# Lab 3 – TCP

1. **Objective**

In this lab, we first get familiar with the format of TCP header, then study the TCP 3-Way Handshake and possibly connection termination.

1. **Introduction**

TCP is the dominant transport layer protocol in the Internet. It provides a reliable, in-order stream of data between two end-points, even if they are connected by a network that may drop, re-order, or corrupt the packets. TCP provides the reliable data streaming service by detecting if packets are lost, delayed, or corrupted during transmission.

In this LAB you will capture a trace of TCP segments exchanged in downloading a web page and you will analyse TCP header fields to identify TCP connection setup, use of sequence numbers, closing of a connection and may be flow control, congestion control etc.

1. **Procedure**
   1. **Capturing the Trace**

Open a command window and type, “nslookup nsbm.lk”. This may show you the IP address of the required server. If it does not succeed, ask your instructor and he/she will give you the IP address of the server.

Create following filter to filter traffic between nsbm.lk web server and your PC.

“(ip.dst == <IP address of nsbm.lk> && ip.src == <IP address of your PC>) or (ip.dst == <IP address of your PC> && ip.src == <IP address of nsbm.lk>)

**Click start capture button** to start new live capture.

Now open a browser window and try to access [www.nsbm.lk](http://www.nsbm.lk/) web site. Now you will see wireshark is capturing all the packets sent from your PC to the server.

**Stop capture and clear the filter.**

Now you should see all the traffic exchanged between the web server and your machine in wireshark packet list.

Select one of HTTP packets from the packet list and expand the 'Transmission control protocol' in the packet details pane which is just below the packet list. You will see values in various fields of TCP protocol. **Record down source port and destination port**.

Update the following filter string by inserting respective port numbers at appropriate places. Then copy filter string into filter box and apply the filter.

“(tcp.dstport == <destination port> && tcp.srcport == <source port>) or (tcp.dstport == <source port> && tcp.srcport == <destination port> )”

Now what you have in the packet list is all the packets related to one TCP connection.

Since this lab is about TCP rather than HTTP, change WireShark’s Packet List Pane window so that it shows information about the TCP segments containing the HTTP messages. To do this, in WireShark, select Analyze => Enabled Protocols. Then uncheck the HTTP box and select OK.

* Select the first packet and explore the details of the TCP segment using the packet details pane and the packet bytes pane.
* Select the Transmission Control Protocol item in the Packet Details Pane then the content of the header is highlighted in the Packet Bytes Pane.
* What is the header length? Check header length field.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. **TCP Connection Setup**

Find the initial three-way handshake in the trace and answer following questions

* Which segments are the initial three-way handshake in the trace? How do you determine this?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* What is the actual initial sequence number in each direction (in hexadecimal format)? How did the client and the server determine these values?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Note: WireShark displays the relative sequence number. You should select the Sequence Number field in the header, the actual value is highlighted in the Packet Bytes Pane.

* What is the value of the acknowledgment number in the SYNACK segment? How does web server determine that value?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* What are the values of the sequence number and the acknowledgment number in the third ACK segments in the three-way handshake? How did the client computer determine these values?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* How did the client and the server announce the maximum TCP payload size that they were willing to accept? What are the values and why did they choose these values?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Is there data sent in the SYN, SYNACK, and ACK segment? How do you determine this?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ctd….

* 1. **TCP Data Flow**

Beginning with the 1st segment, what are the sequence number, acknowledgment number, data length, and the time of the segment sent/received from/to the client computer of three way handshake and 4th, 5th, 6th, ..., 15th segments in the TCP connection? Fill out Table below for the data flow from the client computer to the server. (Note: list both the actual value and relative value of the sequence number and acknowledgment number. Wireshark normally displays relative sequence numbers. To display actual sequence numbers untick relative sequence number check box under TCP protocol preferences.)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pkt Number | Data Segments | | | Acknowledgement Segments | | |
| Src IP: | Dest IP: | | Src IP: | Dest IP: | |
| Seq. No./  Relative Seq. No. | Data  Length | Time (s) | Ack. No./  Relative Ack. No. | Data  Length | Time (s) |
| 1 |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 16 |  |  |  |  |  |  |
| 17 |  |  |  |  |  |  |
| 18 |  |  |  |  |  |  |
| 19 |  |  |  |  |  |  |

**TCP segment exchange table**

* 1. **Retransmission in TCP (Optional)**
* Can you identify any TCP retransmissions in the trace? If there are retransmissions give details.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Save captured trace into your personal storage such as a pen drive.**