

Networking Applications

Dr. Ayman A. Abdel-Hamid

College of Computing and Information Technology

Arab Academy for Science & Technology and
Maritime Transport

File Transfer Protocol (FTP)

Outline

- Introduction
- FTP Model
- Connections
- Setup of data connections
- File Transfer

FTP Objectives

- Promote sharing of files
- Encourage indirect use of remote computers
- Shield user from variations in file storage among hosts
 - Different ways to represent text and data
 - Different directory structure
- Transfer data reliably and efficiently

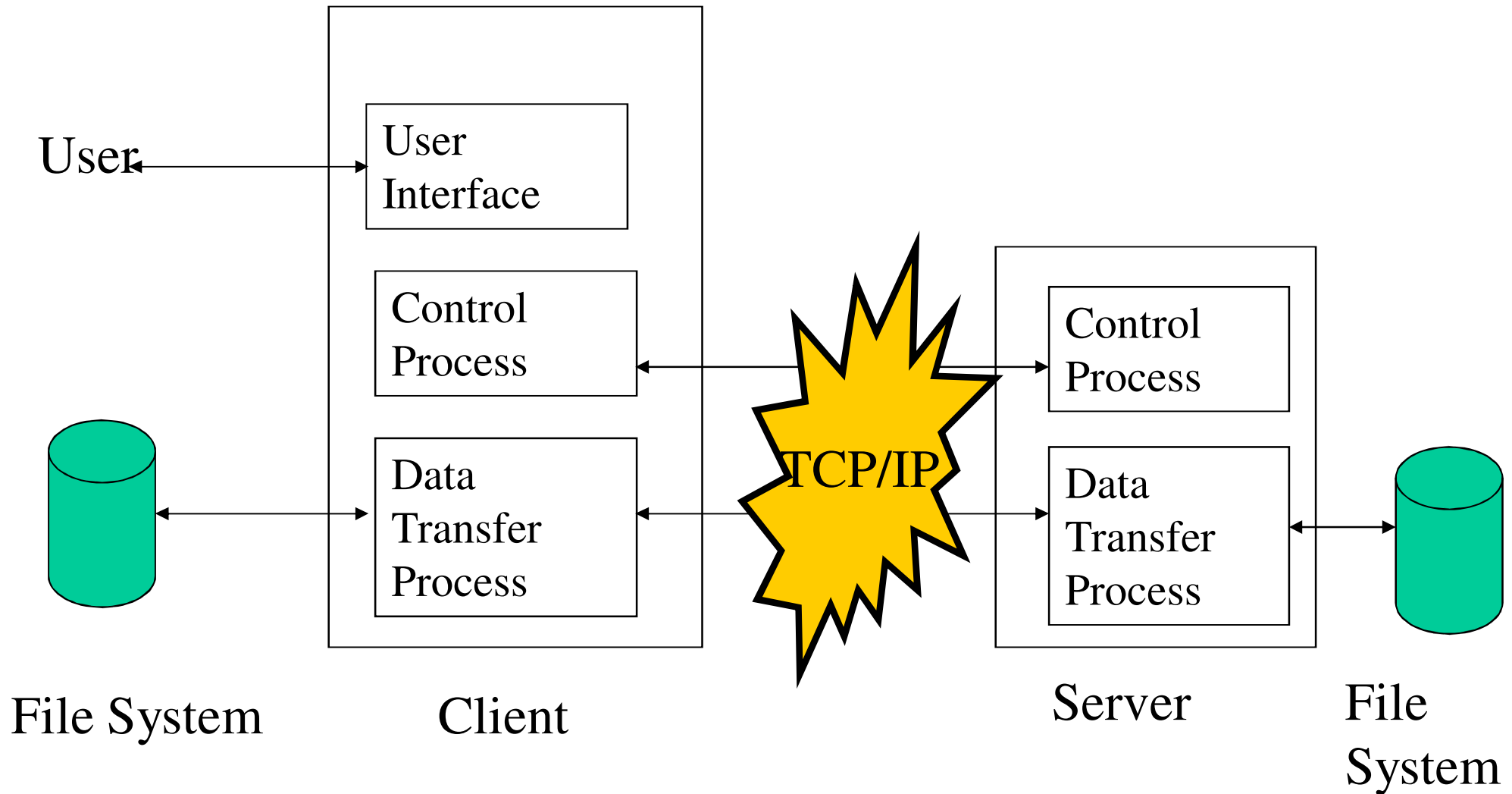
Port Numbers

- Port numbers are 16-bit integers (0 → 65,535)
 - Servers use *well know ports*, 0-1023 are privileged
 - Clients use *ephemeral* (short-lived) ports
- *Internet Assigned Numbers Authority* (IANA) maintains a list of port number assignment.
 - **Well-known ports** (0-1023) → controlled and assigned by IANA
 - **Registered ports** (1024-49151) → IANA registers and lists use of ports as a convenience (49151 is $\frac{3}{4}$ of 65536)
 - **Dynamic ports** (49152-65535) → ephemeral ports
- For well-known port numbers, see /etc/services on a UNIX or Linux machine

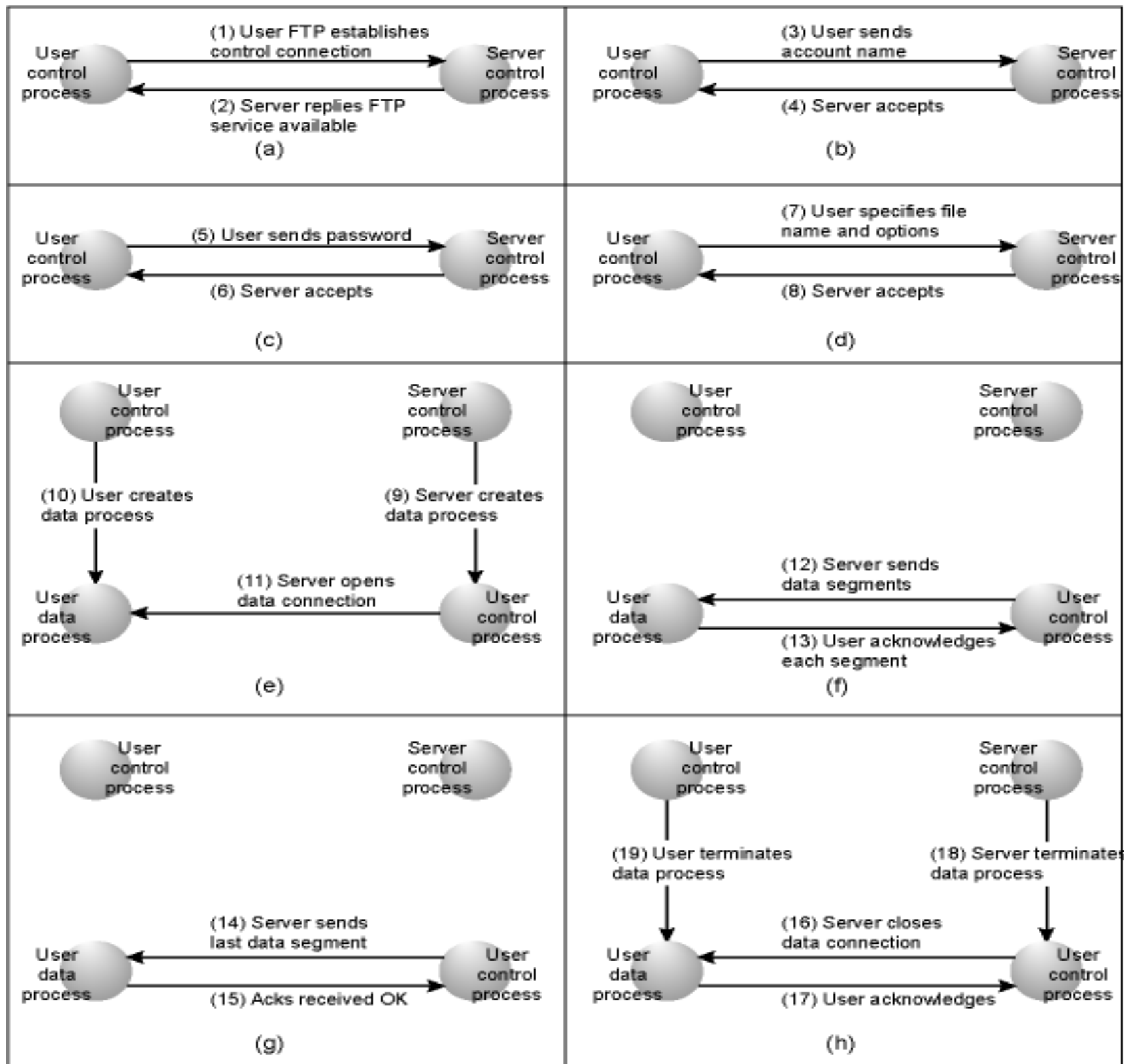
FTP Model ^{1/2}

- Two connections between client and server
 - data transfer connection
 - Control connection (commands and responses)
 - Server uses two well-known TCP ports: 21 for control connection and 20 for data connection
- Client Components
 - User Interface, client control process, and client data transfer process
- Server Components
 - Server control process and server data transfer process

FTP Model _{2/2}



FTP Transfer Overview



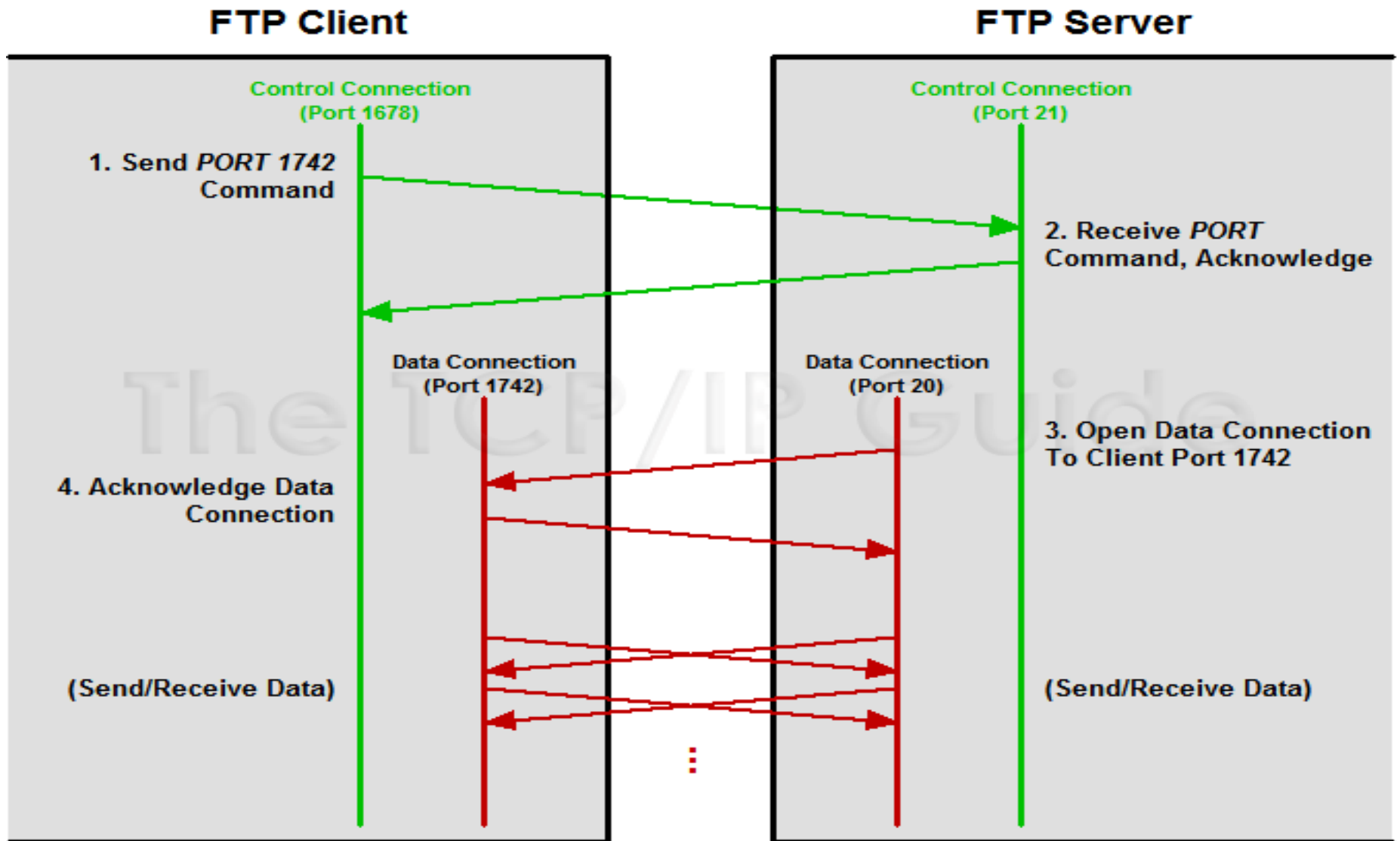
Control Connection

- Control connection maintained during entire FTP session
 - Uses ASCII to communicate commands and responses
 - Send a command as a short line (terminated by carriage return and line feed)
 - Receive response as a short line
 - The objective is to minimize delay
- An FTP session always initiated by the client, however either the client or the server may be the sender of data

Setup of a Data Connection ^{1/4}

- FTP server listening for control connections on port 21
- FTP client uses an ephemeral port for its control connection (for example 1678)
- It connects to the FTP server at port 21
- To setup an active data connection*
 - Client listens on an ephemeral port for data connections from server (for example 1742)
 - Client sends to server port number 1742 on which it is ready to accept data connections
 - The server (using port 20) connects to the client on port 1742

Setup of a Data Connection ^{2/4}



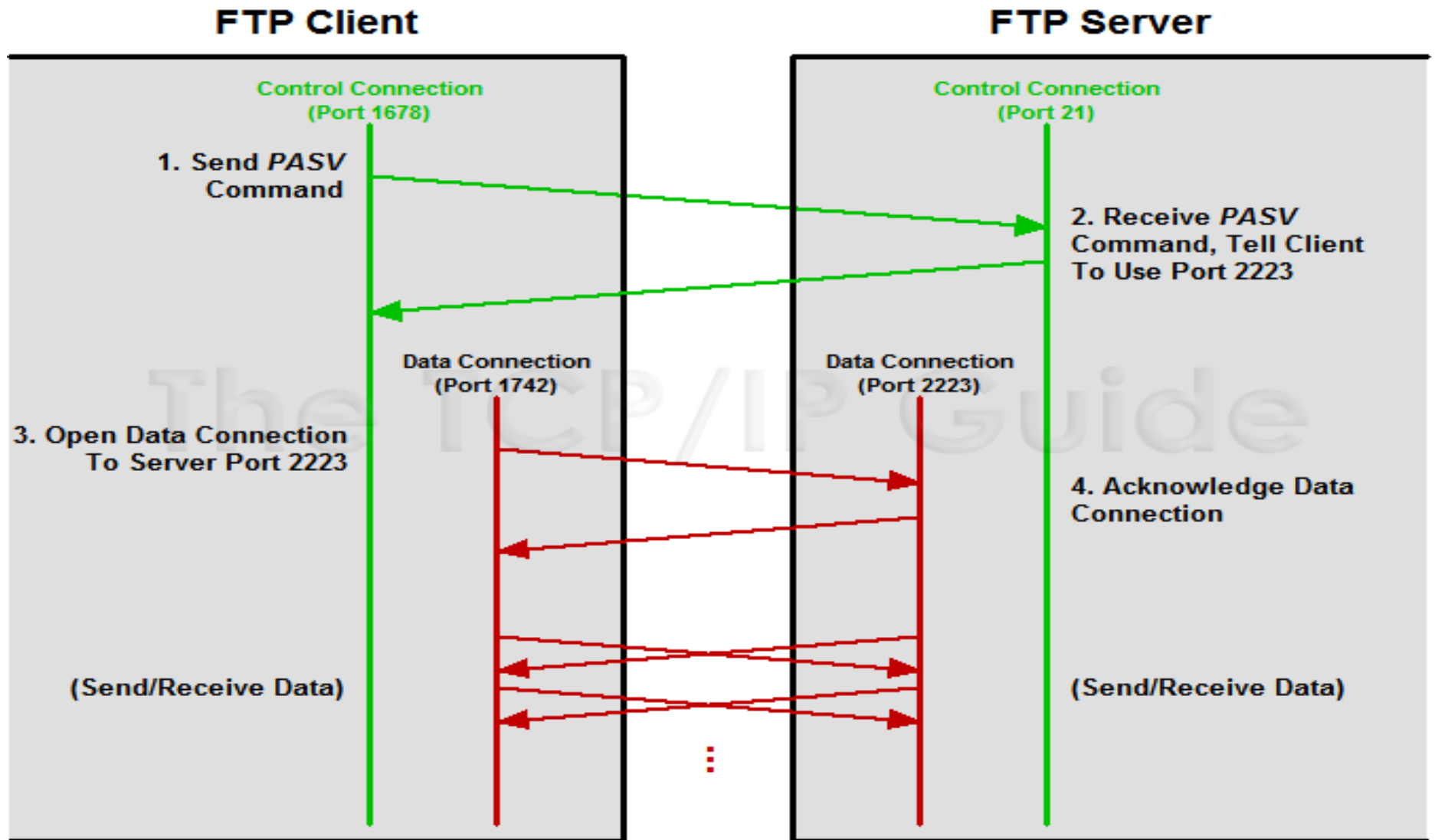
FTP active data connection. Figure from www.tcpipguide.com

Setup of a Data Connection ^{3/4}

- *To setup a passive data connection*

- Client tells server to be passive
- Accept a data connection from client
- Server replies providing IP address and port it is listening to
- Server listens on (IP, port) pair waiting for data connections
- The client chooses an ephemeral port to connect to server at (IP, Port)
- Data transfer can occur

Setup of a Data Connection 4/4



FTP passive data connection. Figure from www.tcpipguide.com

Data Connection

- Data connection opened and closed for each file transfer
 - Opened when data are ready for transfer
 - Closed when not needed
- The objective is to maximize throughput
- To deal with heterogeneity, need to define three attributes for communication (transmitted through the control connection)
 - File type
 - Data structure
 - Transmission mode

Data Connection – File Type

- ASCII File

- Default for transferring text files
- Sender transforms the file from its own representation to ASCII
- Receiver transforms ASCII characters into its own representation

- EBCDIC File (used on IBM computers)

- Image File

- Default for transferring binary files
- File sent as a continuous stream of bits without any interpretation or encoding

Data Connection – Data Structure

- File Structure

- Default. File is a continuous stream of bytes

- Record Structure

- File divided into records
- Only with text files

- Page Structure

- File divided into pages
- Each page has a page number and page header
- Pages can be stored or accessed randomly or sequentially

Data Connection – Transmission Mode ^{1/2}

•Stream Mode

- Default
- Data delivered from FTP to TCP as a continuous stream of bytes
- TCP responsible for chopping data into segments of suitable size

•Block Mode

- Data delivered from FTP to TCP in blocks.
- Each block preceded by a 3-byte header
- First byte is a block descriptor; next 2 bytes define the size of the blocks in bytes

Data Connection – Transmission Mode ^{2/2}

•Compressed Mode

- Data can be compressed if file is large in size
- Compression method normally used is run-length encoding
- Consecutive appearances of a data unit are replaced by one occurrence and the number of repetitions
 - ✓ Spaces in a text file
 - ✓ Null characters in a binary file

File Transfer

- A file is to be copied from server to client (*get*)
 - Retrieving a file
- File copied from client to server (*put*)
 - Storing a file
- List of directory names sent from server to client (*ls*)
 - FTP treats a list of directory or file names as a file
 - It is sent over the data connection

User Interface

- Most operating systems provide a user interface to access FTP services
- Lots of tools to provide GUI for FTP commands
- Popular FTP commands
 - put (mput)
 - get (mget)
 - cd
 - lcd
 - bye

Anonymous FTP

- To use FTP, an user name and password are needed on the remote server
- Some sites have files available for *public access*
- A user can use “*anonymous*” as user name and provide “*guest*” as password (or his email address)
- User access to system is limited
- Can copy files, but limited navigation capabilities

FTP in Action ^{1/3}

```
bash$ ftp plaza.aarnet.edu.au
```

```
Connected to plaza.aarnet.edu.au.
```

```
220 plaza.aarnet.EDU.AU FTP server (Version wu-2.4(2) Fri Apr 15  
14:04:20 EST 1994) ready.
```

```
Name (plaza.aarnet.edu.au:jphb): ftp
```

```
331 Guest login ok, send your complete e-mail address as password.
```

```
Password:
```

```
230-
```

```
230- This is the AARNet Archive Server, Melbourne, Australia.
```

```
230-
```

```
230-Local time is Tue Jun 4 17:46:00 2001
```

```
230-
```

```
230-Please read the file /info/welcome-ftpuser
```

```
230- it was last modified on Fri Apr 22 14:47:05 1999 - 774 days ago
```

```
230 Guest login ok, access restrictions apply.
```

FTP in Action ^{2/3}

ftp> pwd

257 "/" is current directory.

ftp> ls

200 PORT command successful.

150 Opening ASCII mode data connection for file list. lost+found

usr

etc

java

pub

micros

bin

ACS

usenet

X11

rfc

graphics

226 Transfer complete.

214 bytes received in 0.018 seconds (11 Kbytes/s)

FTP in Action ^{3/3}

ftp> cd rfc

250 CWD command successful.

ftp> get rfc1048.txt.gz

200 PORT command successful.

150 Opening ASCII mode data connection for rfc1048.txt.gz (5141 bytes).

226 Transfer complete.

local: rfc1048.txt.gz remote: rfc1048.txt.gz

5161 bytes received in 1.6 seconds (3.2 Kbytes/s)

ftp> quit

221 Goodbye.

FTP Connections in Action ^{1/2}

```
C:\WINDOWS\system32>netstat
```

Active Connections

...

...

```
TCP    Office:4948          host-197.218.56.2.tedata.net:ftp ESTABLISHED
```

```
C:\WINDOWS\system32>netstat -n
```

Active Connections

```
TCP    188.9.140.10:4948      197.218.56.2:21        ESTABLISHED
```

FTP Connections in Action 2/2

//After File Transfer

C:\WINDOWS\system32>**netstat -n**

Active Connections

TCP	188.9.140.10:4948	197.218.56.2:21	ESTABLISHED
TCP	188.9.140.10:1296	197.218.56.2:20	TIME_WAIT

Further Information

- RFC 959: File Transfer Protocol (FTP), October 1985