Network Programming

Lab 6 Submission

Team

Josh Beals (bealsj2) - bealsj2@rpi.edu
Arron Look (looka) - looka@rpi.edu
Mike Yang (yangy25) - yangy25@rpi.edu
Abdul-Muiz Yusuff (yusufa2) - yusufa2@rpi.edu

Part I

1. Using the filter of ftp.command, determine the FTP commands that the user has used:
   The user used the following FTP commands: LIST, STOR, and RETR.

2. Using the filter of ftp.response, determine the FTP codes that have been returned:
   The unique FTP codes that have been returned are: 220, 331, 230, 215, 211, 257, 200, 227, 125, 226, 331, and 250.

3. What is the username and password for the access to the FTP server:
   The username is “Administrator”. The password is “napier”.

4. What is the name of the file which is uploaded:
   The file that is uploaded is “111.png”.

5. What is the name of the file which is downloaded:
   The file that is downloaded is “manual.txt”.

6. Using the filter of ftp.request.command=="LIST", determine the first packet number which performs a “LIST”:
   The first packet number that requests the LIST command is packet number 21.

7. In performing in the list of the files on the FTP server, which TCP port is used on the server for the transfer:
   The TCP ports used on the server for data transfer are 1078, 1080, 1081.

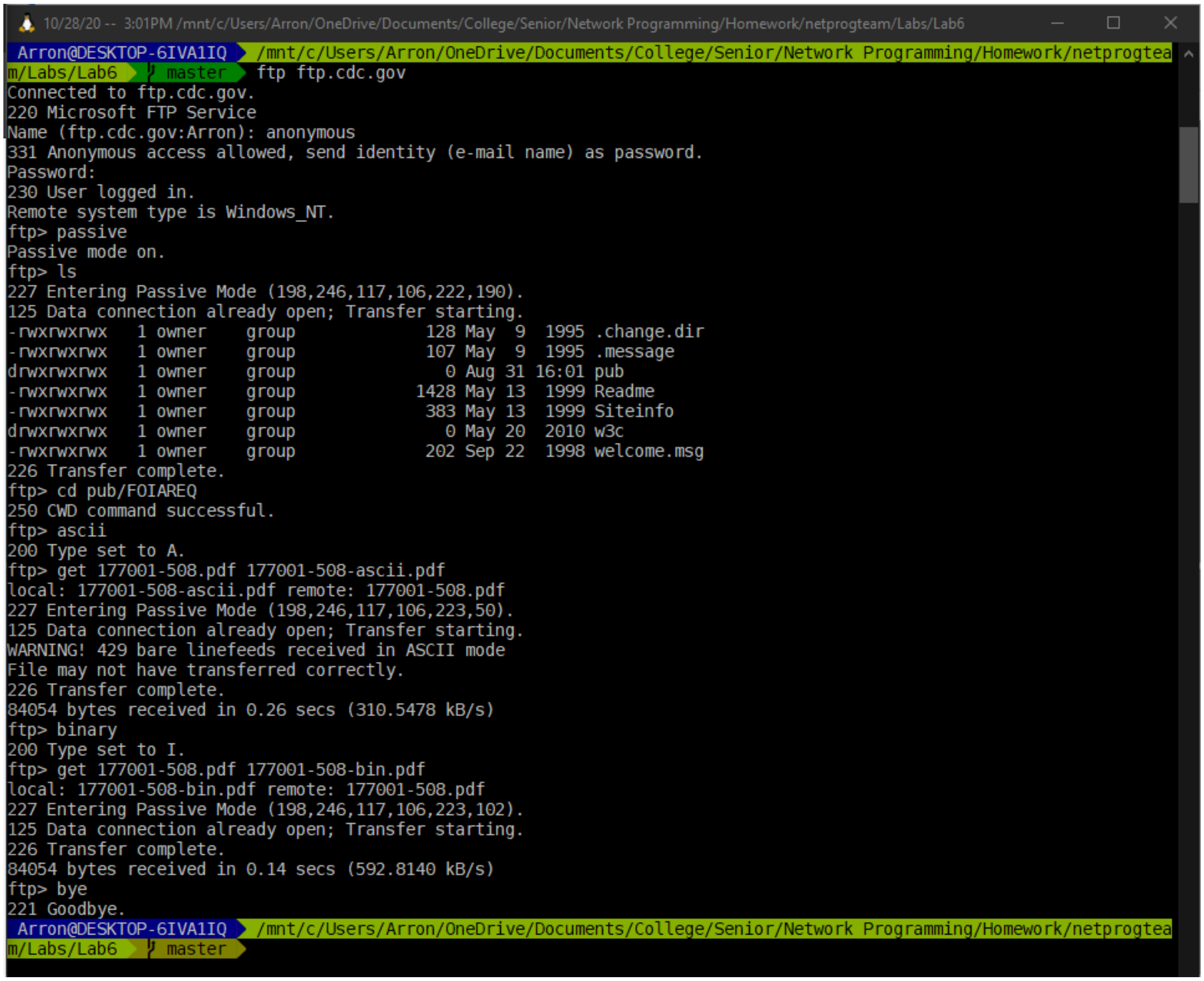
8. From the final “LIST” command, which are the files on the server?
   From the final LIST command, the response (ftp-data line-based text data) tells us that “1.docx” and “manual.txt” are the files.

9. What does the filter ftp.response.code==227 identify in terms of the ports that are used for the transfer:

This filter tells us that the server set up ports, on its end, for data connection starting from port 1078, incrementing the port number for every additional data connection that was established. The last port number negotiated is port 1082.

Part II

10. Include all terminal output in your Lab6_answers.pdf, showing that you were able to execute the commands listed above and download both versions of the file.



```
10/28/20 --  3:01PM /mnt/c/Users/Arron/OneDrive/Documents/College/Senior/Network Programming/Homework/netprogteam/Labs/Lab6
 Arron@DESKTOP-6IVA1IQ  /mnt/c/Users/Arron/OneDrive/Documents/College/Senior/Network Programming/Homework/netprogtea
m/Labs/Lab6   master  ftp ftp.cdc.gov
Connected to ftp.cdc.gov.
220 Microsoft FTP Service
Name (ftp.cdc.gov:Arron): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> passive
Passive mode on.
ftp> ls
227 Entering Passive Mode (198,246,117,106,222,190).
125 Data connection already open; Transfer starting.
-rwxrwxrwx   1 owner    group             128 May  9  1995 .change.dir
-rwxrwxrwx   1 owner    group             107 May  9  1995 .message
drwxrwxrwx   1 owner    group               0 Aug 31 16:01 pub
-rwxrwxrwx   1 owner    group            1428 May 13  1999 Readme
-rwxrwxrwx   1 owner    group             383 May 13  1999 Siteinfo
drwxrwxrwx   1 owner    group               0 May 20  2010 w3c
-rwxrwxrwx   1 owner    group             202 Sep 22  1998 welcome.msg
226 Transfer complete.
ftp> cd pub/FOIAREQ
250 CWD command successful.
ftp> ascii
200 Type set to A.
ftp> get 177001-508.pdf 177001-508-ascii.pdf
local: 177001-508-ascii.pdf remote: 177001-508.pdf
227 Entering Passive Mode (198,246,117,106,223,50).
125 Data connection already open; Transfer starting.
WARNING! 429 bare linefeeds received in ASCII mode
File may not have transferred correctly.
226 Transfer complete.
84054 bytes received in 0.26 secs (310.5478 kB/s)
ftp> binary
200 Type set to I.
ftp> get 177001-508.pdf 177001-508-bin.pdf
local: 177001-508-bin.pdf remote: 177001-508.pdf
227 Entering Passive Mode (198,246,117,106,223,102).
125 Data connection already open; Transfer starting.
226 Transfer complete.
84054 bytes received in 0.14 secs (592.8140 kB/s)
ftp> bye
221 Goodbye.
 Arron@DESKTOP-6IVA1IQ  /mnt/c/Users/Arron/OneDrive/Documents/College/Senior/Network Programming/Homework/netprogtea
m/Labs/Lab6   master
```

11. Open both downloaded PDF files in a PDF viewer and compare (try looking at the image in the upper left of the first page). note: Acrobat/Firefox will probably fail to open the ASCII version, but Edge, Chrome and Safari will open it just fine. Explain why the two files are different (if they are), or why ascii vs binary mode didn't matter. Also state your operating system, since this can affect your results.

\r\n←line feed in windows   \n ← line feed in "linux"

We used the built-in ftp client on the Windows WSL1 terminal (providing a "linux" like environment). The ftp server is running a Windows operating system and our actual operating system running WSL1 is Windows. When specifying ASCII mode, the ftp server will encode it's copy of the file and the ftp client on WSL1 will decode it, thinking that our system is using linux, which is why getting the file with ASCII mode resulting in using the linux line feed character when the original final on the windows ftp server used a windows line feed character. On the other hand, binary mode just transfers the file from the windows ftp server, as is, so the file we downloaded via binary mode preserves the windows line feed character. So when we open the two pdfs obtained through different modes, one uses the windows line feed character (the binary mode file) and correctly displays the image while the other file uses the linux line feed character (the ascii mode file), which messes up our Windows operating system's rendering of the image.