

Lab 1 Wireshark Exercises

Hint: See the accompanying document titled the “QuickStart Guide” – Look under the appendix describing “Hits Versus Page Views”.

Hint: a favicon.ico is a small graphic that can be used as an icon to identify a web page. In the following graphic the colorful “G” to the left is a favicon.ico.



I) **Exercise One**

Open “Wireshark”, then use the “File” menu and the “Open” command to open the file “Exercise One.pcap”. You should see 26 packets listed.

This set of packets describes a ‘conversation’ between a user’s client and a central server. This entire conversation happens automatically, after a user types something and hits enter. Look at the packets to answer the following questions in relation to this conversation.

In answering the following questions, use brief descriptions. For example, “In frame X, the client requests a web page, and in frame Y, the server delivers the content of the page.”

a) What is the IP address of the client that initiates the conversation?

- 131.247.95.216

a) Use the first two packets to identify the server that is going to be contacted. List the common name, and three IP addresses that can be used for the server.

- google
- 64.233.161.99 , 64.233.161.104, and 64.233.161.147

b) What is happening in frames 3, 4, and 5?

- they are doing a three way handshake

c) What is happening in frames 6 and 7?

- in frame 6 the client requests server for URI and frame 7 the server acknowledges the requests

d) Ignore frame eight. However, for your information, frame eight is used to manage flow control.

e) What is happening in frames nine and ten? How are these two frames related?

- In frame 9, acknowledge is set and forwarded to client
- In frame 10, server is sending requested URI to client

f) What happens in packet 11?

- it acknowledges the packets received in frame 10

g) After the initial set of packets is received, the client sends out a new request in packet 12. This occurs automatically without any action by the user. Why does this occur? See the first “hint” to the left.

- there is an image which was not sent in frame 10, so the client asks for the image in another package

h) What is occurring in packets 13 through 22?

- its repeating the above
- Packet 13 acknowledges packet 12. Packets 14 to 21 are requests and acknowledge related to the requested image file.
- Packet 22 contains the image file that is finally sent to the client.

i) Explain what happens in packets 23 through 26. See the second “hint” to the left.

- Frame 23 is an automatic request sent by the client, for the favicon that was not sent.
- Frame 24 is the acknowledge to frame 23.
- Frame 25 contains the image file requested by client.
- Frame 26 is the acknowledge for received packet in frame 25.

j) In one sentence describe what the user was doing (Reading email? Accessing a web page? FTP? Other?).

- accessing a web page

Review: The TCP/IP network does not know how to route using names, such as www.yahoo.com. It only knows how to route using IP addresses. Therefore, a common name (CNAME), such as www.yahoo.com must be translated to an IP address, like 216.109.117.106 before your computer can request a web page.

Your computer uses a DNS request to lookup a CNAME and it gets back a set of IP addresses (a primary address and one or more backup addresses) that can be used to contact the server.

Hint: See the accompanying document titled the “QuickStart Guide” – Look under the appendix describing “Hits Versus Page Views”.

II) Exercise Two

Open “Wireshark”, then use the “File” menu and the “Open” command to open the file “Exercise Two.pcap”. You should see 176 packets listed.

- a) In the first few packets, the client machine is looking up the common name (cname) of a web site to find its IP address. What is the cname of this web site? Give two IP addresses for this web site.

- yahoo.com
- 216.109.117.106
- 216.109.117.109

- b) How many packets/frames does it take to receive the web page (the answer to the first http get request only)?

- 22 packets

- c) Does this web site use gzip to compress its data for sending? Does it write cookies? In order to answer these questions, look under the payload for the reassembled packet that represents the web page. This will be the last packet from question b above. Look to see if it has “Content-Encoding” set to gzip, and to see if it has a “Set-Cookie” to write a cookie.

- No, yahoo doesn't use gzip, but uses cookies

- d) What is happening in packets 26 and 27? Does every component of a web page have to come from the same server? See the Hint to the left.

- In packet 26, the server is sending query to another server.
- In packet 27, the next server is responding to the main server.
- No, not every component come from same server

- e) In packet 37 we see another DNS query, this time for us.il.yimg.com. Why does the client need to ask for this IP address? Didn't we just get this address in packet 26? (This is a trick question; carefully compare the two common names in packet 26 and 37.)

- The DNS query are different for 26 and 37

- f) In packet 42 we see a HTTP “Get” statement, and in packet 48 a new HTTP “Get” statement. Why didn't the system need another DNS request before the second get

statement?
Click on
packet 42
and look in
the middle
window.
Expand the
line titled
“Hypertext
Transfer
Protocol”
and read the
“Host:” line.
Compare
that line to
the “Host:”
line for
packet 48.

packet 42 and 48 the host is the
same, so there was no need for
another query in the same session

- g) Examine packet 139. It is one segment of a PDU that is reassembled with several other segments in packet 160. Look at packets 141, 142, and 143. Are these three packets also part of packet 160? What happens if a set of packets that are supposed to be reassembled do not arrive in a continuous stream or do not arrive in the proper order?

- Packets 141 and 142 are not the part of packet 160, however packet 143 is apart of packet 160.
- If a set of packets that are supposed to be reassembled do not arrive in a continuous stream or do not arrive in the proper order, it does not effect the main packet.

• b

- e h) Return to examine frames 141 and 142. Both of these
c are graphics (GIF files) from the same source IP
a address. How does the client know which graphic to
u match up to each get statement? Hint: Click on each
s and look in the middle window for the heading line that
e starts with “Transmission Control Protocol”. What
difference do you see in the heading lines for the two
files? Return to the original “Get” statements. Can you
see the same difference in the “Get” statements?

- They know the difference from their stream index, each one has a different stream index.

III) Exercise Three

Open “Wireshark”, then use the “File” menu and the “Open” command to open the file “Exercise Three.pcap”. You should see 22 packets listed.

These packets represent two different requests for web pages. Packets 1-7 involve the request for the web page www.yahoo.com. Packets 8-22 involve the request for the web page my.usf.edu.

- a) Compare the destination port in the TCP packet in frame 3 with the destination port in the TCP packet in frame 12. What difference do you see? What does this tell you about the difference in the two requests?

i) the port for frame 3 is different for the port in frame 12. this tells me that they use different protocols

The following table compares the two requests for web pages. For example, row i) shows that frames 1-2 and frames 8-9 represent the DNS lookups for each of the web requests.

Row	www.yahoo.com frames	my.usf.edu frames	Brief Explanation of Activity
i)	1-2	8-9	DNS Request to find IP address for common name & DNS Response
ii)	3-5	10-12	Three-way handshake
iii)	--	13-20	
iv)	6	21	“Get” request for web page
v)	7	22	First packet from web server with web page content.

- b) Explain what is happening in row “iii” above. Why are there no frames listed for yahoo in row “iii”?
- i there is no frame because it is encrypted
- c) Look at the “Info” column on frame 6. It says: “GET / HTTP / 1.1. What is the corresponding Info field for the my.usf.com web request (frame 21)? Why doesn’t it read the same as in frame 6?
- i Because frame 6 is HTTP packet which is unencrypted where as packet 21 is uses HTTPS packet which is encrypted

First, you must decide on a web site to visit. Pick a web site, such as www.yahoo.com, or www.cnn.com that you haven’t visited yet today.

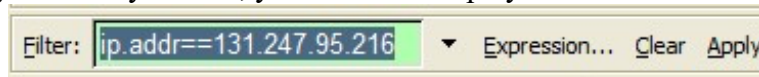
If you pick one that you recently visited, your system may not need to send out a DNS request, since the IP address may be cached (saved temporarily) on your machine. For this exercise, we would prefer to have a DNS request as part of the detail of transactions.

IV) Exercise Four

In this exercise, you are going to capture live traffic from your computer. Open up Wireshark and use the “Capture” menu to save live traffic. The Wireshark “QuickStart” guide distributed with these exercises contains more instructions on using Wireshark.

Start capturing data, visit a live web site using your standard Internet browser, and stop capturing data.

If you have a large amount of network traffic, the relevant data may be hidden among a lot of broadcast messages. To focus on just the key frames, you can set a display filter like this.



For the IP number enter the IP number of your client machine. Type it as shown (ip.addr==your.ip.address) in the graphic above. Then click on “Apply”.

Using an approach similar to the approach in Exercise One, describe the set of frames that you captured.

- For this description think of this as a conversation – every discussion starts with a question and follows with an answer.
- For example, two of the frames will contain the DNS request for an IP address for the web site, and the DNS answer with the IP number.
- Remember that some answers may take several frames if they need to be reassembled from segmented packets.

No.	Time	Source	Destination	Protocol	Length	Info
1344	11.849368	192.168.1.254	192.168.1.93	DNS	Standard query response 0xc236 AAAA encrypted-tbnl.gstatic.com AAAA 2607:f8b0:4009:881::200e	
1345	13.371101	192.168.1.93	192.168.1.254	DNS	Standard query 0x30b6 AAAA www.yahoo.com	
1346	13.383203	192.168.1.254	192.168.1.93	DNS	Standard query response 0x1f9e A www.yahoo.com CNAME new-fp-shed.wgl.b.yahoo.com A 98.137.11.163 A 74.	
1387	13.705983	192.168.1.93	192.168.1.254	DNS	Standard query 0xd507 A s.yimg.com	
1388	13.706025	192.168.1.93	192.168.1.254	DNS	Standard query 0xb916 AAAA s.yimg.com	
1389	13.706094	192.168.1.93	192.168.1.254	DNS	Standard query 0xd3b0 A search.yahoo.com CNAME edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001.	
1390	13.706127	192.168.1.93	192.168.1.254	DNS	Standard query 0xb11e AAAA search.yahoo.com	
1391	13.706182	192.168.1.93	192.168.1.254	DNS	Standard query 0x87ff A 11.at.atwola.com	
1392	13.706210	192.168.1.93	192.168.1.254	DNS	Standard query 0xa042 AAAA 11.at.atwola.com	
1398	13.719099	192.168.1.254	192.168.1.93	DNS	Standard query response 0xd507 A s.yimg.com CNAME edge.gycpi.b.yahoodns.net A 69.147.86.12 A 69.147.86.	
1399	13.719099	192.168.1.254	192.168.1.93	DNS	Standard query response 0xb916 AAAA s.yimg.com CNAME edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001.	
1400	13.719100	192.168.1.254	192.168.1.93	DNS	Standard query response 0xd3b0 A search.yahoo.com CNAME ds-global3.l7.search.ystgl.b.yahoo.com A 66.21.	
1401	13.719100	192.168.1.254	192.168.1.93	DNS	Standard query response 0xb11e AAAA search.yahoo.com CNAME ds-global3.l7.search.ystgl.b.yahoo.com AAAA	
1402	13.719100	192.168.1.254	192.168.1.93	DNS	Standard query response 0x87ff A 11.at.atwola.com CNAME edge.gycpi.b.yahoodns.net A 69.147.86.12 A 69.	
1403	13.719100	192.168.1.254	192.168.1.93	DNS	Standard query response 0xa042 AAAA 11.at.atwola.com CNAME edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.	

No.	Time	Source	Destination	Protocol	Length	Info
1254	12.187315	192.168.1.93	17.42.251.56	TCP	52208 → 993 [ACK] Seq=789 Ack=1390 Win=159 Len=0 TSval=4125136587 TSecr=3509212619	
1270	12.253901	192.168.1.93	199.187.193.179	TCP	[TCP Retransmission] 54346 → 443 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0	
1271	12.293207	17.42.251.56	192.168.1.93	TLSv1	Application Data	
1272	12.293414	192.168.1.93	17.42.251.56	TCP	52208 → 993 [ACK] Seq=789 Ack=1563 Win=159 Len=0 TSval=4125136693 TSecr=3509212726	
1273	12.300408	192.168.1.93	17.42.251.56	TLSv1	Application Data	
1284	12.353061	17.42.251.56	192.168.1.93	TCP	993 → 52208 [ACK] Seq=1563 Ack=832 Win=110 Len=0 TSval=3509212787 TSecr=4125136700	
1300	12.478835	192.168.1.93	66.225.223.95	TCP	[TCP Retransmission] 54327 → 443 [FIN, ACK] Seq=1 Ack=1 Win=2048 Len=0 TSval=3544835972 TSecr=3807575	
1319	12.597995	17.42.251.56	192.168.1.93	TLSv1	Application Data	
1320	12.598151	192.168.1.93	17.42.251.56	TCP	52208 → 993 [ACK] Seq=832 Ack=1602 Win=159 Len=0 TSval=4125136998 TSecr=3509213031	
1341	13.242013	192.168.1.93	224.0.0.251	MDNS	Standard query response 0x0000 TXT, cache flush NSEC, cache flush MacBook Air, companion-Link_tcp.lo	
1343	13.371035	192.168.1.93	192.168.1.254	DNS	Standard query 0x1f9e A www.yahoo.com	
1344	13.371103	192.168.1.93	192.168.1.254	DNS	Standard query 0x50b6 AAAA www.yahoo.com	
1345	13.383203	192.168.1.254	192.168.1.93	DNS	Standard query response 0x1f9e A www.yahoo.com CNAME new-fp-shed.wgl.b.yahoo.com A 98.137.11.163 A 74.	
1346	13.383204	192.168.1.254	192.168.1.93	DNS	Standard query response 0x50b6 AAAA www.yahoo.com CNAME new-fp-shed.wgl.b.yahoo.com AAAA 2001:4998:12.	
1387	13.705983	192.168.1.93	192.168.1.254	DNS	Standard query 0xd507 A s.yimg.com	
1388	13.706025	192.168.1.93	192.168.1.254	DNS	Standard query 0xb916 AAAA s.yimg.com	

No.	Time	Source	Destination	Protocol	Len	Info
1389	13.706094	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xd3b0 A search.yahoo.com
1390	13.706127	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xb11e AAAA search.yahoo.com
1391	13.706182	192.168.1.93	192.168.1.254	DNS	-	Standard query 0x87ff A 11.at.atwola.com
1392	13.706210	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xa042 AAAA 11.at.atwola.com
1398	13.719099	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xd507 A s.yimg.com CNAME edge.gycpi.b.yahoodns.net A 69.147.86.12 A 69.147.86.12
1399	13.719099	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xb916 AAAA s.yimg.com CNAME edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001
1400	13.719100	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xd3b0 A search.yahoo.com CNAME ds-global3.l7.search.ystgl.b.yahoo.com A 66.21.14.16 A 66.21.14.16
1401	13.719100	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xb11e AAAA search.yahoo.com CNAME ds-global3.l7.search.ystgl.b.yahoo.com AAAA 2001:4998:20.0001
1402	13.719100	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0x87ff A 11.at.atwola.com CNAME edge.gycpi.b.yahoodns.net A 69.147.86.12 A 69.147.86.12
1403	13.719100	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xa042 AAAA 11.at.atwola.com CNAME edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001
1407	13.720499	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xb0a6 A video-api.yql.yahoo.com
1408	13.720536	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xc511 AAAA video-api.yql.yahoo.com CNAME v4-edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001
1409	13.720506	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xb0a6 A geo.yahoo.com
1410	13.720614	192.168.1.93	192.168.1.254	DNS	-	Standard query 0xb0a6 A geo.yahoo.com
1416	13.734644	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xb0a6 A video-api.yql.yahoo.com CNAME v4-edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001
1417	13.734645	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xc511 AAAA video-api.yql.yahoo.com CNAME v4-edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001
1418	13.734645	192.168.1.254	192.168.1.93	DNS	-	Standard query response 0xb0a6 A video-api.yql.yahoo.com CNAME v4-edge.gycpi.b.yahoodns.net AAAA 2001:4998:20.0001

frame 1343 and 1344 initiated the request and 1345-1346 responded to the request. the next following frames are requests and response to things like the search bar and other thing. We can the ip address of those behind the scenes in the queries for the following frames

V) Exercise Five

In this exercise, you are going to capture live traffic from your email service. Begin by opening up your email service and preparing an email from you to yourself – you can use a subject line of “Homework” and a body of “Homework” if you chose. **Don’t hit the send button yet.**

Open up Wireshark, go to the “Capture” menu and tell it to “Start” capturing data. Now return to your email service and hit the send button. After the email is sent, tell Wireshark to “Stop” capturing data.

Look in the protocol column and see what protocol your email server uses. There are several choices.

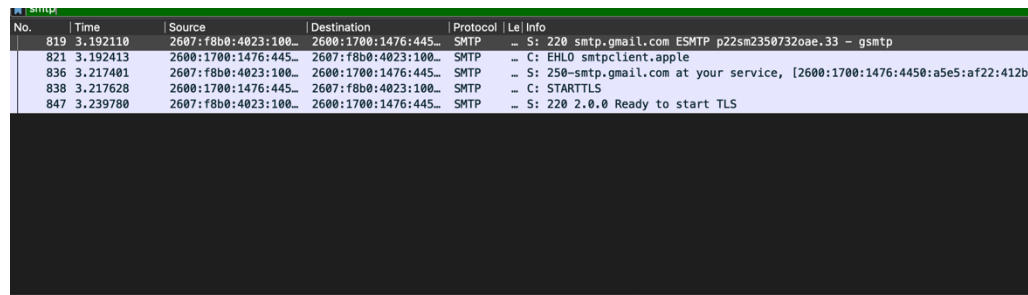
- If you use a local client, and your email server **does not** support secure communications, then you might see

something like SMTP. SMTP is an application running on top of TCP. Wireshark will recognize SMTP and report that the packet is in SMTP format.

- b. If you use a local client, and your email server **does** support secure communications, then you might only see TCP. Since communications are secure, Wireshark will not be able to break the encryption. It will not recognize the format of the TCP payload; therefore, it will simply report a TCP message and show the encrypted data.
- c. If you use a web browser to access email through a regular http:// web address, you will see commands that look like you are using web forms. For example, you might see a HTTP Post command.
- d. If you use a secure web page (https://), you will see TCP with an encrypted payload, similar to item 2) above. However, the process of setting up the encryption will look different since setting up a secure web page is different from setting up a secure email connection.

What protocol did your system use?

My mail service uses SMPT to send the email



No.	Time	Source	Destination	Protocol	Length	Info
819	3.192110	2607:f8b0:4023:100...	2600:1700:1476:445...	SMTP	5	220 smtp.gmail.com ESMTP p22sm23507320ae.33 - gsmt
821	3.192413	2600:1700:1476:445...	2607:f8b0:4023:100...	SMTP	100	C: EHLO smtpclient.apple
836	3.217401	2607:f8b0:4023:100...	2600:1700:1476:445...	SMTP	100	S: 250-smtp.gmail.com at your service, [2600:1700:1476:4450:a5e5:af22:412b...
838	3.217628	2600:1700:1476:445...	2607:f8b0:4023:100...	SMTP	100	C: STARTTLS
847	3.239780	2607:f8b0:4023:100...	2600:1700:1476:445...	SMTP	100	S: 220 2.0.0 Ready to start TLS