MARTIN MWANGI NJOROGE

mwangimartins650@gmail.com

CS-SA08-24031

L2 MAC Flooding & ARP Spoofing

Here's a link to my finished room,

https://tryhackme.com/r/room/layer2

1.0 Introduction

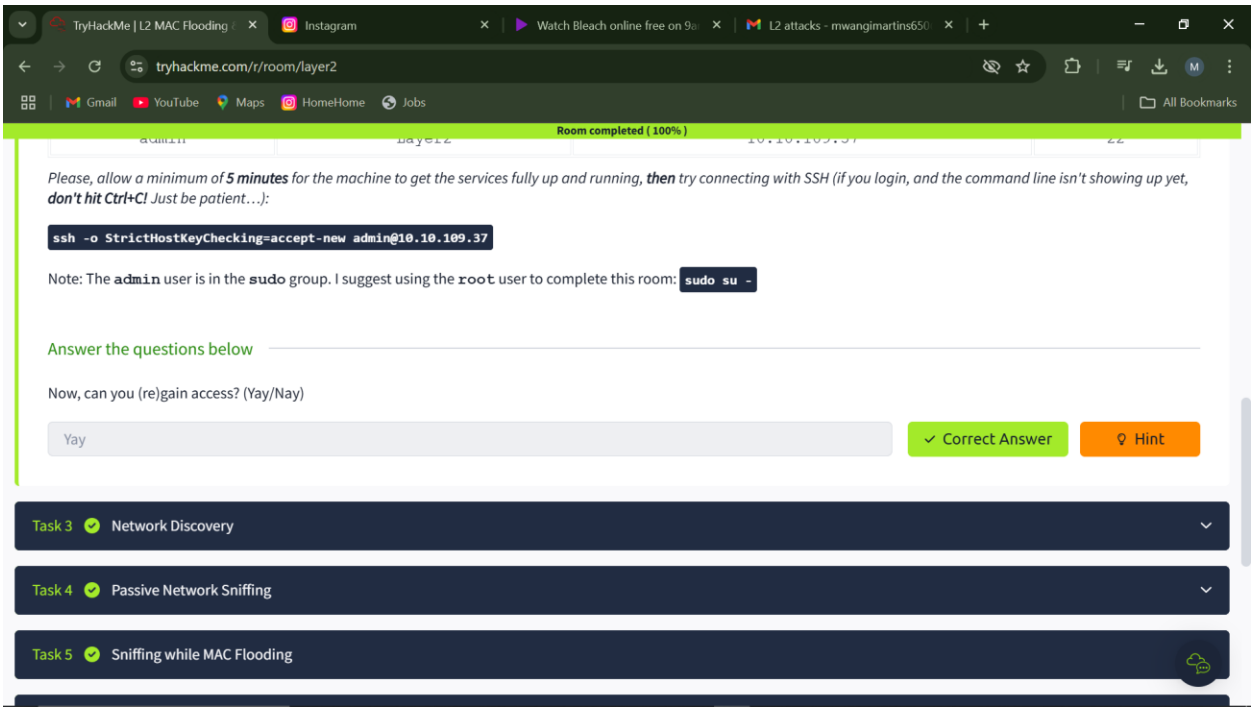
This room focuses on exploit Layer 2, Data Link Layer, of the OSI model, that is responsible to frame forwarding and MAC addressing. Some these Layer 2 exploits are MAC address flooding, ARP spoofing and Man-In-The-Middle Attacks.

1.1 Initial Access

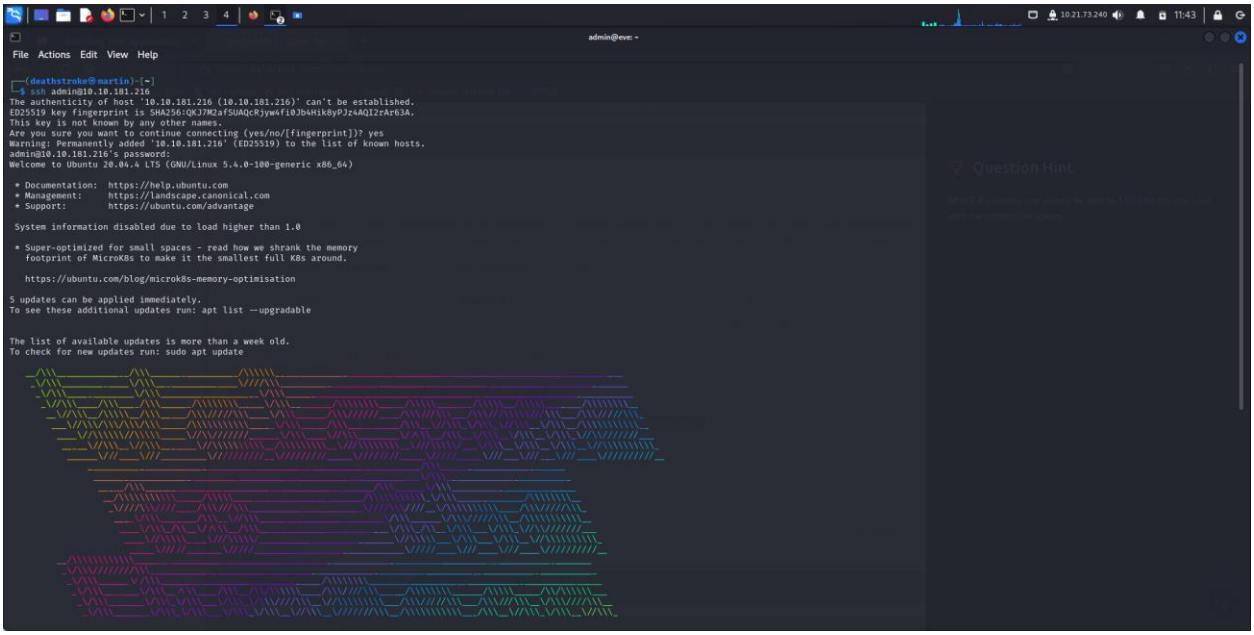
Scenario

While conducting a pentest, you have gained initial access to a network and escalated privileges to root on a Linux machine. During your routine OS enumeration, you realize it's a dual-honed host, meaning it is connected to two (or more) networks. Being the curious hacker you are, you decided to explore this network to see if you can move laterally.

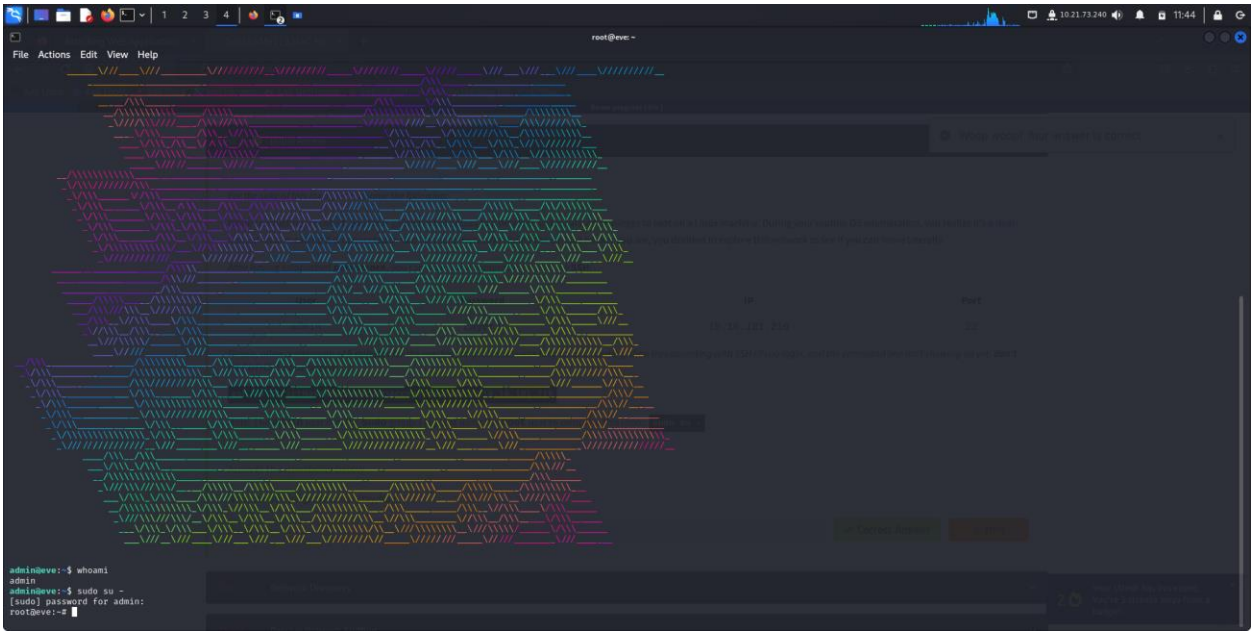
Questions



I login using ssh protocol and username(admin) , Target IP, and Password(Layer2),

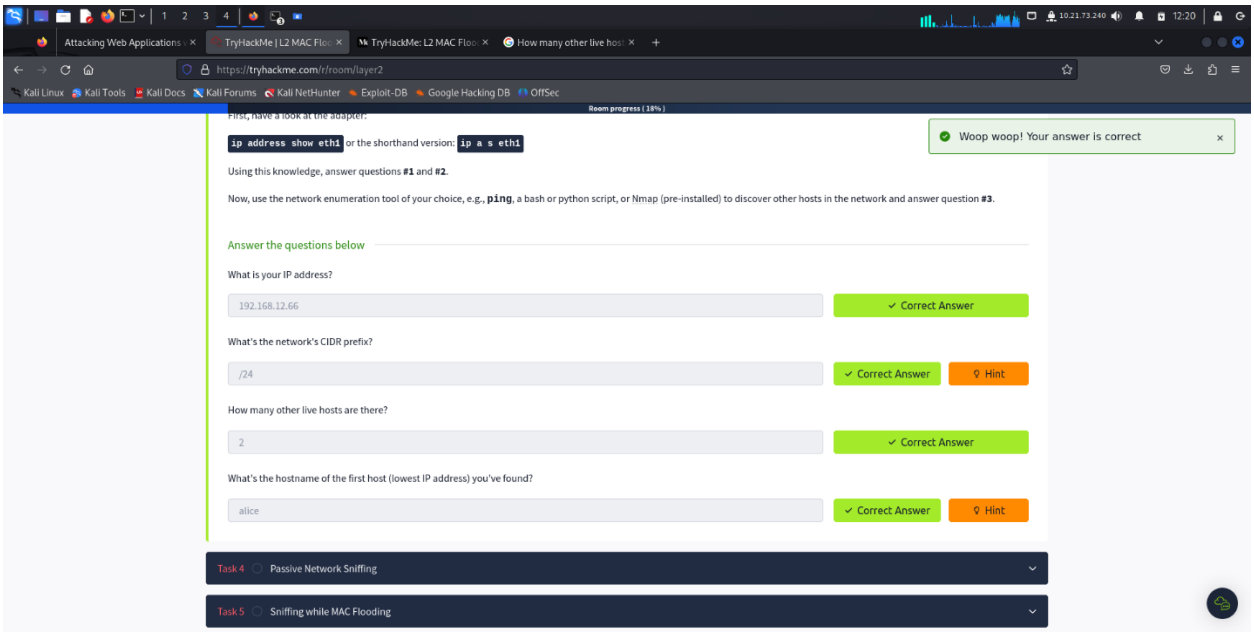


Using command *sudo su* - I regained access as root user,



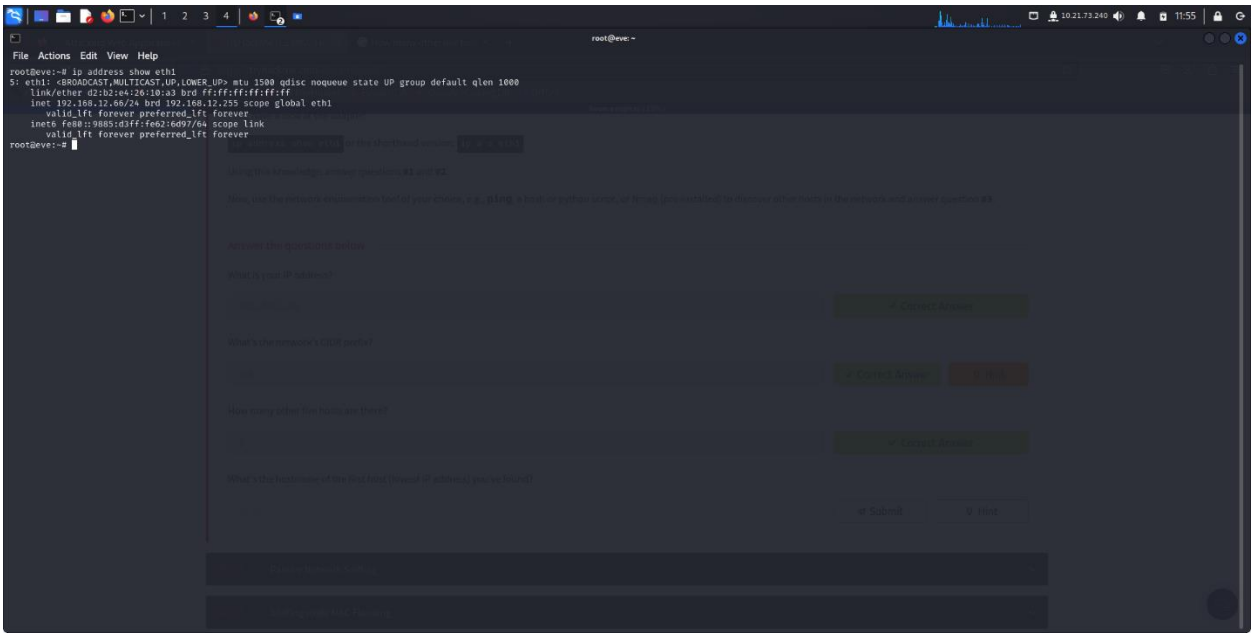
1.2 Network Discovery

Questions

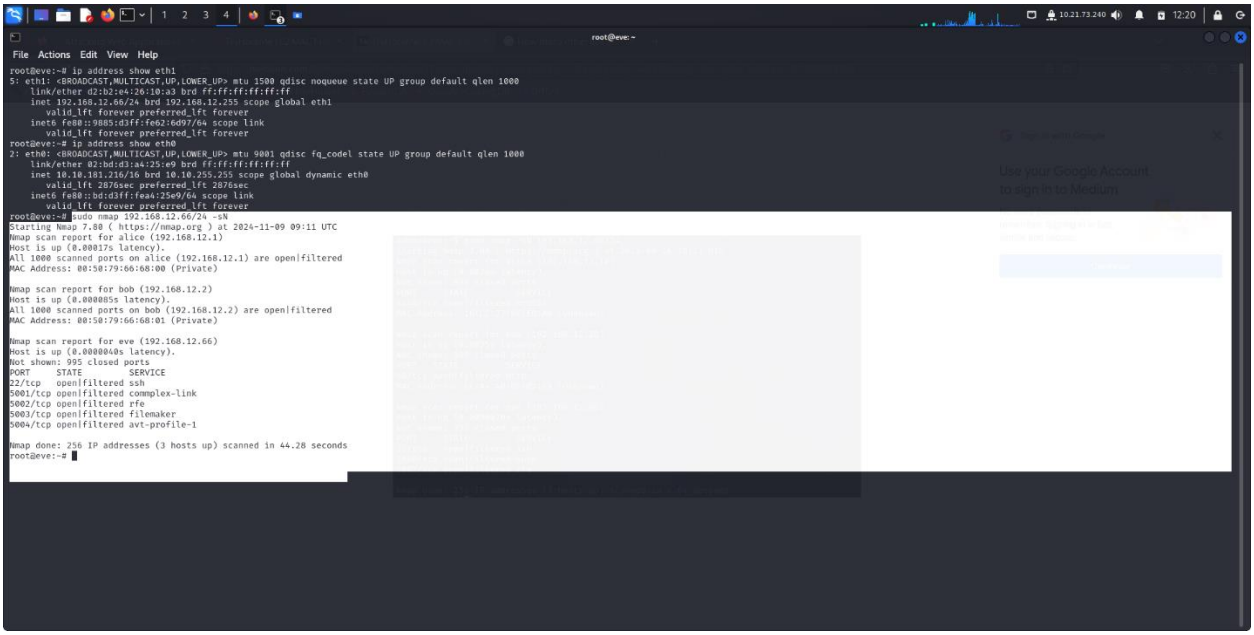


With already using eth0 for access to the system, I explored for eth1 being used with the target's network to communicate with other targets in that network.

Using command *IP address show eth1* I found the target having a class C IP address with /24 subnet mask and Broadcast address in the same class C.

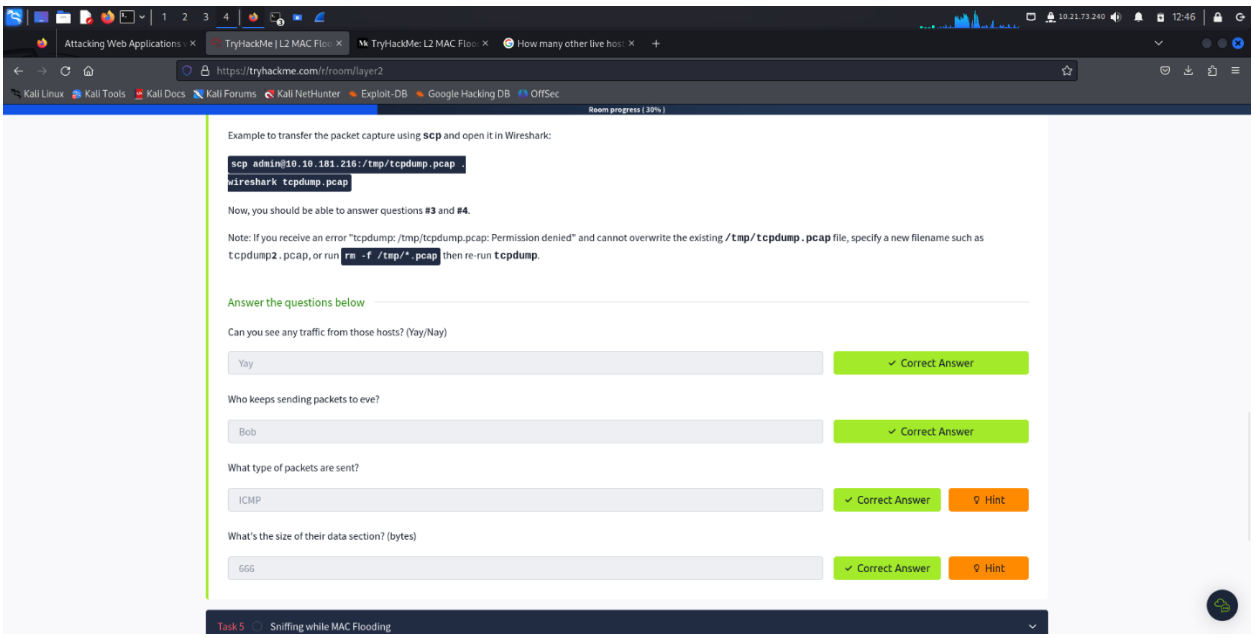


To find the host of the lowest IP, I did a host discovery scan with nmap, by including flag, *-sN* to prevent port scanning, and the target IP with Subnet. As the results came found the host to be alice,

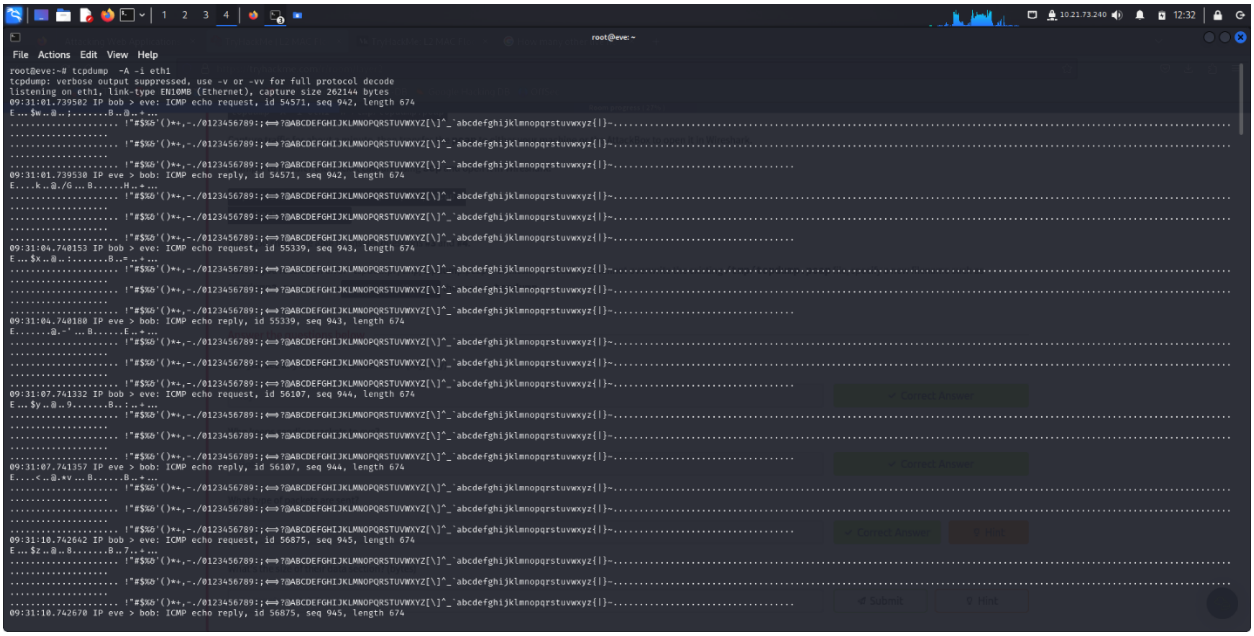


1.3 Passive Network Sniffing

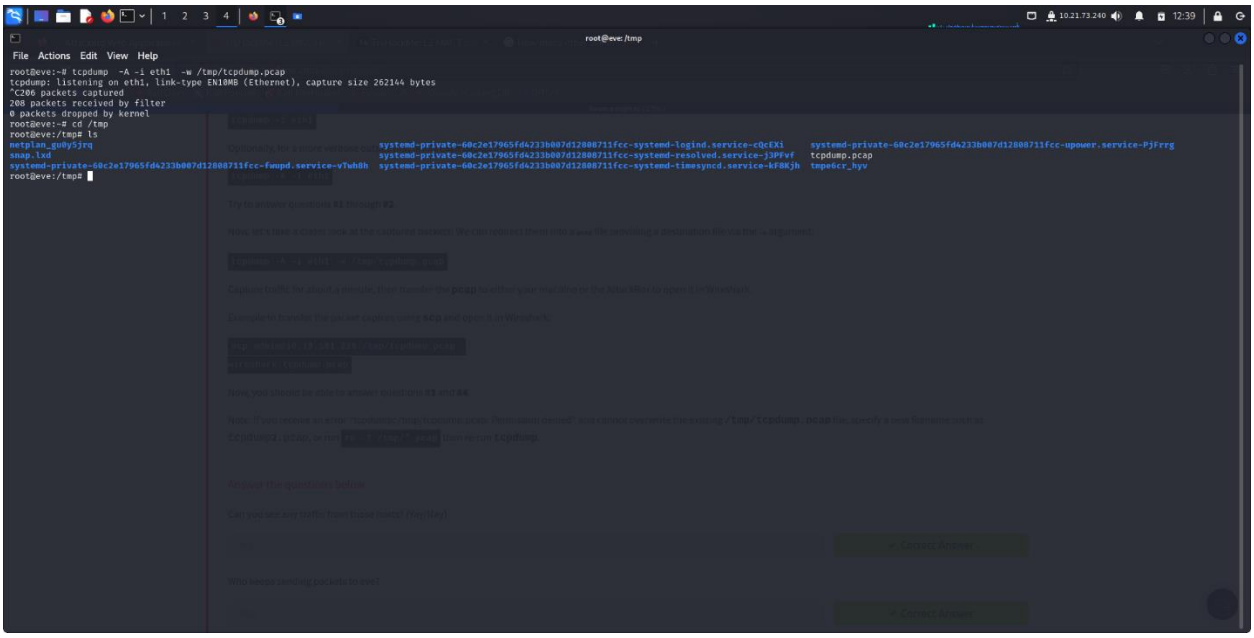
Questions



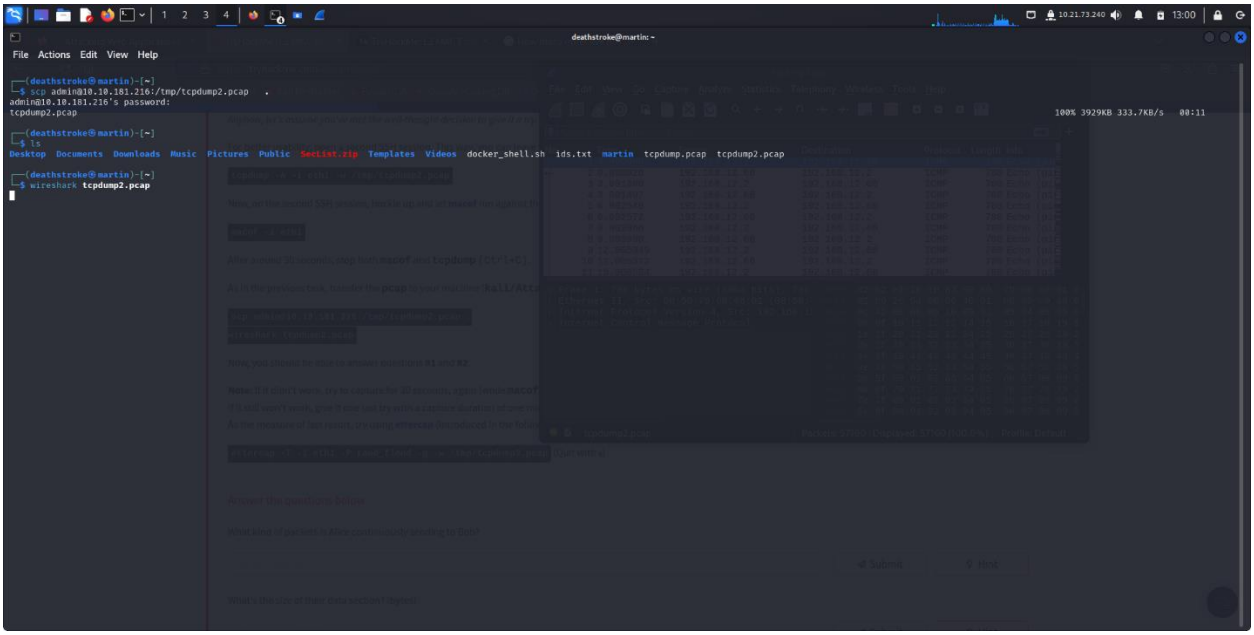
Using tcpdump tool to passively sniff the target, and flag -A more verbose output and -i for interface, I sniffed eth1 interface and found one could view the traffic, and on the traffic I found the one sending the packets(Bob) and receiving them,



After sniffing the network with tcpdump, and saving the file on /tmp/tcpdump.pcap, I had to download it to my machine and use wireshark to deeply analyze the traffic packets sent.

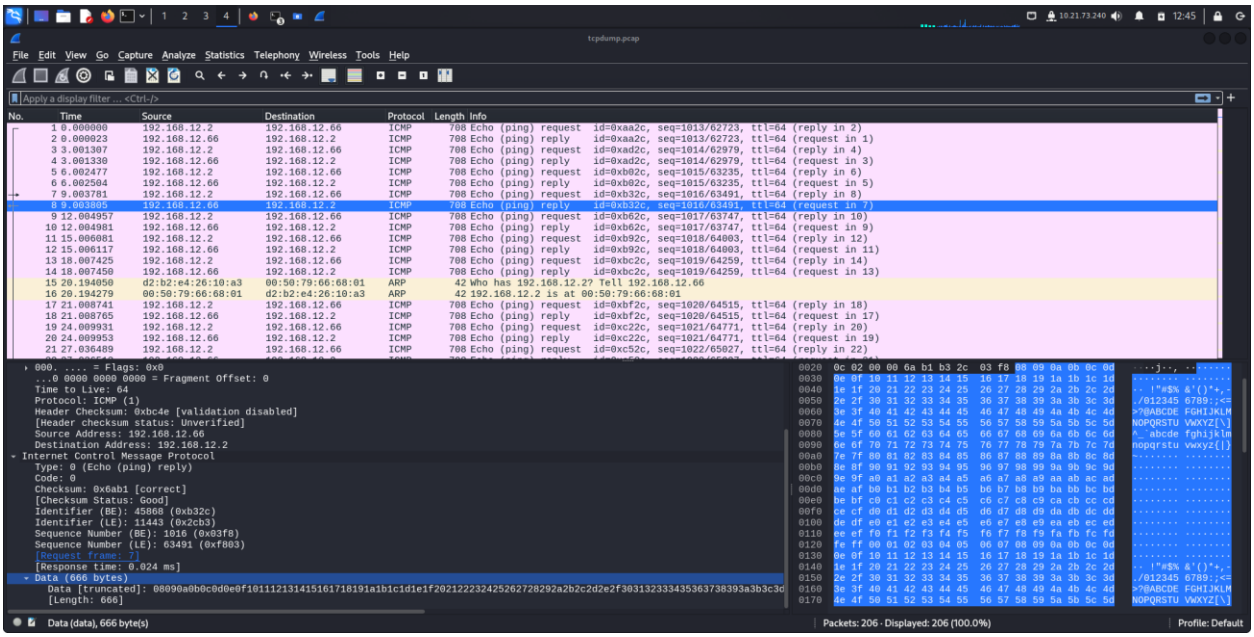


Used *scp* tool to with the username@targetIP:file path, and the file was downloaded.



Opened the file with wireshark, and viewed the traffic, found the protocols used to be ICMP, plus I could view the parts of ICMP packets, like checksum(for errors), Sequence No.(packets to come in order they were sent).

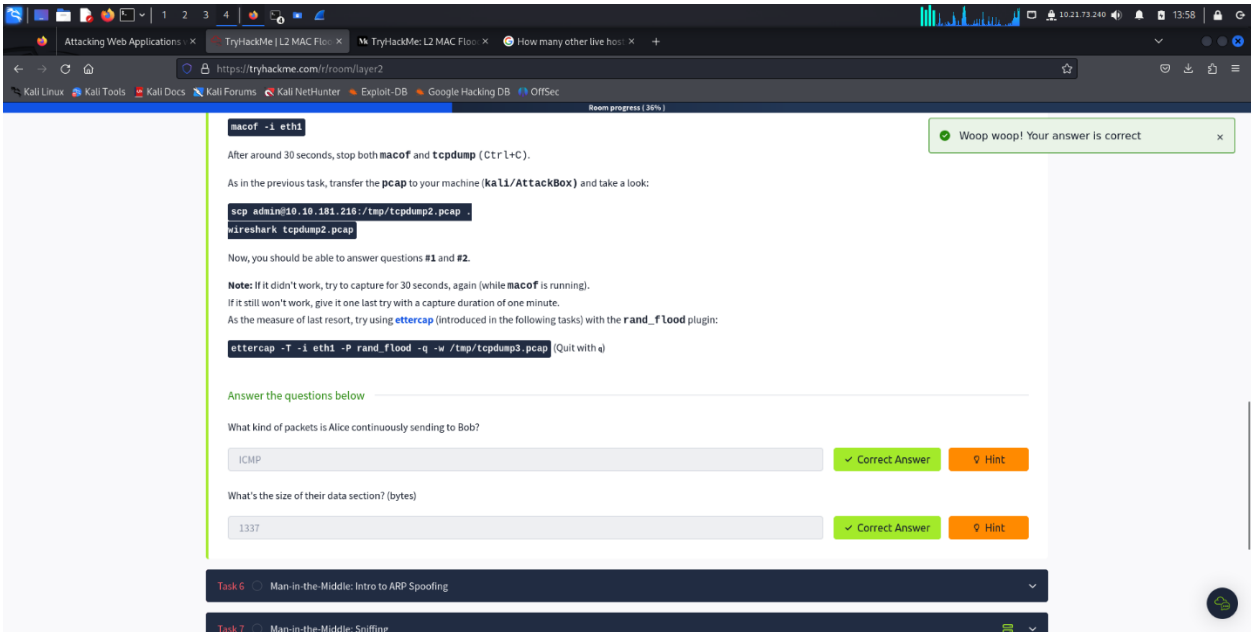
And also the data itself sent, found it size to 666 bytes.



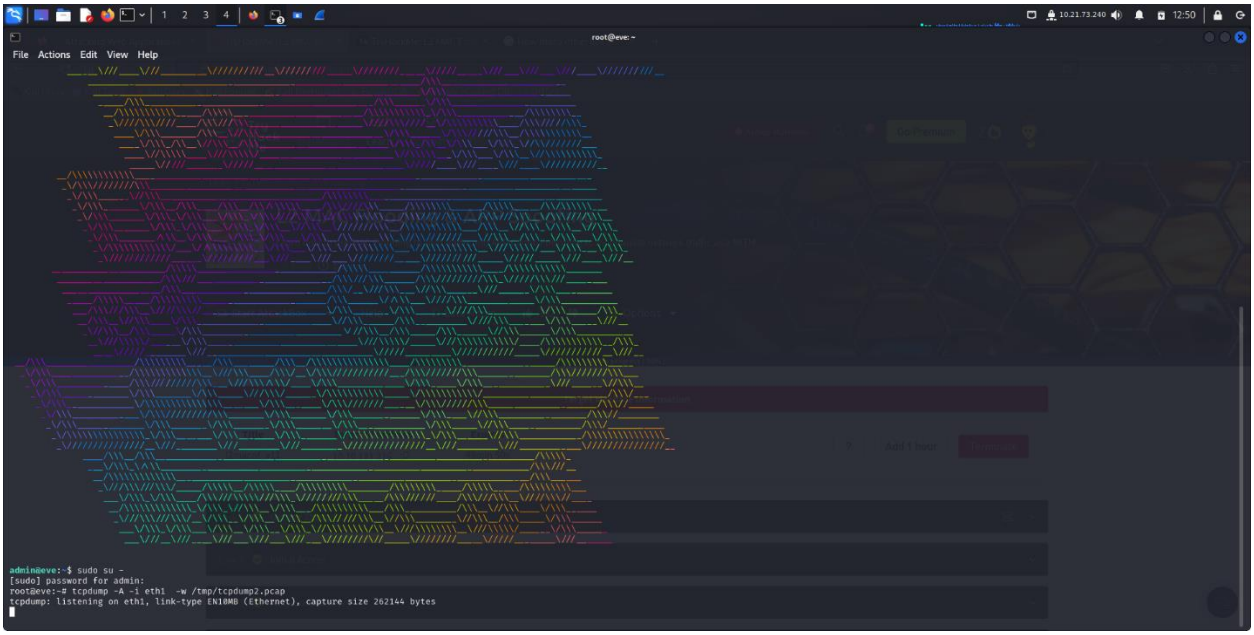
1.4 Sniffing while MAC Flooding

MAC flooding could trigger an alarm in a SOC. As suspicious layer 2 traffic can easily be detected and reported by state-of-the-art and properly configured network devices.

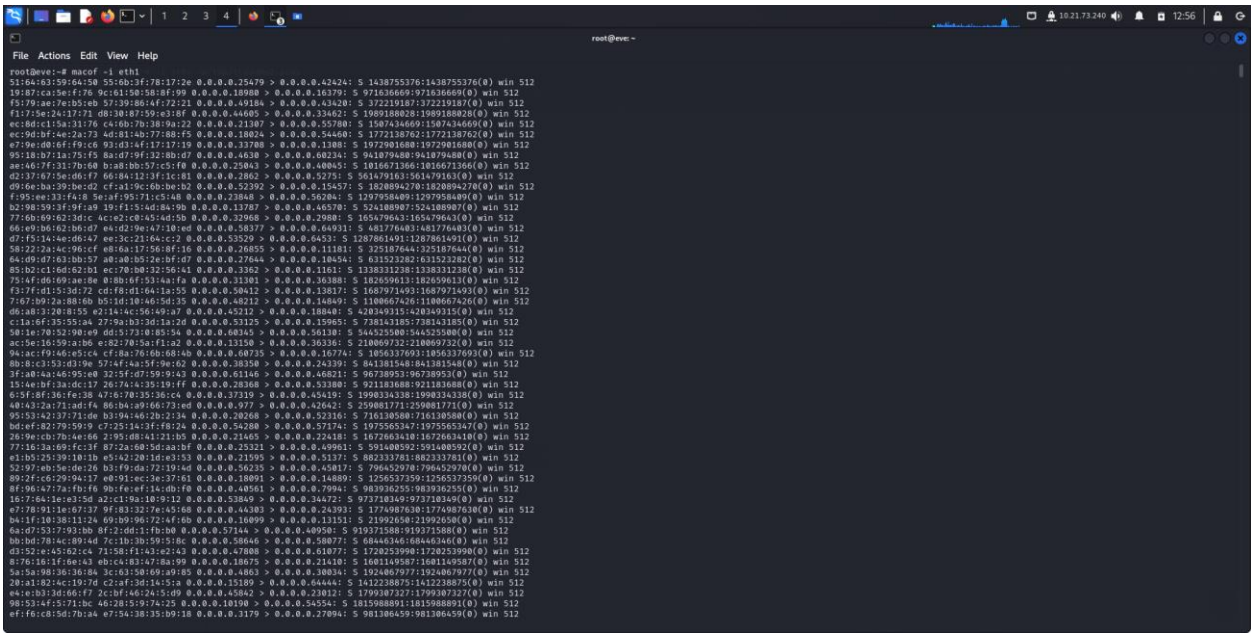
Questions



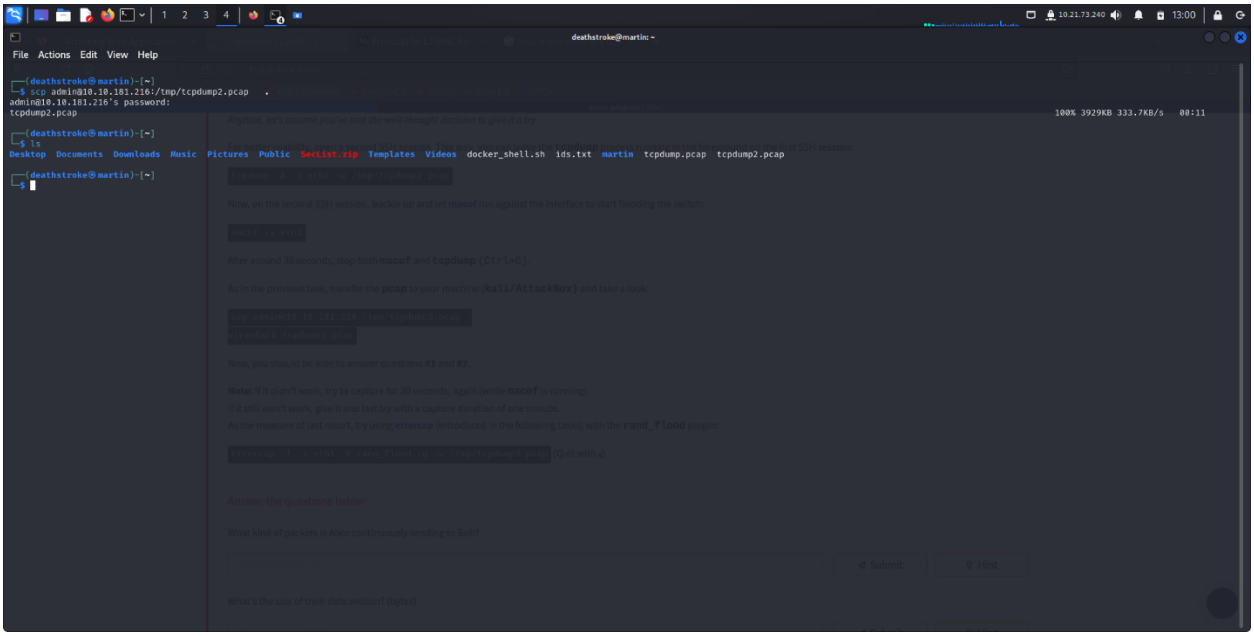
Started off by running my sniffing tool *TCPDUMP* and saving the results on file tcpdump2.pcap,



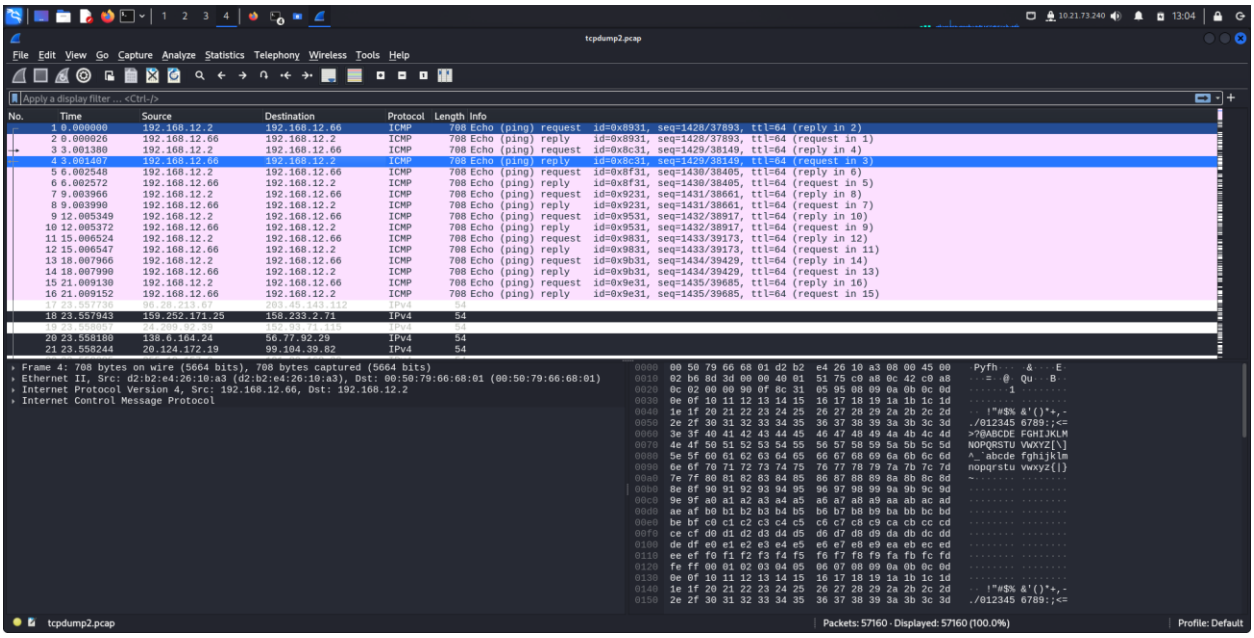
I went by to flood the MAC address table, using *macof* tool,



After about 1 minute I closed the macof service running and tcpdump service running, with use of scp tool I downloaded the file with results,



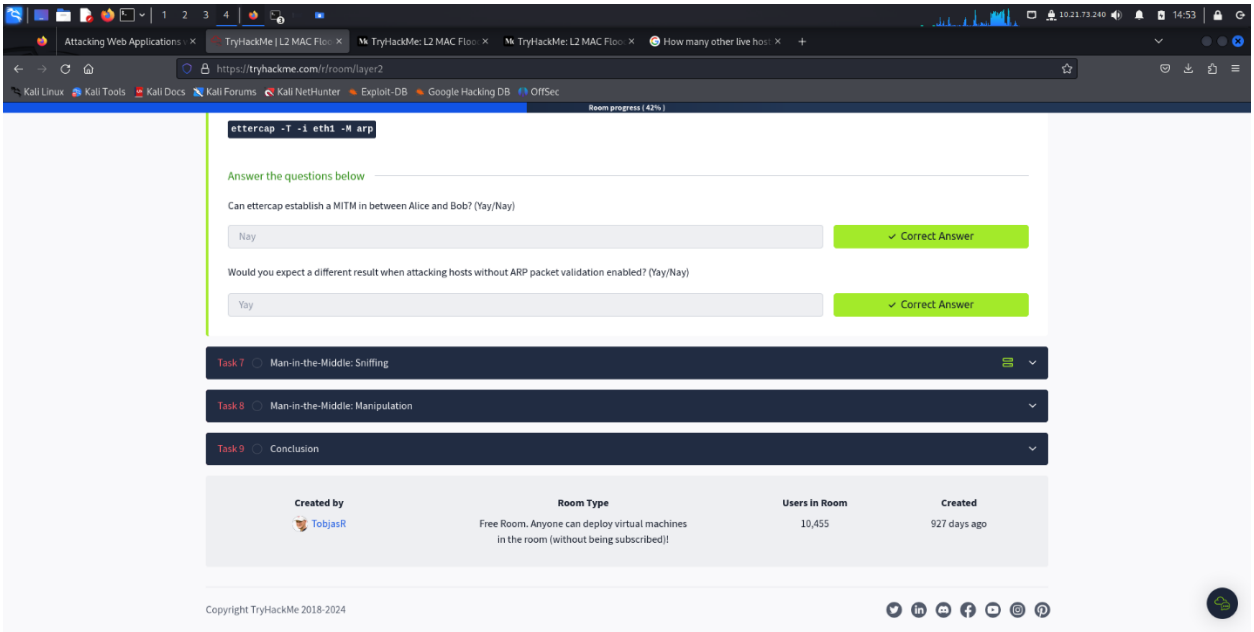
After downloading the file I opened it with wireshark to analyze the traffic and found the protocol used was ICMP and data size was 1337, due to the MAC address floading,



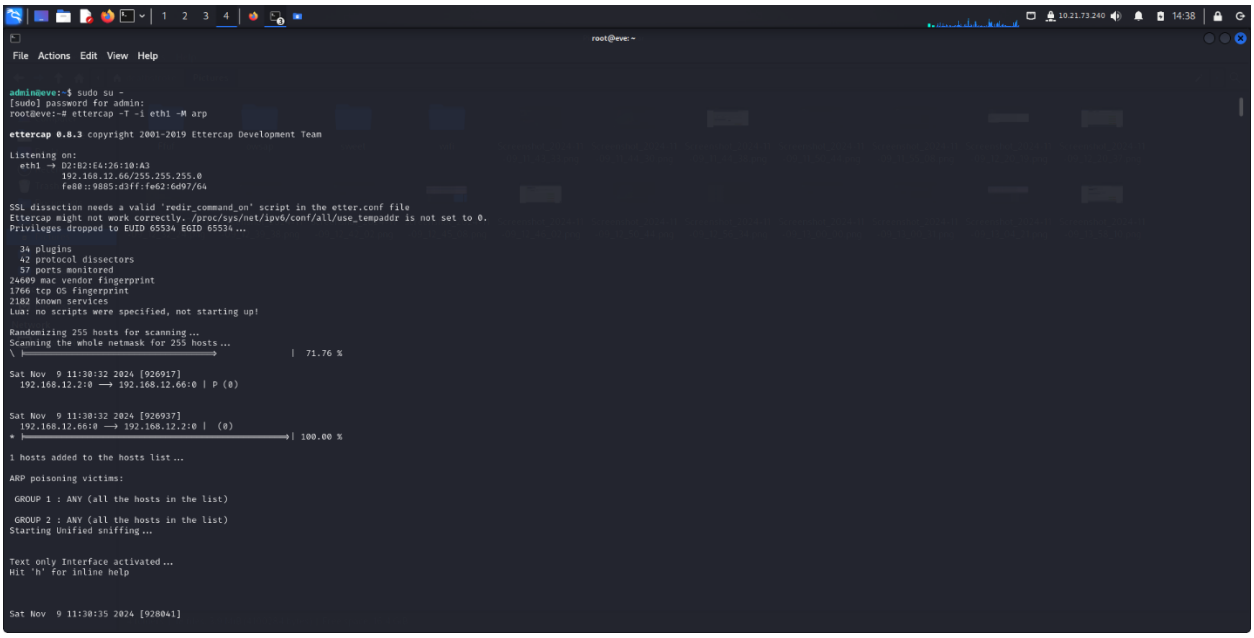
1.5 Man-in-the-Middle: Intro to ARP Spoofing

MAC Flooding can be considered a real "noisy" technique. In order to reduce the risk of detection and DoS we will leave *macof* and perform *ARP cache poisoning* attacks.

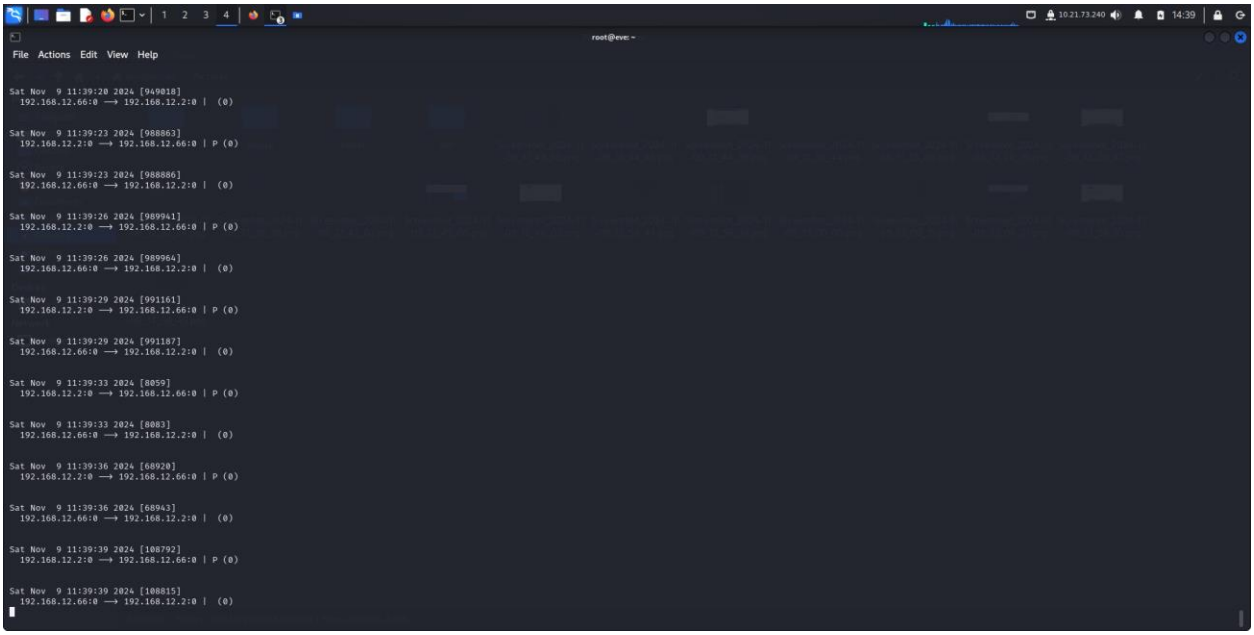
Questions



Using *Ettercap tool* could perform ARP cache poisoning and using flag -M to enable ARP packet validation, and found one can establish a MITM attack,

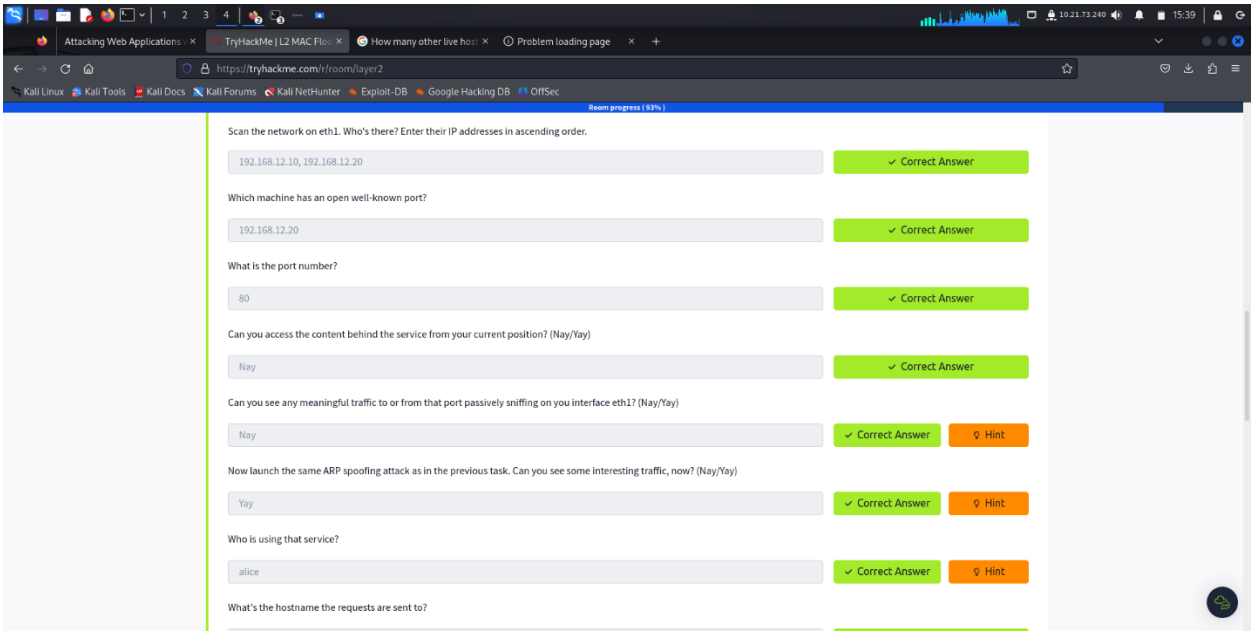


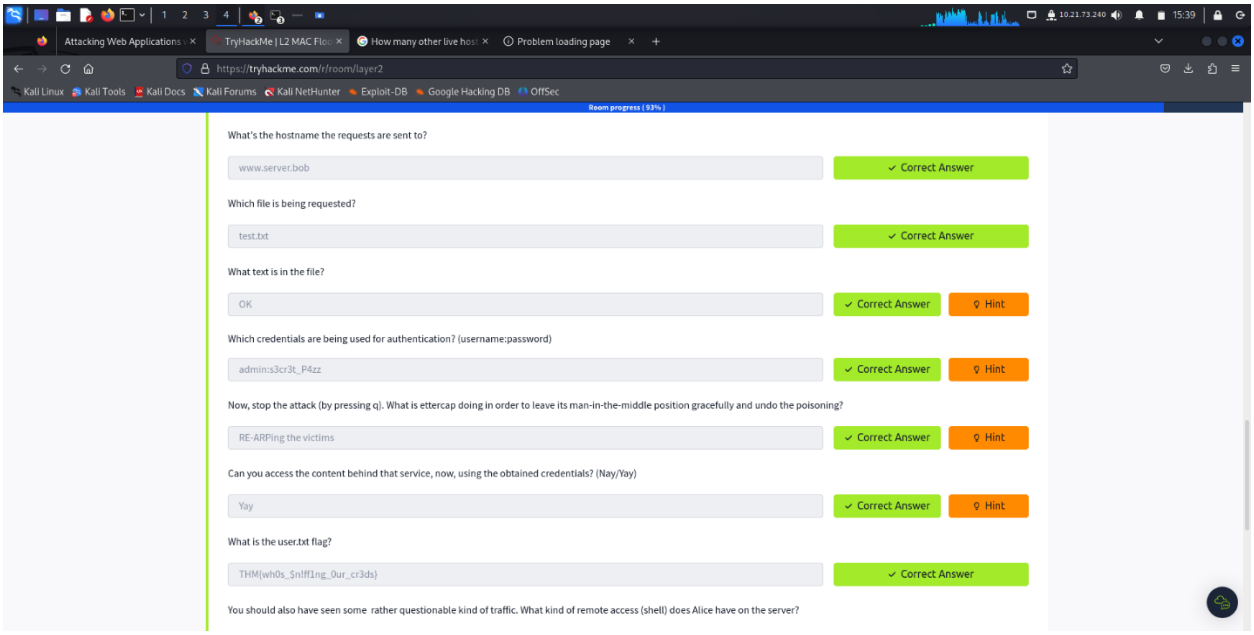
And also without enabling arp validation the, result are different.

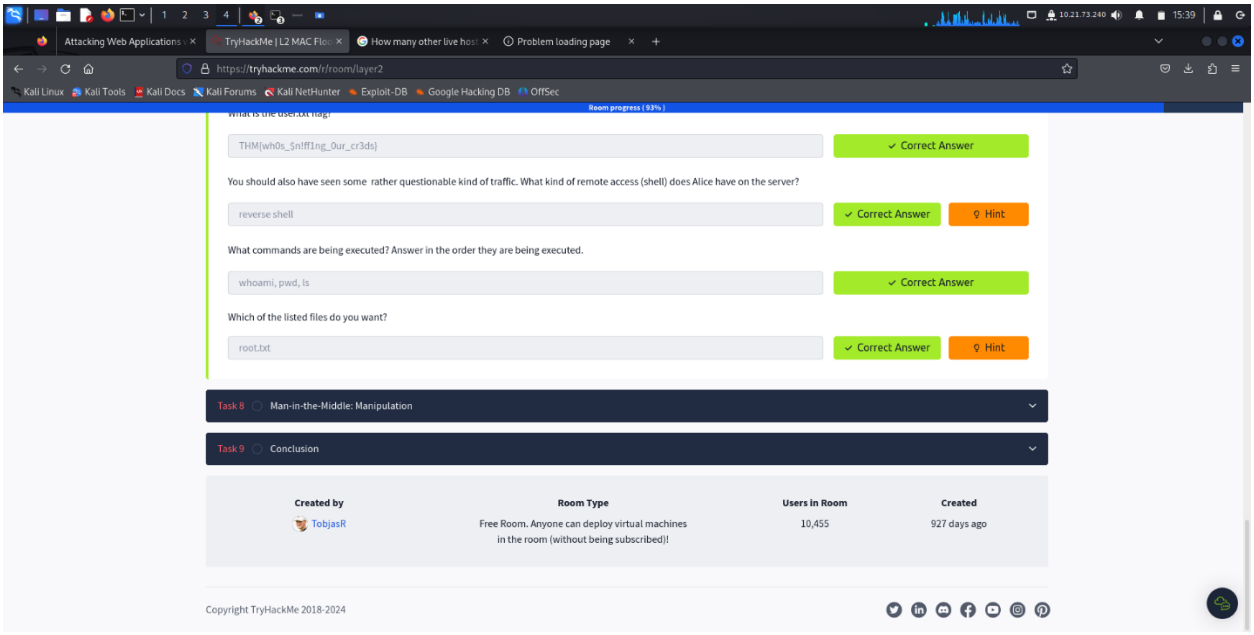


1.6 Man-in-the-Middle: Sniffing

Questions



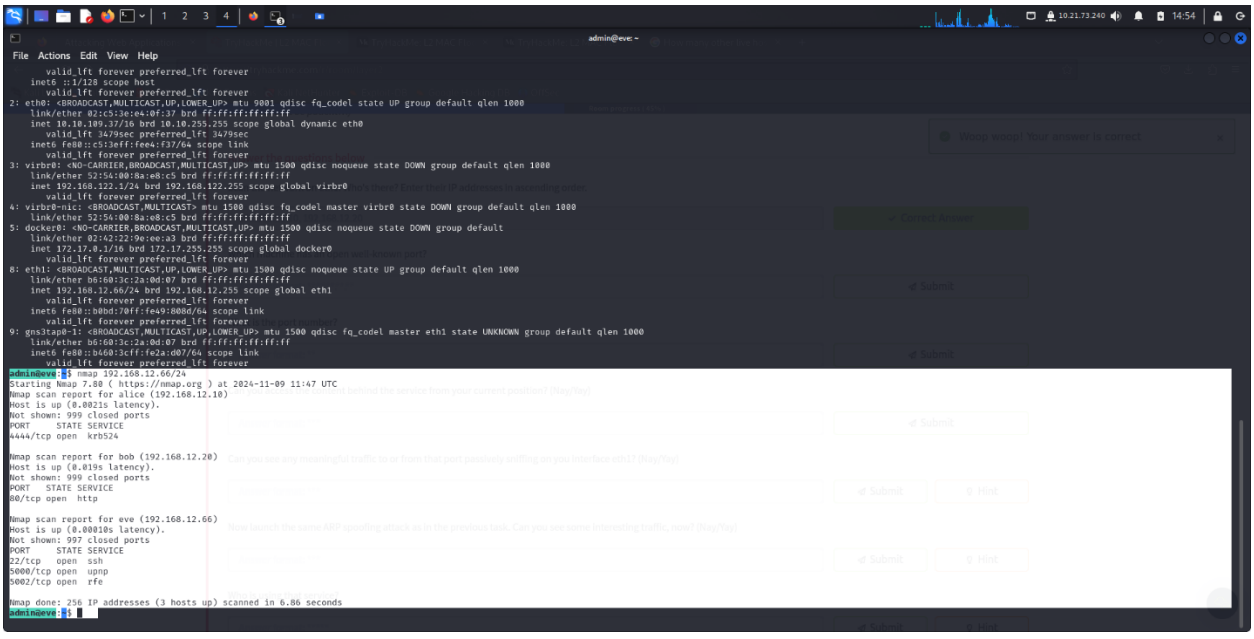




Quiz 1

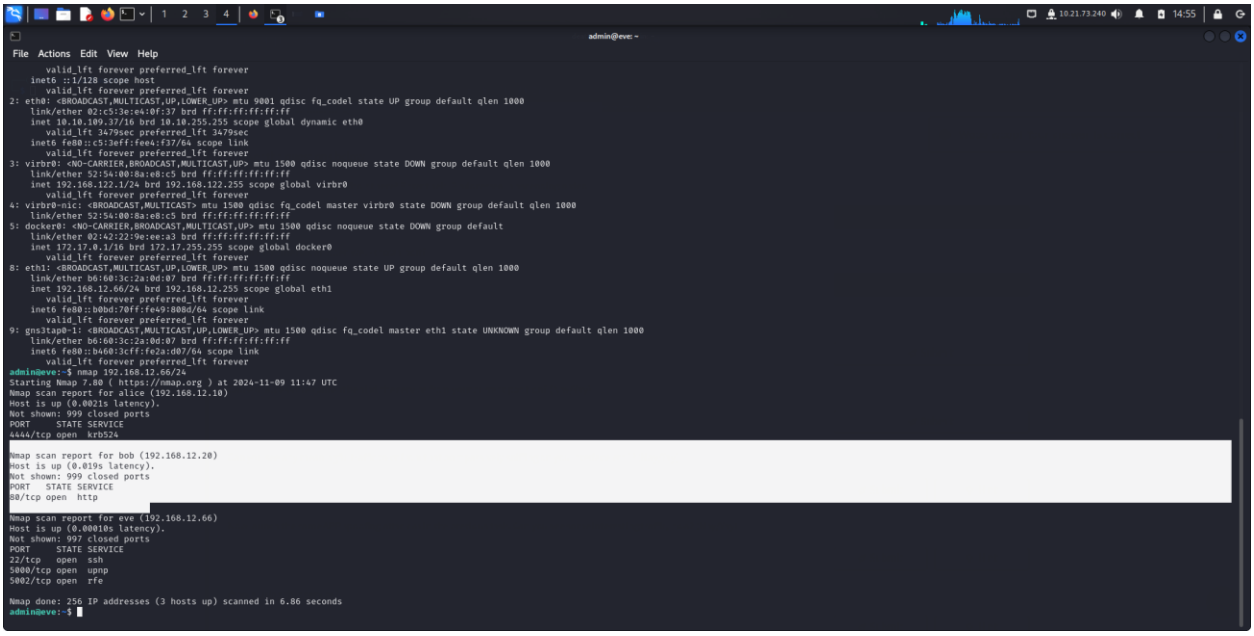
I used nmap to scan the target with it subnet, I found the target IPs in that network and services running in those IPs and port open.

From this scan I found the IPs.



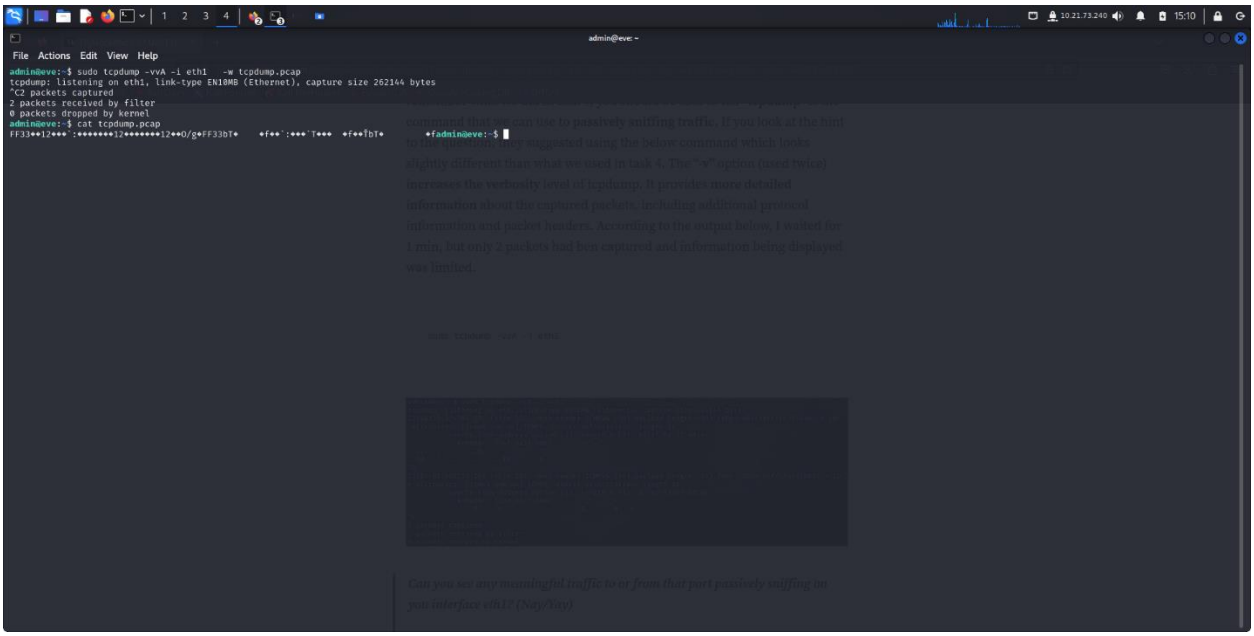
Quiz 2, 3

From the scan I found the target with the open port and service running to be 192.168.12.20, port 80, service http.



Quiz 4

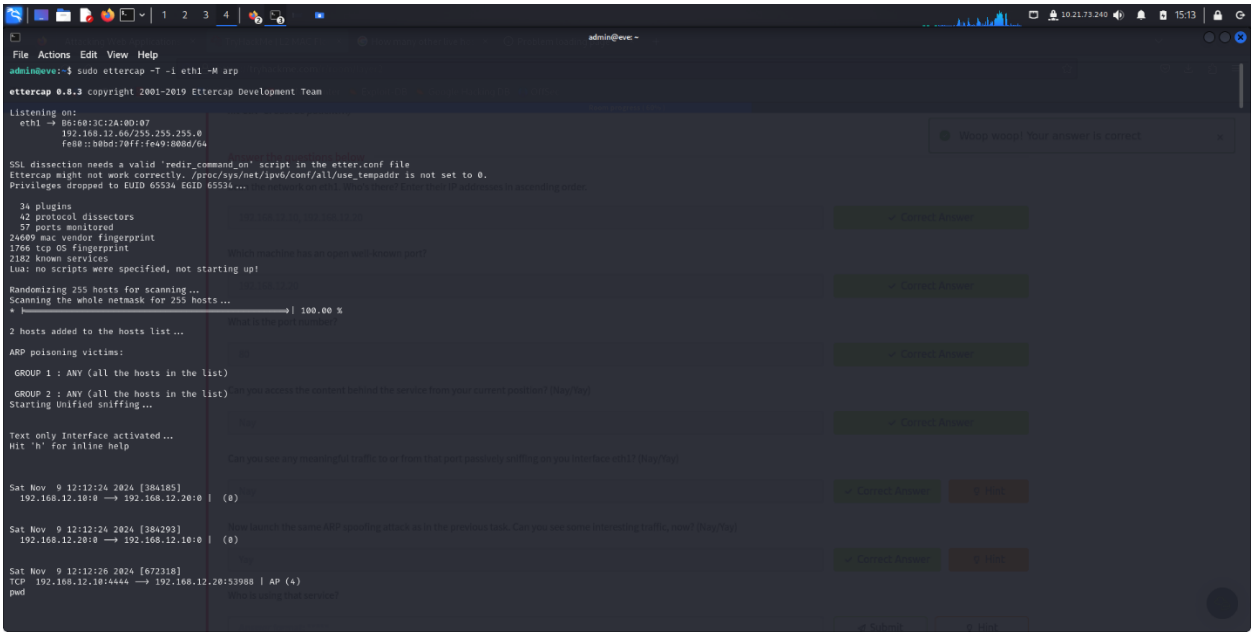
With tcpdump sniffing I could not find any useful data,



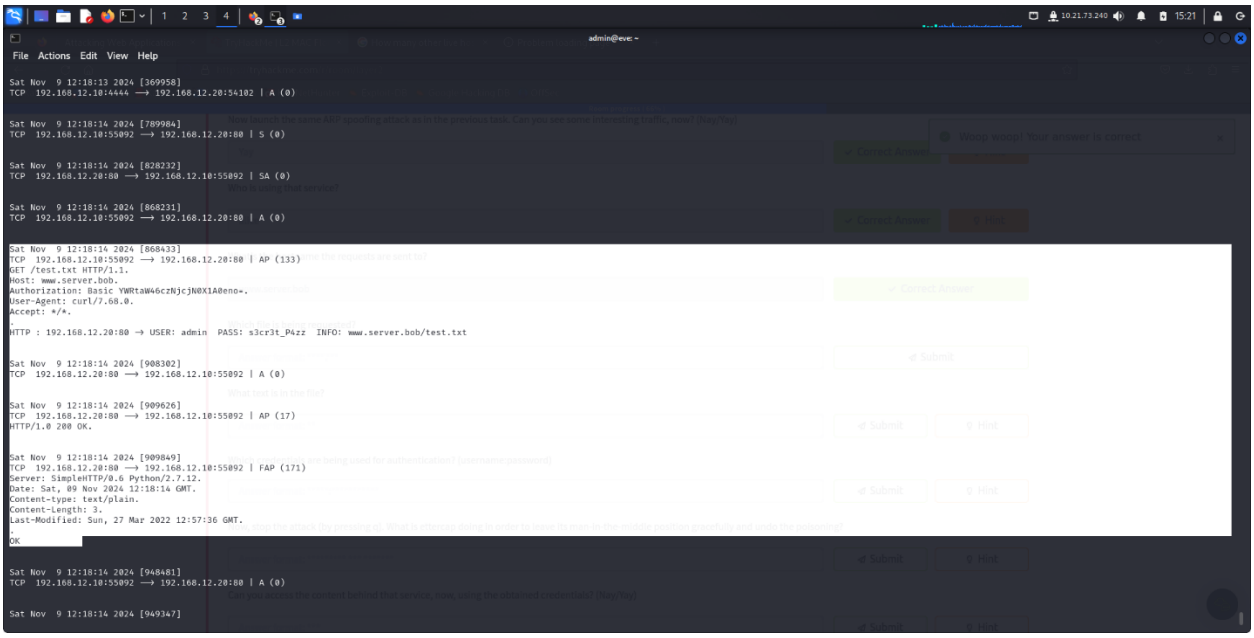
Quiz 5, 6,7,8,9,10

With use of ARP cache poisoning, with using Ettercap tool, I was able to get useful data, as I was man-In-The-Middle, Found some on the commands being run, *whoami, pwd, ls* .

And analyzing the data found the IP using the service to Alice IPs,

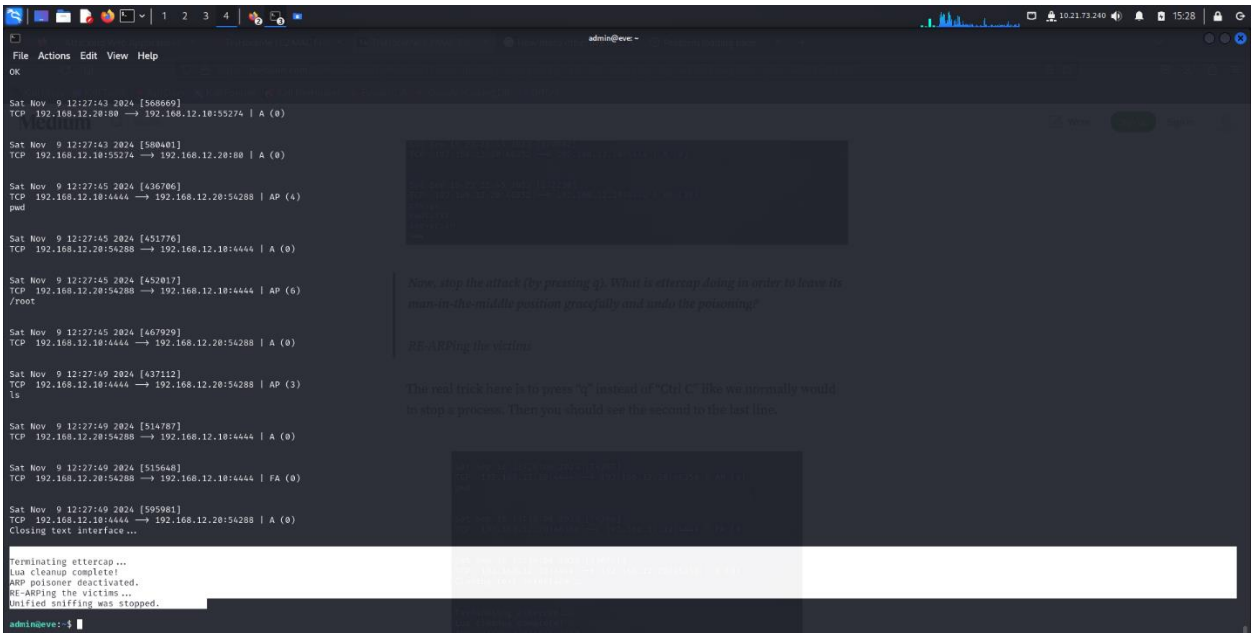


This output is from the ARP cache poisoning, I found useful information such as the file being accessed, its contents, the username and password to that server. And also the server name/hostname.



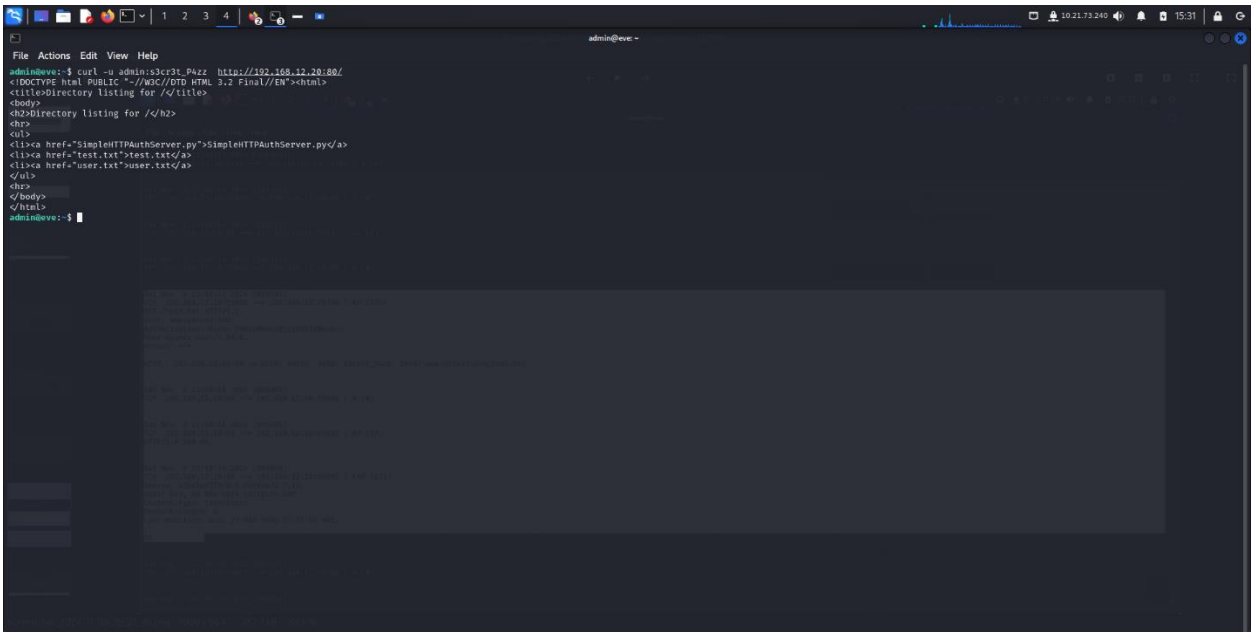
Quiz 11

With use o letter q to quit the ARP cache poisoning, this was the result.



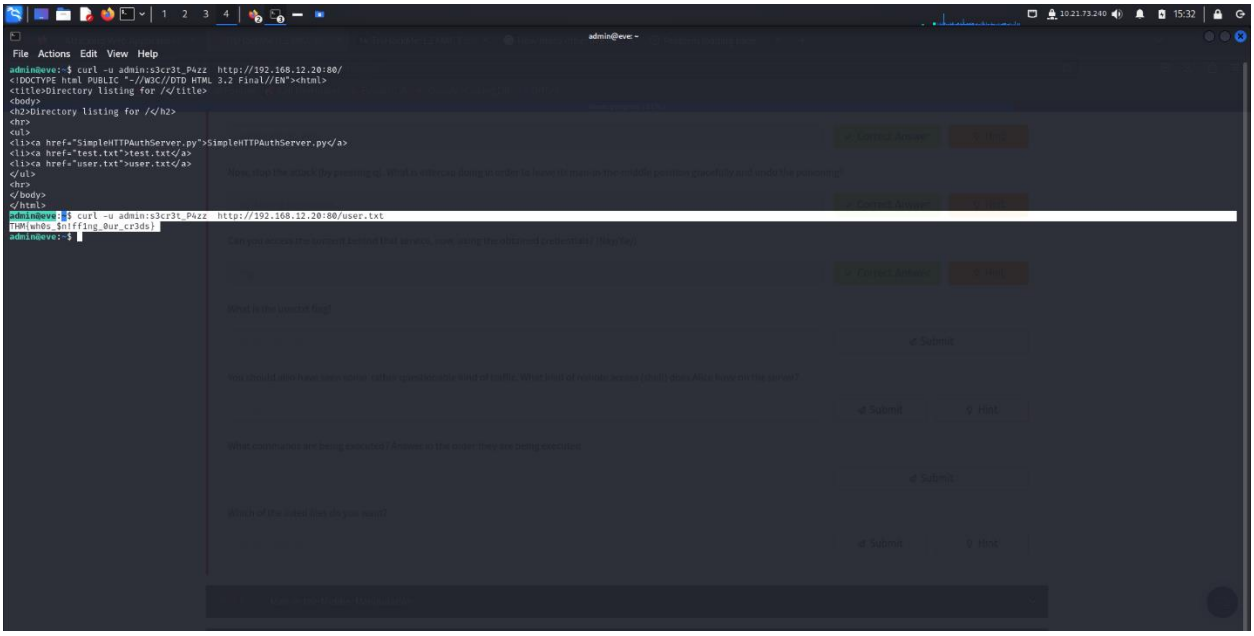
Quiz 12

With use of the username and password found I could access the http service on port 80, using curl command, and found it had files like previous one I found the arp poisoning(test.txt) and user.txt.



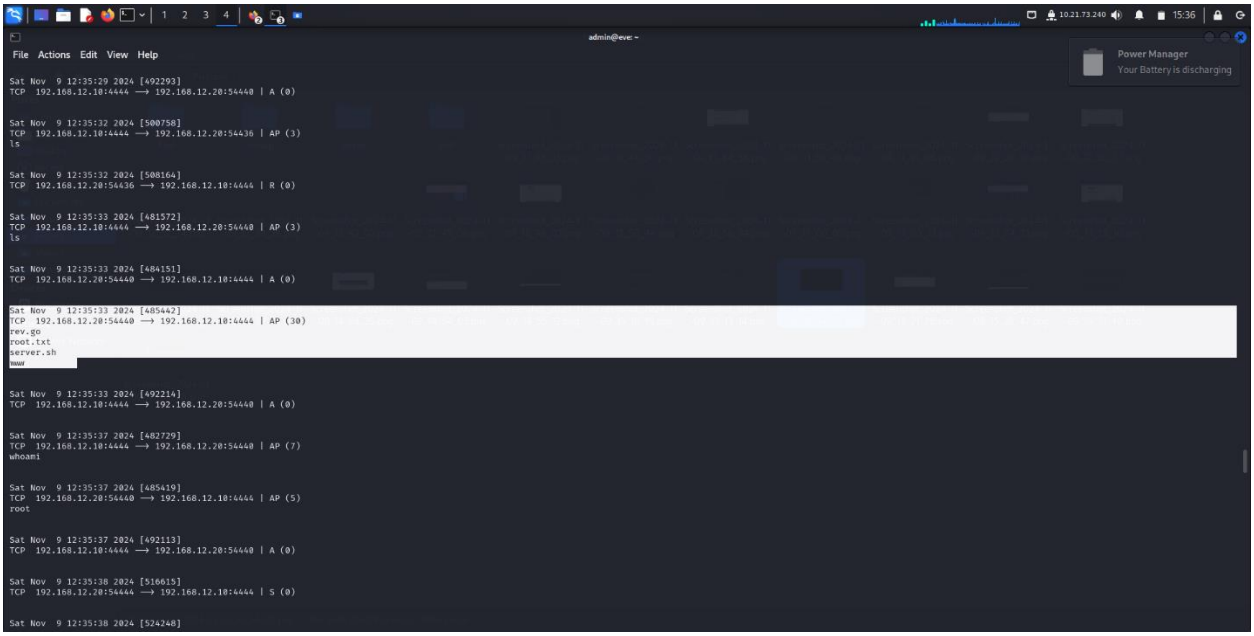
Quiz 13

With use of curl command and the username and password, I explored the contents of the user.txt file and found the flag,



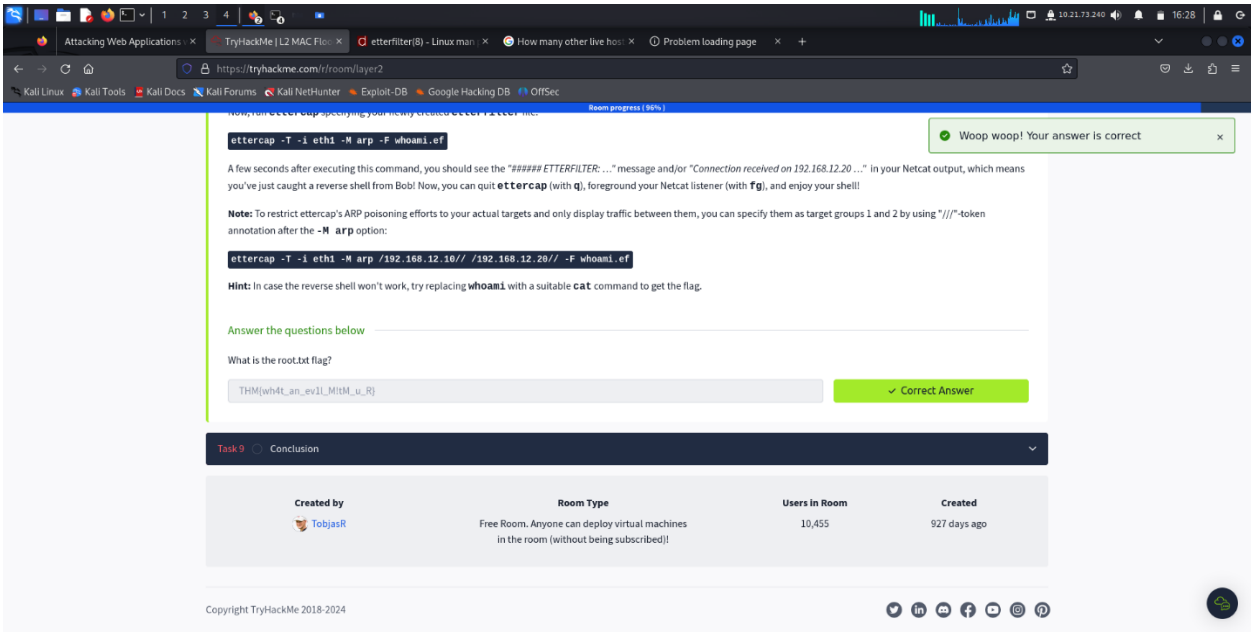
Quiz 14, 15, 16.

From this traffic we can see that alice remote access to bob server is through reverse shell, such that she's using port 4444 as listening port on netcat, and there some file displayed after the ls command. And other commands being used *whoami, pwd.*

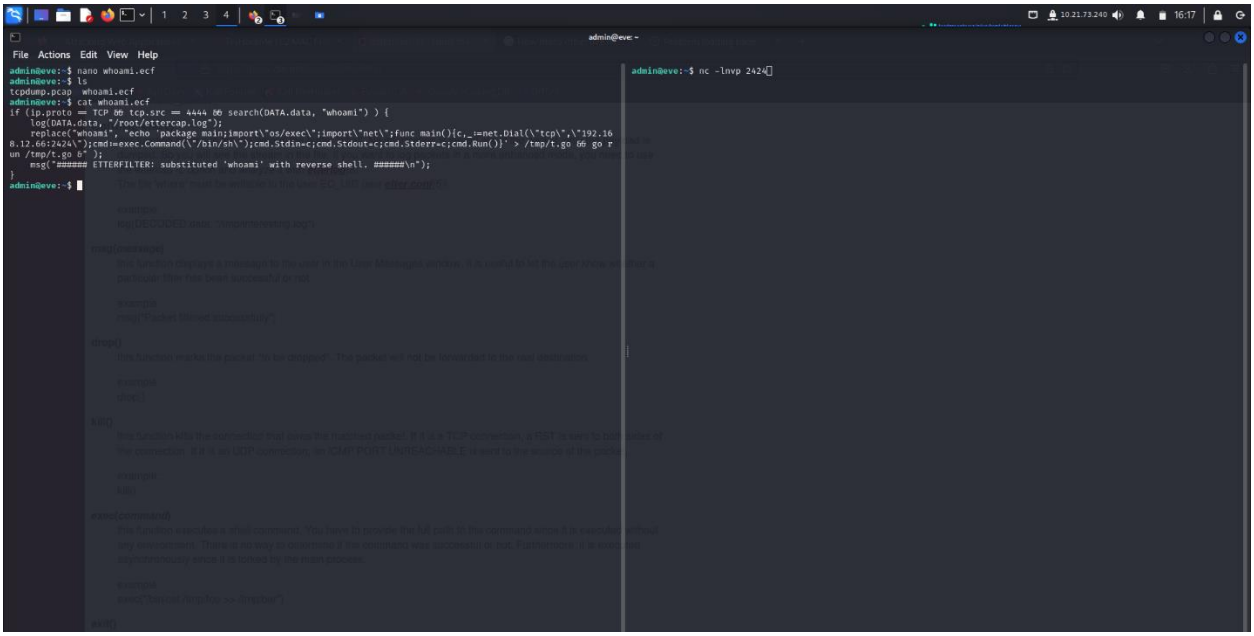


1.7 Man-in-the-Middle: Manipulation

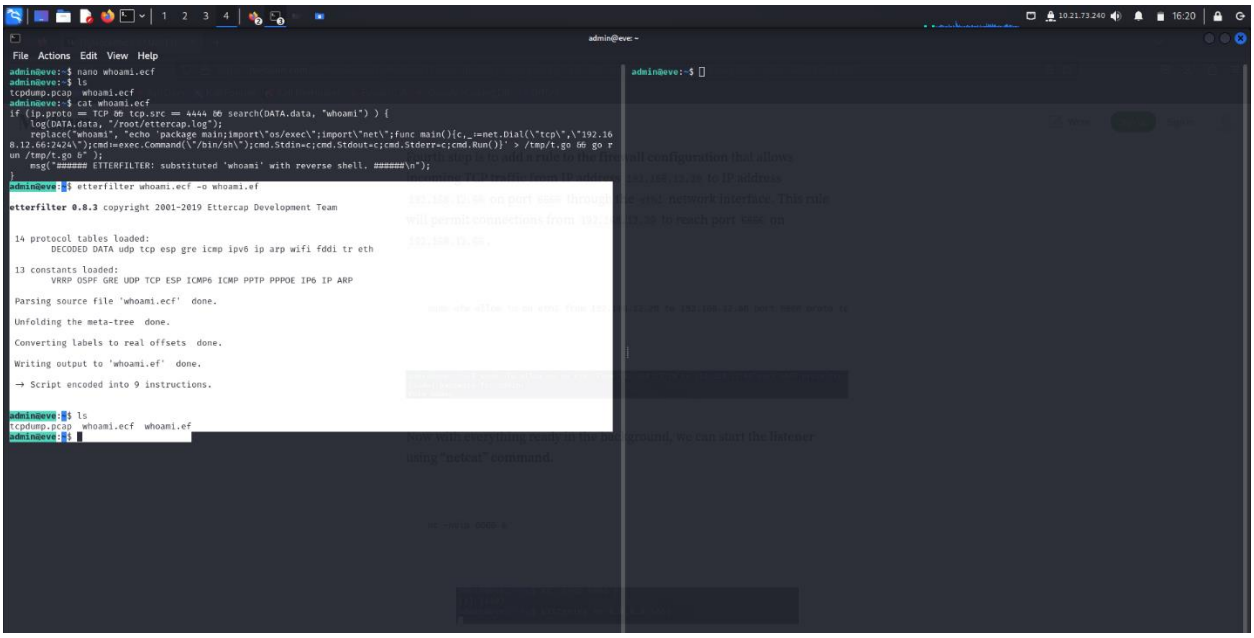
Questions



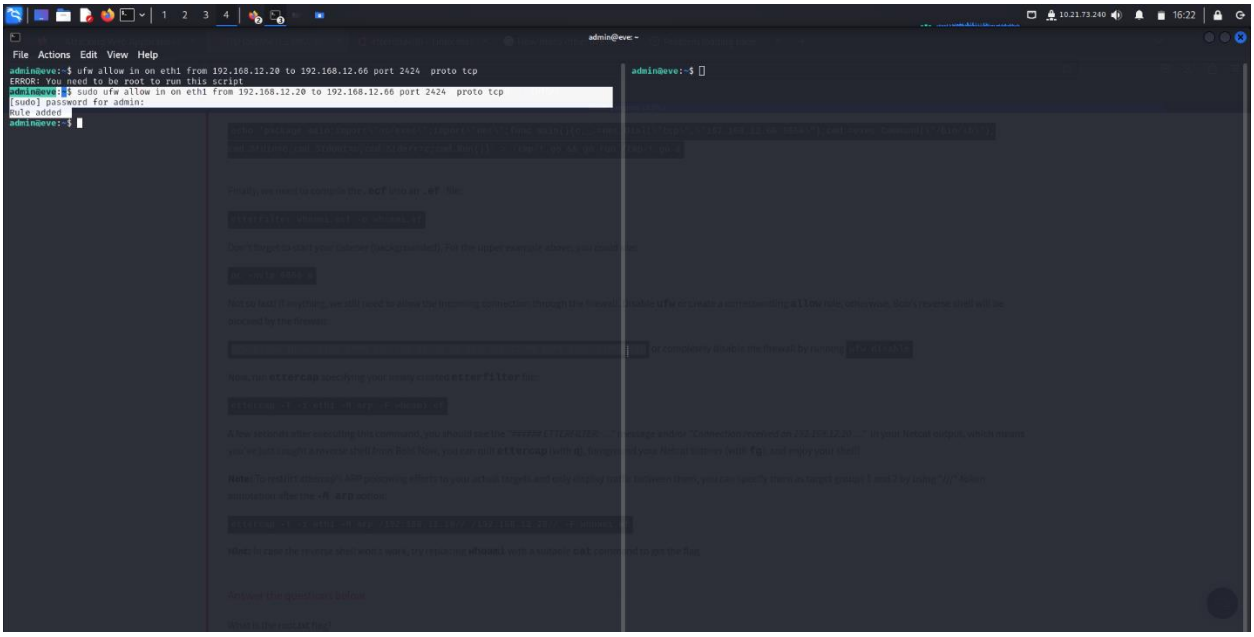
Started of by creating a file with script in it to trigger a revers shell to me so as to have elevated privileges to root. This is the syntax of the script.



After successful writing and saving the script, I had to convert it to .ef format to use it with the Ettercap tool during Arp cache poisoning,



I used this rule to instruct my machine firewall to allow the reverse shell coming from the target IP back to my machine,



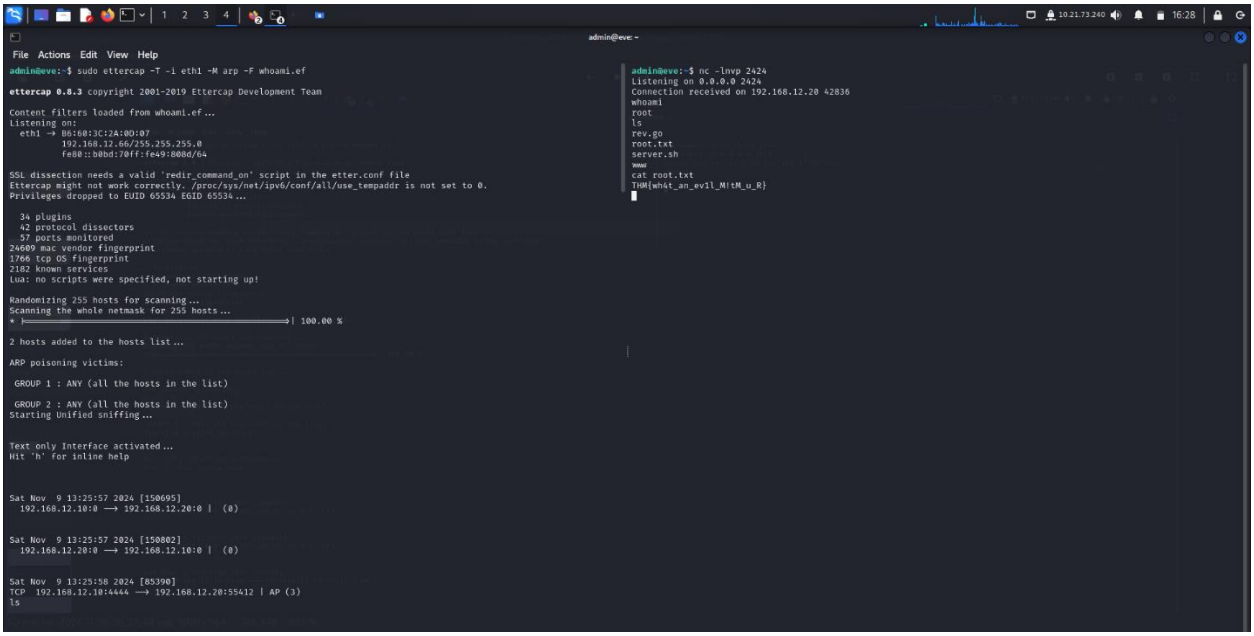
After all the parameters required it was time to run the ARP poisoning with the file.

On the left side I started my netcat to listen on port 2424, and when I ran the Ettercap, I got a reverse connection of the system.

To check which user I was I used whoami and I was root.

Used ls command to list the files and directories within root and found the root.txt file.

Used cat and viewd the contents of the root file, thus found the flag.



2.0 In conclusion

This room has be help full in solidifying my understanding in the layer 2 threats in a network and what tools and how I can exploit them to check if they security measure configured to prevent such threats or not.