Week 06 – Task 6.1P

Question A:

1. This domain was created in 2020, is this correct? If not, what date was this domain created first?
    - Yes, The creation date of this domain 2020.08.04

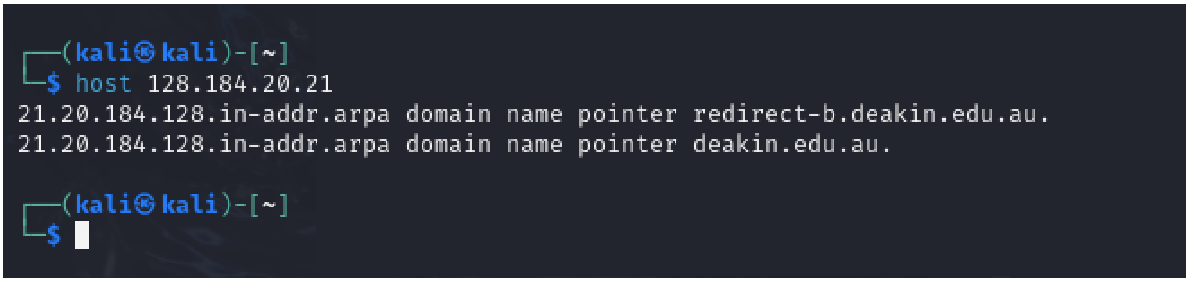
2. When does the registration of the domain expire?
    - The register will expire on 2024.08.04

3. What is a "Name Server" and what is it used for? (Include a reference for your answer) What are the Name Servers for this domain?
    - "Traffic on the internet is organized and routed by name servers." - Forbes
    - Name server – NS1.ATOM.COM
        - NS2.ATOM.COM

4. What is a Registrar? Who is the registrar of this domain?
    - A register is a company or organization that also manages domain name reservation
    - As the IP addresses assigned to those domain names.

Question B:

1. Include a screenshot of the output you get.

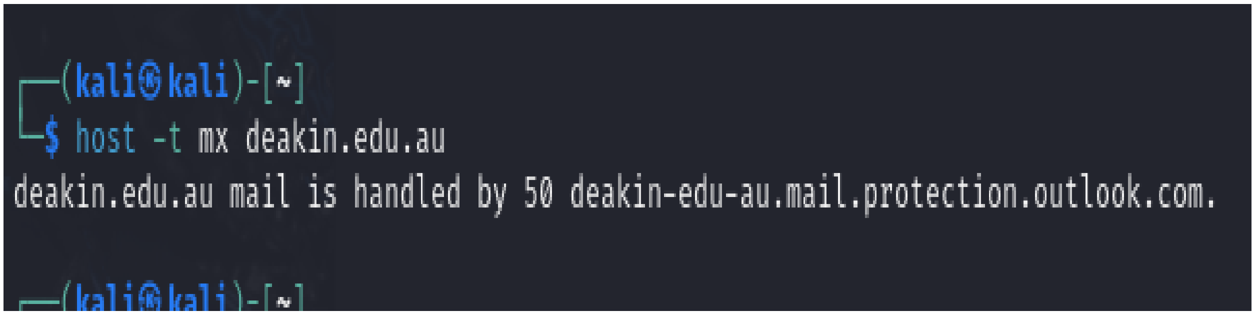


```
┌──(kali㉿kali)-[~]
└─$ host 128.184.20.21
21.20.184.128.in-addr.arpa domain name pointer redirect-b.deakin.edu.au.
21.20.184.128.in-addr.arpa domain name pointer deakin.edu.au.

┌──(kali㉿kali)-[~]
└─$ █
```

2. What is the host command used for? (Include a reference for your answer)
    - In Linux systems, DNS (Domain Name System) lookup operations are performed using the host command. – Geeksforgeeks.

Question C:

1. Include a screenshot of the output you get.

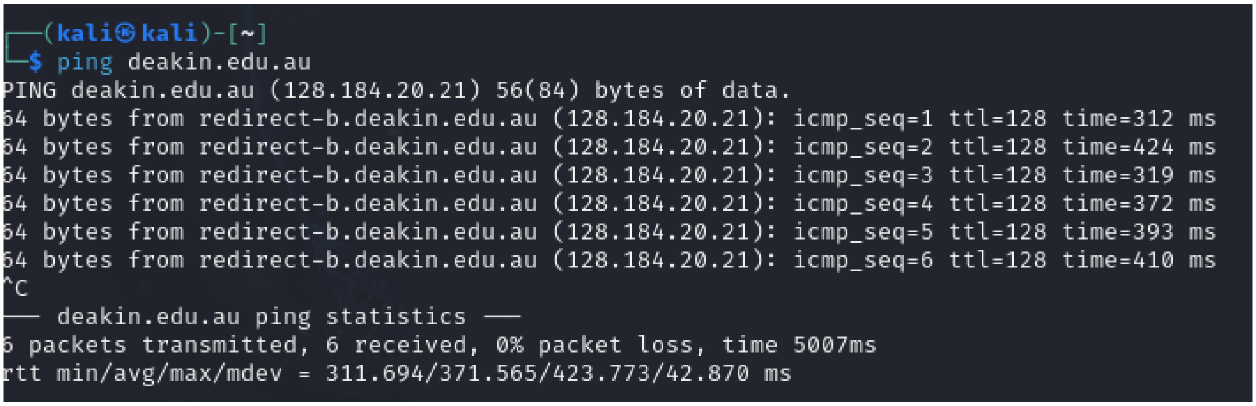


```
┌──(kali㉿kali)-[~]
└─$ host -t mx deakin.edu.au
deakin.edu.au mail is handled by 50 deakin-edu-au.mail.protection.outlook.com.

┌──(kali㉿kali)-[~]
```

2. What is an MX record in DNS?
    - Email to a mail server is directed by an MX (Mail Exchange) record.

Question D:

1. Include a screenshot of the output you get.



```
┌──(kali㉿kali)-[~]
└─$ ping deakin.edu.au
PING deakin.edu.au (128.184.20.21) 56(84) bytes of data.
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=1 ttl=128 time=312 ms
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=2 ttl=128 time=424 ms
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=3 ttl=128 time=319 ms
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=4 ttl=128 time=372 ms
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=5 ttl=128 time=393 ms
64 bytes from redirect-b.deakin.edu.au (128.184.20.21): icmp_seq=6 ttl=128 time=410 ms
^C
--- deakin.edu.au ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5007ms
rtt min/avg/max/mdev = 311.694/371.565/423.773/42.870 ms
```

Investigate about ping command and answer the following questions:

2. What is ICMP?
   - ICMP - Internal Control Massage Protocol

3. Fill in the blanks: A correctly-formed ping packet is typically_56__ bytes in size, or__64__ bytes when the ICMP header is considered, and__84__ including Internet Protocol version 4 header.
   - 56
   - 64
   - 84

4. What does `ttl' refer to in the ping command output?
   - The term “ttl” describe the number of network hops a packet has before the router discards it.

Question E:

1. Using the “host” you learned about earlier, find the IP address for localhost. What is the IPv4 address for localhost?
    - 127.0.0.1

2. Do you need Internet access to retrieve the “localhost” domain?
    - The localhost domains can be retrieved without internet connectivity because it is a network address that belongs on your own device.

Question F:

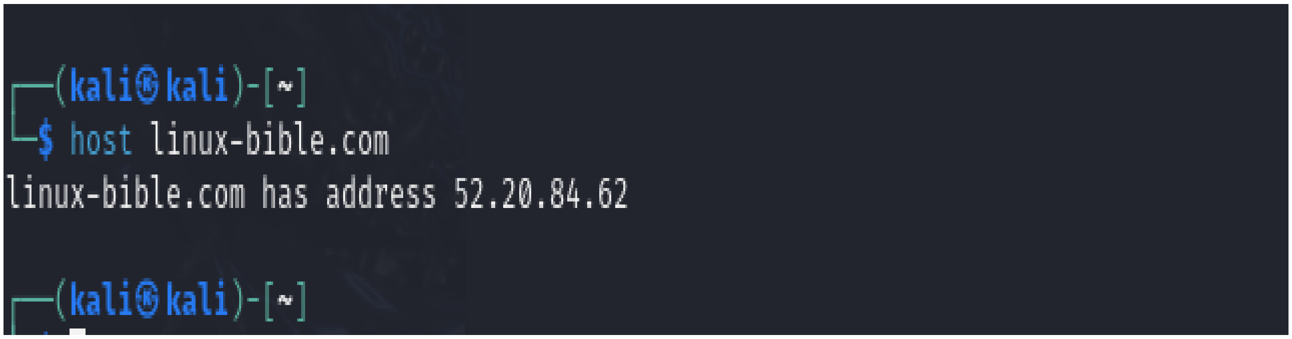
1. What is a `hop' referring to in the output for the traceroute command?
    - Every step that a packet takes to go from its source to its destination is referred to as a hop.

2. What happens if one of the servers/routers in the hops is not listening for ICMP echo requests?
    - Traceroute displays *** indicating a timeout or the possibility that it will terminate before traveling the whole distance.

3. How can an attacker use "traceroute" when targeting computer networks?
    - To aid plan possible assaults, an attacker can use the "traceroute" command to learn more about a target's network structure. To put it another way, it server as a crucial reconnaissance instrument in hacking.

Question G – Challenge 1

1. The Challenge 1 is to crack a password. Using the host command, find the IP address of the domain linux-bible.com. Include the screenshot of your host command and the results.

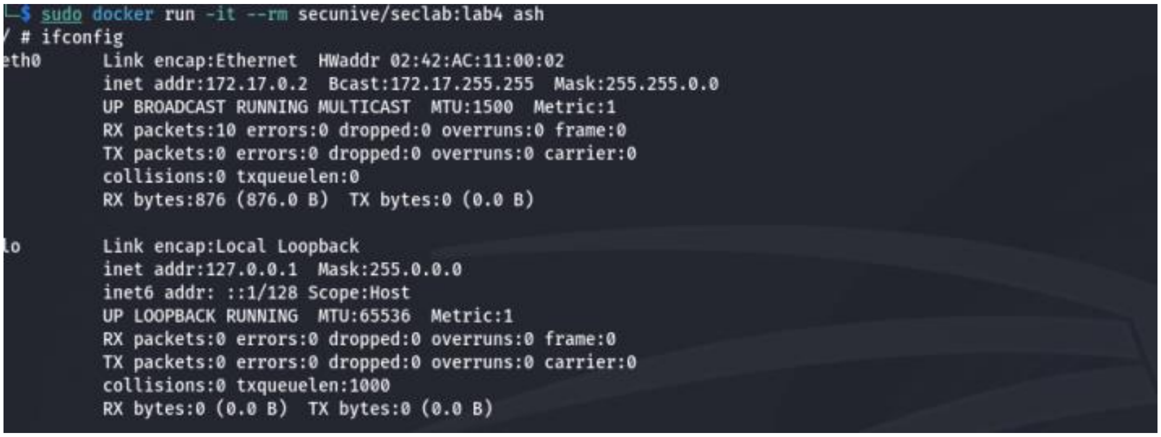


```
┌──(kali㉿kali)-[~]
└─$ host linux-bible.com
linux-bible.com has address 52.20.84.62

┌──(kali㉿kali)-[~]
```

Question H:

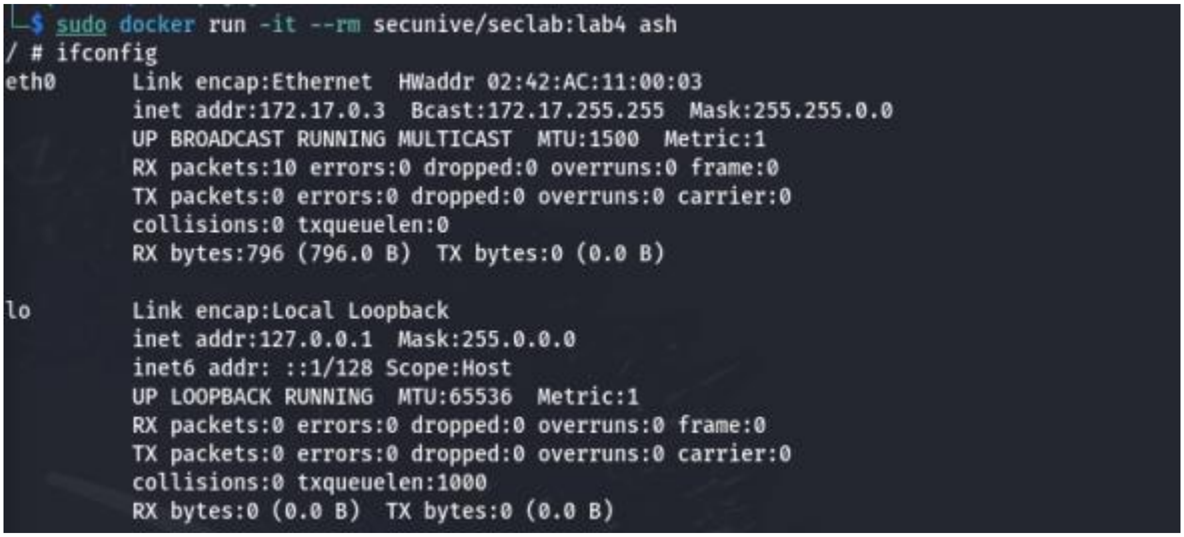
1. Include a screenshot of running the “ifconfig” command in each of the terminals. What is the IP address for eth0 in Terminal 1 and what is the IP address for eth0 in Terminal 2?

Terminal 1



```
└$ sudo docker run -it --rm secunive/seclab:lab4 ash
/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:AC:11:00:02
          inet addr:172.17.0.2  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:876 (876.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Terminal 2



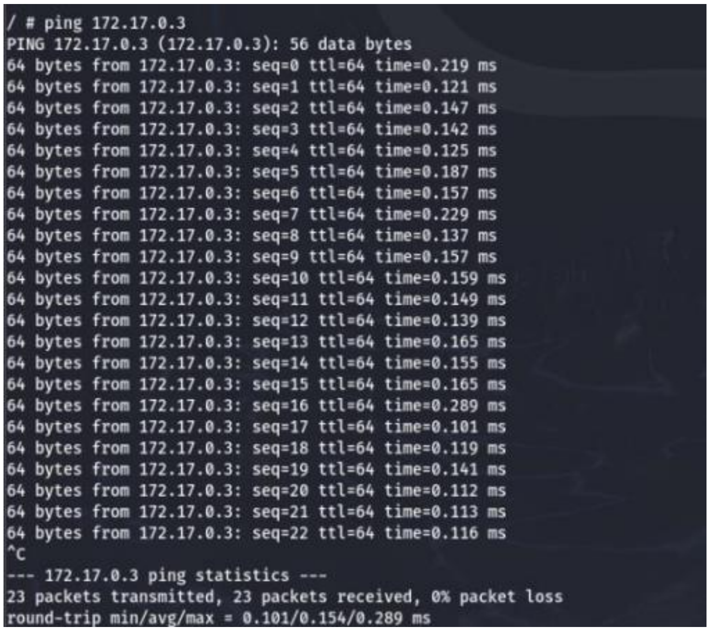
```
└$ sudo docker run -it --rm secunive/seclab:lab4 ash
/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:AC:11:00:03
          inet addr:172.17.0.3  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:10 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:796 (796.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

- In terminal 1, eth0's IP address is 127.17.0.2.
- In terminal 2, eth0's IP address is 127.17.0.3.

Question I:

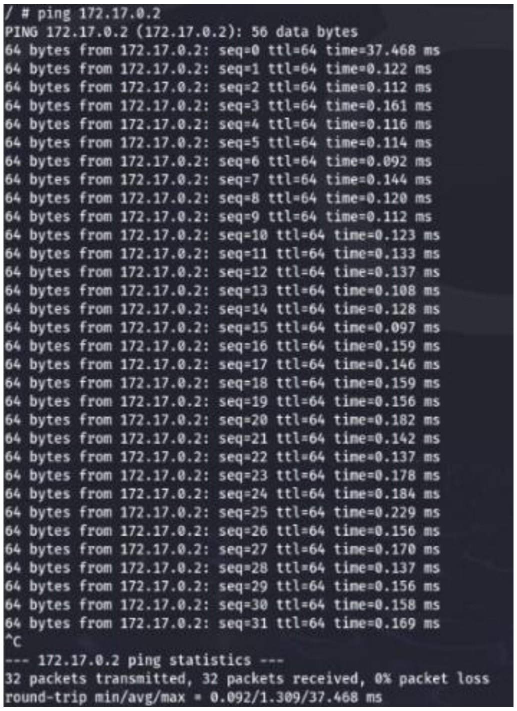
1. Ping host 172.17.0.2 from the second Docker instance in the second terminal (that should have the address 172.17.0.3) and vice-versa, to check network connectivity between the two hosts. Include screenshots confirming that you have executed the commands and received ping replies confirming connectivity between the two hosts.

   Terminal 1



```
/ # ping 172.17.0.3
PING 172.17.0.3 (172.17.0.3): 56 data bytes
64 bytes from 172.17.0.3: seq=0 ttl=64 time=0.219 ms
64 bytes from 172.17.0.3: seq=1 ttl=64 time=0.121 ms
64 bytes from 172.17.0.3: seq=2 ttl=64 time=0.147 ms
64 bytes from 172.17.0.3: seq=3 ttl=64 time=0.142 ms
64 bytes from 172.17.0.3: seq=4 ttl=64 time=0.125 ms
64 bytes from 172.17.0.3: seq=5 ttl=64 time=0.187 ms
64 bytes from 172.17.0.3: seq=6 ttl=64 time=0.157 ms
64 bytes from 172.17.0.3: seq=7 ttl=64 time=0.229 ms
64 bytes from 172.17.0.3: seq=8 ttl=64 time=0.137 ms
64 bytes from 172.17.0.3: seq=9 ttl=64 time=0.157 ms
64 bytes from 172.17.0.3: seq=10 ttl=64 time=0.159 ms
64 bytes from 172.17.0.3: seq=11 ttl=64 time=0.149 ms
64 bytes from 172.17.0.3: seq=12 ttl=64 time=0.139 ms
64 bytes from 172.17.0.3: seq=13 ttl=64 time=0.165 ms
64 bytes from 172.17.0.3: seq=14 ttl=64 time=0.155 ms
64 bytes from 172.17.0.3: seq=15 ttl=64 time=0.165 ms
64 bytes from 172.17.0.3: seq=16 ttl=64 time=0.289 ms
64 bytes from 172.17.0.3: seq=17 ttl=64 time=0.101 ms
64 bytes from 172.17.0.3: seq=18 ttl=64 time=0.119 ms
64 bytes from 172.17.0.3: seq=19 ttl=64 time=0.141 ms
64 bytes from 172.17.0.3: seq=20 ttl=64 time=0.112 ms
64 bytes from 172.17.0.3: seq=21 ttl=64 time=0.113 ms
64 bytes from 172.17.0.3: seq=22 ttl=64 time=0.116 ms
^C
--- 172.17.0.3 ping statistics ---
23 packets transmitted, 23 packets received, 0% packet loss
round-trip min/avg/max = 0.101/0.154/0.289 ms
```

Terminal 2



```
/ # ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2): 56 data bytes
64 bytes from 172.17.0.2: seq=0 ttl=64 time=37.468 ms
64 bytes from 172.17.0.2: seq=1 ttl=64 time=0.122 ms
64 bytes from 172.17.0.2: seq=2 ttl=64 time=0.112 ms
64 bytes from 172.17.0.2: seq=3 ttl=64 time=0.161 ms
64 bytes from 172.17.0.2: seq=4 ttl=64 time=0.116 ms
64 bytes from 172.17.0.2: seq=5 ttl=64 time=0.114 ms
64 bytes from 172.17.0.2: seq=6 ttl=64 time=0.092 ms
64 bytes from 172.17.0.2: seq=7 ttl=64 time=0.144 ms
64 bytes from 172.17.0.2: seq=8 ttl=64 time=0.120 ms
64 bytes from 172.17.0.2: seq=9 ttl=64 time=0.112 ms
64 bytes from 172.17.0.2: seq=10 ttl=64 time=0.123 ms
64 bytes from 172.17.0.2: seq=11 ttl=64 time=0.133 ms
64 bytes from 172.17.0.2: seq=12 ttl=64 time=0.137 ms
64 bytes from 172.17.0.2: seq=13 ttl=64 time=0.108 ms
64 bytes from 172.17.0.2: seq=14 ttl=64 time=0.128 ms
64 bytes from 172.17.0.2: seq=15 ttl=64 time=0.097 ms
64 bytes from 172.17.0.2: seq=16 ttl=64 time=0.159 ms
64 bytes from 172.17.0.2: seq=17 ttl=64 time=0.146 ms
64 bytes from 172.17.0.2: seq=18 ttl=64 time=0.159 ms
64 bytes from 172.17.0.2: seq=19 ttl=64 time=0.156 ms
64 bytes from 172.17.0.2: seq=20 ttl=64 time=0.182 ms
64 bytes from 172.17.0.2: seq=21 ttl=64 time=0.142 ms
64 bytes from 172.17.0.2: seq=22 ttl=64 time=0.137 ms
64 bytes from 172.17.0.2: seq=23 ttl=64 time=0.178 ms
64 bytes from 172.17.0.2: seq=24 ttl=64 time=0.184 ms
64 bytes from 172.17.0.2: seq=25 ttl=64 time=0.229 ms
64 bytes from 172.17.0.2: seq=26 ttl=64 time=0.156 ms
64 bytes from 172.17.0.2: seq=27 ttl=64 time=0.170 ms
64 bytes from 172.17.0.2: seq=28 ttl=64 time=0.137 ms
64 bytes from 172.17.0.2: seq=29 ttl=64 time=0.156 ms
64 bytes from 172.17.0.2: seq=30 ttl=64 time=0.158 ms
64 bytes from 172.17.0.2: seq=31 ttl=64 time=0.169 ms
^C
--- 172.17.0.2 ping statistics ---
32 packets transmitted, 32 packets received, 0% packet loss
round-trip min/avg/max = 0.092/1.309/37.468 ms
```

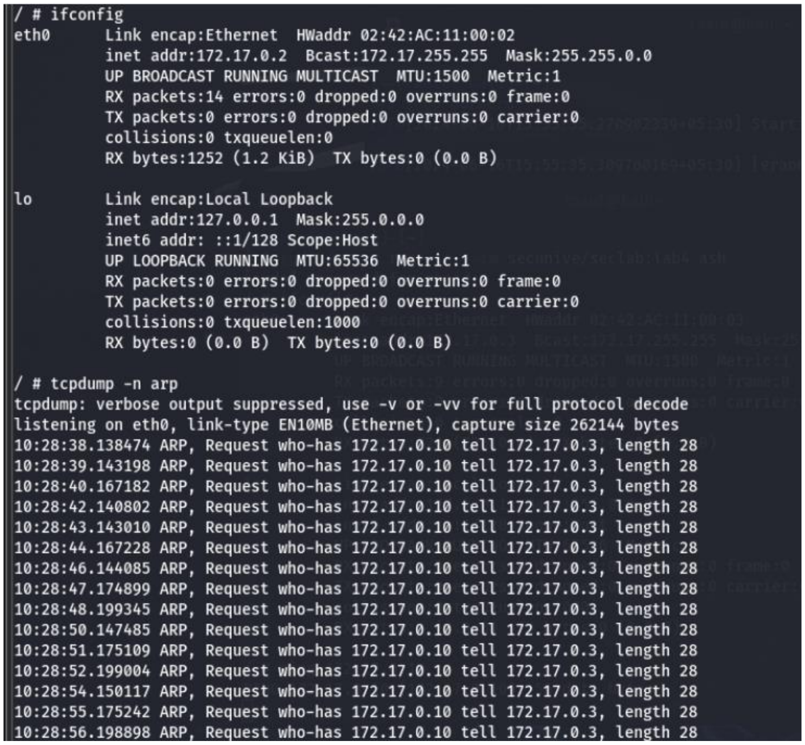
Question J Challenge – 2:

1. ARP protocol is used to discover the Media Access Control (MAC) address corresponding to a certain IP address. Whenever a host needs to connect to an IP that has not used recently (for which it has a cached MAC address), it broadcasts an ARP request.

ARP protocol is used to discover the Media Access Control (MAC) address corresponding to a certain IP address. Whenever a host needs to connect to an IP that has not used recently (for which it has a cached MAC address), it broadcasts an ARP request.

The.......... is the password.

What is the password you obtained?



```
/ # ifconfig
eth0      Link encap:Ethernet  HWaddr 02:42:AC:11:00:02
          inet addr:172.17.0.2  Bcast:172.17.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:14 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1252 (1.2 KiB)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

/ # tcpdump -n arp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
10:28:38.138474 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:39.143198 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:40.167182 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:42.140802 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:43.143010 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:44.167228 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:46.144085 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:47.174899 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:48.199345 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:50.147485 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:51.175109 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:52.199004 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:54.150117 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:55.175242 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
10:28:56.198898 ARP, Request who-has 172.17.0.10 tell 172.17.0.3, length 28
```

- who-has is the password.

Question K:

1. How did this task complement the theoretical concepts you learned in Week 4 and Week 5? What did you learn that was most exciting for you? Are you finding it easier to use the shell for hands-on activities?

    - I learned the fundamentals of packets inspection-the most crucial idea in network reconnaissance and docker through this assignment endeavor. How to efficiently use the tcpdump command to check domain this work helped me become proficient in these important areas.