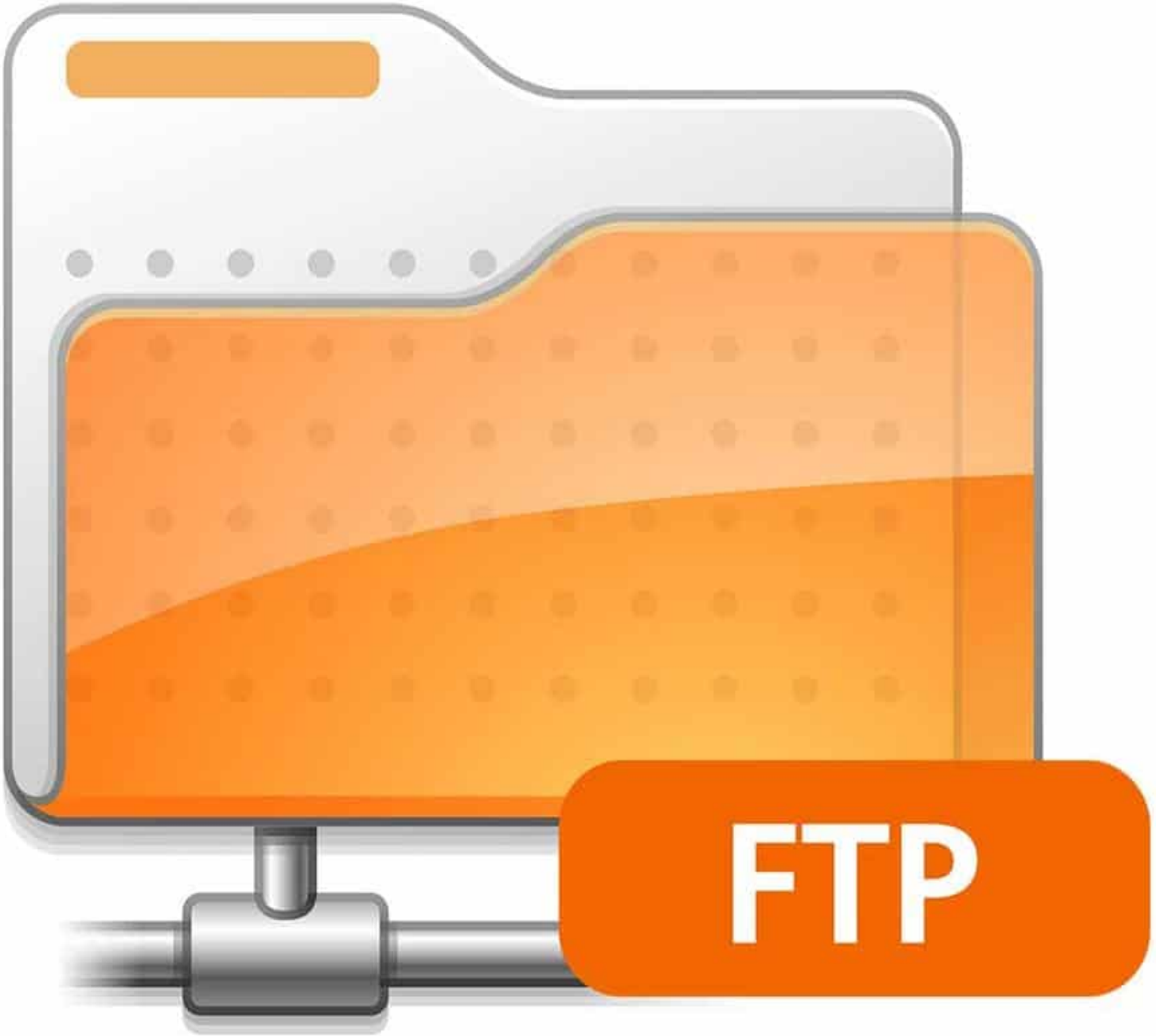
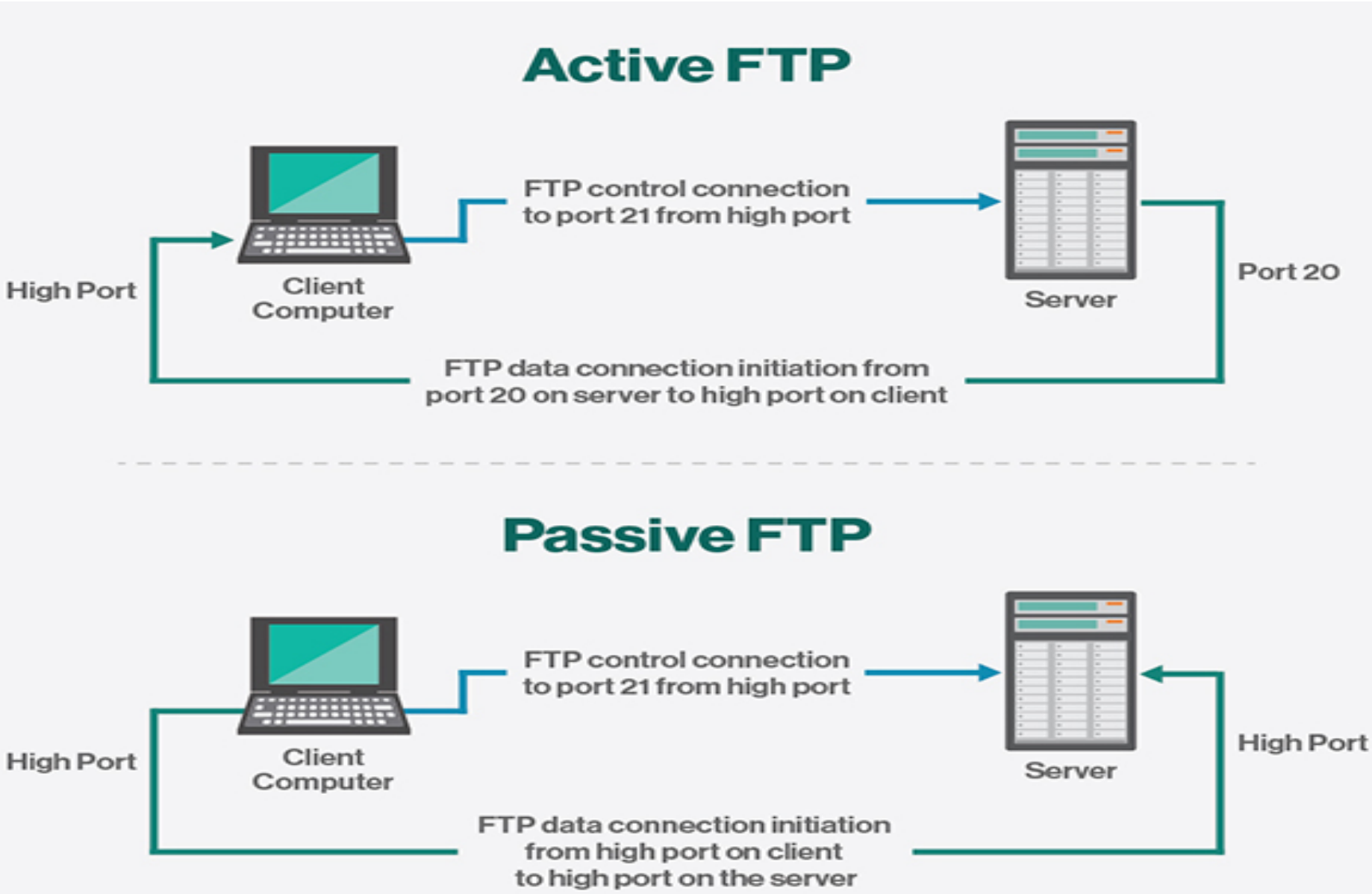

FTP

File Transfer Protocol

- File Transfer Protocol (FTP) :

  File Transfer Protocol (FTP) is a standard Internet protocol for transmitting files between computers on the Internet over TCP/IP connections.
  FTP is a client-server protocol that relies on two communications channels between client and server: a command channel for controlling the conversation and a data channel for transmitting file content. Clients initiate conversations with servers by requesting to download a file. Using FTP, a client can upload, download, delete, rename, move and copy files on a server. A user typically needs to log on to the FTP server, although some servers make some or all of their content available without login, also known as anonymous FTP.


Active FTP
FTP control connection
to port 21 from high port
High Port
Client
Computer
Server
Port 20
FTP data connection initiation from
port 20 on server to high port on client
Passive FTP
FTP control connection
to port 21 from high port
High Port
Client
Computer
Server
High Port
FTP data connection initiation
from high port on client
to high port on the server
TechTarget
DESIGN: TECHTARGET/CHRISTOPHER SEERO

History of FTP server

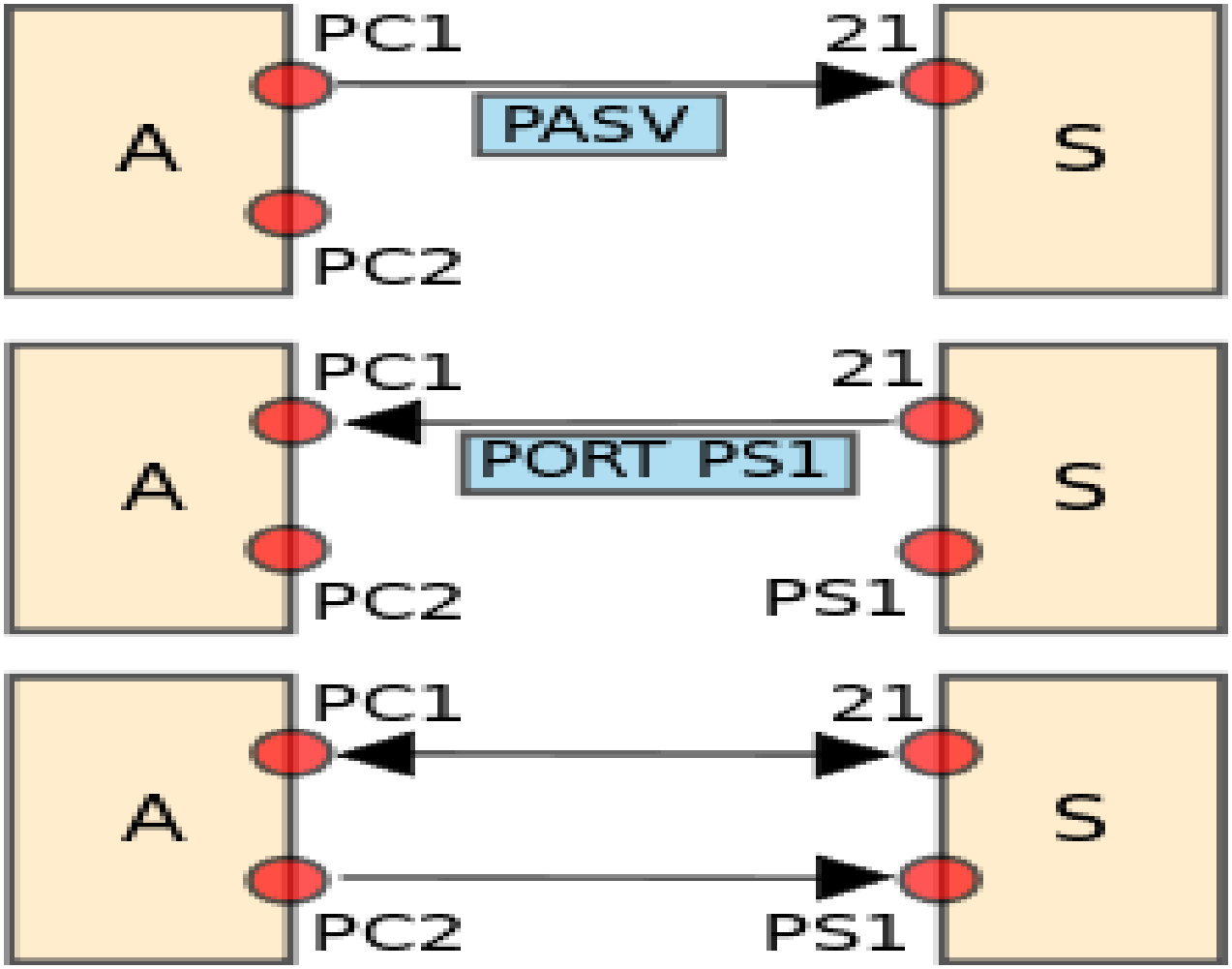
- The original specification for the File Transfer Protocol was written by Abhay Bhushan and published as RFC 114 on 16 April 1971. Until 1980, FTP ran on NCP, the predecessor of TCP/IP. The protocol was later replaced by a TCP/IP version, RFC 765 (June 1980) and RFC 959 (October 1985), the current specification. Several proposed standards amend RFC 959, for example RFC 1579 (February 1994) enables Firewall-Friendly FTP (passive mode), RFC 2228 (June 1997) proposes security extensions, RFC 2428 (September 1998) adds support for IPv6 and defines a new type of passive mode.

Communication and data transfer

FTP may run in *active* or *passive* mode, which determines how the data connection is established. In both cases, the client creates a TCP control connection from a random, usually an unprivileged, port N to the FTP server command port 21.

- In active mode, the client starts listening for incoming data connections from the server on port M. It sends the FTP command PORT M to inform the server on which port it is listening. The server then initiates a data channel to the client from its port 20, the FTP server data port.
- In situations where the client is behind a firewall and unable to accept incoming TCP connections, *passive mode* may be used. In this mode, the client uses the control connection to send a PASV command to the server and then receives a server IP address and server port number from the server, which the client then uses to open a data connection from an arbitrary client port to the server IP address and server port number received.

Illustration of starting a passive connection using port 21



Data Representation

While transferring data over the network, four data representations can be used:

- **ASCII mode**: Used for text. Data is converted, if needed, from the sending host's character representation to "8-bit ASCII" before transmission, and (again, if necessary) to the receiving host's character representation. As a consequence, this mode is inappropriate for files that contain data other than plain text.
- **Image mode (commonly called Binary mode):** The sending machine sends each file byte by byte, and the recipient stores the bytestream as it receives it. (Image mode support has been recommended for all implementations of FTP).
- **EBCDIC mode:** Used for plain text between hosts using the EBCDIC character set.
- **Local mode**: Allows two computers with identical setups to send data in a proprietary format without the need to convert it to ASCII.

Data transfer

Data transfer can be done in any of three modes:

- Stream mode: Data is sent as a continuous stream, relieving FTP from doing any processing. Rather, all processing is left up to TCP. No End-of-file indicator is needed, unless the data is divided into records.
- Block mode: FTP breaks the data into several blocks (block header, byte count, and data field) and then passes it on to TCP.
- Compressed mode: Data is compressed using a simple algorithm (usually run-length encoding).

FTP Protocol Design

FTP Protocol Design

FTP commands, responses

- Sample commands:
  - » Sent as ASCII text on control socket

```
USER <username>
PASS <password>
LIST
    Return list of file in current directory
RETR <filename>
    Retrieves (gets) file
STOR <filename>
    Stores (puts) file onto remote host
```

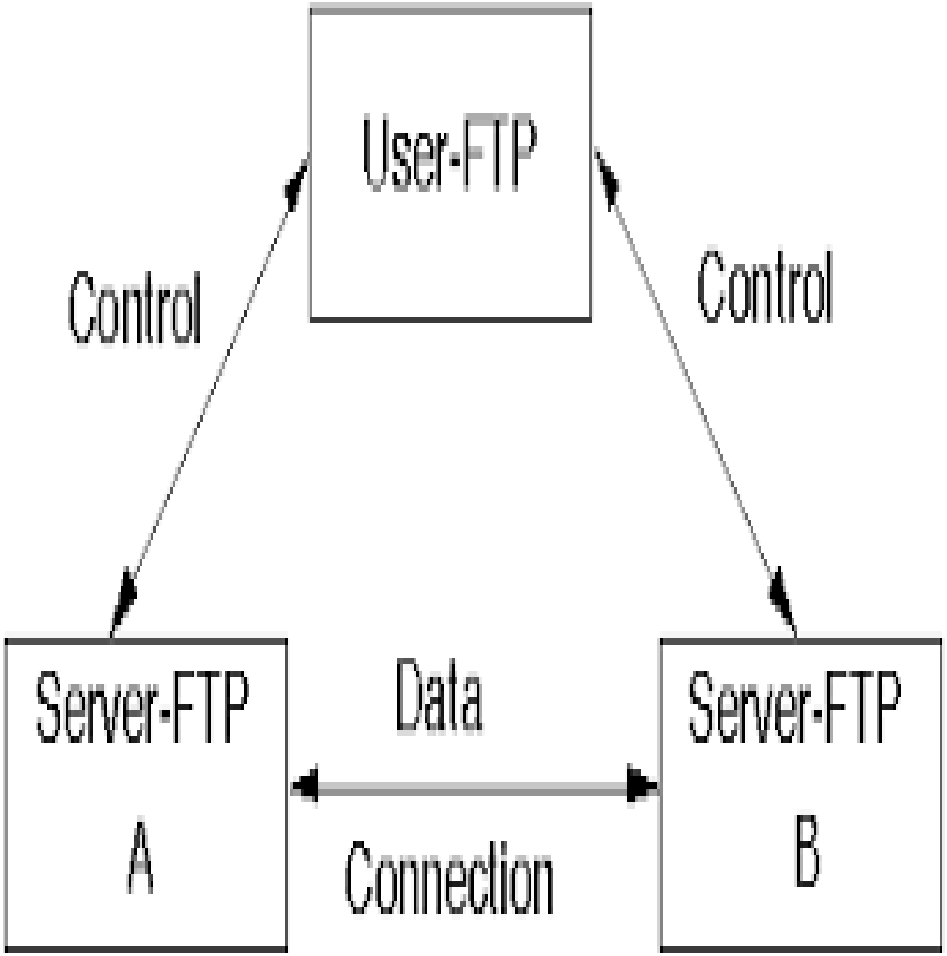
- Sample return codes
  - » Status code and phrase (as in HTTP)

```
331 Username OK, password required
125 data connection already open;
   transfer starting
425 Can't open data connection
452 Error writing file
```

Data/Representation Types

- Important ones ASCII and Image (binary)
- FTP command to change data type is "TYPE"
  - parameter is either A or I
- Text files normally stored as character string
  - 8-bit ASCII on most machines
- Image transfer is bit-by-bit replication of file from the source machine on the target machine
  - that is why in most ftp clients the corresponding command is called "binary"

FTP Data Connection



FTP Commands and Replies...

- Some of the more common commands are:
  - **USER *username*** *: Used to send the user identification to server.*
  - **PASS *password* :** *Used to send the user password to the server.*
  - **LIST :** Used to ask the server to send back a list of all the files in the current remote directory. The list of files is sent over a (new and non-persistent) data TCP connection and not over the control TCP connection.
  - **RETR *filename* :** *Used to retrieve (i.e., get) a file from the current directory of the remote host.*
  - **STOR *filename* :** *Used to store (i.e., put) a file into the current directory of the remote host.*

Common FTP Commands

```
ftp> ?
Commands may be abbreviated.  Commands are:

!               debug           mdir            sendport        site
$               dir             mget            put             size
account         disconnect      mkdir           pwd             status
append          exit            mls             quit            struct
ascii           form            mode            quote           system
bell            get             modtime         recv            sunique
binary          glob            mput            reget           tenex
bye             hash            newer           rstatus         tick
case            help            nmap            rhelp           trace
cd              idle            nlist           rename          type
cdup            image           ntrans          reset           user
chmod           lcd             open            restart         umask
close           ls              prompt          rmdir           verbose
cr              macdef          passive         runique         ?
delete          mdelete         proxy           send
ftp> help mkdir
mkdir           make directory on the remote machine
ftp> ftp.piglipuw.com
```

Common FTP Commands

- **?:** *To request help or information about the FTP commands.*
- **Ascii:** *To set the mode of file transfer to ASCII (this is the default and transmits seven bits per character).*
- **Binary:** T*o set the mode of file transfer to binary (the binary mode transmits all eight bits per byte and thus provides less chance of a transmission error and must be used to transmit files other than ASCII files).*
- **Bye:** T*o exit the FTP environment (same as quit).*

Common FTP Commands

- **Cd:** *To change directory on the remote machine.*
- **Close:** T*o terminate a connection with another computer.*
- **Close Brubeck:** Closes the current FTP connection with Brubeck, but still leaves you within the FTP environment.
- **Delete:** *To delete (remove) a file in the current remote directory (same as rm in UNIX).*

Common FTP Commands

- **Get:** T*o copy one file from the remote machine to the local machine.*

- **get ABC DEF:** Copies file ABC in the current remote directory to (or on top of) a file named DEF in your current local directory.
- **get ABC:** Copies file ABC in the current remote directory to (or on top of) a file with the same name, ABC, in your current local directory.

Common FTP Commands

- **Help:** *To request a list of all available FTP commands.*
- **Lcd:** *To change directory on your local machine (same as UNIX cd).*
- **Ls:** *To list the names of the files in the current remote directory.*
- **Mkdir:** *To make a new directory within the current remote directory.*

Common FTP Commands

- **mget:** *To copy multiple files from the remote machine to the local machine; you are prompted for a y/n answer before transferring each file.*
- **mget *:** Copies all the files in the current remote directory to your current local directory, using the same filenames. Notice the use of the wild card character, *.
- **mput:** *To copy multiple files from the local machine to the remote machine; you are prompted for a y/n answer before transferring each file.*

Common FTP Commands

- **open:** *To open a connection with another computer.*
- **open brubeck:** Opens a new FTP connection with brubeck; you must enter a username and password for a brubeck account (unless it is to be an anonymous connection).
- **Put:** *To copy one file from the local machine to the remote machine.*

Common FTP Commands

- **Pwd:** *To find out the pathname of the current directory on the remote machine.*
- **Quit:** *To exit the FTP environment (same as bye).*
- **Rmdir:** *To remove (delete) a directory in the current remote directory.*

Features of FTP

File Transfer Protocol

- FTP (File Transfer Protocol) is the simplest and most secure way to exchange files over the Internet.
- Transferring files from a client computer to a server computer is called **"uploading"** and transferring from a server to a client is **"downloading".**
- To access an FTP server, users must be able to connect to the Internet or an intranet (via a modem or local area network) with an FTP client program.

FTP pros and cons

- **Advantages:**
  - Simple implementation
  - Universal application
  - Wide used and standartized

- **Disadvantages:**
  - Clear text passwords, unencrypted data.
  - Multiple TCP/IP connections are used = > Firewalls problems.
  - Hard to filter active mode FTP traffic on the client side by using a firewall.
  - It is possible to abuse the protocol's built-in proxy features to tell a server to send data to an arbitrary port of a third computer; see FXP.
  - High latency.
  - No integrity check on the receiver side.
  - No date/timestamp attribute transfer.

Submitted By: Mrs. Dimple Sharma (Assistant Prof. in Computer Sc. Department.)