hw8

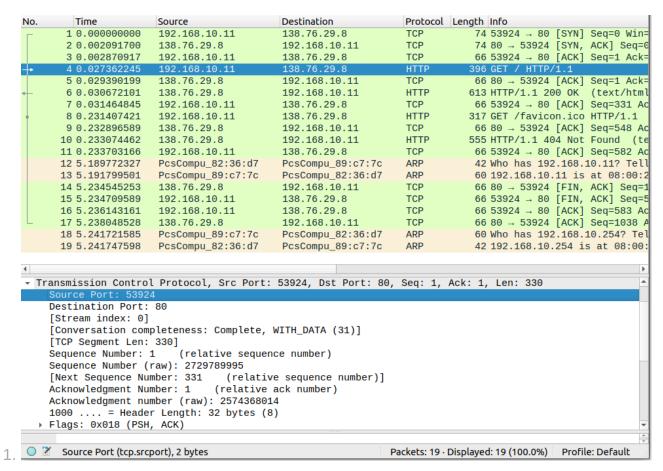
- 1. a) I will assume that the host has received its IP address and knows the IP address of the router since this process is usually done before ARP and the table is already filled
 - 1. Host looks at ARP table to get the MAC address for the first router (router 2) based on the IP address (MAC is 77-77-77-77-77 and dest IP is 182.162.3.1). The host then creates and sends an ethernet frame for the aformentioned MAC address
 - 2. The router (2) gets the frame. It unwraps it to get the datagram and determines the next step to be Router 1 (182.168.2.1). It checks the ARP table and determines this is MAC 33-33-33-33-33. It then creates the ethernet frame and sends it there accordingly
 - 3. Router 1 then gets the frame and follows a similar process. From the datagram it gets the next destination host B with IP 182.168.1.003. From there the ARP table says it is MAC 22-22-22-22-22 where the frame is then created and sent.
 - 4. Host B gets the frame and unwraps it to get the message b) Once again, assume DHCP is done and the address of the router is known 1. The first step is to send an ARP query to FF-FF-FF-FF-FF. This will broadcast that we are looking for where to send the packets to because we do not know where so we need to find the MAC address for Host E to send the packet to. From there, Router 2 will will recieve this packet and send a response to Host B saying to send the packsts to 77-77-77-77-77. This response will go to the sender of the query which is 88-88-88-88-88-88.
 - 2. After that everything is the same

2.

Action	Known MACs	Links sent to	Explanation
A->D	Α	B, C, D	Initially the table is empty. When the packet is received, A is added to the ARP table. From there since the destination B is not known, the links are "flooded" or in other words the info is sent to all the links.
D->A	A, D	Α	The switch recieves the packet and adds D to the ARP table. From there the packet is forwarded to A since A is already in the ARP table
C->A	A, D, C	Α	The switch recieves the packet and adds C to the ARP table. From there the packet is forwarded to A since A is already in the ARP table
A->C	A, D, C	С	The switch recieves the packet. A is in the ARP table so it doesn't change. From there the packet is forwarded to C since C is already in the ARP table

3. Facebook updated their border gateway protocol which withdrew their IP addresses from global routing tables. Since their authoritative DNS was now unreachable, facebook.com was impossible to resolve so you could no longer access the website. This was propogated by a lot of Facebook internal servers being down so it took forever for it to come back.

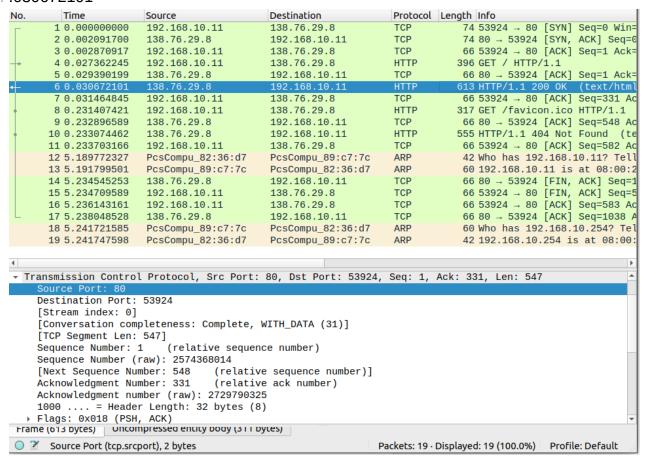
Part 2



The source is 192.168.10.11:53924

The dest is 138.76.29.8:80

2..030672101



3. Same image as 2

The source is 138.76.29.8:80

The dest is 192.168.10.11:53924

4. .027356291

No.	Time	Source	Destination	Protocol	Length Info				
Г	1 0.000000000	10.0.1.254	138.76.29.8	TCP	74 53924 → 80 [SYN] Seq=0 Win=				
	2 0.002058086	138.76.29.8	10.0.1.254	TCP	74 80 → 53924 [SYN, ACK] Seq=6				
	3 0.002853940	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Seq=1 Ack=				
+	4 0.027356291	10.0.1.254	138.76.29.8	HTTP	396 GET / HTTP/1.1				
	5 0.029338911	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Seq=1 Ack=				
4	6 0.030625966	138.76.29.8	10.0.1.254	HTTP	613 HTTP/1.1 200 OK (text/htm)				
	7 0.031448670	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Seq=331 Ac				
+	8 0.231400190	10.0.1.254	138.76.29.8	HTTP	317 GET /favicon.ico HTTP/1.1				
	9 0.232863610	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Seq=548 Ac				
	10 0.233043313	138.76.29.8	10.0.1.254	HTTP	555 HTTP/1.1 404 Not Found (te				
	11 0.233687113	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Seq=582 Ac				
	12 5.189837924	PcsCompu_43:65:cd	PcsCompu_22:fd:74	ARP	42 Who has 10.0.1.253? Tell 10				
	13 5.191700729	PcsCompu_22:fd:74	PcsCompu_43:65:cd	ARP	60 10.0.1.253 is at 08:00:27:2				
	14 5.231662506	PcsCompu_22:fd:74	PcsCompu_43:65:cd	ARP	60 Who has 10.0.1.254? Tell 10				
	15 5.231707677	PcsCompu_43:65:cd	PcsCompu_22:fd:74	ARP	42 10.0.1.254 is at 08:00:27:4				
	16 5.234487950	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [FIN, ACK] Seq=1				
	17 5.234707098	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [FIN, ACK] Seq=5				
	18 5.236144683	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Seq=583 Ac				
L	19 5.238001105	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Seq=1038 A				
4					<u> </u>				
					on interface eth0, id 0				
					_22:fd:74 (08:00:27:22:fd:74)				
			1.254, Dst: 138.76.29						
			: 53924, Dst Port: 80	, Seq: 1, <i>i</i>	Ack: 1, Len: 330				
	Source Port: 5392								
	Destination Port:	80							
[Stream index: 0]									
[Conversation completeness: Complete, WITH_DATA (31)]									
[TCP Segment Len: 330]									
Sequence Number: 1 (relative sequence number)									
Sequence Number (raw): 2729789995									
[Next Sequence Number: 331 (relative sequence number)]									
Acknowledgment Number: 1 (relative ack number)									
0 2	Source Port (tcp.srcport), 2 bytes Packets: 19 · Displayed: 19 (100.0%) Profile: Default								
Source Fore (capital porty) 2 System									

5. Same pic as 4

The source is 10.0.1.254:53924

The dest is 138.76.29.8:80

Interestingly, the source IP is different but the source port is not. The rest are the same

6..030625966

No.	Time	Source	Destination	Protocol	Length Info				
	1 0.000000000	10.0.1.254	138.76.29.8	TCP	74 53924 → 80 [SYN] Se	q=0 Win=			
	2 0.002058086	138.76.29.8	10.0.1.254	TCP	74 80 → 53924 [SYN, AC	K] Seq=0			
	3 0.002853940	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Se	q=1 Ack=			
	4 0.027356291	10.0.1.254	138.76.29.8	HTTP	396 GET / HTTP/1.1				
	5 0.029338911	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Se	q=1 Ack=			
+	6 0.030625966	138.76.29.8	10.0.1.254	HTTP	613 HTTP/1.1 200 OK (t				
	7 0.031448670	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Se				
+	8 0.231400190	10.0.1.254	138.76.29.8	HTTP	317 GET /favicon.ico HT	TP/1.1			
	9 0.232863610	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Se				
+	10 0.233043313	138.76.29.8	10.0.1.254	HTTP	555 HTTP/1.1 404 Not Fo	und (te			
	11 0.233687113	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Se	q=582 Ac			
	12 5.189837924	PcsCompu_43:65:cd	PcsCompu_22:fd:74	ARP	42 Who has 10.0.1.253?	Tell 10			
	13 5.191700729	PcsCompu_22:fd:74	PcsCompu_43:65:cd	ARP	60 10.0.1.253 is at 08	3:00:27:2			
	14 5.231662506	PcsCompu_22:fd:74	PcsCompu_43:65:cd	ARP	60 Who has 10.0.1.254?	Tell 10			
	15 5.231707677	PcsCompu_43:65:cd	PcsCompu_22:fd:74	ARP	42 10.0.1.254 is at 08	3:00:27:4			
	16 5.234487950	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [FIN, AC	K] Seq=1			
	17 5.234707098	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [FIN, AC	K] Seq=5			
	18 5.236144683	10.0.1.254	138.76.29.8	TCP	66 53924 → 80 [ACK] Se	q=583 Ac			
L	19 5.238001105	138.76.29.8	10.0.1.254	TCP	66 80 → 53924 [ACK] Se	q=1038 A			
4									
					on interface eth0, id 0				
					_43:65:cd (08:00:27:43:65:	cd)			
Finternet Protocol Version 4, Src: 138.76.29.8, Dst: 10.0.1.254									
Transmission Control Protocol, Src Port: 80, Dst Port: 53924, Seq: 1, Ack: 331, Len: 547									
Source Port: 80									
Destination Port: 53924									
[Stream index: 0]									
[Conversation completeness: Complete, WITH_DATA (31)]									
[TCP Segment Len: 547]									
Sequence Number: 1 (relative sequence number)									
Sequence Number (raw): 2574368014									
	[Next Sequence Number: 548 (relative sequence number)]								
Acknowledgment Number: 331 (relative ack number)									
Frame (613 byces) Uncompressed entity body (311 bytes)									
	Source Port (tcp.srcport), 2 bytes Packets: 19 · Displayed: 19 (100.0%) Profile: Default								

7. Same image as above

The source is 138.76.29.8:80

The dest is 10.0.1.254:53924

8. Since the NAT protocol essentially replaces the headers at the router with what they're supposed to be, it will be exactly how the host expects

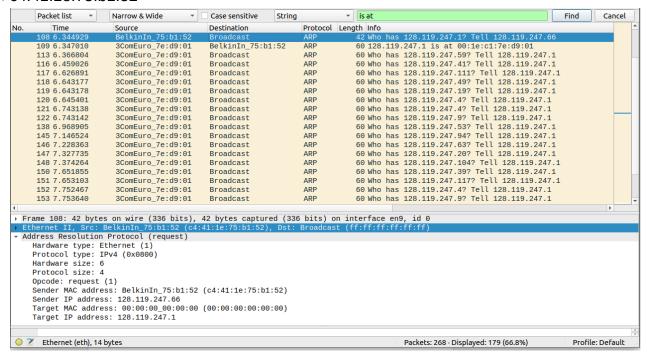
The source is 138.76.29.8:80

The dest is 192.168.10.11:53924

Lab 2

This lab was a bit confusing because there was only one response to any of the ARP messages. I will assume the sender who got a response is the computer

1. c4:41:1e:75:b1:52



- 2. ff:ff:ff:ff:ff (above picture)
- 3. 128.119.247.66 (above pic)
- 4. 128.119.247.1 (above pic)
- 5. 00:1e:c1:7e:d9:01

