

CA169 Networks Assignment Two

Answer Sheets

STUDENT NAME:	CONOR JOYCE
STUDENT NUMBER:	19425804
PROJECT NUMBER:	2
MODULE CODE:	CA169
DEGREE: {CA EC CPSSD ECSA}	CA
LECTURER:	Dr Michael Scriney

Declaration

In submitting this project, I declare that the project material, which I now submit, is my own work. Any assistance received by way of borrowing from the work of others has been cited and acknowledged within the work. I make this declaration in the knowledge that a breach of the rules pertaining to project submission may carry serious consequences.

Part 1: DHCP traffic

Your IP & MAC address for this experiment (use ipconfig)

192.168.1.8

E0-D5-5E-24-E5-32

Screen capture: ipconfig information cmd window

```
Administrator: C:\Windows\System32\cmd.exe

fec0:0:0:ffff::2%1
fec0:0:0:ffff::3%1
NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Ethernet:

Connection-specific DNS Suffix  . : home
Description . . . . . : Realtek PCIe GbE Family Controller
Physical Address. . . . . : E0-D5-5E-24-E5-32
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::1063:e25f:ebc6:bb8d%5(Preferred)
IPv4 Address. . . . . : 192.168.1.8(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Monday 13 April 2020 18:37:16
Lease Expires . . . . . : Tuesday 14 April 2020 18:44:03
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DHCPv6 IAID . . . . . : 81843550
DHCPv6 Client DUID. . . . . : 00-01-00-01-23-A8-43-E2-E0-D5-5E-24-E5-32
DNS Servers . . . . . : 1.1.1.1
                        1.0.0.1
NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Ethernet 3:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix  . :
Description . . . . . : TAP-ProtonVPN Windows Adapter V9
Physical Address. . . . . : 00-FF-7B-C5-2D-30
```

IPv4 Address: my ip

Physical Address: my mac address

Screen capture of Wireshark with DHCP and all ARP packets shown.

*Ethernet						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
dhc or arp						
No.	Time	Source	Destination	Protocol	Length	Info
13	2.092823	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.4? Tell 192.168.1.8
14	2.172741	Apple_1a:08:ce	Giga-Byt_24:e5:32	ARP	60	192.168.1.4 is at 58:e2:8f:1a:08:ce
66	5.760012	192.168.1.8	192.168.1.1	DHCP	342	DHCP Release - Transaction ID 0xf19d8c3e
192	9.486023	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x1aaa2ac7
193	9.487257	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.9? Tell 192.168.1.1
194	9.494770	192.168.1.1	255.255.255.255	DHCP	348	DHCP Offer - Transaction ID 0x1aaa2ac7
195	9.495353	0.0.0.0	255.255.255.255	DHCP	366	DHCP Request - Transaction ID 0x1aaa2ac7
198	9.501822	192.168.1.1	255.255.255.255	DHCP	348	DHCP ACK - Transaction ID 0x1aaa2ac7
199	9.511211	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
214	9.597830	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.8
216	9.598413	HuaweiTe_b6:f5:88	Giga-Byt_24:e5:32	ARP	60	192.168.1.1 is at a8:f5:ac:b6:f5:88
219	9.601038	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
220	9.601050	Giga-Byt_24:e5:32	HuaweiTe_b6:f5:88	ARP	42	192.168.1.8 is at e0:d5:5e:24:e5:32
229	9.636586	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.8
230	9.636895	HuaweiTe_b6:f5:88	Giga-Byt_24:e5:32	ARP	60	192.168.1.1 is at a8:f5:ac:b6:f5:88
240	9.656249	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.8? (ARP Probe)
685	10.472730	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.3? Tell 192.168.1.8
735	10.568303	SamsungE_17:5c:ed	Giga-Byt_24:e5:32	ARP	60	192.168.1.3 is at 38:01:95:17:5c:ed
770	10.656191	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.8? (ARP Probe)
1235	11.656211	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.8? (ARP Probe)
1537	12.517041	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.11? Tell 192.168.1.8
1612	12.617757	BSkyB_10:c1:30	Giga-Byt_24:e5:32	ARP	60	192.168.1.11 is at c0:3e:0f:10:c1:30
1657	12.656219	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 169.254.187.141? (ARP Probe)
1658	12.656236	Giga-Byt_24:e5:32	Broadcast	ARP	42	ARP Announcement for 192.168.1.8
1669	12.662961	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
1670	12.662978	Giga-Byt_24:e5:32	HuaweiTe_b6:f5:88	ARP	42	192.168.1.8 is at e0:d5:5e:24:e5:32

Filter is set to “dhc or arp” to show only these protocols

Packet numbers relevant to the DHCP interaction:

- a. DHCP DISCOVER
 - 192
- b. DHCP OFFER
 - 194
- c. DHCP Request
 - 195
- d. DHCP Acknowledgement
 - 198
- e. DHCP Release (if you release using `ipconfig /release`)
 - 66
- f. All ARP packets used
 - 13, 14, 193, 199-1670

Function of each packet

- a. DHCP DISCOVER
 - Find dhcp server
- b. DHCP OFFER
 - Router offering an ip to a device
- c. DHCP Request
 - Device accepting the ip being offered
- d. DHCP Acknowledgement
 - Router acknowledging device accepting its new ip
- e. DHCP Release (if you release using `ipconfig /release`)
 - Device telling router it doesn't want its ip address anymore
- f. ARP
 - Devices on network figuring out which device has what ip

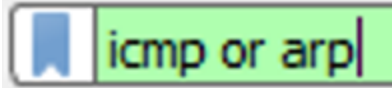
Part 2: ping traffic

Your IP & MAC address for this experiment (use ipconfig)

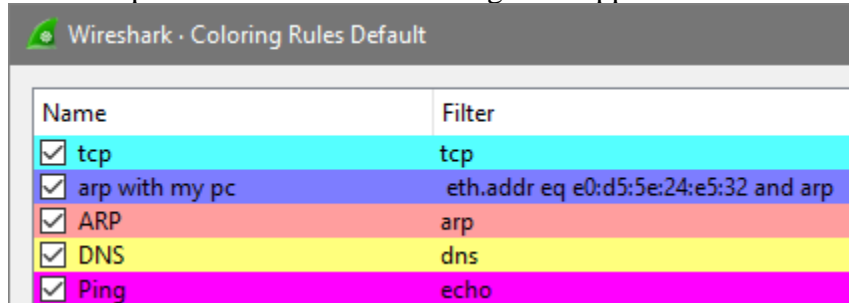
192.168.1.101

E0-D5-5E-24-E5-32

Screen capture of Wireshark filter utilised



Screen capture of Wireshark colouring rules applied



Screen capture of Wireshark packet trace showing all relevant ping generated traffic, including ARP and ICMP traffic.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	SamsungE_17:5c:red	Giga-Byt_24:e5:32	ARP	60	Who has 192.168.1.101? Tell 192.168.1.3
2	0.000019	Giga-Byt_24:e5:32	SamsungE_17:5c:red	ARP	42	192.168.1.101 is at e0:d5:5e:24:e5:32
17	2.343702	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.101
18	2.344038	HuaweiTe_b6:f5:88	Giga-Byt_24:e5:32	ARP	60	192.168.1.1 is at a6:f5:ac:b6:f5:88
33	2.972351	192.168.1.101	74.125.193.99	ICMP	74	Echo (ping) request id=0x0001, seq=11/2816, ttl=128 (reply in 34)
34	2.982549	74.125.193.99	192.168.1.101	ICMP	74	Echo (ping) reply id=0x0001, seq=11/2816, ttl=59 (request in 33)
62	4.466676	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.4? Tell 192.168.1.101
63	4.503015	Apple_1a:08:ce	Giga-Byt_24:e5:32	ARP	60	192.168.1.4 is at 58:e2:8f:1a:08:ce
544	16.472014	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.11? Tell 192.168.1.101
545	16.483113	BSkyB_10:c1:30	Giga-Byt_24:e5:32	ARP	60	192.168.1.11 is at c0:3e:0f:10:c1:30
605	17.654007	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
627	18.653758	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
632	19.653541	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.8? Tell 192.168.1.1
653	20.437819	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.11? Tell 192.168.1.1
656	21.277131	HuaweiTe_b6:f5:88	Broadcast	ARP	60	Who has 192.168.1.11? Tell 192.168.1.1

All pings (echo request and reply) are in pink

ARP packets involving my pc are in blue

ARP broadcast from router is in red

Packet numbers relevant to the experiment:

Explanation for each packet

- Function
- Explain why it is generated
- Explain the data contained in the packet

33:

- echo request (pinging a server)
- generated by device expecting a response from a server
- packet contains type 8 meaning echo request, code 0, checksum, identifier, seq. no. and the payload which is garbage text ending in hi

Payload →

```
..^$.2.. .....E..
.<..s..@.. Q.....
.e..UB.. ..abcdef
ghijklmn opqrstuv
wabcdefg hi
```

34:

- echo reply (response from server)
- sent from server after receiving an echo request
- type 0 (echo reply), code 0, checksum, identifier, seq. no. and the same payload

17:

- arp broadcast
- my pc asking what mac addresses have a certain ip
- packet contains the ip whose mac address is being searched for, and the senders ip

18:

- arp response
- my router responding to my pc's broadcast telling my pc the routers mac address
- packets contains the ip being asked for and the mac address who owns that ip

Part 3:

Your IP & MAC address for this experiment (use ipconfig)

192.168.1.101	E0-D5-5E-24-E5-32
---------------	-------------------

Filter to show only traffic concerning the test machine

Filter	(ip.addr == 192.168.1.101 and ((tcp.stream eq 2 and not http) or dns contains "sothatwemaybefree")) or (eth.addr == E0-D5-5E-24-E5-32 and arp)
--------	--

Explain how you found the start of the interaction between your PC and the website.

- noticed my pc talking to dns server and after dns response, my pc initiates 3-way tcp handshake with the website

Wireshark window showing the start of the interaction (should show ARP, DNS and TCP 3-way handshake)

The image shows a Wireshark packet capture window for a file named '3.cap'. The filter bar contains the expression: (ip.addr == 192.168.1.101 and ((tcp.stream eq 2 and not http) or dns contains "sothatwemaybefree")) or (eth.addr == E0-D5-5E-24-E5-32 and arp). The packet list shows the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
5	2.744411	Giga-Byt_24:e5:32	Broadcast	ARP	42	Who has 192.168.1.1? Tell 192.168.1.101
6	2.744852	HuaweiTe_b6:f5:88	Giga-Byt_24:e5:32	ARP	60	192.168.1.1 is at a8:f5:ac:b6:f5:88
11	4.096244	192.168.1.101	1.1.1.1	DNS	81	Standard query 0xd388 A sothatwemaybefree.com
13	4.205573	1.1.1.1	192.168.1.101	DNS	118	Standard query response 0xd388 A sothatwemaybefree.com A 213.186.33.40
14	4.206318	192.168.1.101	213.186.33.40	TCP	66	6080 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
15	4.235758	213.186.33.40	192.168.1.101	TCP	60	80 → 6080 [SYN, ACK] Seq=0 Ack=1 Win=17520 Len=0 MSS=1412
16	4.235851	192.168.1.101	213.186.33.40	TCP	54	6080 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
20	4.309512	192.168.1.101	213.186.33.40	TCP	54	6080 → 80 [ACK] Seq=438 Ack=477 Win=63764 Len=0

Write down the numbers of the packets with the 3-way handshake.

Explain what is happening with these 3 packets.

- 14: syn, initialise connection
- 15: syn from server + ack of previous syn
- 16: ack from device

Write down a filter to show only these three-way-handshake packets

Filter	tcp.stream eq 2 and not http
--------	------------------------------

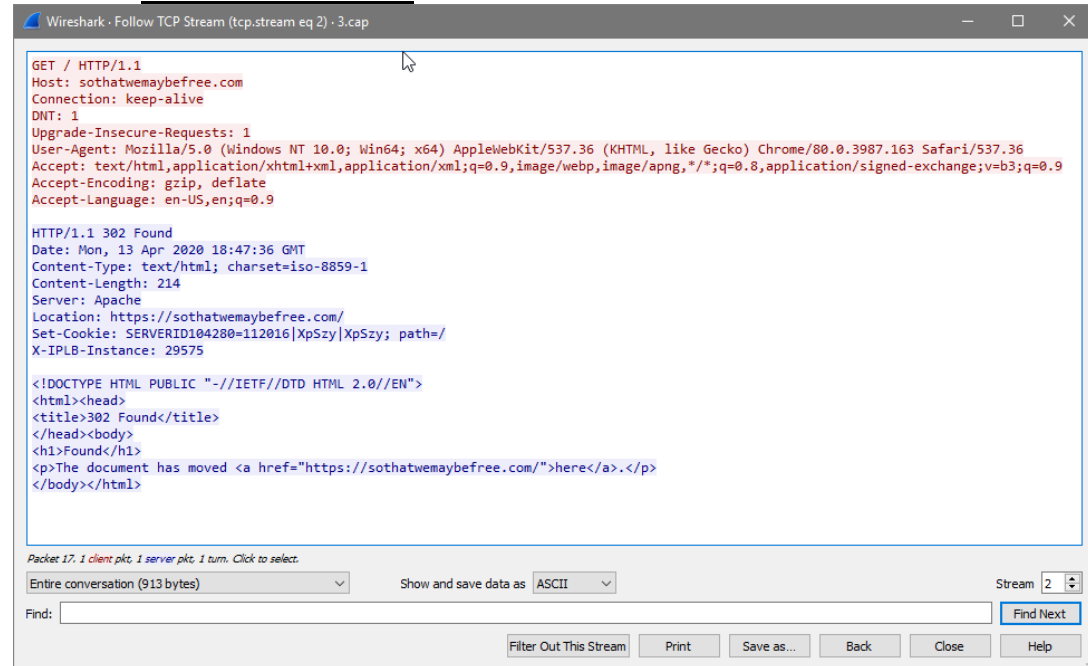
Wireshark window for the 3-way-handshake

The image shows a Wireshark packet capture window for a file named '3.cap'. The filter bar contains the expression: tcp.stream eq 2 and not http. The packet list shows the following packets:

No.	Time	Source	Destination	Protocol	Length	Info
14	4.206318	192.168.1.101	213.186.33.40	TCP	66	6080 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
15	4.235758	213.186.33.40	192.168.1.101	TCP	60	80 → 6080 [SYN, ACK] Seq=0 Ack=1 Win=17520 Len=0 MSS=1412
16	4.235851	192.168.1.101	213.186.33.40	TCP	54	6080 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
20	4.309512	192.168.1.101	213.186.33.40	TCP	54	6080 → 80 [ACK] Seq=438 Ack=477 Win=63764 Len=0

First 3 packets are the handshake (syn, syn/ack, and ack)

Show the **Follow TCP Stream** window here.



Your notes on...

- The GET requests made
 - There is a get request for the html of the website
- The responses from the server
 - After the get request the server responds with the location of the html which is just <https://sothatwemaybefree.com/> - the typical location, so this is a fairly basic website
- The HTTP response codes used in the interaction and what they mean (look them up yourself on the Web)
 - 302 Found – redirects us to the location of the html file