

Advanced Computer Network (4350706)

Unit-5 Application Layer Protocols

Prepared By :
Mr. Chetan C. Kamani
Lecturer, Computer Engineering Department
Government Polytechnic, Jamnagar

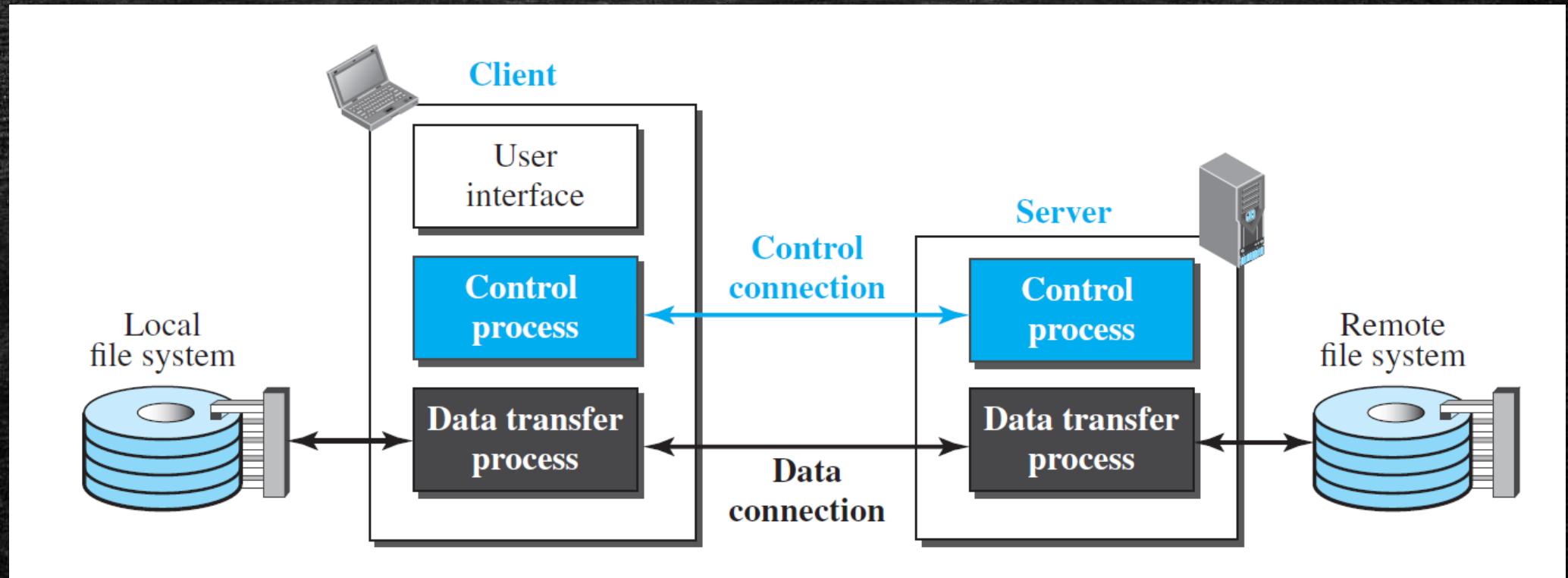
Topics to be discussed

- FTP (File Transfer Protocol)
 - Two Connections
 - Control Connection
 - Data Connection
 - Security for FTP

FTP

- File Transfer Protocol (FTP) is the standard protocol provided by TCP/IP for copying a file from one host to another.
- Although transferring files from one system to another seems simple and straightforward, some problems must be dealt with first. For example,
 - Two systems may use different file name conventions.
 - Two systems may have different ways to represent data.
 - Two systems may have different directory structures.
- All of these problems have been solved by FTP in a very simple and elegant approach.
- Although we can transfer files using HTTP, FTP is a better choice to transfer large files or to transfer files using different formats.

FTP



Basic Model of FTP

FTP

- The client has three components: the user interface, the client control process, and the client data transfer process.
- The server has two components: the server control process and the server data transfer process.
- The control connection is made between the control processes.
- The data connection is made between the data transfer processes.

FTP

- Separation of commands and data transfer makes FTP more efficient.
- The control connection uses very simple rules of communication. We need to transfer only a line of command or a line of response at a time.
- The data connection, on the other hand, needs more complex rules due to the variety of data types transferred.

Two Connections

- The two connections in FTP have different lifetimes.
- The control connection remains connected during the entire interactive FTP session.
- The data connection is opened and then closed for each file transfer activity. It opens each time commands that involve transferring files are used, and it closes when the file is transferred.

Two Connections

- In other words, when a user starts an FTP session, the control connection opens.
- While the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.
- FTP uses two well-known TCP ports:
 - *port 21 is used for the control connection,*
 - *and port 20 is used for the data connection.*

Control Connection

- For control communication, FTP uses the same approach as TELNET. It uses the NVT ASCII character set as used by TELNET.
- Communication is achieved through commands and responses. This simple method is adequate for the control connection because we send one command (or response) at a time. Each line is terminated with a two-character (carriage return and line feed) end-of-line token.

Control Connection

- During this control connection, commands are sent from the client to the server and responses are sent from the server to the client.
- Commands, which are sent from the FTP client control process, are in the form of ASCII uppercase, which may or may not be followed by an argument.

Some FTP commands

<i>Command</i>	<i>Argument(s)</i>	<i>Description</i>
ABOR		Abort the previous command
CDUP		Change to parent directory
CWD	Directory name	Change to another directory
DELE	File name	Delete a file
LIST	Directory name	List subdirectories or files
MKD	Directory name	Create a new directory
PASS	User password	Password
PASV		Server chooses a port
PORT	Port identifier	Client chooses a port
PWD		Display name of current directory
QUIT		Log out of the system
RETR	File name(s)	Retrieve files; files are transferred from server to client
RMD	Directory name	Delete a directory
RNFR	File name (old)	Identify a file to be renamed
RNTO	File name (new)	Rename the file
STOR	File name(s)	Store files; file(s) are transferred from client to server
STRU	F , R , or P	Define data organization (F : file, R : record, or P : page)
TYPE	A , E , or I	Default file type (A : ASCII, E : EBCDIC, I : image)
USER	User ID	User information
MODE	S , B , or C	Define transmission mode (S : stream, B : block, or C : compressed)

Control Connection

- Every FTP command generates at least one response.
- A response has two parts: a three-digit number followed by text.
- The numeric part defines the code; the text part defines needed parameters or further explanations.
- The first digit defines the status of the command. The second digit defines the area in which the status applies. The third digit provides additional information.

Some responses in FTP

<i>Code</i>	<i>Description</i>	<i>Code</i>	<i>Description</i>
125	Data connection open	250	Request file action OK
150	File status OK	331	User name OK; password is needed
200	Command OK	425	Cannot open data connection
220	Service ready	450	File action not taken; file not available
221	Service closing	452	Action aborted; insufficient storage
225	Data connection open	500	Syntax error; unrecognized command
226	Closing data connection	501	Syntax error in parameters or arguments
230	User login OK	530	User not logged in

Data Connection

- The data connection uses the well-known port 20 at the server site.
- However, the creation of a data connection is different from the control connection. The following shows the steps:
 1. The client, not the server, issues a passive open using an ephemeral port. This must be done by the client because it is the client that issues the commands for transferring files.
 2. Using the PORT command, the client sends this port number to the server.
 3. The server receives the port number and issues an active open using the well-known port 20 and the received ephemeral port number.

Data Connection

Communication over Data Connection

- The purpose and implementation of the data connection are different from those of the control connection.
- We want to transfer files through the data connection. The client must define the type of file to be transferred, the structure of the data, and the transmission mode.
- Before sending the file through the data connection, we prepare for transmission through the control connection. The heterogeneity problem is resolved by defining three attributes of communication: file type, data structure, and transmission mode.

Data Connection

- **File Type**

- FTP can transfer one of the following file types across the data connection: ASCII file, EBCDIC file, or image file.

Data Connection

- **Data Structure**

- FTP can transfer a file across the data connection using one of the following interpretations of the structure of the data: file structure, record structure, or page structure.
- The file structure format (used by default) has no structure. It is a continuous stream of bytes.
- In the record structure, the file is divided into records. This can be used only with text files.
- In the page structure, the file is divided into pages, with each page having a page number and a page header. The pages can be stored and accessed randomly or sequentially.

Data Connection

- Transmission Mode
 - FTP can transfer a file across the data connection using one of the following three transmission modes: stream mode, block mode, or compressed mode.
 - The stream mode is the default mode; data are delivered from FTP to TCP as a continuous stream of bytes.
 - In the block mode, data can be delivered from FTP to TCP in blocks. In this case, 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.
 - The compressed mode is used to transfer big files. As we know that, due to the size limit we can not transfer big files on the internet, so the compressed mode is used to decrease the size of the file into small and send it on the internet.

Data Connection

- File Transfer
 - File transfer occurs over the data connection under the control of the commands sent over the control connection.
 - However, we should remember that file transfer in FTP means one of three things: retrieving a file (server to client), storing a file (client to server), and directory listing (server to client).

Security for FTP

- The FTP protocol was designed when security was not a big issue.
- Although FTP requires a password, the password is sent in plaintext (unencrypted), which means it can be intercepted and used by an attacker.
- The data transfer connection also transfers data in plaintext, which is insecure. To be secure, one can add a Secure Socket Layer between the FTP application layer and the TCP layer. In this case FTP is called SSL-FTP.

FTP Examples

References

- Data Communication and Networking, Fifth Edition, BEHROUZA. FOROUZAN
- Data Communication and Networking, Fourth Edition, BEHROUZA. FOROUZA
- <https://www.geeksforgeeks.org/introduction-of-classful-ip-addressing/>

Thank You