

Lab 3: Packet Capture

Introduction

In this lab, you will use a “packet sniffer” called Wireshark to capture and analyze TCP packets generated between the PC browser and a web server, such as **matrix.senecacollege.ca**. When the application layer of the TCP/IP protocol stack creates an HTTP message, that message is “encapsulated” by a transport layer header. The header identifies the protocol TCP which is used to make a reliable connection to a web server. TCP uses a three-way handshake to establish a connection and a three-way handshake to take down a connection between the two hosts. The Internet layer adds a header indicating the logical IP address, but is also responsible to retrieve the MAC address which is passed to the Data Link layer for addition into the LAN header. You will see how the Internet layer uses a protocol called ARP (Address Resolution Protocol) to find the MAC or Ethernet address of the next link. Lastly, you will see the message syntax and sequence of the HTTP protocol.

Objective:

1. Demonstrate basic packet capturing with Wireshark
2. Examining the TCP handshake used to set and take down a reliable connection
3. Examine how the Internet layer uses ARP

Instructions:

1. Use the MyApps folder to locate Wireshark
2. Click the Launch button to open Wireshark
3. Use **ipconfig /all** at a command prompt to get the IP and physical addresses of the local machine.

Physical Address of host	00-68-EB-D6-53-03
IP Address of host	192.168.2.15
IP Address of default gateway	192.168.2.1
Physical address of default gateway	C0-E4-34-57-6C-1D

4. Before we capture packets delete the ARP cache. This area of memory keeps a mapping or IP addresses to MAC addresses. We want to delete any previous entry so that the protocol ARP will need to be used in our capture
5. Open a command line windows as administrator and type the following:
netsh interface ip delete arpcache

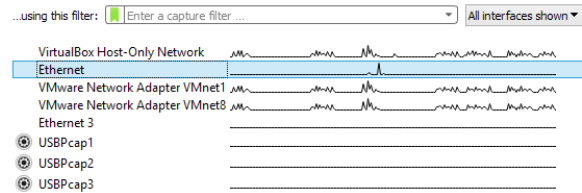
Capturing and Examining TCP Packets

TCP Connection Setup: 3-way Handshake

1. **Close all the browser windows** before starting Wireshark.
2. Select an Interface to capture called "Ethernet" which shows activity on it. Similar to the screen shot above
3. On Wireshark select the interface for packet capturing (ethernet or wifi)
4. On the capture menu click the Start button
5. Open the browser and navigate to **matrix.senecacollege.ca**
6. When the web page loads, close the client window and wait a couple of seconds
7. Return to Wireshark and **Stop** capture.
8. Save the capture as a file called **learnname_L3_capture**. This is important, if you need to return to the original file after applying display filters.
9. Type in the Display filter text box **ip.addr==142.204.165.128** (if accessing from outside Seneca network. If this does not work ping matrix.senecacollege.ca and get the ip address from the ping reply message) or **ip.addr==10.102.108.5** (from Seneca network). This will show the beginning of your conversation with the matrix server. Your Wireshark window should look like the screen shot below.

Welcome to Wireshark

Capture



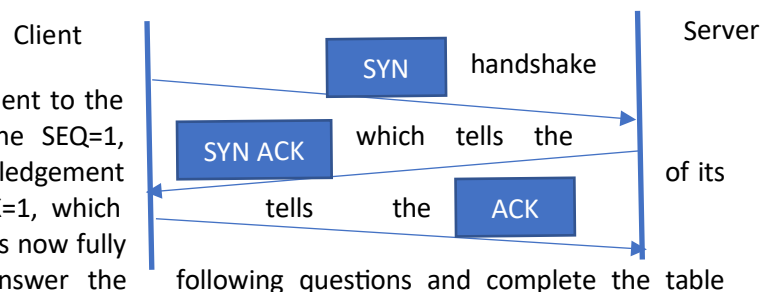
No.	Time	Source	Destination	Protocol	Length	Info
64	18.076855	10.40.105.151	142.204.140.90	TCP	66	49912 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
65	18.076856	10.40.105.151	142.204.140.90	TCP	66	49911 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM=1
66	18.077923	142.204.140.90	10.40.105.151	TCP	66	80 → 49912 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1 WS=128
67	18.077936	142.204.140.90	10.40.105.151	TCP	66	80 → 49911 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1 WS=128
68	18.077958	10.40.105.151	142.204.140.90	TCP	54	49912 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
69	18.077977	10.40.105.151	142.204.140.90	TCP	54	49911 → 80 [ACK] Seq=1 Ack=1 Win=262144 Len=0
70	18.078037	10.40.105.151	142.204.140.90	HTTP	329	GET / HTTP/1.1
71	18.079125	142.204.140.90	10.40.105.151	TCP	60	80 → 49912 [ACK] Seq=1 Ack=276 Win=15744 Len=0
72	18.084523	142.204.140.90	10.40.105.151	HTTP	357	HTTP/1.1 302 Found
73	18.084556	10.40.105.151	142.204.140.90	TCP	54	49912 → 80 [ACK] Seq=276 Ack=304 Win=261632 Len=0

6. Notice the first conversation between your host to the server is a [SYN] packet with an info number of 49912 (**yours will be different**). The latter is a TCP flag which tells the server to open a connection to the host. Notice SEQ=0. Click on the [SYN] packet and open the drop-down arrow on the Transmission Control Protocol in the Details pane in the middle Wireshark window.
7. In the top Wireshark packet list pane, select the second TCP packet, labeled SYN, ACK with the same info number 49912.
8. Observe the packet details in the middle Wireshark packet details pane. Notice that it is an Ethernet II / Internet Protocol Version 4 / Transmission Control Protocol frame.
9. Expand Ethernet II to view Ethernet details. Answer the following questions in the table below:
10. Observe the Destination and Source fields.
11. Expand Internet Protocol Version 4 to view IP details.
12. Observe the Source and Destination IP addresses.
13. Expand Transmission Control Protocol to view TCP details.
14. Observe the Source and Destination ports.

15. Observe the Sequence number. Notice that it is 0 (relative sequence number). To see the actual sequence number, select Sequence number to highlight the sequence number in the bottom Wireshark bytes pane.
16. Observe the Acknowledgement number. Notice that it is 1 (relative ack number). To see the actual acknowledgement number, select Acknowledgement number to highlight the acknowledgement number in the bottom pane. Notice that the actual acknowledgement number is one greater than the sequence number in the previous segment.
17. Expand Flags to view flag details.
18. Observe the flag settings. Notice that SYN and ACK flags are set, indicating the second segment in the TCP three-way handshake.
19. Complete the FIN ACK packet analysis table. **[0.4 Marks]**

FIN ACK Packet Analysis	
What is the source MAC address of this packet? (should be the default gateway physical address)	C0:3C:04:2D:94:6D
What is the destination address of this packet? (should be the host physical address)	C0:E4:34:57:6C:1D
What is the source IP address of this packet? (should be the matrix server IP address)	142.204.165.128
What is the destination IP address of this packet? (should be the host IP address)	192.168.2.15
What is the destination port of this packet? (should be a local dynamic port created for this connection)	57110 56994
What is the source port of this packet? (should be port 80)	443

19. The last step of the 3-way handshake is the host sends an acknowledgement to the server's acknowledgement with the SEQ=1, server that this packet is an acknowledgement previous packet and with the ACK=1, which server the communication channel is now fully open and able to send data. Answer the following questions and complete the table below:



20. Observe the Destination and Source fields.
21. Expand Internet Protocol Version 4 to view IP details.
22. Observe the Source and Destination IP addresses.
23. Expand Transmission Control Protocol to view TCP details.

24. Observe the Source and Destination ports.
25. Observe the Sequence number. Notice that it is 1 (relative sequence number). To see the actual sequence number, select Sequence number to highlight the sequence number in the bottom Wireshark bytes pane.
26. Observe the Acknowledgement number. Notice that it is 1 (relative ack number). To see the actual acknowledgement number, select Acknowledgement number to highlight the acknowledgement number in the bottom pane.
27. Expand Flags to view flag details.
28. Observe the flag settings. Notice that ACK is set, indicating the third segment in the TCP three-way handshake. The client has established a TCP connection with the server.
29. Complete the ACK packet analysis table. **[0.4 Marks]**

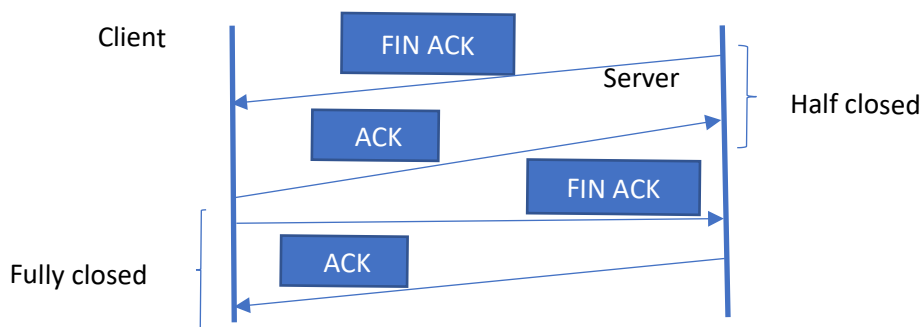
ACK Packet Analysis	
What is the source MAC address of this packet? (should be the host physical address)	C0:E4:34:57:6C:1D
What is the destination address of this packet? (should be the default gateway physical address)	C0:3C:04:2D:94:67
What is the source IP address of this packet? (should be the host IP address)	192.168.2.15
What is the destination IP address of this packet? (should be matrix server IP address)	142.204.165.128
What is the destination port of this packet? (should be a port 80)	443
What is the source port of this packet? (should be the local dynamic port)	57110

TCP Connection Tear Down: 3-way handshake

1. The ending of a TCP connection also uses a 3-way handshake (there are multiple ways to end a TCP connection, we will only discuss one way)
2. Scroll down to after the application data has been sent and look for a TCP packet with the FIN and ACK flags set. This is the beginning step of a tear down. After the server has sent the http response message to the client, it sends a packet with the FIN ACK flags set which tells the client there is nothing more. The client responds with an ACK packet back to the server. Notice the ACK packet copies the SEQ number of the FIN ACK packet as the ACK number of the ACK packet and the SEQ number of the FIN ACK packet is used as the ACK number + 1. Thus, the server knows to which TCP connection the client is referring to and closes the connection.


No.	Time	Source	Destination	Protocol	Length	Info
2161	31.480469	10.247.6.144	142.204.140.90	TCP	54	65496 → 443 [ACK] Seq=1516 Ack=4410 Win=261376 Len=0
2162	31.686422	104.93.160.138	10.247.6.144	TLSv1.2	85	Encrypted Alert
2163	31.686642	10.247.6.144	104.93.160.138	TCP	54	65487 → 443 [ACK] Seq=331 Ack=3073 Win=261632 Len=0
2164	31.686845	104.93.160.138	10.247.6.144	TCP	54	443 → 65487 [FIN, ACK] Seq=3073 Ack=331 Win=30336 Len=0
2165	31.686953	10.247.6.144	104.93.160.138	TCP	54	65487 → 443 [ACK] Seq=331 Ack=3074 Win=261632 Len=0
2166	31.687082	104.93.160.138	10.247.6.144	TCP	54	[TCP Out-Of-Order] 443 → 65487 [FIN, ACK] Seq=3073 Ack=331 Win=30336 Len=0
2167	31.687141	10.247.6.144	104.93.160.138	TCP	54	[TCP Dup ACK 2165#1] 65487 → 443 [ACK] Seq=331 Ack=3074 Win=261632 Len=0
2168	31.773205	142.204.140.90	10.247.6.144	TLSv1	91	Encrypted Alert
2169	31.773537	10.247.6.144	142.204.140.90	TCP	54	65499 → 443 [ACK] Seq=872 Ack=3386 Win=261376 Len=0
2170	31.774925	142.204.140.90	10.247.6.144	TCP	54	443 → 65499 [FIN, ACK] Seq=3386 Ack=872 Win=17824 Len=0
2171	31.775081	10.247.6.144	142.204.140.90	TCP	54	65499 → 443 [ACK] Seq=872 Ack=3387 Win=261376 Len=0
2172	33.334094	13.107.246.254	10.247.6.144	TCP	54	443 → 65454 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2173	35.004594	8.18.25.26	10.247.6.144	TCP	54	443 → 65439 [RST, ACK] Seq=1 Ack=1 Win=51 Len=0
2174	35.345804	10.247.6.144	142.204.140.90	TCP	54	65496 → 443 [FIN, ACK] Seq=1516 Ack=4410 Win=261376 Len=0
2175	35.345930	10.247.6.144	142.204.140.90	TCP	54	65496 → 443 [RST, ACK] Seq=1517 Ack=4410 Win=0 Len=0
2176	35.346157	10.247.6.144	142.204.140.90	TCP	54	65497 → 443 [FIN, ACK] Seq=994 Ack=3712 Win=261120 Len=0
2177	35.346248	10.247.6.144	142.204.140.90	TCP	54	65497 → 443 [RST, ACK] Seq=995 Ack=3712 Win=0 Len=0
2178	35.346600	10.247.6.144	142.204.140.90	TCP	54	65495 → 80 [FIN, ACK] Seq=379 Ack=305 Win=261632 Len=0
2179	35.446219	142.204.140.90	10.247.6.144	TCP	54	443 → 65496 [ACK] Seq=4410 Ack=1517 Win=18896 Len=0
2180	35.446721	142.204.140.90	10.247.6.144	TCP	54	443 → 65497 [ACK] Seq=3712 Ack=995 Win=17824 Len=0
2181	35.447215	142.204.140.90	10.247.6.144	TCP	54	80 → 65495 [ACK] Seq=305 Ack=380 Win=15744 Len=0

- A TCP connection is full duplex. At this state, the connection is half closed. The next step is for the client to send a FIN ACK packet to the server and the server responds with an ACK packet which closes the client side of the connection. Notice again how TCP copies the SEQ number of the FIN ACK packet as the SEQ number of the ACK packet and the ACK number is the SEQ number of the FIN ACK packet + 1.



ARP Protocol

- The ARP protocol is used to map IP addresses to Ethernet addresses which is used on a LAN. Since we deleted the ARP cache the default gateway physical address is not in the ARP cache, so ARP broadcasts to all hosts on the network, "whoever owns this IP address, please forward to me your MAC address" so I can send you some data. Notice it is the Internet layer that gets the physical or MAC address, which is added to the Data Link header. ARP proves that messages are forwarded link-by-link from source to destination.
- Click on the first ARP packet.
- Observe the details pane. Notice that the packet is encapsulated inside an Ethernet frame. Answer the following questions and complete the table below. **[0.4 Marks]**

ARP Request Packet	
What type of ARP packet is this (Request/Reply)?	REQUEST
What is the destination physical address of the ARP packet?	C0:3C:04:2D:94:67
What type of casting is this address? (Unicast, broadcast,multicast)	Unicast 
What is the source physical address of the ARP packet?	C0:E4:34:57:6C:1D
What is the sender's IP address?	192.168.2.15
What is the target's IP address?	192.168.2.1

4. Scroll down to the next ARP packet. Click on the packet and observe the details pane. Answer the following questions: **[0.4 Marks]**

ARP Reply Packet	
What type of ARP packet is this (Request/Reply)?	REPLY
What is the destination physical address of the ARP packet?	C0:E4:34:57:6C:1D
What type of casting is this address? (Unicast, broadcast,multicast)	Unicast
What is the source physical address of the ARP packet?	C0:3C:04:2D:94:67
What is the sender's IP address?	192.168.2.1
What is the target's IP address?	192.168.2.15

Other Common Protocols

Scroll through the list of protocols listed in Wireshark and identify 4 additional protocols. Provide a one sentence description of the protocol's purpose. (use your own words, do not copy and paste) **[0.25 Marks]**

Protocol	Description
DNS	The Domain Name System (ffDNS) is a hierarchical and decentralized naming system for the machines (computers), services or other resources connected to the Internet or a private network.
NBNS	NBNS (sometimes called WINS) stands for NetBIO Name Service. NBNS performs the same function as LLMNR, but using UDP broadcast packets instead of multicast packets.
ICMP	The Internet Control Message Protocol (ffICMP) is a supporting protocol in the Internet protocol suite. It is used by network devices, including routers, to send error messages and operational information indicating success or failure when communicating with another IP address.
LLMNR	The Link-Local Multicast Name Resolution (ffLLMNR) is a protocol based on the Domain Name System (ffDNS) packet format that allows both IPv4 and IPv6 hosts to perform name resolution for hosts on the same local link.

Grading:

- DCF255_L3_packetcapture.docx – with completed tables
- LearnName_L3_packetcapture.pcap

Upload files using link on MySeneca\Graded Work

