COMP247 Data Communications Laboratory
Practical 3B TCP

Objective

1. **Understand the structure of the TCP header and the purpose of the component fields**
2. **Observe the TCP three-way handshake in operation**
3. **Gain a basic understanding of TCP sequence numbers**

Resources

Equipment: Laboratory PCs.

ws-capture-ftp.pcap

Exercise 1. TCP & FTP

1.1 Objective & Background

TCP is used as the transport layer protocol by many applications. In this exercise we are going to examine some of the important parts of TCP. To help with this I'm giving you the capture for this exercise, rather than getting you do capture it yourself. This means your demonstrator knows in advance exactly where to find the material required to answer the questions, so if you have any problems then can rapidly help you.

In this exercise we are going to use FTP as a vehicle for investigating TCP. FTP is only one of an enormous range of application layer protocols that use TCP. It is a fairly simple one, and so shouldn't get in the way of us looking at TCP. FTP stands for File Transfer Protocol and is used for transferring files between computers (oddly enough).

Protocols Examined

- Application Layer – FTP for file transfer
- Transport Layer – TCP connections, sequence numbering, and graceful close

1.2 Theory

Reminder: Please remember to read these notes and review the web links **before the lab session.**

Textbook pages: Chapter 5 (the sections on TCP).

RFCs: FTP (RFC 959, try http://www.faqs.org/rfcs/rfc959.html)

You should review the material on TCP from the lecture notes and textbook (chapter 5). Note the diagrams showing the structure of a TCP packet.

Recall that we have two ways of communicating – connection-oriented and connectionless. These are both suited to different kinds of applications.

Connectionless communication is fast but unreliable. Connection-oriented removes this unreliability, but has overhead in setting up and tearing down a connection. In a perfect world we would not need connection-oriented communication, but we get lost packets and corrupted packets. The larger a packet, the more likely an error will occur, so for any line, the best packet size – not too small, and not too large, but just right – is used. Thus a message exceeding this size is broken up and the resulting sequence of packets sent over a connection.

Connection-oriented communication is like the telephone system – lift the receiver, dial the number, the target hears the phone ringing and lifts their receiver, thereby acknowledging willingness to participate, you talk over this connection, and then hang up. TCP is the layer 4 protocol for establishing connections which is the subject of this session. The 3-way (better called 3-message because there are only two parties involved) handshake is the TCP way of establishing this connection.

Connectionless is like the postal service. You place each package you want to send in an envelope, completely specify the address on each package, send it and hope it gets there, but maybe you never hear back, so there is no way you can tell if this worked. (You can send a registered letter, in which case you get an acknowledgement that the recipient received your letter.) A message sent over a connectionless service is often called a datagram. The protocol used for a connectionless service is UDP. We will not be looking at UDP further, but you might see UDP packets in your Wireshark traces.

Please note that there is a connection vs connectionless debate (aka religious war) that persists in the industry. These debates often arise because people on both sides have limited understanding of the issues and want to make things too simple. However, what will get you good marks in the course and other courses is that you know why these different approaches exist and what different situations you use them in.

You should also do some research on the connection establishment used by TCP. This was treated in the TCP lecture. You need to know a little more. Here's some links to get you started:

http://www.inetdaemon.com/tutorials/internet/tcp/3-way_handshake.shtml

http://en.wikipedia.org/wiki/Transmission_Control_Protocol#Connection_establishment

http://www.pccitizen.com/threewayhandshake.htm

http://support.microsoft.com/kb/172983

You probably won't need to read all of them thoroughly. You can find your own sources – there is much useful information to be found around the web. Make sure you understand TCP connection establishment before proceeding – ask you demonstrator if you need some more explanation.

Remember the transport layer provides connectivity from the application layer to the IP layer by means of ports (for both TCP and UDP). You will notice in this section that the FTP server uses the well-known ports 21 and 20. However, the FTP client is provided a port number by TCP.

This exercise examines an entire FTP and TCP session. This illustrates both how TCP works and an application-level protocol.

1.3 Procedure

Download the wireshark FTP capture *ws-capture-ftp* from the practicals web page. Open it up in wireshark. You will find the following filter useful to isolate the relevant packets:

tcp.port==21 || tcp.port==20

Port 21 is the ftp command channel and port 20 is the data channel. Recall that transport-layer ports are the final part of a destination address that tells us what application to deliver to on a machine – the IP address has delivered the message to that machine.

Examine the capture file and answer the following questions.

Documentation Task 1.

ftp.ftpplanet.com

1. What is the name and IP address of the FTP server contacted? In which packets can this information be found (hint: you will need to clear the filter and examine the packets just before the first packets on port 21)? 85
2. What is the port used on the server? 20,21
3. What port is being used on the client? 2011
4. Identify the three-way handshake that established the TCP connection between the client and server. Which packet numbers does the three-way handshake appear in? 79,83,84
5. The TCP connection is set by a SYN message from the FTP client, a SYN/ACK from the FTP server and an ACK from the client. What is happening in each of these messages?
6. Once the connection is established the FTP server sends a message saying it is ready. What packet carries this message? package 85
7. What information in the packet makes you think you have the correct packet? it contain the ACK message, and this package is sending to the server port 21
8. The user then types a user ID and a password. What packet numbers are involved in this exchange? Are the characters in the user ID sent individually or in a single packet? What is the user name? 124,126,149,152
9. Can you see the password? If so, what is it? No
10. What is your conclusion about the security of FTP? Isn't wiresniffing fun? all the credentials are encrypted
11. Find the single FTP command issued to the server. What is it and what message is it in? Request: LIST
12. In what packet(s) is the actual data sent from the server in response to that command? NOTE: that this will be in the FTP-DATA packets sent from the server on port 20. 173
13. Why do FTP and other protocols use a separate channel for command and data communications? Continue sending and receiving control instruction on the control connection while you are transfering data. Have more than one data connection active at the same time. The server decides when it's ready to send you data.
14. In what packet does the client say it wants to end the FTP session? 302

client node sends a S
the same or an externa
if the s
The target server mus
connections. When t
node, it responds and

The client node receive

15. Which flag is set in this packet? PUSH, ACK
16. In what packet does the server respond? 303
17. What messages are used to close the TCP connection? RST
18. Which end – client or server – actually terminates the TCP connection? client
19. What are the sequence numbers used in these messages? 71
20. Compare the sequence numbers displayed in Wireshark's dissection against the actual values shown in the raw packet in the bottom window – are they the same? Can you suggest why?
21. Select the first message in the command (port 21) conversation. Use the "Follow TCP Stream" option in the Wireshark Analyze Menu to see a summary of the information that is exchanged between the client and server. Does the result show the password? No
22. Select the first message on the data port (20). Display it as a stream – what data is being transferred?
    txt,html editors image, internet access mp3 software, other software, sample files, tutorials, users, welcome html
23. Identify the three-way handshake for this connection – which end is initiating the connection: the client or the server? client
24. Find the PORT command issued by the FTP client. How does the FTP server know where to send the data from this message? ip address and port value
25. In packets 174 and 178, the server sends two messages to the client ("125 Transferring directory" and "226 Transfer complete") but there is only one ACK message from the client following both of these. Explain why two messages can have only one ACK.