. If the authentication is successful, the user sends the IP address and port number of its ephemeral port for the data connection.

The IP address and port number is sent in dotted-decimal notation, where the first four numbers specify the IP address and the last two numbers specify the port number.

By default, files are transmitted as text files using ASCII characters. The FTP client can change this default so that files are transmitted bit-by-bit without any interpretation. While the file transfer has been completed, the FTP server sends a message to the FTP client.

At this 13 ASCII (American Standard code for Information Interchange) is an encoding format for textual data, which represents an alphanumeric character or special character by seven bits.

Application layer protocols transmit each character in a byte (octet) with the most significant bit set to zero.

The FTP client can download or upload another files, or end the FTP sessions by issuing the command “Quit”.

ii. Trivial File Transfer Protocol (TFTP)

TFTP stands for Trivial File Transfer Protocol.

The TFTP is a minimal protocol for transferring files without authentication. In TFTP, there is no separation of control information and data as in FTP. Therefore TFTP must not be used on computer where sensitive/confidential information is stored. TFTP is frequently used by devices without permanent storage for copying an initial memory image from a remote server when the devices are powered on. Due to the lack of security features, the use of TFTP is generally restricted.

TFTP uses the unreliable transport protocol UDP(User Datagram Protocol) for data transport, whereas FTP uses TCP(Transmission Control Protocol). Each TFTP message is carried in a separate UDP datagram. The first two bytes of a TFTP message specify the type of a message, which can be a request to download a file, request to upload a file, a data message, or an acknowledgement or error message. A TFTP session is initiated when a TFTP client sends a request to upload or download a file from an ephemeral UDP port to the (well-known) UDP port 69 of a TFTP server. When the request is received the TFTP server picks an ephemeral UDP port of its own and uses this port to communicate with the TFTP client. Thus, both client and server communicate using ephemeral ports.

Since UDP does not recover lost or corrupted data, TFTP is responsible for maintaining the integrity of the data exchange. TFTP transfers data in blocks of 512 bytes. Each block is assigned a 2-byte long sequence number and is transmitted in a separate UDP datagram. A block must be acknowledged before the next block can be sent. When an acknowledgment is not received before a timer expires, the block is retransmitted.