# Week1

## Lab1

Armant Touche

**Class/Instructor:** CS430P/ Dr. Wu-Chang
**Date:** 9/30/22

# Table of Contents

1. Lab1, HW1

   1.1. Homework #1

   1.2. ARP, Wireshark, Netsim

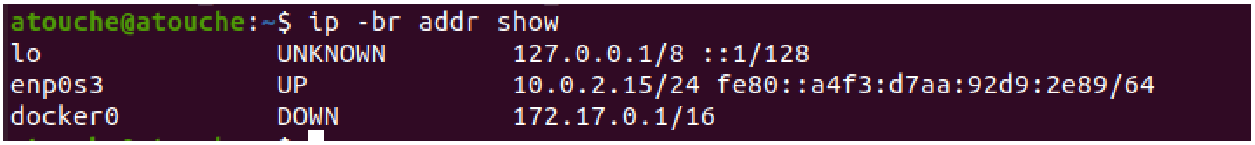
   1.3. Cloud networking

# 1.1. Homework #1 (Link)

1. Linux VM setup
    - □ Download Ubuntu 20.04 VM
        - i. Link: https://releases.ubuntu.com/20.04/
    - □ VirtualBox (VB)
        - i. Install
            1. Link: https://www.virtualbox.org/wiki/Downloads
2. Slack Account
    - □ Join
3. GitLab Account
    - □ Signup
    - □ Add `id_rsa.pub` key in Preferences -> SSH Keys
4. GitLab repo
    - □ Create Project
    - □ Invite Instructor and TA
    - □ Setup local git in VB, clone, and add README
5. Git
    - □ Init notebook
6. Docker Hub account
    - □ Add `dockerhub.txt` after signing up with @pdx.edu email
7. Google Cloud Platform account
    - □ Get Coupon
    - □ Create Project
8. AWS Academy
    - □ Wait for Canvas invite (NOTE: did not receive yet)
9. AWS via Vora
10. AWS CLI
    - □ Using Option **(1)**

# 1.2 ARP, Wireshark, Netsim (Link)

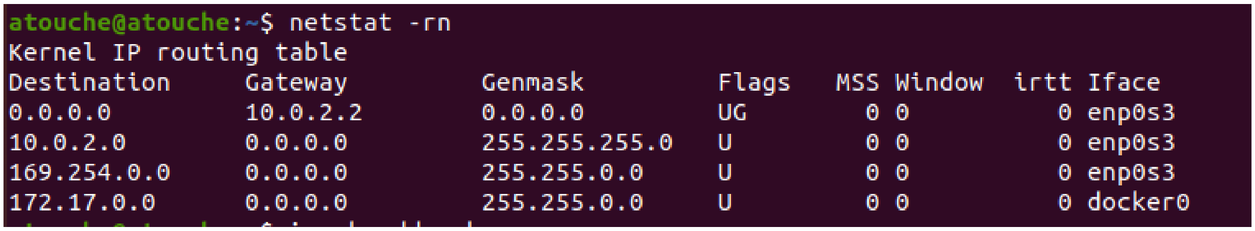
1. ARP #1
   - ☐ Install Wireshark
   - ☐ Perform tasks
     i. Use `ip` cmd to find virtual `lo` interface address



```
atouche@atouche:~$ ip -br addr show
lo               UNKNOWN        127.0.0.1/8 ::1/128
enp0s3           UP             10.0.2.15/24 fe80::a4f3:d7aa:92d9:2e89/64
docker0          DOWN           172.17.0.1/16
```

1. lo: 10.0.2.15/24

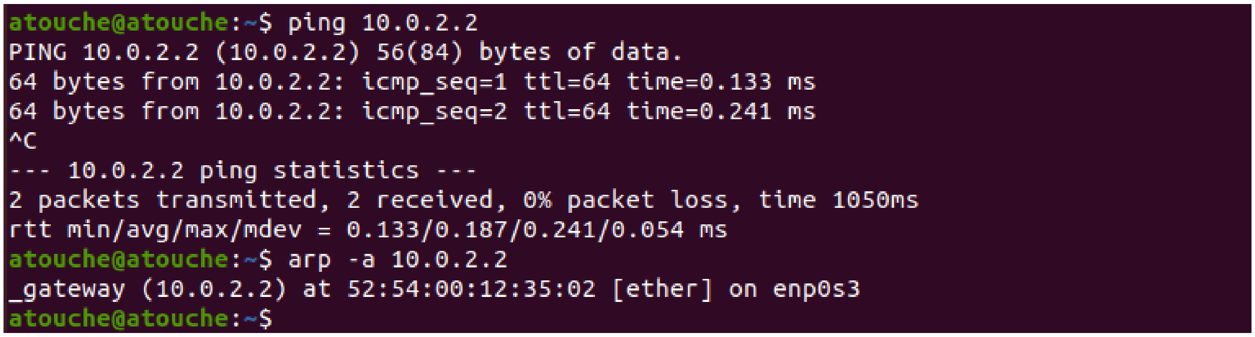
ii. Perform `netstat -rn` d to find default router’s IP address



```
atouche@atouche:~$ netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.0.2.2        0.0.0.0         UG        0 0          0 enp0s3
10.0.2.0        0.0.0.0         255.255.255.0   U         0 0          0 enp0s3
169.254.0.0     0.0.0.0         255.255.0.0     U         0 0          0 enp0s3
172.17.0.0      0.0.0.0         255.255.0.0     U         0 0          0 docker0
```

1. gateway: 10.0.2.2

iii. `ping` default and use `arp` to find it’s hardware address (`MAC`)



```
atouche@atouche:~$ ping 10.0.2.2
PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data.
64 bytes from 10.0.2.2: icmp_seq=1 ttl=64 time=0.133 ms
64 bytes from 10.0.2.2: icmp_seq=2 ttl=64 time=0.241 ms
^C
--- 10.0.2.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1050ms
rtt min/avg/max/mdev = 0.133/0.187/0.241/0.054 ms
atouche@atouche:~$ arp -a 10.0.2.2
_gateway (10.0.2.2) at 52:54:00:12:35:02 [ether] on enp0s3
atouche@atouche:~$
```

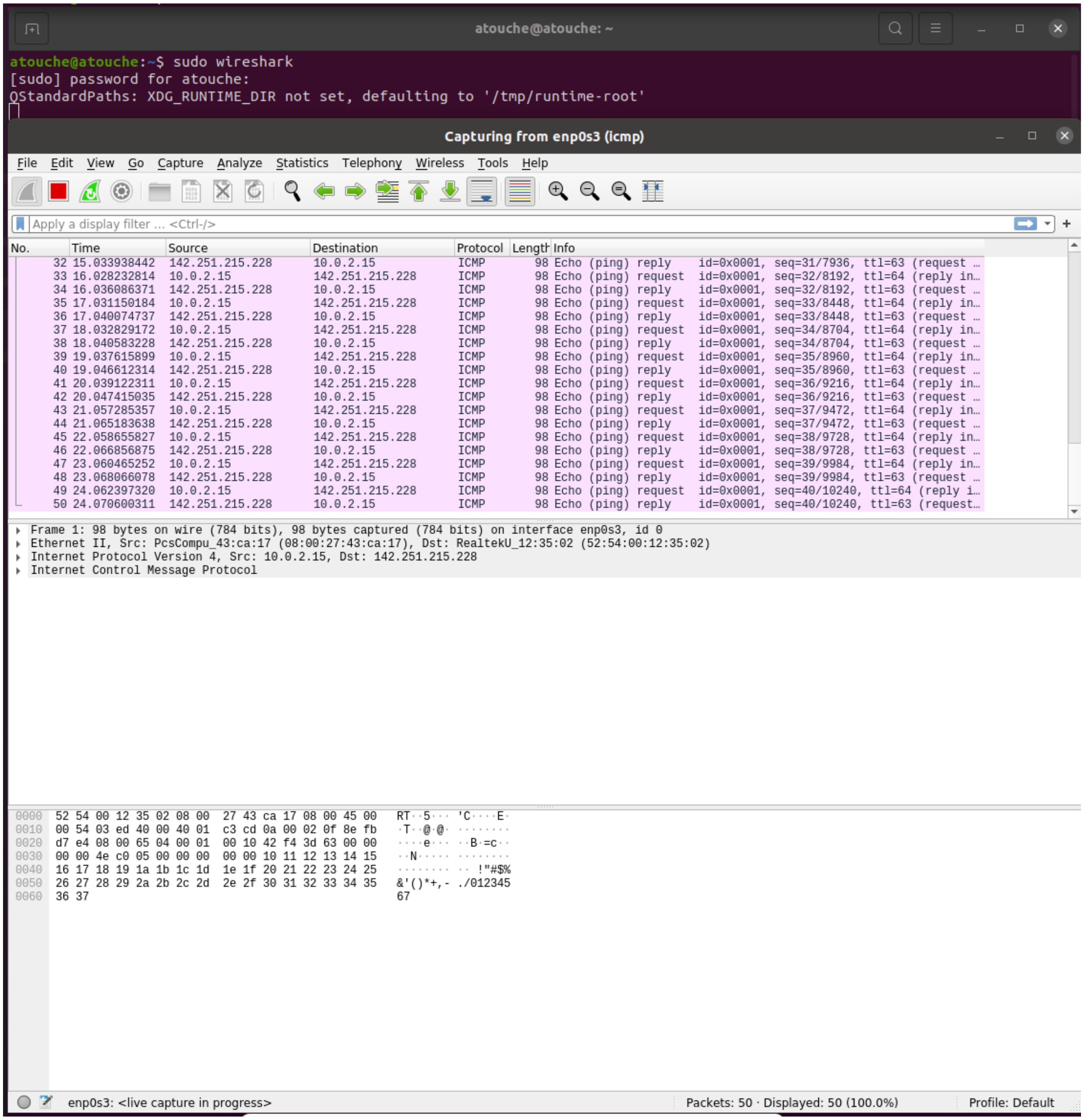
1. defaultMAC: 52:54:00:12:35:02

iv. Wireshark

1. Add `icmp` and ping [www.google.com](http://www.google.com)
2. Answer following questions for `request`:
    a. **Which hardware manufacturer does the destination hardware address of the packet indicate?**

        In the data-link layer (L2) packet or frame, the MAC address for destination is 52:54:00:12:35:02 which is the MAC address of the default gateway (10.0.0.2). L2 frame indicates source device is the VM and destination is the router/default gateway.

b. **Take a screenshot of the bytes in the packet dump window as shown below**



```
atouche@atouche:~$ sudo wireshark
[sudo] password for atouche:
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

Capturing from enp0s3 (icmp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.     Time         Source           Destination      Protocol Length Info
     32 15.033938442 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=31/7936, ttl=63 (request …
     33 16.028232814 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=32/8192, ttl=64 (reply in…
     34 16.036086371 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=32/8192, ttl=63 (request …
     35 17.031150184 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=33/8448, ttl=64 (reply in…
     36 17.040074737 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=33/8448, ttl=63 (request …
     37 18.032829172 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=34/8704, ttl=64 (reply in…
     38 18.040583228 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=34/8704, ttl=63 (request …
     39 19.037615899 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=35/8960, ttl=64 (reply in…
     40 19.046612314 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=35/8960, ttl=63 (request …
     41 20.039122311 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=36/9216, ttl=64 (reply in…
     42 20.047415035 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=36/9216, ttl=63 (request …
     43 21.057285357 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=37/9472, ttl=64 (reply in…
     44 21.065183638 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=37/9472, ttl=63 (request …
     45 22.058655827 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=38/9728, ttl=64 (reply in…
     46 22.066856875 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=38/9728, ttl=63 (request …
     47 23.060465252 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=39/9984, ttl=64 (reply in…
     48 23.068066078 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=39/9984, ttl=63 (request …
     49 24.062397320 10.0.2.15        142.251.215.228  ICMP        98 Echo (ping) request  id=0x0001, seq=40/10240, ttl=64 (reply i…
     50 24.070600311 142.251.215.228  10.0.2.15        ICMP        98 Echo (ping) reply    id=0x0001, seq=40/10240, ttl=63 (request…
```

```
▸ Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: PcsCompu_43:ca:17 (08:00:27:43:ca:17), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▸ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 142.251.215.228
▸ Internet Control Message Protocol
```

```
0000  52 54 00 12 35 02 08 00  27 43 ca 17 08 00 45 00   RT··5··· 'C····E·
0010  00 54 03 ed 40 00 40 01  c3 cd 0a 00 02 0f 8e fb   ·T··@·@· ········
0020  d7 e4 08 00 65 04 00 01  00 10 42 f4 3d 63 00 00   ····e··· ··B·=c··
0030  00 00 4e c0 05 00 00 00  00 00 10 11 12 13 14 15   ··N····· ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········ ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
0060  36 37                                              67
```

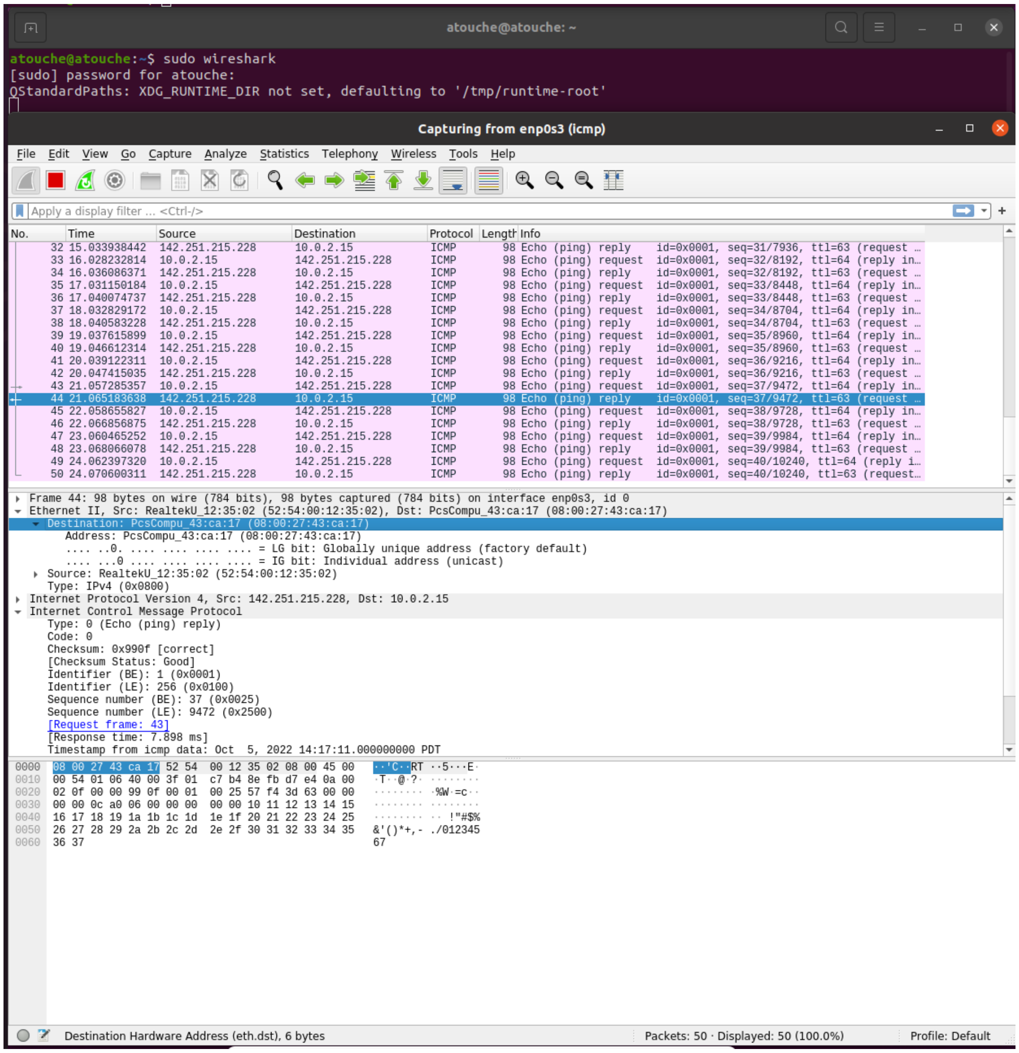
enp0s3: <live capture in progress>
Packets: 50 · Displayed: 50 (100.0%)
Profile: Default

3. Answer following questions for `reply`:

   a. **Which hardware manufacturer does the destination hardware address of the packet indicate?**

      In the data-link layer (L2) packet or frame, the MAC address for destination is 08:00:27:43:ca:17 which is the MAC address of Google's servers (142.251.33.68). L2 frame indicates source device is the default gateway and destination is Google’s server.

   b. **Take a screenshot of the bytes in the packet dump window as shown below**



```
atouche@atouche:~$ sudo wireshark
[sudo] password for atouche:
QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
```

Capturing from enp0s3 (icmp)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.  Time          Source           Destination      Protocol Length Info
  32 15.033938442  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=31/7936, ttl=63 (request …
  33 16.028232814  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=32/8192, ttl=64 (reply in…
  34 16.036086371  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=32/8192, ttl=63 (request …
  35 17.031150184  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=33/8448, ttl=64 (reply in…
  36 17.040074737  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=33/8448, ttl=63 (request …
  37 18.032829172  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=34/8704, ttl=64 (reply in…
  38 18.040583228  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=34/8704, ttl=63 (request …
  39 19.037615899  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=35/8960, ttl=64 (reply in…
  40 19.046612314  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=35/8960, ttl=63 (request …
  41 20.039122311  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=36/9216, ttl=64 (reply in…
  42 20.047415035  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=36/9216, ttl=63 (request …
  43 21.057285357  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=37/9472, ttl=64 (reply in…
  44 21.065183638  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=37/9472, ttl=63 (request …
  45 22.058655827  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=38/9728, ttl=64 (reply in…
  46 22.066856875  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=38/9728, ttl=63 (request …
  47 23.060465252  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=39/9984, ttl=64 (reply in…
  48 23.068066078  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=39/9984, ttl=63 (request …
  49 24.062397320  10.0.2.15        142.251.215.228  ICMP     98 Echo (ping) request  id=0x0001, seq=40/10240, ttl=64 (reply i…
  50 24.070600311  142.251.215.228  10.0.2.15        ICMP     98 Echo (ping) reply    id=0x0001, seq=40/10240, ttl=63 (request…
```

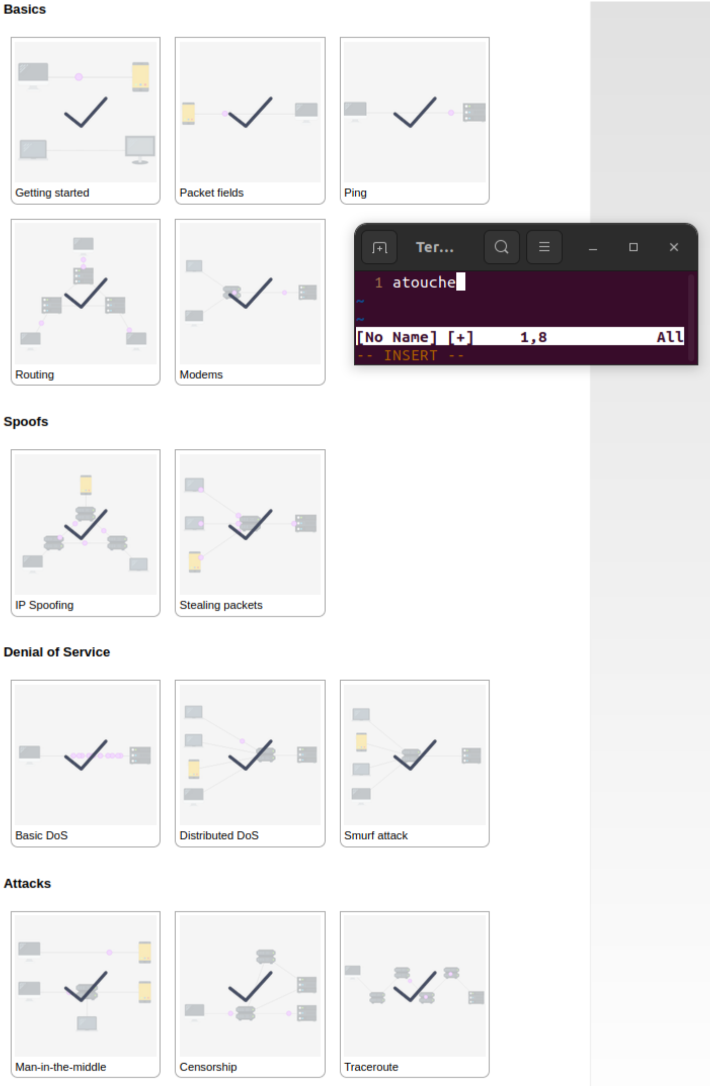
```
▸ Frame 44: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface enp0s3, id 0
▾ Ethernet II, Src: RealtekU_12:35:02 (52:54:00:12:35:02), Dst: PcsCompu_43:ca:17 (08:00:27:43:ca:17)
  ▾ Destination: PcsCompu_43:ca:17 (08:00:27:43:ca:17)
      Address: PcsCompu_43:ca:17 (08:00:27:43:ca:17)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  ▸ Source: RealtekU_12:35:02 (52:54:00:12:35:02)
    Type: IPv4 (0x0800)
▸ Internet Protocol Version 4, Src: 142.251.215.228, Dst: 10.0.2.15
▾ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x990f [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence number (BE): 37 (0x0025)
    Sequence number (LE): 9472 (0x2500)
    [Request frame: 43]
    [Response time: 7.898 ms]
    Timestamp from icmp data: Oct  5, 2022 14:17:11.000000000 PDT
```

```
0000  08 00 27 43 ca 17 52 54  00 12 35 02 08 00 45 00   ··'C··RT ··5···E·
0010  00 54 01 06 40 00 3f 01  c7 b4 8e fb d7 e4 0a 00   ·T··@·?· ········
0020  02 0f 00 00 99 0f 00 01  00 25 57 f4 3d 63 00 00   ········ ·%W·=c··
0030  00 00 0c a0 06 00 00 00  00 00 10 11 12 13 14 15   ········ ········
0040  16 17 18 19 1a 1b 1c 1d  1e 1f 20 21 22 23 24 25   ········ ·· !"#$%
0050  26 27 28 29 2a 2b 2c 2d  2e 2f 30 31 32 33 34 35   &'()*+,- ./012345
0060  36 37                                              67
```

Destination Hardware Address (eth.dst), 6 bytes     Packets: 50 · Displayed: 50 (100.0%)     Profile: Default

2. Netsim #2

□ Complete all levels



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

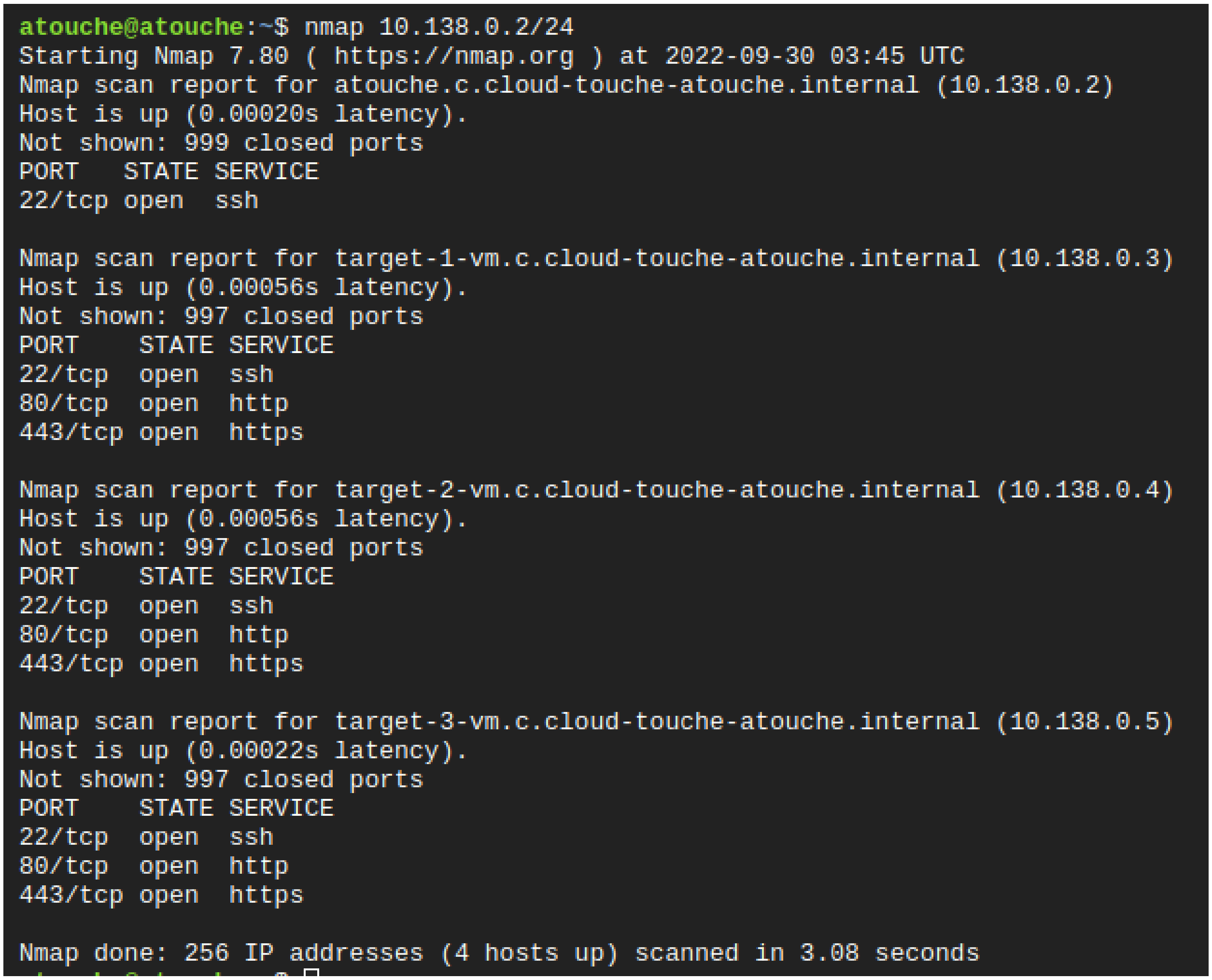
Censorship

Traceroute

```
atouche
~
~
[No Name] [+]       1,8            All
-- INSERT --
```

# 1.3 Cloud Networking

1. Network scanning (`nmap`) #1
   - ☐ Create VM
   - ☐ Install nmap
2. Launch targets
3. Scan targets for services



```
atouche@atouche:~$ nmap 10.138.0.2/24
Starting Nmap 7.80 ( https://nmap.org ) at 2022-09-30 03:45 UTC
Nmap scan report for atouche.c.cloud-touche-atouche.internal (10.138.0.2)
Host is up (0.00020s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap scan report for target-1-vm.c.cloud-touche-atouche.internal (10.138.0.3)
Host is up (0.00056s 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 target-2-vm.c.cloud-touche-atouche.internal (10.138.0.4)
Host is up (0.00056s 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 target-3-vm.c.cloud-touche-atouche.internal (10.138.0.5)
Host is up (0.00022s 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 3.08 seconds
```

4. CIDR and subnets #2

5. Navigating default network

   □ Answer following questions:

      i. **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`)**

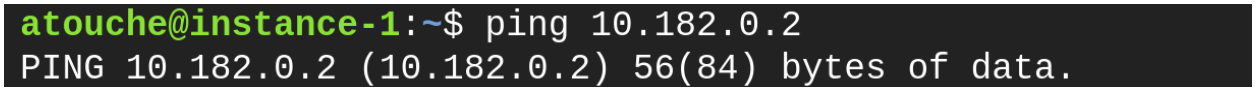
         72 subnetworks are created initially with default.

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

         $2^{(32-20)}$ = 4096 host(s) / subnetwork

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

         In 10.150.0.0/instance-1 and 10.182.0.0/instance-2. Yes



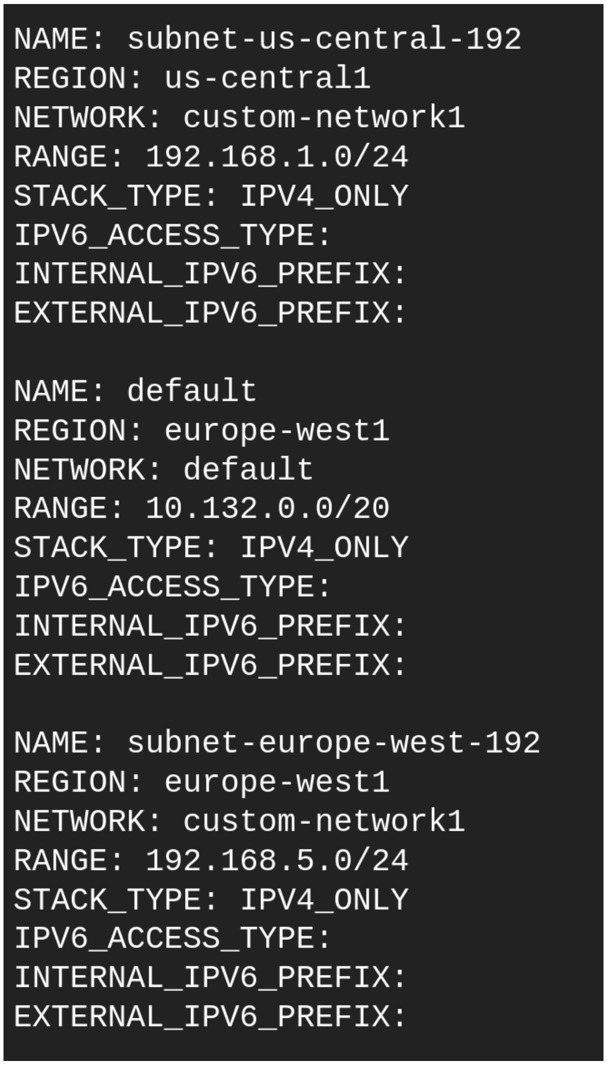
```
atouche@instance-1:~$ ping 10.182.0.2
PING 10.182.0.2 (10.182.0.2) 56(84) bytes of data.
```

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

         Virtual switch

6. Creating custom networks

   □ custom-network1

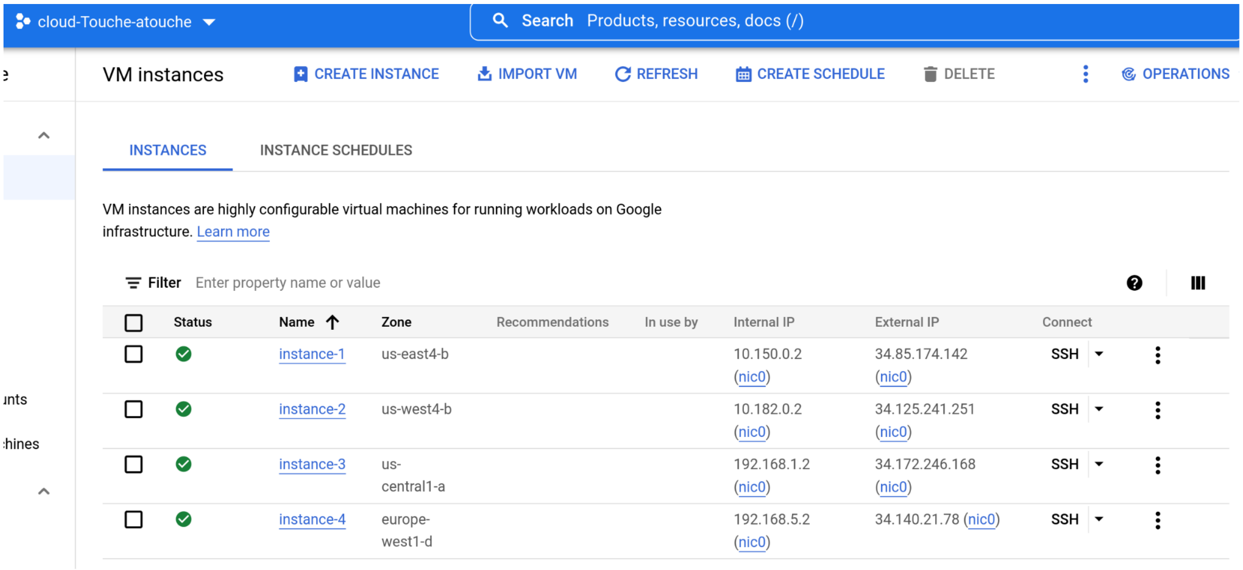


```
NAME: subnet-us-central-192
REGION: us-central1
NETWORK: custom-network1
RANGE: 192.168.1.0/24
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: default
REGION: europe-west1
NETWORK: default
RANGE: 10.132.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

NAME: subnet-europe-west-192
REGION: europe-west1
NETWORK: custom-network1
RANGE: 192.168.5.0/24
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:
```

   □ Ping instance-3&4 from instance-1

      i. Cannot ping outside of region since they are not on the same subnet and do not share the same virtual network switch like instance1&2 does.

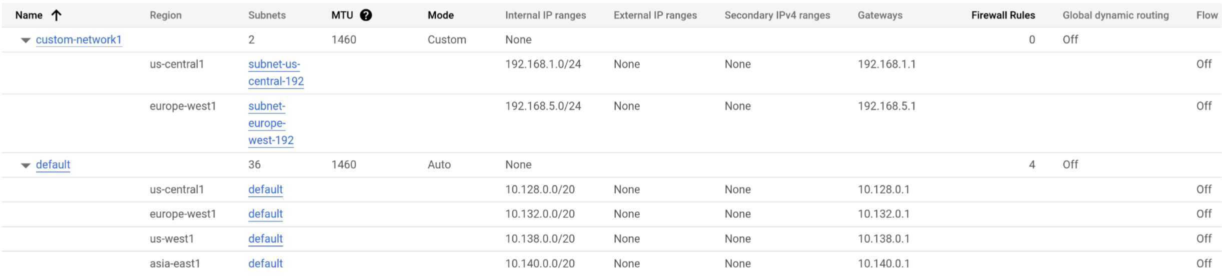
□ Take screenshot of all instances



VM instances are highly configurable virtual machines for running workloads on Google infrastructure. Learn more

| □ | Status | Name ↑ | Zone | Recommendations | In use by | Internal IP | External IP | Connect | |
|---|---|---|---|---|---|---|---|---|---|
| □ | ✓ | instance-1 | us-east4-b | | | 10.150.0.2 (nic0) | 34.85.174.142 (nic0) | SSH ▾ | ⋮ |
| □ | ✓ | instance-2 | us-west4-b | | | 10.182.0.2 (nic0) | 34.125.241.251 (nic0) | SSH ▾ | ⋮ |
| □ | ✓ | instance-3 | us-central1-a | | | 192.168.1.2 (nic0) | 34.172.246.168 (nic0) | SSH ▾ | ⋮ |
| □ | ✓ | instance-4 | europe-west1-d | | | 192.168.5.2 (nic0) | 34.140.21.78 (nic0) | SSH ▾ | ⋮ |

□ Then visit VPC Network



| Name ↑ | Region | Subnets | MTU | Mode | Internal IP ranges | External IP ranges | Secondary IPv4 ranges | Gateways | Firewall Rules | Global dynamic routing | Flow |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ▾ custom-network1 | | 2 | 1460 | Custom | None | | | | 0 | Off | |
| | us-central1 | subnet-us-central-192 | | | 192.168.1.0/24 | None | None | 192.168.1.1 | | | Off |
| | europe-west1 | subnet-europe-west-192 | | | 192.168.5.0/24 | None | None | 192.168.5.1 | | | Off |
| ▾ default | | 36 | 1460 | Auto | None | | | | 4 | Off | |
| | us-central1 | default | | | 10.128.0.0/20 | None | None | 10.128.0.1 | | | Off |
| | europe-west1 | default | | | 10.132.0.0/20 | None | None | 10.132.0.1 | | | Off |
| | us-west1 | default | | | 10.138.0.0/20 | None | None | 10.138.0.1 | | | Off |
| | asia-east1 | default | | | 10.140.0.0/20 | None | None | 10.140.0.1 | | | Off |

7. Clean up