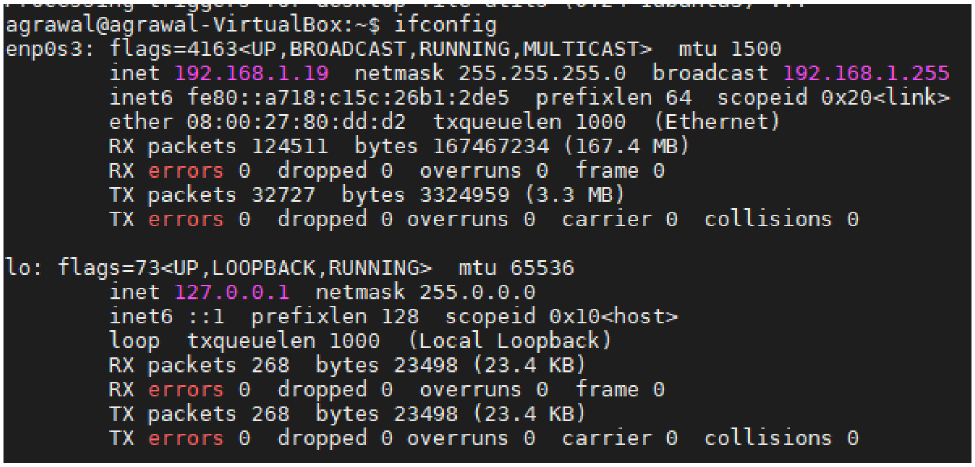
Week1 Lab work

Neha Agrawal

01.2: ARP, Wireshark, Netsim

1. ARP #1

- Use the ifconfig command to find the IP address and hardware address of the local virtual ethernet card interface.Ifconfig:



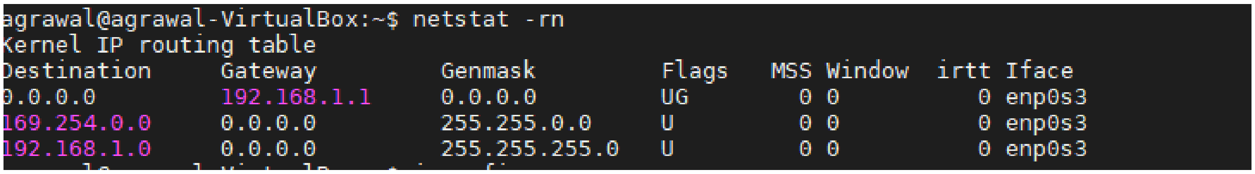
```
agrawal@agrawal-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.19  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a718:c15c:26b1:2de5  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:80:dd:d2  txqueuelen 1000  (Ethernet)
        RX packets 124511  bytes 167467234 (167.4 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32727  bytes 3324959 (3.3 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 268  bytes 23498 (23.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 268  bytes 23498 (23.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

IP address: 192.168.1.19

Hardware address: 08:00:27:80:dd:d2

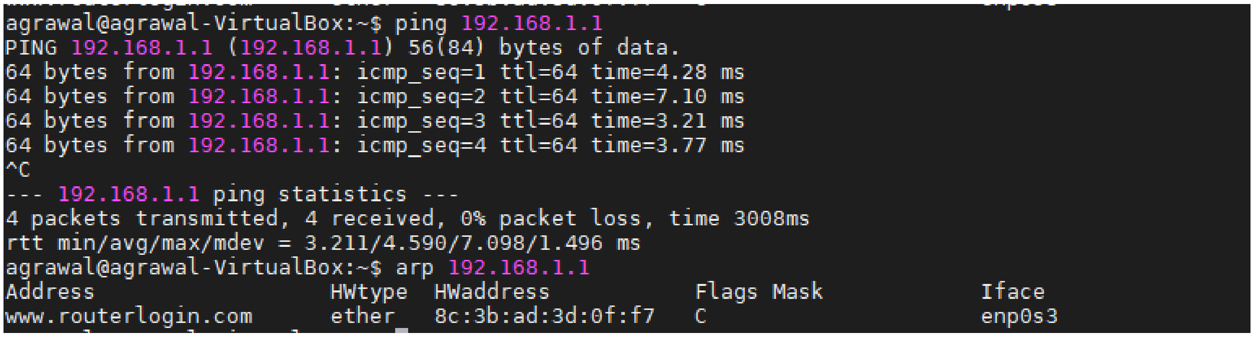
- Perform a netstat -rn to find default router's IP address



```
agrawal@agrawal-VirtualBox:~$ netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG        0 0          0 enp0s3
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 enp0s3
192.168.1.0     0.0.0.0         255.255.255.0   U         0 0          0 enp0s3
```

Default gateway (router) IP address: 192.168.1.1

- Ping the default router and use arp to find its hardware address

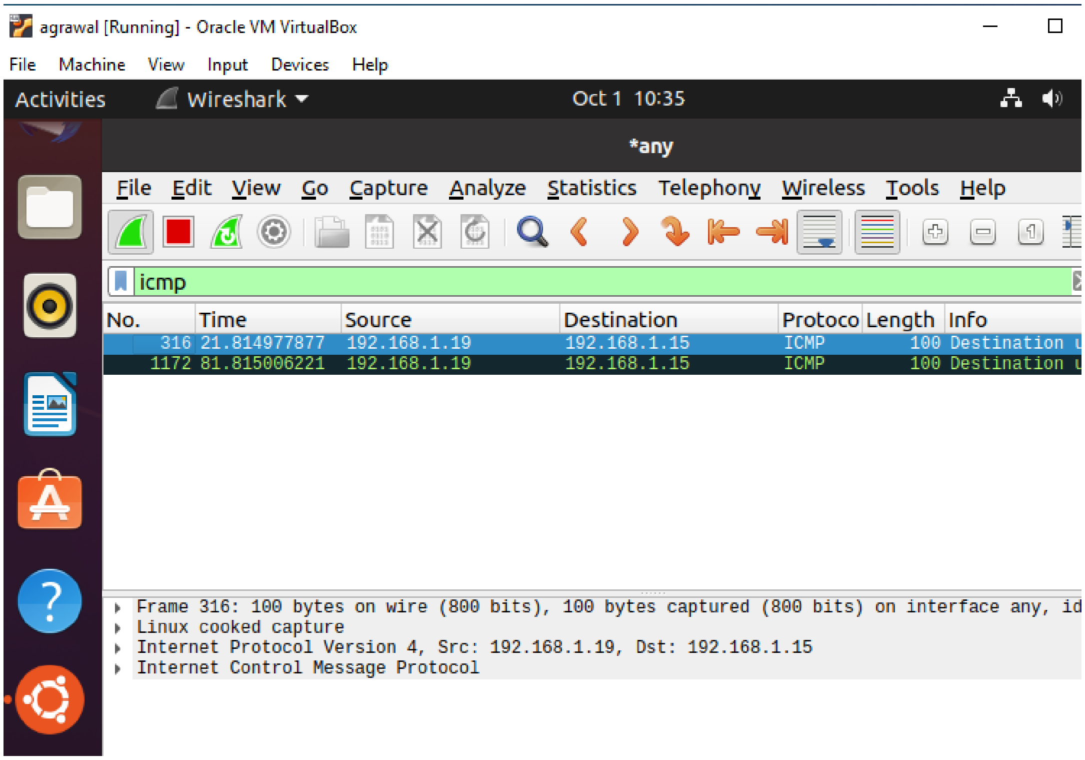


```
agrawal@agrawal-VirtualBox:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=4.28 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=7.10 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=3.21 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=3.77 ms
^C
--- 192.168.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3008ms
rtt min/avg/max/mdev = 3.211/4.590/7.098/1.496 ms
agrawal@agrawal-VirtualBox:~$ arp 192.168.1.1
Address                  HWtype  HWaddress           Flags Mask            Iface
www.routerlogin.com      ether   8c:3b:ad:3d:0f:f7   C                     enp0s3
```

HW address: 8c:3b:ad:3d:0f:f7

WIRESHARK:

- Use a "Capture Filter" to capture icmp (the protocol used by the ping command)

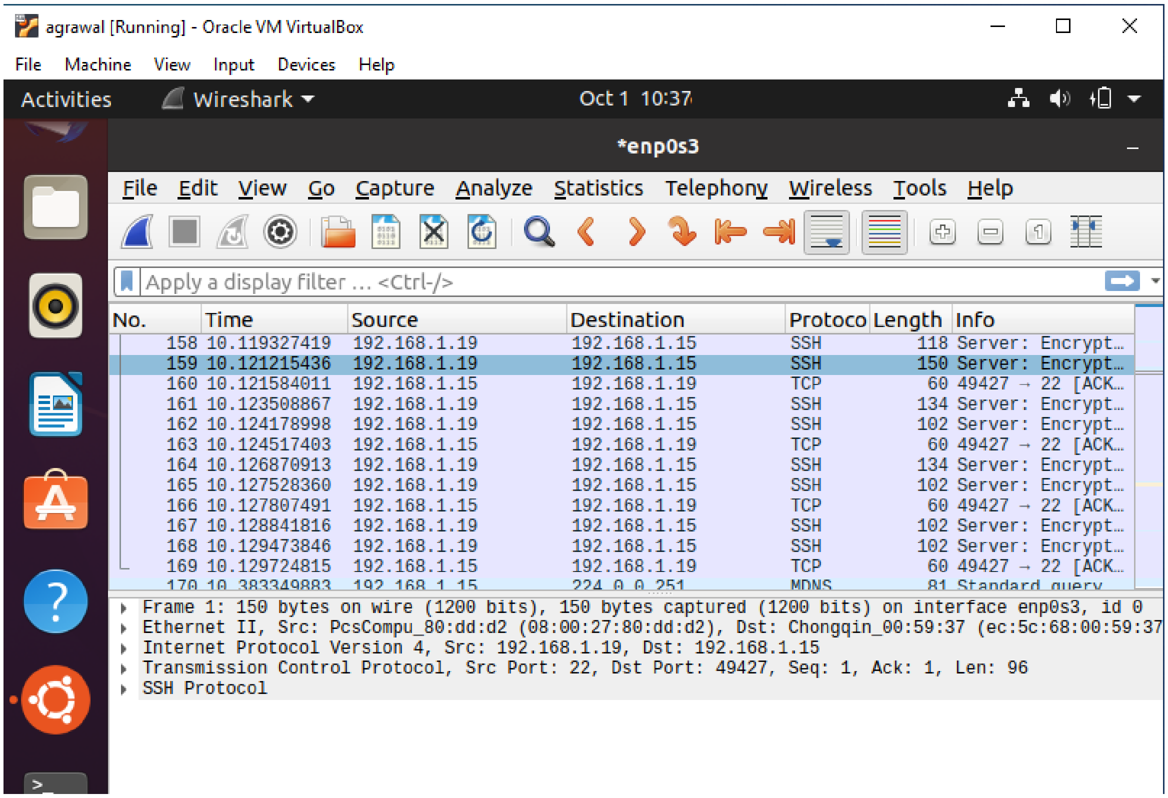

agrawal [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Wireshark
Oct 1 10:35
*any
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

icmp

No.	Time	Source	Destination	Protoco	Length	Info
316	21.814977877	192.168.1.19	192.168.1.15	ICMP	100	Destination u
1172	81.815006221	192.168.1.19	192.168.1.15	ICMP	100	Destination u

```
Frame 316: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface any, id
Linux cooked capture
Internet Protocol Version 4, Src: 192.168.1.19, Dst: 192.168.1.15
Internet Control Message Protocol
```

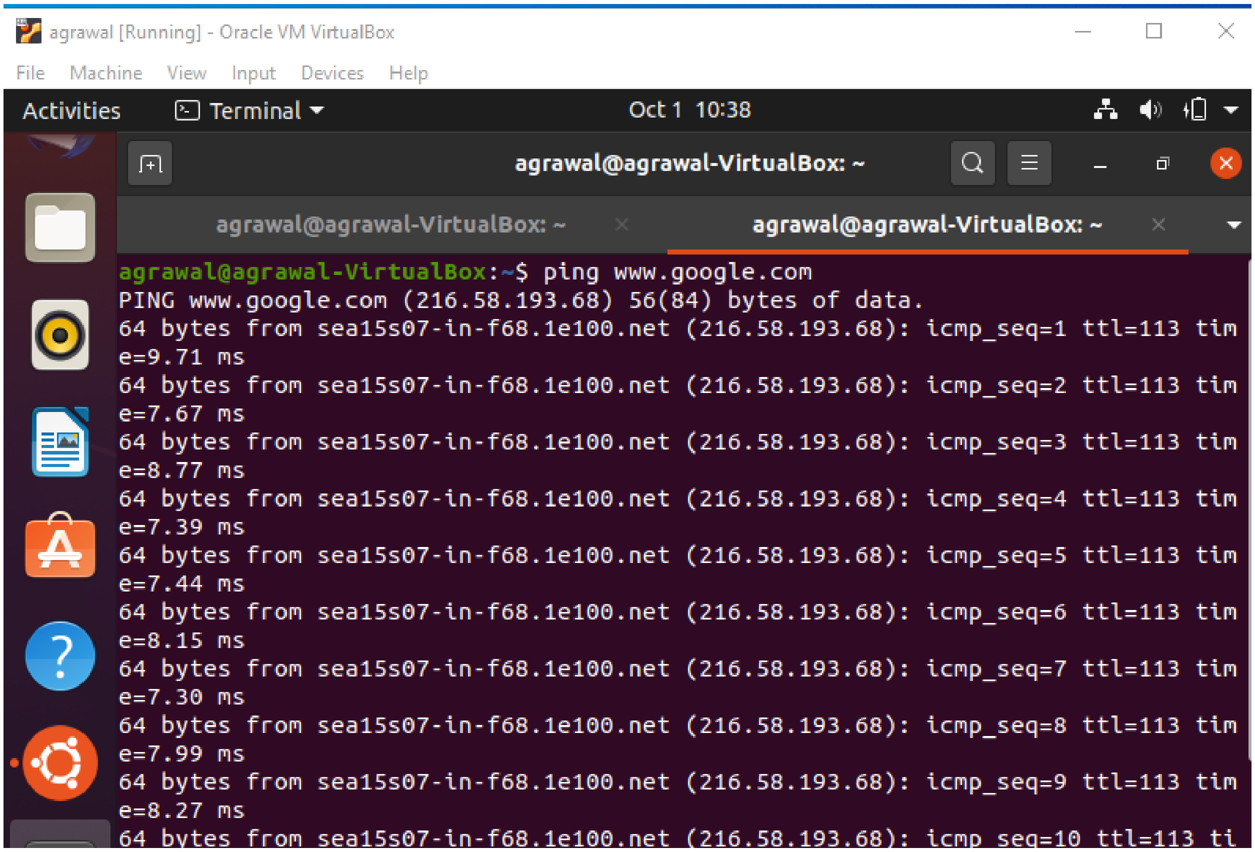
- Select your VMs virtual ethernet interface (e.g. enp0s3), then begin a capture


agrawal [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Wireshark
Oct 1 10:37
*enp0s3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

No.	Time	Source	Destination	Protoco	Length	Info
158	10.119327419	192.168.1.19	192.168.1.15	SSH	118	Server: Encrypt…
159	10.121215436	192.168.1.19	192.168.1.15	SSH	150	Server: Encrypt…
160	10.121584011	192.168.1.15	192.168.1.19	TCP	60	49427 → 22 [ACK…
161	10.123508867	192.168.1.19	192.168.1.15	SSH	134	Server: Encrypt…
162	10.124178998	192.168.1.19	192.168.1.15	SSH	102	Server: Encrypt…
163	10.124517403	192.168.1.15	192.168.1.19	TCP	60	49427 → 22 [ACK…
164	10.126870913	192.168.1.19	192.168.1.15	SSH	134	Server: Encrypt…
165	10.127528360	192.168.1.19	192.168.1.15	SSH	102	Server: Encrypt…
166	10.127807491	192.168.1.15	192.168.1.19	TCP	60	49427 → 22 [ACK…
167	10.128841816	192.168.1.19	192.168.1.15	SSH	102	Server: Encrypt…
168	10.129473846	192.168.1.19	192.168.1.15	SSH	102	Server: Encrypt…
169	10.129724815	192.168.1.15	192.168.1.19	TCP	60	49427 → 22 [ACK…
170	10.383349883	192.168.1.15	224.0.0.251	MDNS	81	Standard query

```
Frame 1: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits) on interface enp0s3, id 0
Ethernet II, Src: PcsCompu_80:dd:d2 (08:00:27:80:dd:d2), Dst: Chongqin_00:59:37 (ec:5c:68:00:59:37
Internet Protocol Version 4, Src: 192.168.1.19, Dst: 192.168.1.15
Transmission Control Protocol, Src Port: 22, Dst Port: 49427, Seq: 1, Ack: 1, Len: 96
SSH Protocol
```

- In a separate terminal, ping www.google.com



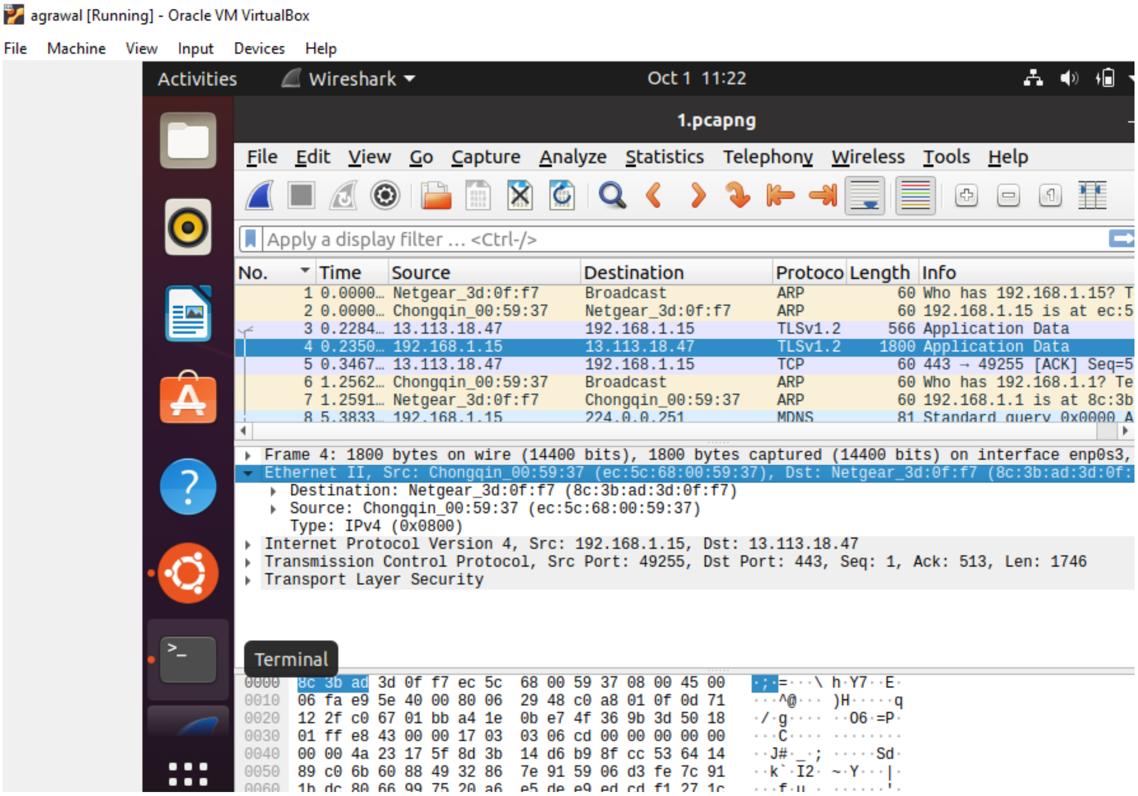
agrawal [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
Oct 1 10:38
agrawal@agrawal-VirtualBox: ~
agrawal@agrawal-VirtualBox: ~
agrawal@agrawal-VirtualBox: ~

```
agrawal@agrawal-VirtualBox:~$ ping www.google.com
PING www.google.com (216.58.193.68) 56(84) bytes of data.
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=1 ttl=113 tim
e=9.71 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=2 ttl=113 tim
e=7.67 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=3 ttl=113 tim
e=8.77 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=4 ttl=113 tim
e=7.39 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=5 ttl=113 tim
e=7.44 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=6 ttl=113 tim
e=8.15 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=7 ttl=113 tim
e=7.30 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=8 ttl=113 tim
e=7.99 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=9 ttl=113 tim
e=8.27 ms
64 bytes from sea15s07-in-f68.1e100.net (216.58.193.68): icmp_seq=10 ttl=113 ti
```

Click on the request packet in the top window of the wireshark UI. Then, in the middle window, expand the data-link layer packet and click on the source and destination hardware addresses.

- Which hardware manufacturer does the destination hardware address of the packet indicate?
- Show the bytes in the packet dump window as shown below

A. Request Packet:


agrawal [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Wireshark
Oct 1 11:22
1.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Terminal

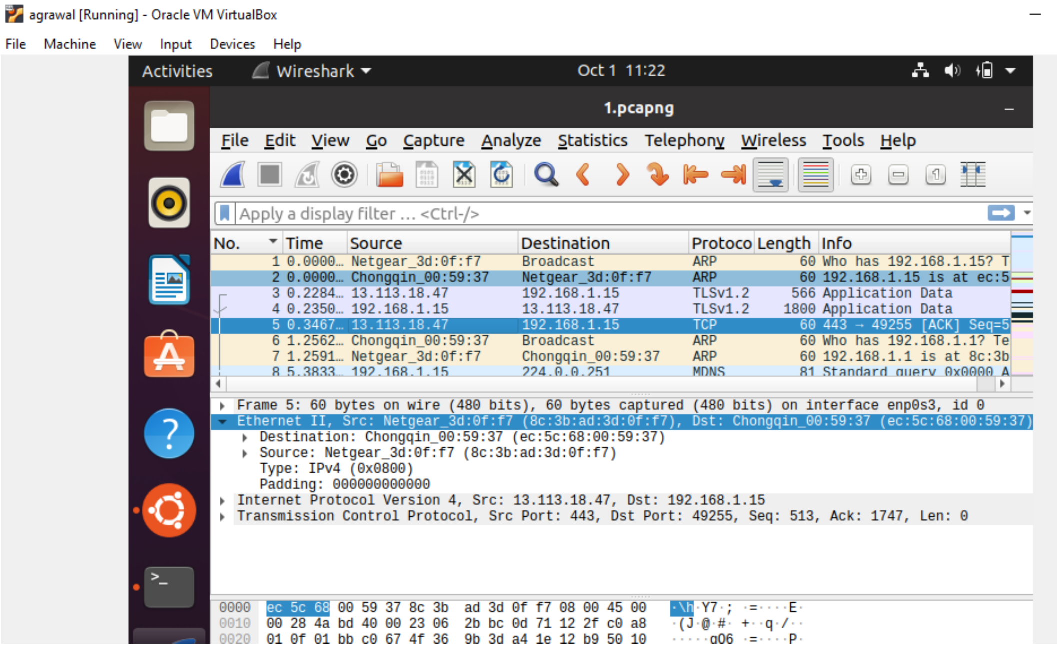
```
No.   Time      Source              Destination         Protoco Length Info
    1 0.0000…  Netgear_3d:0f:f7    Broadcast           ARP         60 Who has 192.168.1.15? T
    2 0.0000…  Chongqin_00:59:37   Netgear_3d:0f:f7    ARP         60 192.168.1.15 is at ec:5
    3 0.2284…  13.113.18.47        192.168.1.15        TLSv1.2    566 Application Data
    4 0.2350…  192.168.1.15        13.113.18.47        TLSv1.2   1800 Application Data
    5 0.3467…  13.113.18.47        192.168.1.15        TCP         60 443 → 49255 [ACK] Seq=5
    6 1.2562…  Chongqin_00:59:37   Broadcast           ARP         60 Who has 192.168.1.1? Te
    7 1.2591…  Netgear_3d:0f:f7    Chongqin_00:59:37   ARP         60 192.168.1.1 is at 8c:3b
    8 5.3833…  192.168.1.15        224.0.0.251         MDNS        81 Standard query 0x0000 A
```

```
▸ Frame 4: 1800 bytes on wire (14400 bits), 1800 bytes captured (14400 bits) on interface enp0s3,
▾ Ethernet II, Src: Chongqin_00:59:37 (ec:5c:68:00:59:37), Dst: Netgear_3d:0f:f7 (8c:3b:ad:3d:0f:
   ▸ Destination: Netgear_3d:0f:f7 (8c:3b:ad:3d:0f:f7)
   ▸ Source: Chongqin_00:59:37 (ec:5c:68:00:59:37)
     Type: IPv4 (0x0800)
▸ Internet Protocol Version 4, Src: 192.168.1.15, Dst: 13.113.18.47
▸ Transmission Control Protocol, Src Port: 49255, Dst Port: 443, Seq: 1, Ack: 513, Len: 1746
▸ Transport Layer Security
```

```
0000  8c 3b ad 3d 0f f7 ec 5c  68 00 59 37 08 00 45 00   ·;·=···\ h·Y7··E·
0010  06 fa e9 5e 40 00 80 06  29 48 c0 a8 01 0f 0d 71   ···^@···  )H·····q
0020  12 2f c0 67 01 bb a4 1e  0b e7 4f 36 9b 3d 50 18   ·/·g····  ··O6·=P·
0030  01 ff e8 43 00 00 17 03  03 06 cd 00 00 00 00 00   ···C····  ········
0040  00 00 4a 23 17 5f 8d 3b  14 d6 b9 8f cc 53 64 14   ··J#·_·;  ·····Sd·
0050  89 c0 6b 60 88 49 32 86  7e 91 59 06 d3 fe 7c 91   ··k`·I2·  ~·Y···|·
0060  1b dc 80 66 99 75 20 a6  e5 de e9 ed cd f1 27 1c   ···f·u·  ·······'·
```

Hardware manufacturer: Netgear_3d:0f:f7

B. Response Packet:


agrawal [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Wireshark
Oct 1 11:22
1.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.   Time      Source              Destination         Protoco Length Info
    1 0.0000…  Netgear_3d:0f:f7    Broadcast           ARP         60 Who has 192.168.1.15? T
    2 0.0000…  Chongqin_00:59:37   Netgear_3d:0f:f7    ARP         60 192.168.1.15 is at ec:5
    3 0.2284…  13.113.18.47        192.168.1.15        TLSv1.2    566 Application Data
    4 0.2350…  192.168.1.15        13.113.18.47        TLSv1.2   1800 Application Data
    5 0.3467…  13.113.18.47        192.168.1.15        TCP         60 443 → 49255 [ACK] Seq=5
    6 1.2562…  Chongqin_00:59:37   Broadcast           ARP         60 Who has 192.168.1.1? Te
    7 1.2591…  Netgear_3d:0f:f7    Chongqin_00:59:37   ARP         60 192.168.1.1 is at 8c:3b
    8 5.3833…  192.168.1.15        224.0.0.251         MDNS        81 Standard query 0x0000 A
```

```
▸ Frame 5: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface enp0s3, id 0
▾ Ethernet II, Src: Netgear_3d:0f:f7 (8c:3b:ad:3d:0f:f7), Dst: Chongqin_00:59:37 (ec:5c:68:00:59:37)
   ▸ Destination: Chongqin_00:59:37 (ec:5c:68:00:59:37)
   ▸ Source: Netgear_3d:0f:f7 (8c:3b:ad:3d:0f:f7)
     Type: IPv4 (0x0800)
     Padding: 000000000000
▸ Internet Protocol Version 4, Src: 13.113.18.47, Dst: 192.168.1.15
▸ Transmission Control Protocol, Src Port: 443, Dst Port: 49255, Seq: 513, Ack: 1747, Len: 0
```

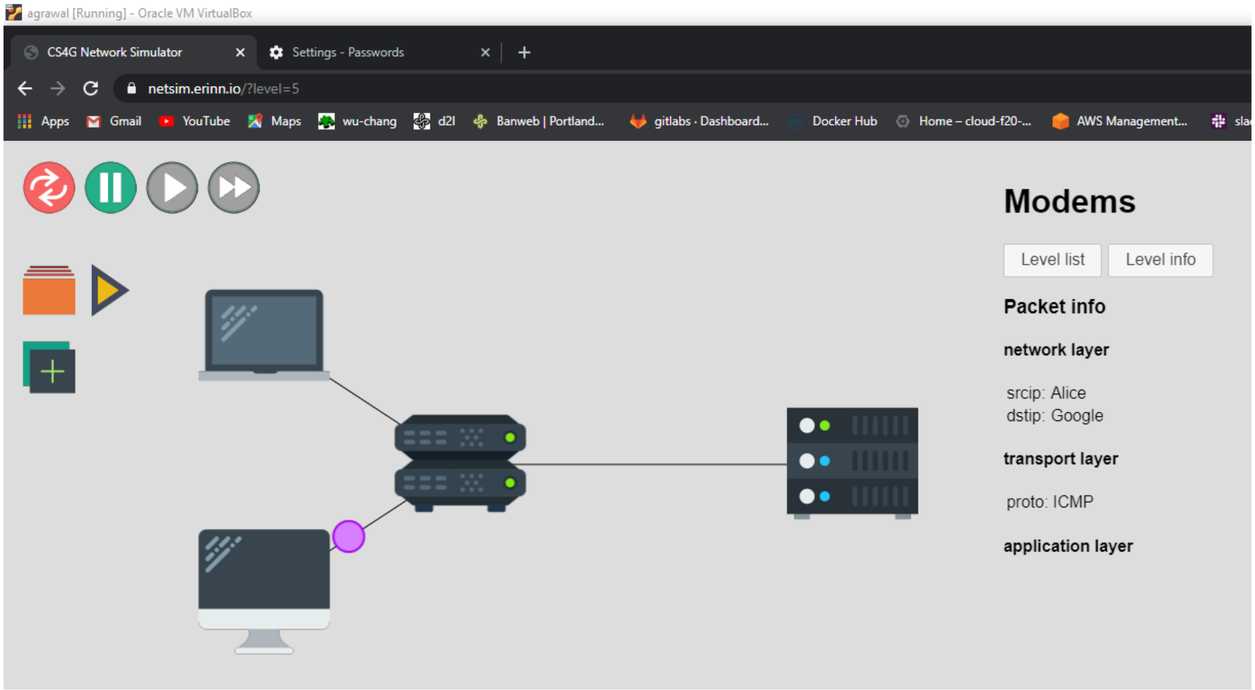
```
0000  ec 5c 68 00 59 37 8c 3b  ad 3d 0f f7 08 00 45 00   ·\h·Y7·; ·=····E·
0010  00 28 4a bd 40 00 23 06  2b bc 0d 71 12 2f c0 a8   ·(J·@·#· +··q·/··
0020  01 0f 01 bb c0 67 4f 36  9b 3d a4 1e 12 b9 50 10   ·····gO6 ·=····P·
```

Hardware manufacturer: Chongquin_00:59:37

2. Netsim #2

Modem Level #5

Before:



Packet info

network layer

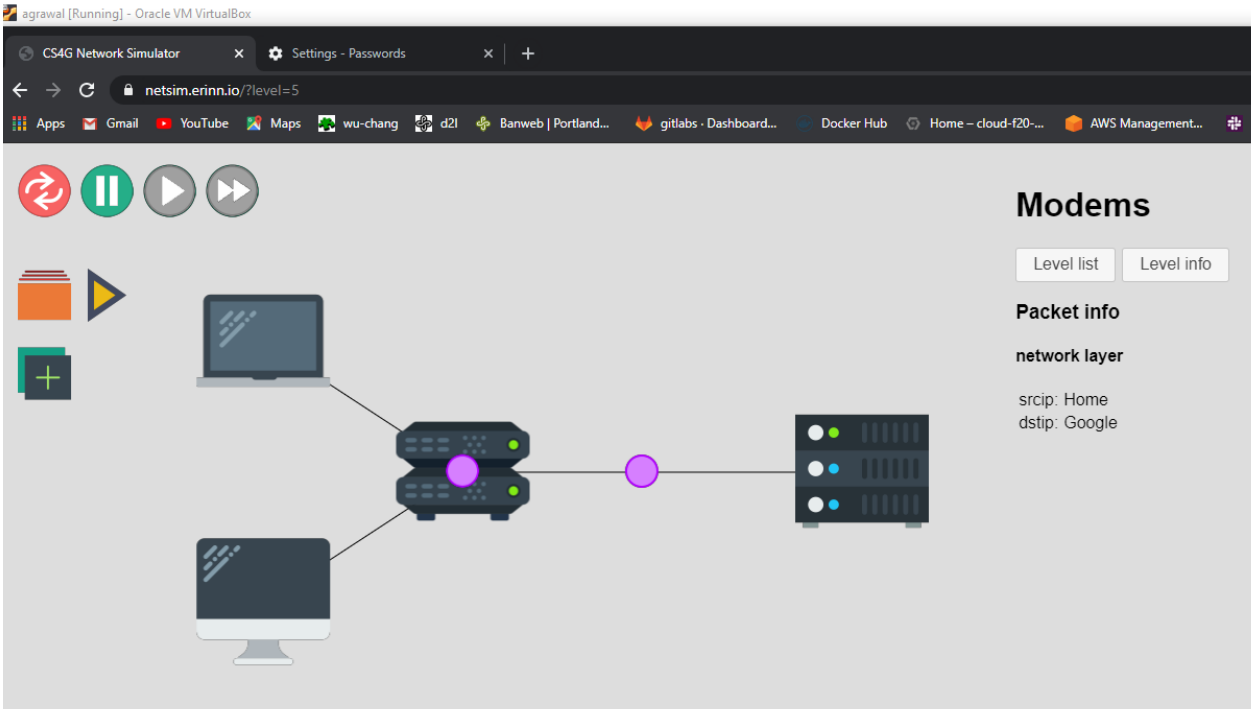
srcip: Alice
dstip: Google

transport layer

proto: ICMP

application layer

After:

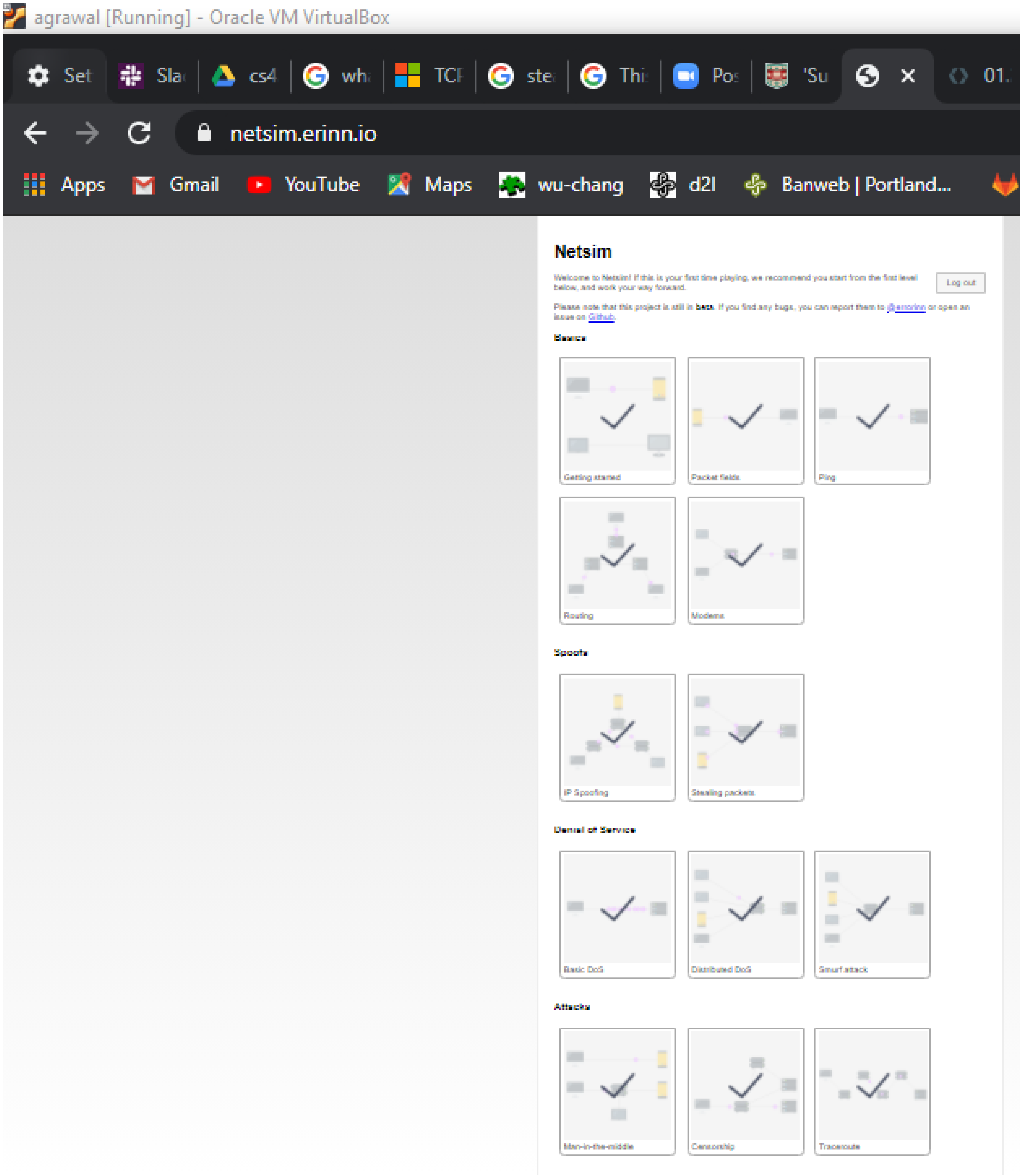


Packet info

network layer

srcip: Home
dstip: Google

All the levels completed:



Netsim

Welcome to Netsim! If this is your first time playing, we recommend you start from the first level below, and work your way forward.

Please note that this project is still in **beta**. If you find any bugs, you can report them to @errorinn or open an issue on Github.

Basics

- ✓ Getting started
- ✓ Packet fields
- ✓ Ping
- ✓ Routing
- ✓ Modems

Spoofs

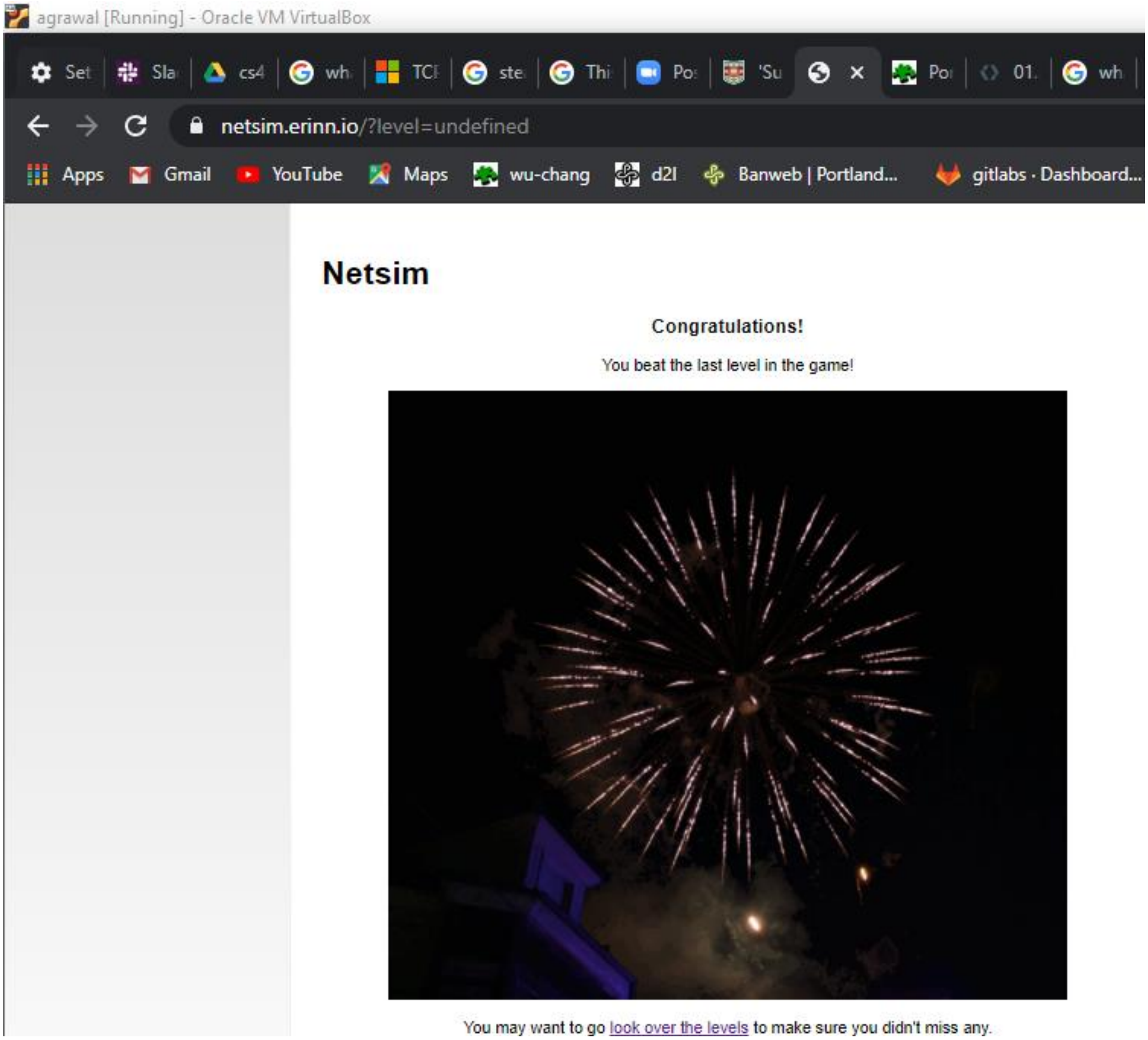
- ✓ IP Spoofing
- ✓ Stealing packets

Denial of Service

- ✓ Basic DoS
- ✓ Distributed DoS
- ✓ Smurf attack

Attacks

- ✓ Man-in-the-middle
- ✓ Censorship
- ✓ Traceroute



Netsim

Congratulations!

You beat the last level in the game!

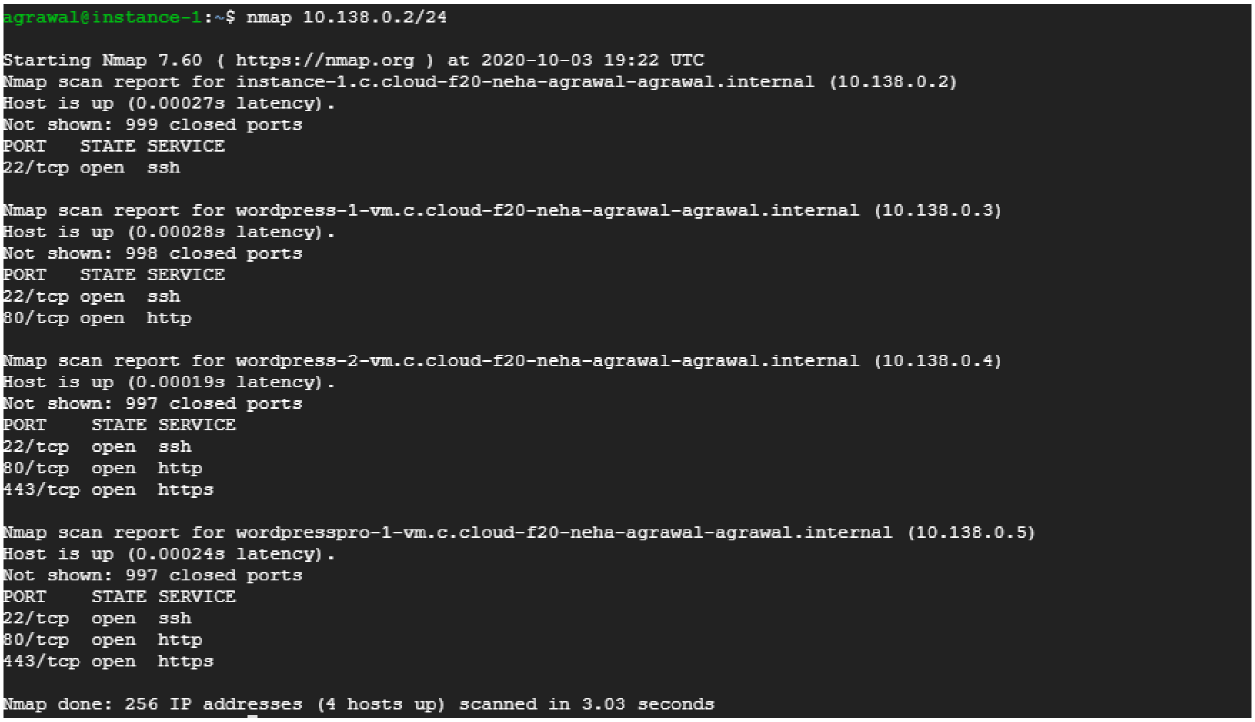
You may want to go look over the levels to make sure you didn't miss any.

01.3: Cloud Networking

1. Network scanning (nmap) #1

- **Show a screenshot of the output for the scan for your lab notebook.**

  You should see a list of ports that each machine exposes over the network. This provides administrators important data for taking an inventory of their infrastructure in order to ensure only a minimal set of services are exposed.



```
agrawal@instance-1:~$ nmap 10.138.0.2/24

Starting Nmap 7.60 ( https://nmap.org ) at 2020-10-03 19:22 UTC
Nmap scan report for instance-1.c.cloud-f20-neha-agrawal-agrawal.internal (10.138.0.2)
Host is up (0.00027s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap scan report for wordpress-1-vm.c.cloud-f20-neha-agrawal-agrawal.internal (10.138.0.3)
Host is up (0.00028s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap scan report for wordpress-2-vm.c.cloud-f20-neha-agrawal-agrawal.internal (10.138.0.4)
Host is up (0.00019s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
443/tcp open  https

Nmap scan report for wordpresspro-1-vm.c.cloud-f20-neha-agrawal-agrawal.internal (10.138.0.5)
Host is up (0.00024s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
443/tcp open  https

Nmap done: 256 IP addresses (4 hosts up) scanned in 3.03 seconds
```

4. CIDR and subnets #2

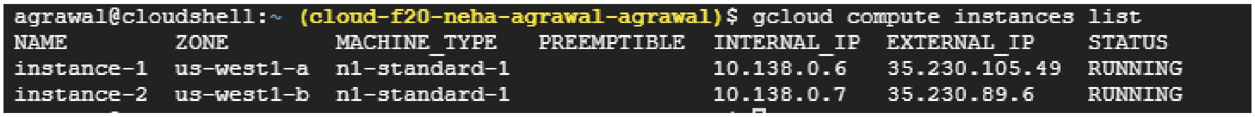
- **How many subnetworks are created initially on the default network? How many regions does this correspond to? (Use a pipe to pass output to grep in order to return specific lines of output and then another to pass output to wc to count them: | grep default | wc -l)**

  24

- **Given the CIDR prefix associated with each subnetwork, how many hosts does each subnetwork support?**

  2^{12}= *4096*

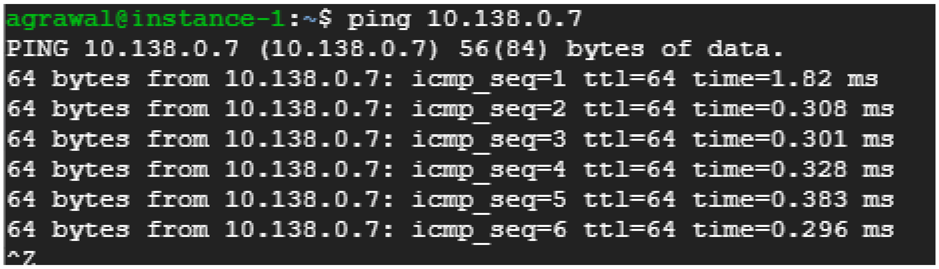
- **Which CIDR subnetworks are these instances brought up in? Do they correspond to the appropriate region based on the prior commands?**



```
agrawal@cloudshell:~ (cloud-f20-neha-agrawal-agrawal)$ gcloud compute instances list
NAME        ZONE        MACHINE_TYPE   PREEMPTIBLE  INTERNAL_IP  EXTERNAL_IP    STATUS
instance-1  us-west1-a  n1-standard-1               10.138.0.6   35.230.105.49  RUNNING
instance-2  us-west1-b  n1-standard-1               10.138.0.7   35.230.89.6    RUNNING
```

Both the instances are brought up in 10.138.0.0/20 CIDR subnetwork range.
Yes, they both corresponds to appropriate region which is us-west1

From instance-1, perform a ping to the Internal IP address of instance-2. Take a screenshot of the output.

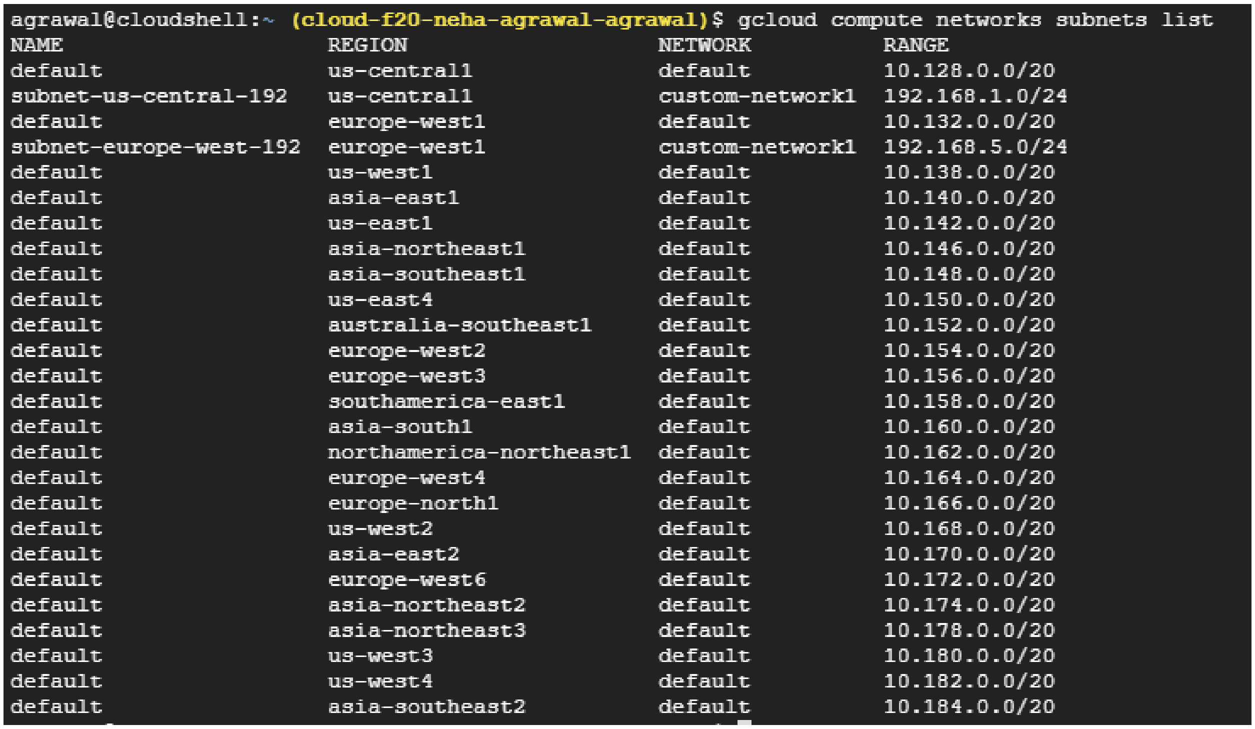


```
agrawal@instance-1:~$ ping 10.138.0.7
PING 10.138.0.7 (10.138.0.7) 56(84) bytes of data.
64 bytes from 10.138.0.7: icmp_seq=1 ttl=64 time=1.82 ms
64 bytes from 10.138.0.7: icmp_seq=2 ttl=64 time=0.308 ms
64 bytes from 10.138.0.7: icmp_seq=3 ttl=64 time=0.301 ms
64 bytes from 10.138.0.7: icmp_seq=4 ttl=64 time=0.328 ms
64 bytes from 10.138.0.7: icmp_seq=5 ttl=64 time=0.383 ms
64 bytes from 10.138.0.7: icmp_seq=6 ttl=64 time=0.296 ms
^Z
```

- **From the figure in the previous step. What facilitates this connectivity: the virtual switch or the VPN Gateway?**

  The virtual switch

- **Include a screenshot of the new subnets created in custom-network1 alongside the default subnetworks in those regions assigned to the default network.**

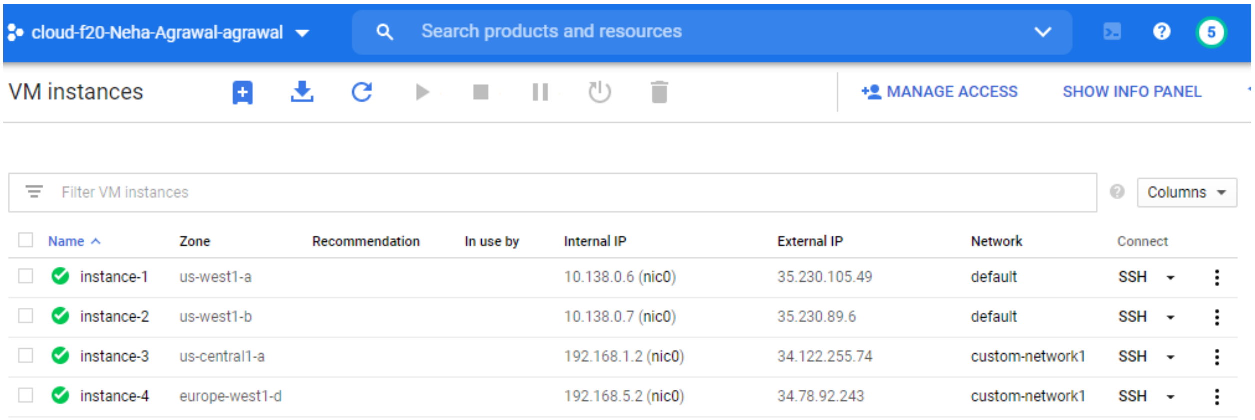


```
agrawal@cloudshell:~ (cloud-f20-neha-agrawal-agrawal)$ gcloud compute networks subnets list
NAME                    REGION                   NETWORK          RANGE
default                 us-central1              default          10.128.0.0/20
subnet-us-central-192   us-central1              custom-network1  192.168.1.0/24
default                 europe-west1             default          10.132.0.0/20
subnet-europe-west-192  europe-west1             custom-network1  192.168.5.0/24
default                 us-west1                 default          10.138.0.0/20
default                 asia-east1               default          10.140.0.0/20
default                 us-east1                 default          10.142.0.0/20
default                 asia-northeast1          default          10.146.0.0/20
default                 asia-southeast1          default          10.148.0.0/20
default                 us-east4                 default          10.150.0.0/20
default                 australia-southeast1     default          10.152.0.0/20
default                 europe-west2             default          10.154.0.0/20
default                 europe-west3             default          10.156.0.0/20
default                 southamerica-east1       default          10.158.0.0/20
default                 asia-south1              default          10.160.0.0/20
default                 northamerica-northeast1  default          10.162.0.0/20
default                 europe-west4             default          10.164.0.0/20
default                 europe-north1            default          10.166.0.0/20
default                 us-west2                 default          10.168.0.0/20
default                 asia-east2               default          10.170.0.0/20
default                 europe-west6             default          10.172.0.0/20
default                 asia-northeast2          default          10.174.0.0/20
default                 asia-northeast3          default          10.178.0.0/20
default                 us-west3                 default          10.180.0.0/20
default                 us-west4                 default          10.182.0.0/20
default                 asia-southeast2          default          10.184.0.0/20
```

- **Explain why the result is different from instance-2.**

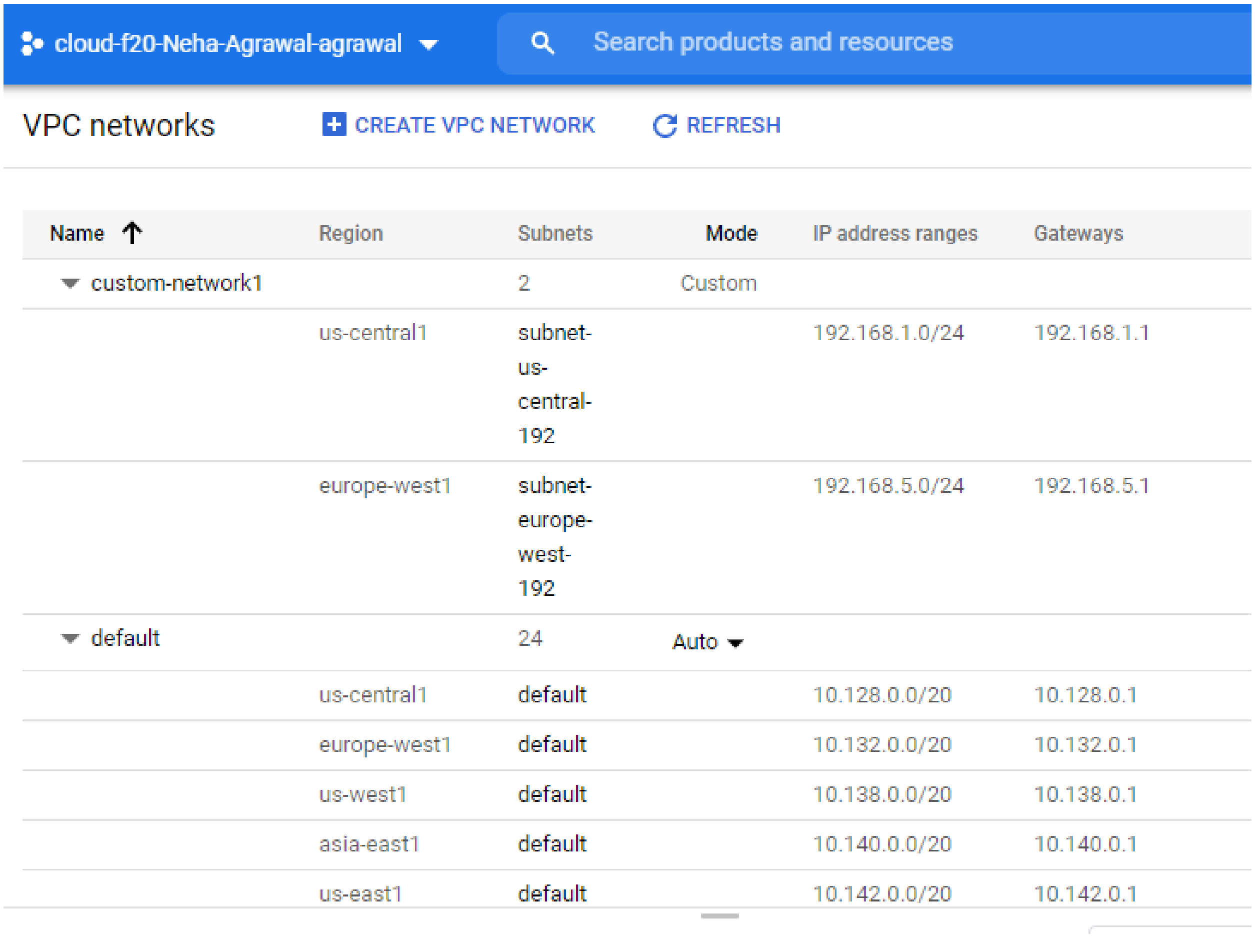
  We are not able to perform the ping from instance 1 to instance 3 and 4, this is because they both belong to different networks. To enable communication amongst all 4 instances we need to set up peering between the two networks.

- **Take screenshots of all 4 instances in the UI including the network they belong to.**



	Name	Zone	Recommendation	In use by	Internal IP	External IP	Network	Connect
□	✓ instance-1	us-west1-a			10.138.0.6 (nic0)	35.230.105.49	default	SSH
□	✓ instance-2	us-west1-b			10.138.0.7 (nic0)	35.230.89.6	default	SSH
□	✓ instance-3	us-central1-a			192.168.1.2 (nic0)	34.122.255.74	custom-network1	SSH
□	✓ instance-4	europe-west1-d			192.168.5.2 (nic0)	34.78.92.243	custom-network1	SSH

- **Then visit "VPC Network" and take a screenshot of the subnetworks created.**



Name ↑	Region	Subnets	Mode	IP address ranges	Gateways
custom-network1		2	Custom		
	us-central1	subnet-us-central-192		192.168.1.0/24	192.168.1.1
	europe-west1	subnet-europe-west-192		192.168.5.0/24	192.168.5.1
default		24	Auto		
	us-central1	default		10.128.0.0/20	10.128.0.1
	europe-west1	default		10.132.0.0/20	10.132.0.1
	us-west1	default		10.138.0.0/20	10.138.0.1
	asia-east1	default		10.140.0.0/20	10.140.0.1
	us-east1	default		10.142.0.0/20	10.142.0.1