

Odin ID: jtn7

Use the ip address command to find the IPv4 address and hardware address of the local ethernet card interface (Typically beginning with eth, ens, or enp). Include both in your lab notebook.

```
jtn7@ada:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:13:a0:c6 brd ff:ff:ff:ff:ff:ff
    altname enp93s0
    inet 131.252.208.103/24 brd 131.252.208.255 scope global dynamic ens3
        valid_lft 13070sec preferred_lft 13070sec
```

Ethernet

IPv4

Perform a netstat -rn to list the route table for the machine.

What is the default router's IP address (e.g. the gateway address for the default route 0.0.0.0/0)?

```
jtn7@ada:~$ netstat -rn
Kernel IP routing table
Destination        Gateway            Genmask           Flags   MSS Window  irtt Iface
0.0.0.0            131.252.208.1     0.0.0.0           UG        0  0          0 ens3
131.252.208.0      0.0.0.0           255.255.255.0     U        0  0          0 ens3
169.254.0.0        0.0.0.0           255.255.0.0       U        0  0          0 ens3
```

Perform an arp command (both with and without the -n flag) on the IP address of the router.

What is the name of the default router and its hardware address?

```
jtn7@ada:~$ arp 131.252.208.1
Address                  HWtype  HWaddress           Flags Mask            Iface
router.seas.pdx.edu      ether    00:00:5e:00:01:01   C                    ens3
jtn7@ada:~$ arp -n 131.252.208.1
Address                  HWtype  HWaddress           Flags Mask            Iface
131.252.208.1           ether    00:00:5e:00:01:01   C                    ens3
```

We can pipe the output of the command to wc -l to determine the number of entries in the table.

How many entries are there in the ARP table?

33

```
jtn7@ada:~$ arp -a | wc -l
33
```

List any IP addresses share the same hardware address:

131.252.208.20

131.252.208.121

131.252.208.121

131.252.208.121

How many less hardware addresses are there than IP addresses in the ARP table? 2

```
jtn7@ada:~$ arp -a | wc -l
33
jtn7@ada:~$ arp -a | sort -k 4 | awk '{print $4}' | uniq | wc -l
31
jtn7@ada:~$ arp -a | sort -k 4
router.seas.pdx.edu (131.252.208.1) at 00:00:5e:00:01:01 [ether] on ens3
rdns.cat.pdx.edu (131.252.208.53) at 00:00:5e:00:01:35 [ether] on ens3
cs302lab.cs.pdx.edu (131.252.208.83) at 00:00:5e:00:01:53 [ether] on ens3
cs163lab.cs.pdx.edu (131.252.208.84) at 00:00:5e:00:01:54 [ether] on ens3
cs299lab.cs.pdx.edu (131.252.208.86) at 00:00:5e:00:01:56 [ether] on ens3
vhost-therest.cat.pdx.edu (131.252.208.114) at 00:00:5e:00:01:72 [ether] on ens3
gitlab.cecs.pdx.edu (131.252.208.138) at 00:00:5e:00:01:8a [ether] on ens3
glados.cat.pdx.edu (131.252.208.21) at 3c:08:cd:4a:26:a0 [ether] on ens3
linuxlab.cs.pdx.edu (131.252.208.125) at 52:54:00:25:06:08 [ether] on ens3
omr-rdns-01.cat.pdx.edu (131.252.208.118) at 52:54:00:30:e3:f2 [ether] on ens3
quizor5.cs.pdx.edu (131.252.208.55) at 52:54:00:58:b5:8e [ether] on ens3
jammy.cecs.pdx.edu (131.252.208.11) at 52:54:00:59:3e:39 [ether] on ens3
babbage.cs.pdx.edu (131.252.208.23) at 52:54:00:5c:6f:6e [ether] on ens3
mirrors.cat.pdx.edu (131.252.208.20) at 52:54:00:5f:45:5f [ether] on ens3
simirror.cat.pdx.edu (131.252.208.121) at 52:54:00:5f:45:5f [ether] on ens3
focal.cecs.pdx.edu (131.252.208.94) at 52:54:00:78:73:00 [ether] on ens3
tanto.cs.pdx.edu (131.252.208.5) at 52:54:00:87:21:c4 [ether] on ens3
quizor6.cs.pdx.edu (131.252.208.60) at 52:54:00:a3:46:7f [ether] on ens3
dc-rdns-01.cat.pdx.edu (131.252.208.117) at 52:54:00:a9:30:9f [ether] on ens3
quizor4.cs.pdx.edu (131.252.208.36) at 52:54:00:cf:4c:1b [ether] on ens3
rita.cecs.pdx.edu (131.252.208.28) at 52:54:00:eb:9a:42 [ether] on ens3
ruby.cecs.pdx.edu (131.252.208.85) at 52:54:00:f2:09:bc [ether] on ens3
mircle.cat.pdx.edu (131.252.208.54) at 52:54:00:f6:f8:54 [ether] on ens3
cs162lab.cs.pdx.edu (131.252.208.81) at cc:aa:77:06:98:2b [ether] on ens3
quizor2.cs.pdx.edu (131.252.208.172) at cc:aa:77:06:98:2b [ether] on ens3
silverfish.cat.pdx.edu (131.252.208.77) at cc:aa:77:0b:76:be [ether] on ens3
destiny.cat.pdx.edu (131.252.208.17) at cc:aa:77:50:b9:5d [ether] on ens3
expn.cat.pdx.edu (131.252.208.110) at cc:aa:77:5f:de:0e [ether] on ens3
web-therest-lum.cat.pdx.edu (131.252.208.100) at cc:aa:77:8f:61:cb [ether] on ens3
stargate.cat.pdx.edu (131.252.208.43) at cc:aa:77:ed:72:3e [ether] on ens3
mirapo.cat.pdx.edu (131.252.208.63) at cc:aa:77:f1:d3:21 [ether] on ens3
? (169.254.169.254) at e0:89:9d:a8:0a:dd [ether] on ens3
shodan.seas.pdx.edu (131.252.208.3) at f4:cc:55:0c:71:00 [ether] on ens3
```

Use a single command-line to create a file that contains each IP address that appears in the machine's ARP table and places the results in a file called `arp_entries`. Include the command in your lab notebook. What network prefix do most of the IP addresses in the ARP table share? **131.252.208**

```
Terminal
jtn7@ada:~$ arp -an | awk -F '[]' '{print $2}' > arp_entries
jtn7@ada:~$ cat arp_entries
131.252.208.43
131.252.208.5
131.252.208.11
131.252.208.100
169.254.169.254
131.252.208.54
131.252.208.55
131.252.208.17
131.252.208.110
131.252.208.53
131.252.208.114
131.252.208.23
131.252.208.20
131.252.208.21
131.252.208.118
131.252.208.83
131.252.208.63
131.252.208.77
131.252.208.81
131.252.208.138
131.252.208.60
131.252.208.117
131.252.208.86
131.252.208.172
131.252.208.84
131.252.208.28
131.252.208.85
131.252.208.121
131.252.208.3
131.252.208.36
131.252.208.125
131.252.208.1
131.252.208.94
jtn7@ada:~$
```

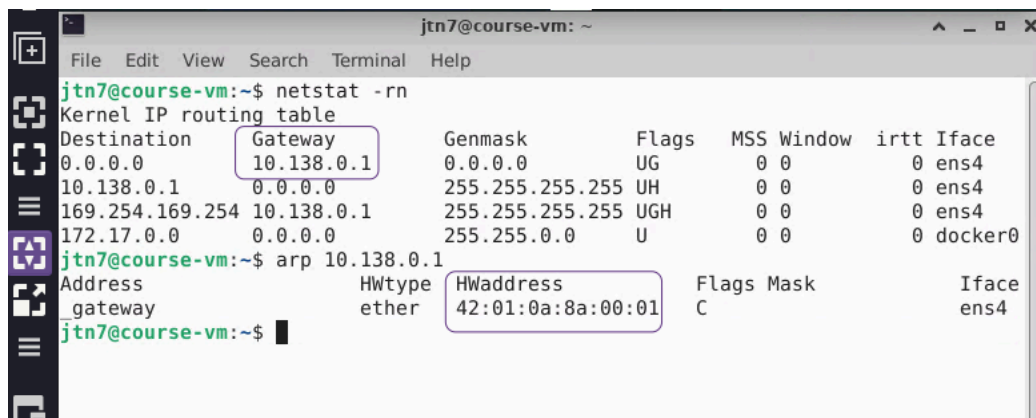
Find the IP address and hardware address of the local ethernet card interface (Typically beginning with `eth`, `ens`, or `enp`). Include both in your lab notebook

```
jtn7@course-vm: ~
File Edit View Search Terminal Help
jtn7@course-vm:~$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1460 qdisc mq state UP group default qlen 1000
    link/ether 42:01:0a:8a:00:02 brd ff:ff:ff:ff:ff:ff
    inet 10.138.0.2/32 metric 100 scope global dynamic ens4
    valid_lft 86172sec preferred_lft 86172sec
    inet6 fe80::4001:aff:fe8a:2/64 scope link
    valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:b1:ba:c7:78 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
    valid_lft forever preferred_lft forever
jtn7@course-vm:~$
```

Ethernet
IPv4
IPv6

What is the default router's IP address (e.g. the gateway address for the default route 0.0.0.0/0)
10.138.0.1

What is the default router's hardware address? 42:01:0a:8a:00:01

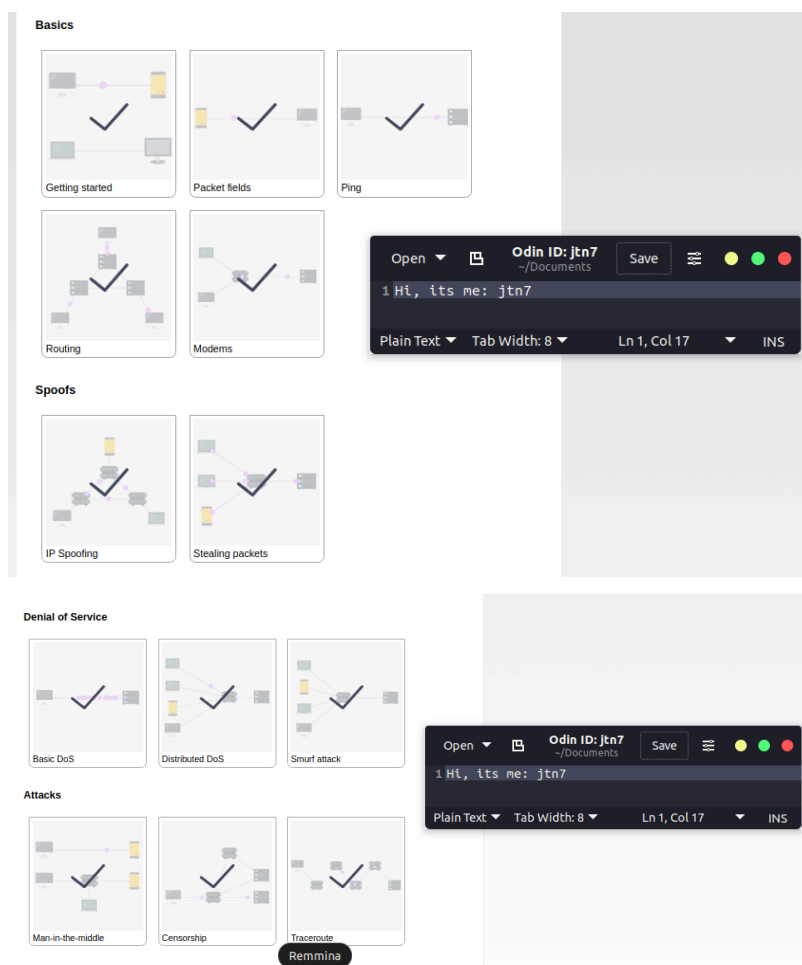


```
jtn7@course-vm:~$ netstat -rn
Kernel IP routing table
Destination        Gateway           Genmask          Flags   MSS Window  irtt Iface
0.0.0.0            10.138.0.1       0.0.0.0          UG        0 0        0 ens4
10.138.0.1         0.0.0.0          255.255.255.255  UH        0 0        0 ens4
169.254.169.254    10.138.0.1       255.255.255.255  UGH       0 0        0 ens4
172.17.0.0         0.0.0.0          255.255.0.0      U        0 0        0 docker0

jtn7@course-vm:~$ arp 10.138.0.1
Address                  HWtype  HWaddress           Flags Mask            Iface
gateway                  ether    42:01:0a:8a:00:01   C                 ens4

jtn7@course-vm:~$
```

Netsim: upon completion of all levels take a screenshot of the completed list of levels



Run nmap on the internal subnet the instances have been placed on. Show a screenshot of the output for the scan for your lab notebook.

```
jtn7@course-vm:~$ nmap 10.138.0.3
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-13 19:25 UTC
Nmap scan report for tikiwiki-20-07-2020-1-vm.c.cloud-nguyen-jtn7.internal (10.138.0.3)
Host is up (0.00022s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
jtn7@course-vm:~$ nmap 10.138.0.4
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-13 19:25 UTC
Nmap scan report for orocommerce-1-vm.c.cloud-nguyen-jtn7.internal (10.138.0.4)
Host is up (0.00027s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
jtn7@course-vm:~$ nmap 10.138.0.5
Starting Nmap 7.80 ( https://nmap.org ) at 2024-01-13 19:25 UTC
Nmap scan report for joomla-1-vm.c.cloud-nguyen-jtn7.internal (10.138.0.5)
Host is up (0.00031s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
jtn7@course-vm:~$
```

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`)

84 subnetworks were created and this corresponds to 84 regions.

Given the CIDR prefix associated with each subnetwork, how many hosts does each subnetwork support? **20**

Create two instances in different zones in separate regions of your choice. List both instances.

```
jtn7@cloudshell:~ (cloud-nguyen-jtn7)$ gcloud compute instances list
NAME: instance-2
ZONE: us-central1-a
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.128.0.2
EXTERNAL_IP: 34.171.224.44
STATUS: RUNNING

NAME: instance-1
ZONE: us-west1-a
MACHINE_TYPE: n1-standard-1
PREEMPTIBLE:
INTERNAL_IP: 10.138.0.6
EXTERNAL_IP: 34.168.230.196
STATUS: RUNNING

NAME: course-vm
ZONE: us-west1-b
MACHINE_TYPE: e2-medium
PREEMPTIBLE:
INTERNAL_IP: 10.138.0.2
EXTERNAL_IP: 34.145.38.62
STATUS: RUNNING
```

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

The /20 prefix CIDR and yes, they correspond to the appropriate region.

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

```
jtn7@instance-1:~$ ping 10.128.0.2
PING 10.128.0.2 (10.128.0.2) 56(84) bytes of data.
64 bytes from 10.128.0.2: icmp_seq=1 ttl=64 time=39.1 ms
64 bytes from 10.128.0.2: icmp_seq=2 ttl=64 time=38.7 ms
64 bytes from 10.128.0.2: icmp_seq=3 ttl=64 time=38.6 ms
64 bytes from 10.128.0.2: icmp_seq=4 ttl=64 time=38.6 ms
64 bytes from 10.128.0.2: icmp_seq=5 ttl=64 time=38.6 ms
64 bytes from 10.128.0.2: icmp_seq=6 ttl=64 time=38.6 ms
64 bytes from 10.128.0.2: icmp_seq=7 ttl=64 time=38.3 ms
64 bytes from 10.128.0.2: icmp_seq=8 ttl=64 time=35.0 ms
64 bytes from 10.128.0.2: icmp_seq=9 ttl=64 time=35.1 ms
64 bytes from 10.128.0.2: icmp_seq=10 ttl=64 time=35.0 ms
64 bytes from 10.128.0.2: icmp_seq=11 ttl=64 time=35.0 ms
64 bytes from 10.128.0.2: icmp_seq=12 ttl=64 time=35.0 ms
^C
--- 10.128.0.2 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11016ms
rtt min/avg/max/mdev = 34.959/37.139/39.091/1.797 ms
```

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

The VPN gateway because instance 1 and instance 2 are not in the same region, so the VPN is used to route traffic between GCP infrastructure (see highlighted below).

The infrastructure that is deployed to implement this is shown in red. Because these subnetworks were initially private, virtual switches that handle traffic within GCP infrastructure must be used to encrypt traffic between the 3 subnetworks. The figure also shows VPN gateways that must be used to encrypt and route traffic between GCP infrastructure and external destinations such as the customer site. Note that the CIDR prefixes for each subnetwork employ private IP address ranges that are not reachable externally (e.g. 10.240.0.0/24, 192.168.1.0/24, and 10.2.0.0/16).

Explain why the result of this ping is different from when you performed the ping to instance-2.

Because instance-3 and instance-4 are on the custom network we created, custom-network-1 with the 24 CIDR prefix while instance-1 is on the default network with the 20 CIDR prefix, each subnetwork employ private IP address ranges that are not reachable externally. Thus instance-1 cannot reach instance-3 and instance-4 and the packets are dropped. (See highlighted text above).


```

jtn7@cloudshell:~ (cloud-nguyen-jtn7)$ gcloud compute networks subnets list --regions=us-central1,europe-west1
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:

NAME: default
REGION: us-central1
NETWORK: default
RANGE: 10.128.0.0/20
STACK_TYPE: IPV4_ONLY
IPV6_ACCESS_TYPE:
INTERNAL_IPV6_PREFIX:
EXTERNAL_IPV6_PREFIX:

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:

```

<input type="checkbox"/>		instance-1	us-west1-a	10.138.0.6 (nic0)	34.168.230.196 (nic0)	default	SSH	⌵	⋮
<input type="checkbox"/>		instance-2	us-central1-a	10.128.0.2 (nic0)	34.171.224.44 (nic0)	default	SSH	⌵	⋮
<input type="checkbox"/>		instance-3	us-central1-a	192.168.1.2 (nic0)	34.173.104.3 (nic0)	custom-network1	SSH	⌵	⋮
<input type="checkbox"/>		instance-4	europe-west1-d	192.168.5.2 (nic0)	34.34.183.41 (nic0)	custom-network1	SSH	⌵	⋮

custom-network1													
OVERVIEW	SUBNETS	STATIC INTERNAL IP ADDRESSES	FIREWALLS	FIREWALL ENDPOINTS	ROUTES	VPC NETWORK PEERING	PRIVATE SERVICES AI						
Subnets													
<div> <div>Filter</div> <div>Enter property name or value</div> <div> <div></div> <div></div> </div> </div>													
<input type="checkbox"/>	Name ↑	Region	Stack Type	Internal IP ranges	External IP ranges	Secondary IPv4 ranges ↑	Gateway	Private Google Access	Flow logs				
<input type="checkbox"/>	subnet-europe-west-192	europe-west1	IPv4	192.168.5.0/24	None	None	192.168.5.1	Off	Off				
<input type="checkbox"/>	subnet-us-central-192	us-central1	IPv4	192.168.1.0/24	None	None	192.168.1.1	Off	Off				
default													
OVERVIEW	SUBNETS	STATIC INTERNAL IP ADDRESSES	FIREWALLS	FIREWALL ENDPOINTS	ROUTES	VPC NETWORK PEERING	PRIVATE SERVICES						
Subnets													
<div> <div>Filter</div> <div>Enter property name or value</div> <div> <div></div> <div></div> </div> </div>													
<input type="checkbox"/>	Name ↑	Region	Stack Type	Internal IP ranges	External IP ranges	Secondary IPv4 ranges	Gateway	Private Google Access	Flow logs				
<input type="checkbox"/>	default	us-central1	IPv4	10.128.0.0/20	None	None	10.128.0.1	Off	Off				
<input type="checkbox"/>	default	europe-west1	IPv4	10.132.0.0/20	None	None	10.132.0.1	Off	Off				
<input type="checkbox"/>	default	us-west1	IPv4	10.138.0.0/20	None	None	10.138.0.1	Off	Off				
<input type="checkbox"/>	default	asia-east1	IPv4	10.140.0.0/20	None	None	10.140.0.1	Off	Off				
<input type="checkbox"/>	default	us-east1	IPv4	10.142.0.0/20	None	None	10.142.0.1	Off	Off				
<input type="checkbox"/>	default	asia-northeast1	IPv4	10.146.0.0/20	None	None	10.146.0.1	Off	Off				