Exercise 3: Telnet and HTTP (Telnet, TCP, HTTP)

Objective

In this exercise we introduce the Telnet protocol for accessing resources in a remote machine. We first examine the Telnet protocol by logging into a remote host. We then show how Telnet can be used to connect to the well-known port of a server and to interact with the server through a command line. In the process we investigate additional details of the TCP and HTTP protocols.

Protocols Examined

- Telnet protocol for remote login
- HTTP access using telnet
- TCP windows, sequence numbering, and graceful close

Background Material

Refere correspondig chaper of your Textbook

RFCs: Telnet (RFC 854, try http://www.freesoft.org/CIE/RFC/854/index.htm)

HTTP (RFC 1945, try http://www.freesoft.org/CIE/RFC/1945/index.htm)

Telnet command: Consult your system documentation for information on using Telnet. For example, in Windows XP, start "Help and Support", then enter "telnet" in the search window and press Enter.

Prerequisite

Exercise 1: Web Browsing.

Procedure

Logging into a Telnet server.

1. Start an Ethereal packet capture.

2. Open a Command Prompt window. (In Windows XP, from the "All Programs" Menu, select "Accessories" and then "Command Prompt").

3. Type "`telnet <hostname>`". Check to see if your university library provides telnet access to its catalogue. For example, to access the University of Calgary Library you can telnet the host clavis.ucalgary.ca; to access the U.S. Library of Congress you can telnet locis.loc.gov. See www.telnet.org for other hosts you can telnet to.

4. Follow the instructions in the text returned from the telnet server. For example, if you telnet to clavis, you login as "clavis". You then press "Enter" twice to go to the main menu. Type "E", then press Enter to logout from the system.

5. Stop the packet capture.

6. Save the command screen using "Alt PrtSc" for future reference.

Accessing a Web Server using telnet.

1. Many application servers can be accessed by using telnet to connect to the server well-known port. The user can then interact with the server by entering text through a command line interface.

2. For example, to log into the yahoo website, type "`telnet www.google.co.in 80`" in the command interface. You will then obtain a blank screen. Type GET and then ENTER. The server will then send the HTTP response.

3. Stop the packet capture.

Protocol Analysis Questions

To answer the following questions, start Ethereal and open the packet capture file created above.

Logging into a Telnet server

1. Find the IP address of the Telnet server you are contacting.

2. Identify the three-way handshake that established the TCP connection between the client and server. Note the well-known port number for the server. Note the initial sequence numbers of the connections in each direction.

3. The TCP connection setup is followed by an exchange of packets between the telnet server and client negotiating options. Identify these packets and examine the contents of the telnet messages to see what options where negotiated, e.g. terminal type, terminal speed, and window size.

4. Once the options have been negotiated, the server usually sends a login prompt. Find the sequence of packets that carry this information from the server.

5. The user then types a user ID and sometimes a password. Trace the sequence of packets that carries the user id. Are the characters in the user ID sent individually or as a group? Does the client TCP set the PUSH flag?

6. If telnet was set up with the echo option, the server will echo each character received back to the client. Check to see if the characters are being echoed.

7. Use the "Follow TCP Stream" option in the Ethereal Tools Menu to see the information that is exchanged between the client and server.

8. In the Tools Menu, try the TCP Stream Analysis tools provided by Ethereal. The time-sequence graph provides a plot of sequence number versus time. The throughput graph provides an indication of short-term transmission rate versus time. The RTT graph shows how the round-trip time varies during the course of the session.

9. Finally examine the exchange of segments that execute the TCP graceful close for the connection in each direction. Can you find the FIN segment that is sent by the server to initiate the closing from its direction? Confirm the setting of the sequence numbers in the segments as the graceful close proceeds.

Accessing a Web Server using telnet.

1. Examine the DNS packets to identify the IP address of the server that is being contacted.

2. Identify the three-way handshake that sets up the TCP connection between the client and server. Note the well-known port number for the server. Note the TCP initial sequence numbers and the window sizes.

3. Identify the packets that carry the GET command message from the client. Does TCP use the PUSH command? Note the corresponding ACK segments from the server and determine whether the character is being echoed.

4. Next identify the sequence of packets that carry the HTTP response from the server. What is the text carried in the first segment?

5. Do the packets from the server carry maximum size TCP segments? Do you see a pattern as to how frequently the TCP client acknowledges segments?

6. Observe the advertised window sizes in the sequence of segments. Do you see any variation in the window size over time?

Exercise 3: Telnet and HTTP (Telnet, TCP, HTTP)

7. What is the size of the file transferred from the server to the client? How much time was required to carry out the transfer? Estimate the average transmission bit rate.

8. In the Ethereal Tools Menu, use the TCP stream analysis tools to observe the evolution of sequence number versus time, the throughput versus time, and the RTT versus time.

9. In the Ethereal Tools Menu, select "Follow TCP stream". The resulting window shows the client's "get" and the server's response. Save the window as an HTML file. Open the HTML file using a web browser to see what would have been displayed in your browser window.

10. Identify the graceful close that terminates the TCP connection.