Lab 11.2.4 Protocol Inspector, TCP, and HTTP – Instructor Version

Objective

The objective of this lab is to use Protocol Inspector, or equivalent software, to view dynamic Transmission Control Protocol (TCP) operations. The operation that will be specifically looked at is HTTP during web page access.

Background / Preparation

Protocol analysis software has a feature called **capture**. This feature allows all frames through an interface to be captured for analysis. With this feature, it is possible to see how the TCP moves segments filled with user data across the network. TCP may seem to be a bit abstract, but the protocol analyzer shows just how important TCP is to network processes such as e-mail and web browsing.

At least one of the hosts must have the Protocol Inspector software installed. If the lab is done in pairs, having the software installed on both machines means that each person can run the lab steps. However, each host may display slightly different results.

Step 1 Start Protocol Inspector and your browser

Step 2 Go to detail view

Step 3 Start a capture

Step 4 Request a Web Page

Step 5 Watch the monitor view while the web page is requested and delivered

Step 6 Stop the capture

Step 7 Study the TCP frames, HTTP frames, and statistics using various views, especially the detail view

Step 8 Using the detail view, explain what evidence it provides about the following:

(Answers vary. See notes and sample output below)

- TCP handshakes
- TCP acknowledgments
- TCP segmentation and segment size
- TCP sequence numbers
- TCP sliding windows
- HTTP protocol

Reflection

How did this lab help to visualize the TCP protocol in action?

Answers will vary. See notes and sample output below.

Instructor Notes

There is a 250 packet limit for captures in the education version of Protocol Inspector. It is important that a very small network is used for this lab to examine the first few packets and thus the 3-way handshake. A workstation running PI connected to a router or switch with **`ip http server`** enabled is a very simple way to capture clean, usable output. Adding Steps 6A and Step 6B will clarify to the student what is occurring with the captured screen information.

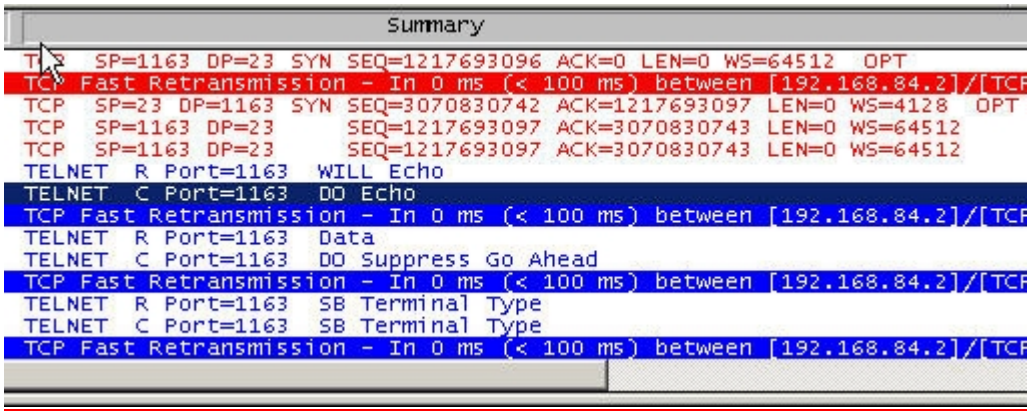
Step 6A – After the capture is stopped, go to File and Save Capture. Save the captured packets with a name like HTTP_Capture1.

Step 6B – Go to File and Open, then open one of the captures you saved. You'll be alerted that the file was truncated because of the 250 packet limit. Now you can see details as shown in these screen shots.

The following screen shots from captures of a Telnet and an HTTP session illustrate some of key concepts of TCP/IP:

Three-way handshake

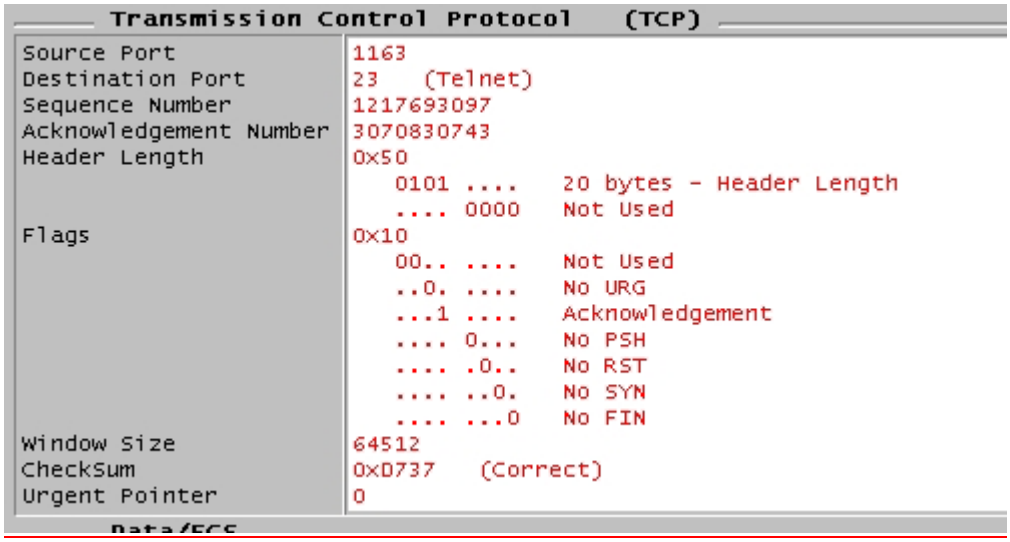
Line 1 shows a SYN; Line 3 shows a SYN-ACK; and Line 4 is an ACK. Also note sequence numbers for segmentation (SEQ), and window size (WS).



TCP Sliding Windows

Note the difference in window size in the packet from the workstation to the host, and the packet from the host to the workstation. Note also the port numbers, and the changing of source and destination port for the return packet.

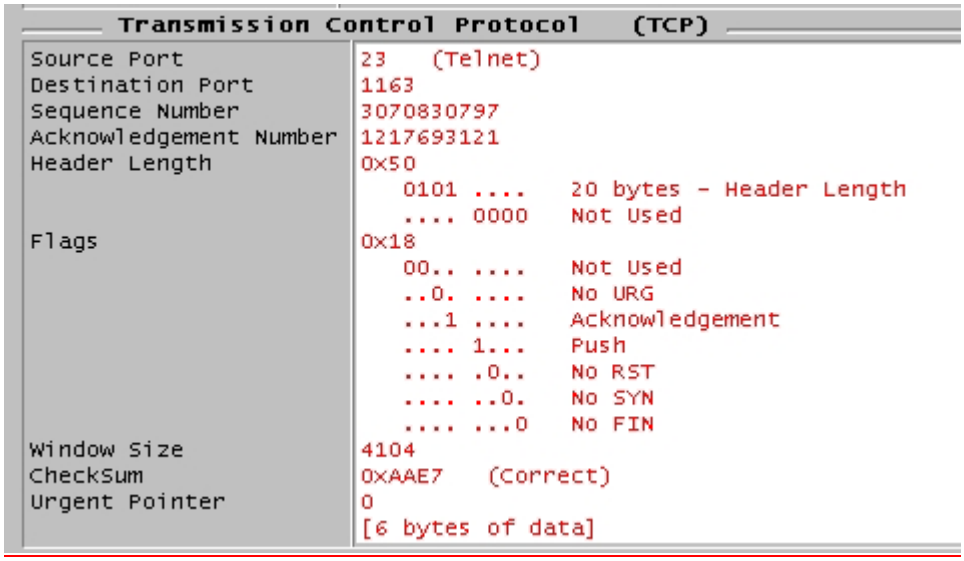
From workstation to host:



Transmission Control Protocol (TCP)

Source Port	1163
Destination Port	23 (Telnet)
Sequence Number	1217693097
Acknowledgement Number	3070830743
Header Length	0x50
	0101 .... 20 bytes - Header Length
	.... 0000 Not Used
Flags	0x10
	00.. .... Not Used
	..0. .... No URG
	...1 .... Acknowledgement
	.... 0... No PSH
	.... .0.. No RST
	.... ..0. No SYN
	....0 No FIN
Window Size	64512
CheckSum	0xD737 (Correct)
Urgent Pointer	0

From host to workstation:

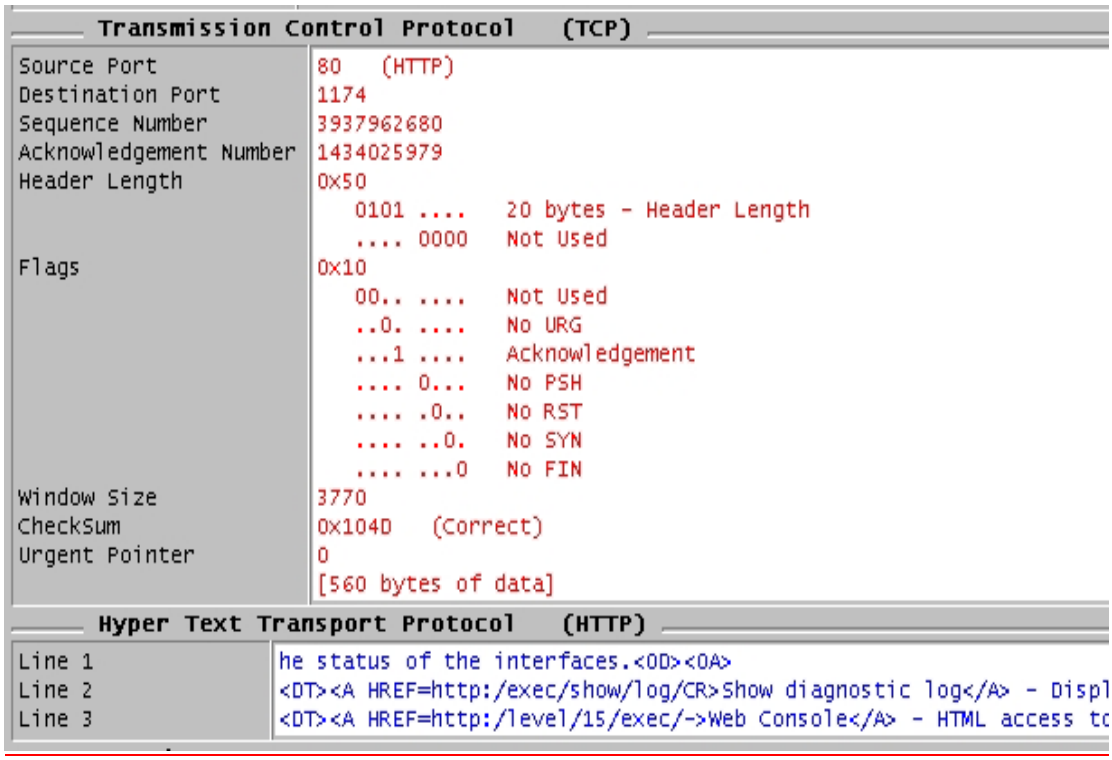


Transmission Control Protocol (TCP)

Source Port	23 (Telnet)
Destination Port	1163
Sequence Number	3070830797
Acknowledgement Number	1217693121
Header Length	0x50
	0101 .... 20 bytes - Header Length
	.... 0000 Not Used
Flags	0x18
	00.. .... Not Used
	..0. .... No URG
	...1 .... Acknowledgement
	.... 1... Push
	.... .0.. No RST
	.... ..0. No SYN
	....0 No FIN
Window Size	4104
CheckSum	0xAAE7 (Correct)
Urgent Pointer	0
	[6 bytes of data]

HTTP

This is a data packet returning from the Web server. Note that the source port is 80 (HTTP). The destination port (1174) was generated by the requesting workstation for this session only. The page (or portion of a page) that was delivered by this packet can be seen below under **Hyper Text Transfer Protocol (HTTP)**.



Transmission Control Protocol (TCP)

Source Port	80 (HTTP)
Destination Port	1174
Sequence Number	3937962680
Acknowledgement Number	1434025979
Header Length	0x50
	0101 .... 20 bytes - Header Length
	.... 0000 Not Used
Flags	0x10
	00.. .... Not Used
	..0. .... No URG
	...1 .... Acknowledgement
	.... 0... No PSH
	.... .0.. No RST
	.... ..0. No SYN
	....0 No FIN
Window Size	3770
CheckSum	0x104D (Correct)
Urgent Pointer	0
	[560 bytes of data]

Hyper Text Transport Protocol (HTTP)

Line 1	he status of the interfaces.<0D><0A>
Line 2	<DT><A HREF=http:/exec/show/log/CR>Show diagnostic log</A> - Displ
Line 3	<DT><A HREF=http:/level/15/exec/->Web Console</A> - HTML access to

