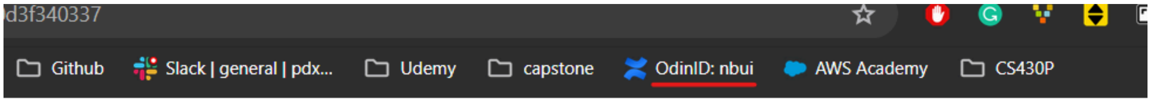
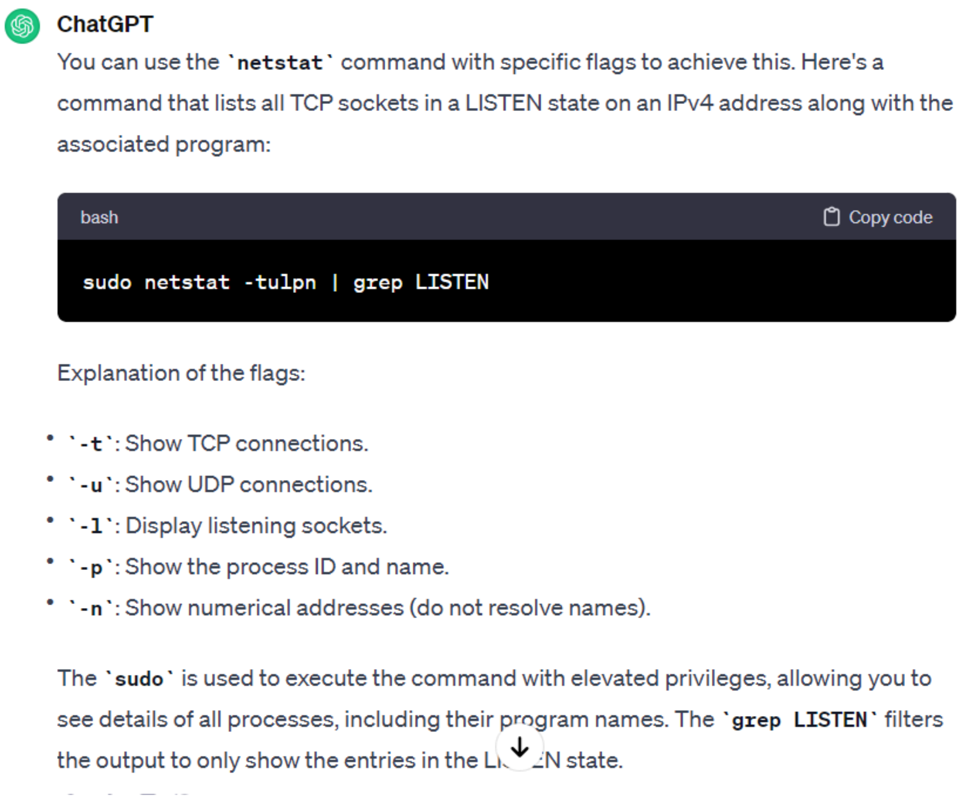
Lab 02.1 **2**
1. TCP #1: Sockets 2
2.- lsof 5
3. TCP #2: Throughput 6
4. - iperf 6
5. HTTP #3: Requests 7
6. Asynchronous HTTP requests 11
Lab 02.2 **12**
1. DNS reconnaissance #1 (dig) 12
2. Iterative DNS lookups 15
3. Reverse DNS lookups 16
4. Host enumeration 17
5. Geographic DNS #2 18
7. 20
10. Analyze trace 21
ARP 22
DNS 23
TCP 23
HTTP 24

Lab 02.1

1. TCP #1: Sockets

- **Take a screenshot of the prompt and the command that ChatGPT generates**
- **Run the command using `sudo` and take a screenshot of the output to include in your lab notebook.**





ChatGPT

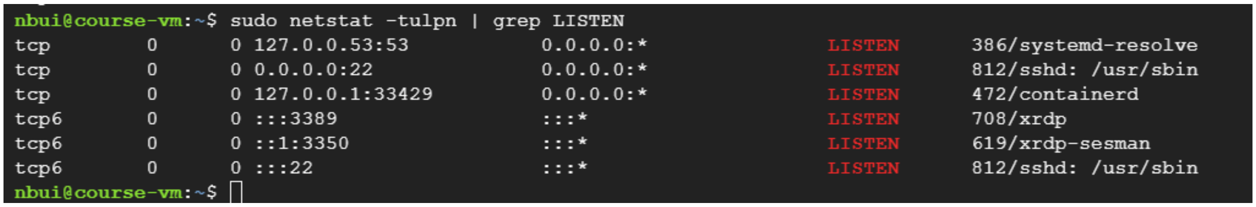
You can use the `netstat` command with specific flags to achieve this. Here's a command that lists all TCP sockets in a LISTEN state on an IPv4 address along with the associated program:

```bash
sudo netstat -tulpn | grep LISTEN
```

Explanation of the flags:

- `-t`: Show TCP connections.
- `-u`: Show UDP connections.
- `-l`: Display listening sockets.
- `-p`: Show the process ID and name.
- `-n`: Show numerical addresses (do not resolve names).

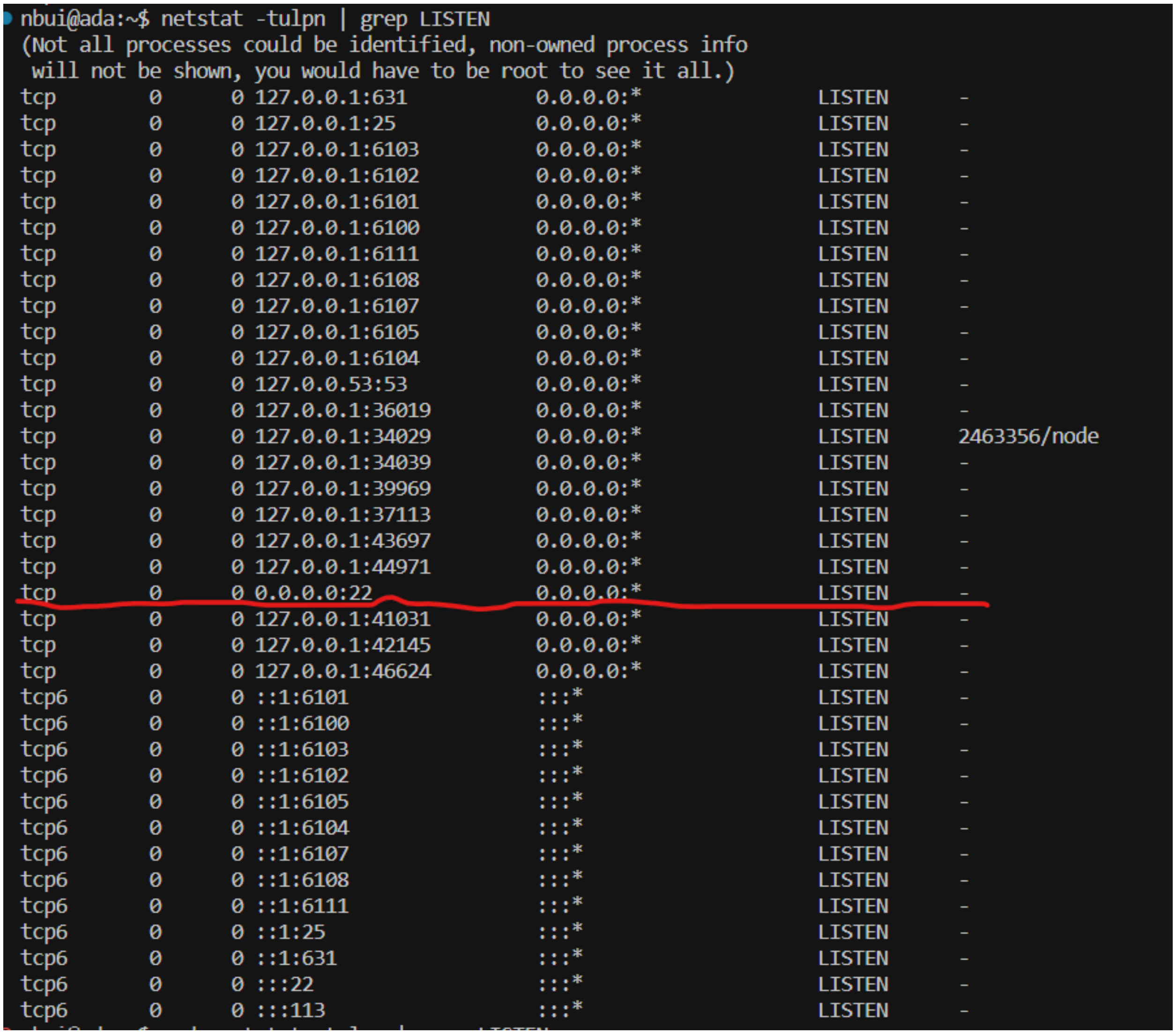
The `sudo` is used to execute the command with elevated privileges, allowing you to see details of all processes, including their program names. The `grep LISTEN` filters the output to only show the entries in the L EN state.



```
nbui@course-vm:~$ sudo netstat -tulpn | grep LISTEN
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      386/systemd-resolve
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      812/sshd: /usr/sbin
tcp        0      0 127.0.0.1:33429         0.0.0.0:*               LISTEN      472/containerd
tcp6       0      0 :::3389                 :::*                    LISTEN      708/xrdp
tcp6       0      0 ::1:3350                :::*                    LISTEN      619/xrdp-sesman
tcp6       0      0 :::22                   :::*                    LISTEN      812/sshd: /usr/sbin
nbui@course-vm:~$ 
```

- **List a service that can be contacted from any interface on the machine. List a service that can only be contacted by local processes.**

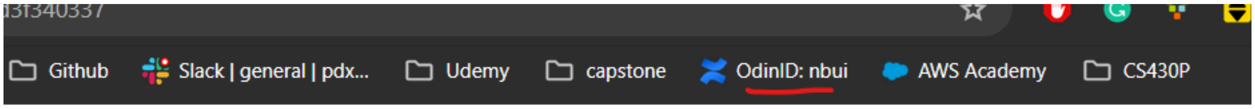
    - **List a service that can be contacted from any interface on the machine:** `sshd: /usr/sbin`
    - **List a service that can only be contacted by local processes:** systemd-resolve, containerd, xrdp-sesman
- **Run the command again, but do not use sudo as this is a machine managed by CAT. Include a screenshot of the output.**
- **List the services that this machine provides for external access**

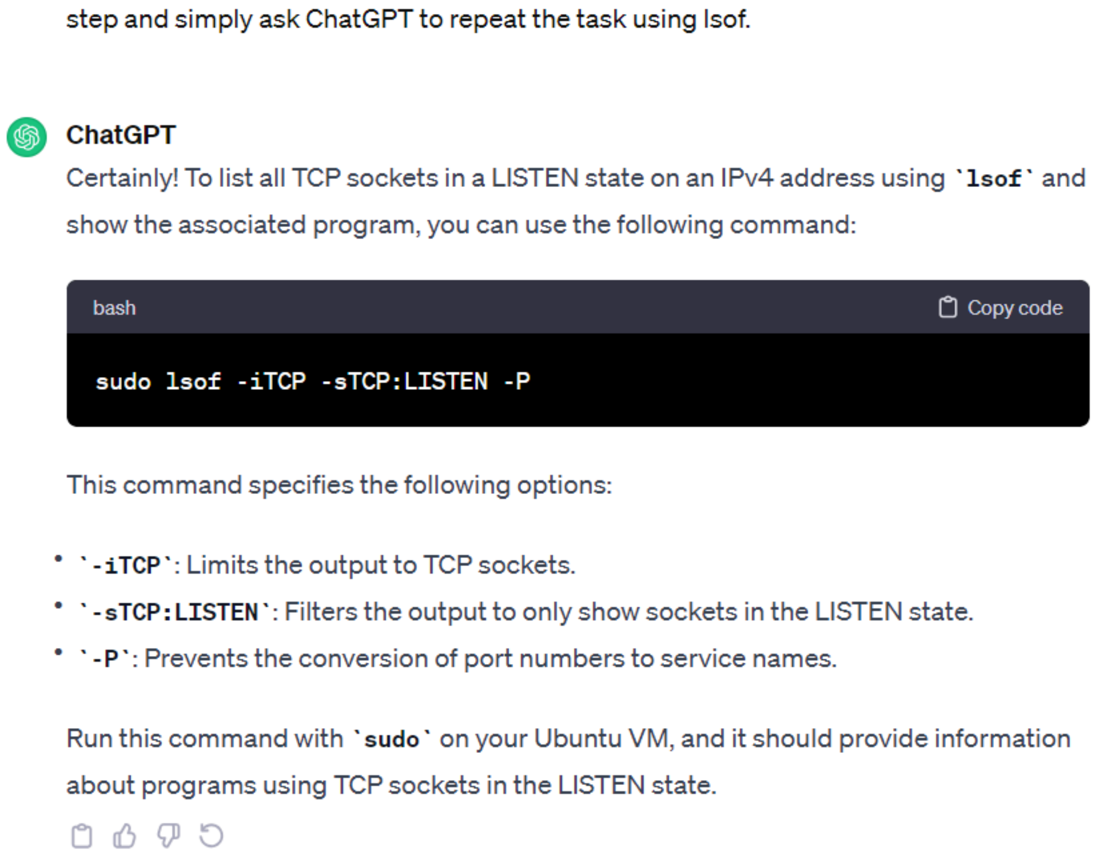


```
nbui@ada:~$ netstat -tulpn | grep LISTEN
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6103          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6102          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6101          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6100          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6111          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6108          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6107          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6105          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6104          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:36019         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:34029         0.0.0.0:*               LISTEN      2463356/node
tcp        0      0 127.0.0.1:34039         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:39969         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:37113         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:43697         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:44971         0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:41031         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:42145         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:46624         0.0.0.0:*               LISTEN      -
tcp6       0      0 ::1:6101                :::*                    LISTEN      -
tcp6       0      0 ::1:6100                :::*                    LISTEN      -
tcp6       0      0 ::1:6103                :::*                    LISTEN      -
tcp6       0      0 ::1:6102                :::*                    LISTEN      -
tcp6       0      0 ::1:6105                :::*                    LISTEN      -
tcp6       0      0 ::1:6104                :::*                    LISTEN      -
tcp6       0      0 ::1:6107                :::*                    LISTEN      -
tcp6       0      0 ::1:6108                :::*                    LISTEN      -
tcp6       0      0 ::1:6111                :::*                    LISTEN      -
tcp6       0      0 ::1:25                  :::*                    LISTEN      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 :::113                  :::*                    LISTEN      -
```

2.- lsof

- **Take a screenshot of the prompt and the command that ChatGPT generates**
- **Run the command using sudo and take a screenshot of the output to include in your lab notebook.**




step and simply ask ChatGPT to repeat the task using lsof.

ChatGPT

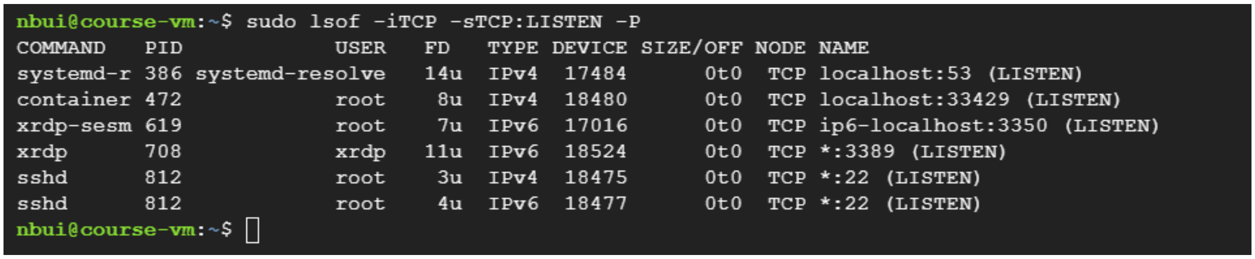
Certainly! To list all TCP sockets in a LISTEN state on an IPv4 address using `lsof` and show the associated program, you can use the following command:

```bash
sudo lsof -iTCP -sTCP:LISTEN -P
```

This command specifies the following options:

- `-iTCP`: Limits the output to TCP sockets.
- `-sTCP:LISTEN`: Filters the output to only show sockets in the LISTEN state.
- `-P`: Prevents the conversion of port numbers to service names.

Run this command with `sudo` on your Ubuntu VM, and it should provide information about programs using TCP sockets in the LISTEN state.

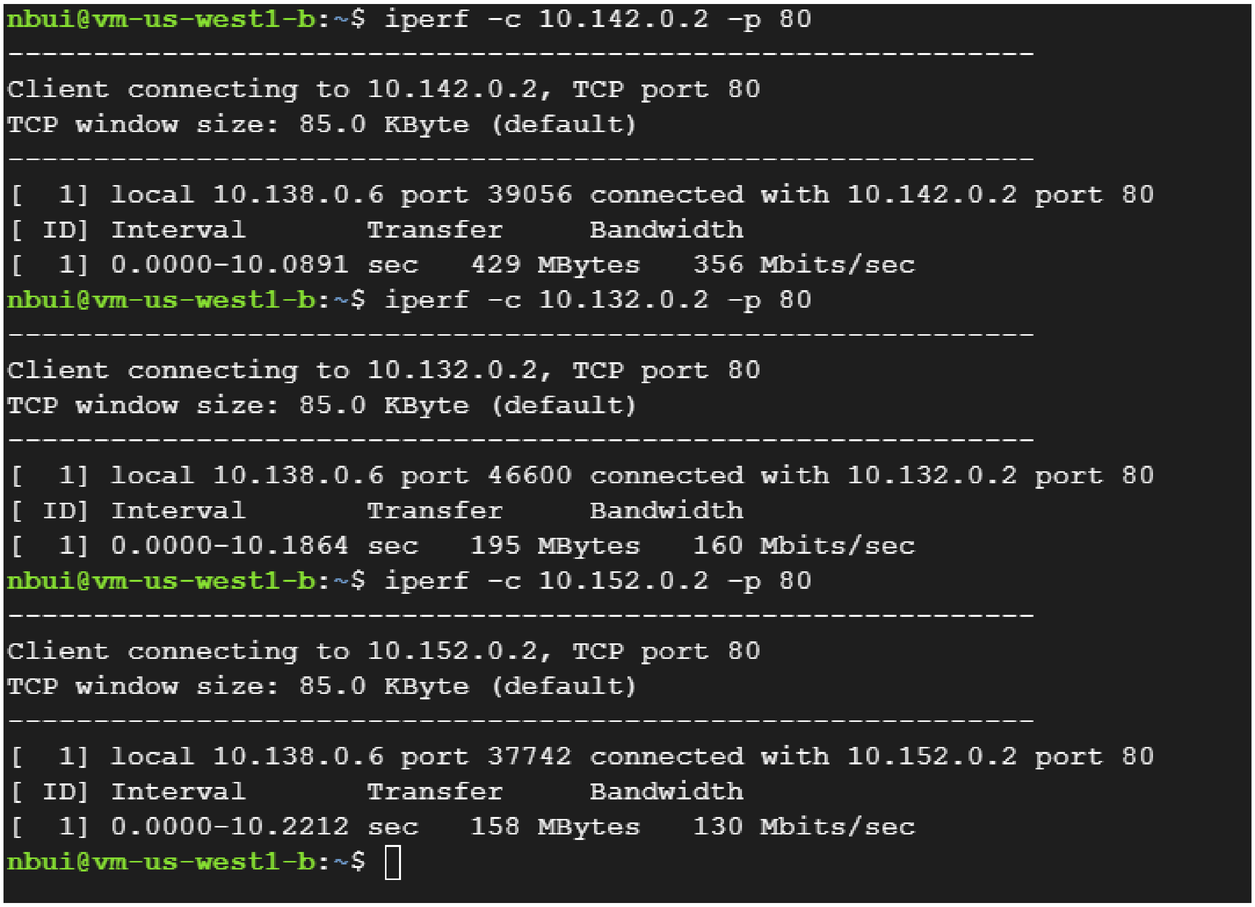


```
nbui@course-vm:~$ sudo lsof -iTCP -sTCP:LISTEN -P
COMMAND   PID            USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
systemd-r 386 systemd-resolve   14u  IPv4  17484      0t0  TCP localhost:53 (LISTEN)
container 472            root    8u  IPv4  18480      0t0  TCP localhost:33429 (LISTEN)
xrdp-sesm 619            root    7u  IPv6  17016      0t0  TCP ip6-localhost:3350 (LISTEN)
xrdp      708            xrdp   11u  IPv6  18524      0t0  TCP *:3389 (LISTEN)
sshd      812            root    3u  IPv4  18475      0t0  TCP *:22 (LISTEN)
sshd      812            root    4u  IPv6  18477      0t0  TCP *:22 (LISTEN)
nbui@course-vm:~$ 
```

3. TCP #2: Throughput

4. - iperf

- **Show a screenshot of the measured bandwidth available between your us-west1-b VM and each of the other Compute Engine VMs.**

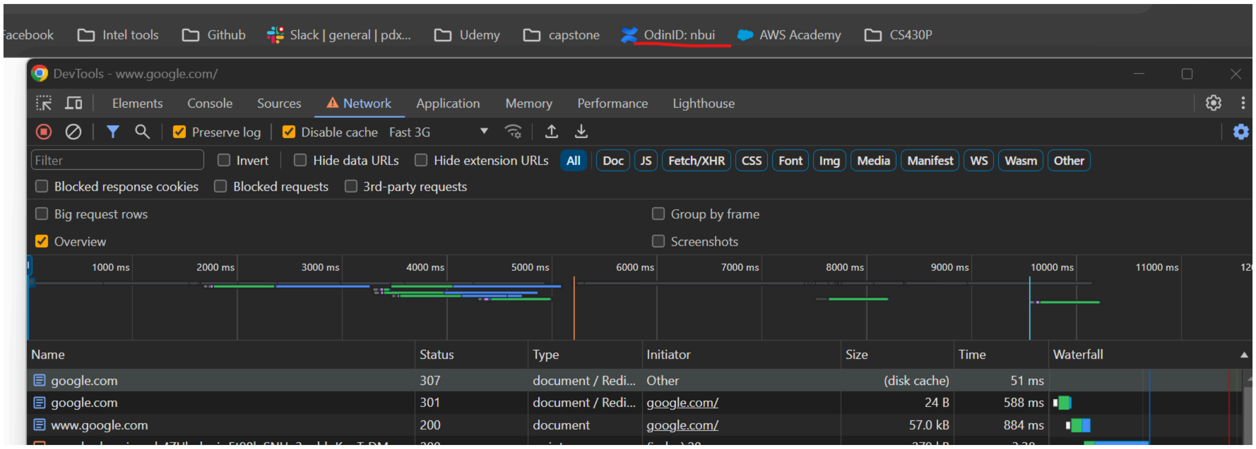


```
nbui@vm-us-west1-b:~$ iperf -c 10.142.0.2 -p 80
------------------------------------------------------------
Client connecting to 10.142.0.2, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.6 port 39056 connected with 10.142.0.2 port 80
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.0891 sec   429 MBytes   356 Mbits/sec
nbui@vm-us-west1-b:~$ iperf -c 10.132.0.2 -p 80
------------------------------------------------------------
Client connecting to 10.132.0.2, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.6 port 46600 connected with 10.132.0.2 port 80
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.1864 sec   195 MBytes   160 Mbits/sec
nbui@vm-us-west1-b:~$ iperf -c 10.152.0.2 -p 80
------------------------------------------------------------
Client connecting to 10.152.0.2, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.6 port 37742 connected with 10.152.0.2 port 80
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.2212 sec   158 MBytes   130 Mbits/sec
nbui@vm-us-west1-b:~$ 
```

- **Explain the relative differences (or lack thereof) in your results**: the bandwidth and transfer difference in the screenshots are affected by the geographic location of those instances. Between the US (US West to US-East 10.142.0.2) the bandwidth and transfer are higher with low latency. The further geographic between instances, the lower bandwidth transmission and higher latency.

5. HTTP #3: Requests

- Take a screenshot of the initial 3 requests that the browser makes for your lab notebook.

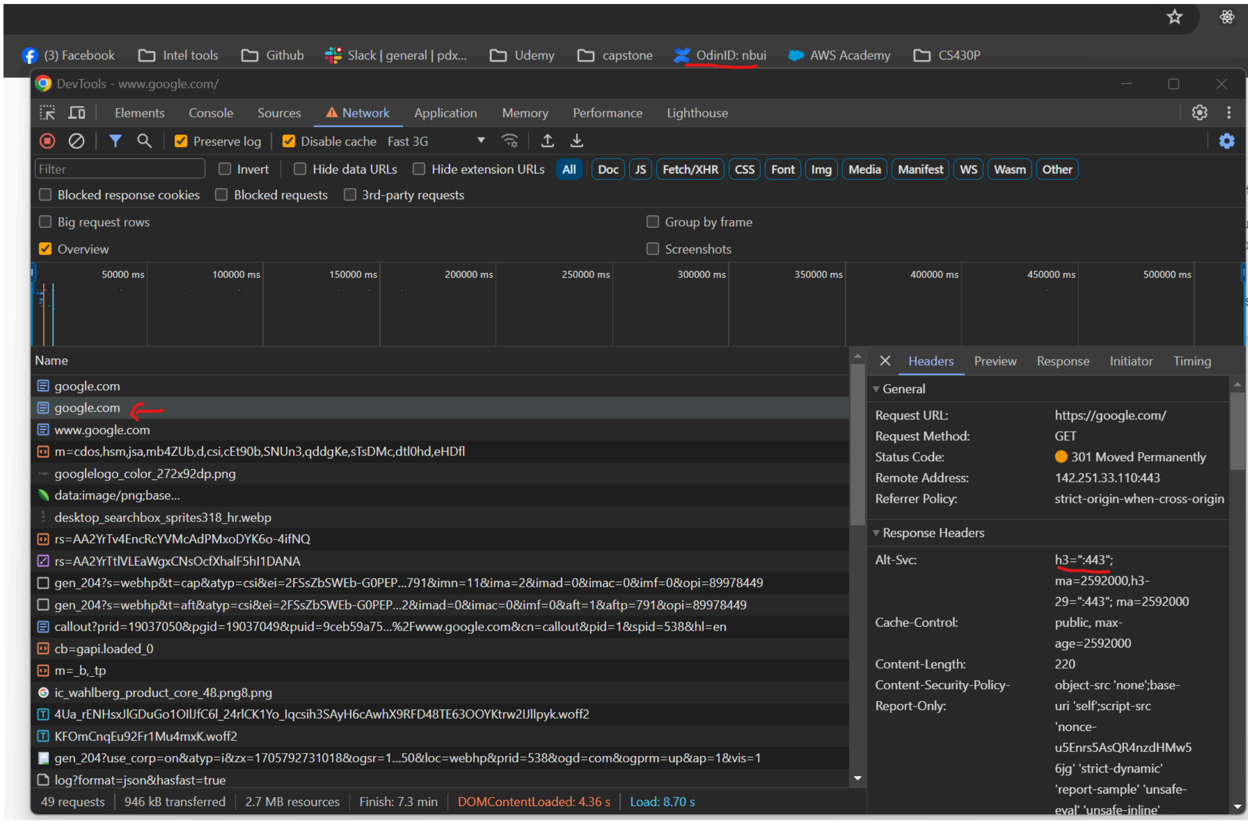


☑ Preserve log
☑ Disable cache
☐ Invert
☐ Hide data URLs
☐ Hide extension URLs
☐ Blocked response cookies
☐ Blocked requests
☐ 3rd-party requests
☐ Big request rows
☐ Group by frame
☑ Overview
☐ Screenshots

Name	Status	Type	Initiator	Size	Time
google.com	307	document / Redi...	Other	(disk cache)	51 ms
google.com	301	document / Redi...	google.com/	24 B	588 ms
www.google.com	200	document	google.com/	57.0 kB	884 ms

- **What is the URL being requested?**
  - http://google.com/
  - https://google.com/
  - https://www.google.com/
- **Explain the HTTP status code that is returned and what the code indicates**
  - 307: Internal Redirect (from disk cache): it is sent when a document is temporarily available at a different URL
  - 301: permanent redirect (the destination URL will never change)
  - 200: request succeeded
- **Take a screenshot indicating the version of the HTTP protocol that is used for each request. (Hint: look at the response status line and `alt-svc:` HTTP response headers indicating HTTP/2 or HTTP/3).**

- ○



☑ Preserve log ☑ Disable cache Fast 3G

☐ Invert ☐ Hide data URLs ☐ Hide extension URLs

☐ Blocked response cookies ☐ Blocked requests ☐ 3rd-party requests

☐ Big request rows ☐ Group by frame

☑ Overview ☐ Screenshots

```
google.com
google.com
www.google.com
m=cdos,hsm,jsa,mb4ZUb,d,csi,cEt90b,SNUn3,qddgKe,sTsDMc,dtl0hd,eHDfl
googlelogo_color_272x92dp.png
data:image/png;base...
desktop_searchbox_sprites318_hr.webp
rs=AA2YrTv4EncRcYVMcAdPMxoDYK6o-4ifNQ
rs=AA2YrTtlVLEaWgxCNsOcfXhalF5hl1DANA
gen_204?s=webhp&t=cap&atyp=csi&ei=2FSsZbSWEb-G0PEP...791&imn=11&ima=2&imad=0&imac=0&imf=0&opi=89978449
gen_204?s=webhp&t=aft&atyp=csi&ei=2FSsZbSWEb-G0PEP...2&imad=0&imac=0&imf=0&aft=1&aftp=791&opi=89978449
callout?prid=19037050&pgid=19037049&puid=9ceb59a75...%2Fwww.google.com&cn=callout&pid=1&spid=538&hl=en
cb=gapi.loaded_0
m=_b,_tp
ic_wahlberg_product_core_48.png8.png
4Ua_rENHsxJlGDuGo1OIlJfC6l_24rlCK1Yo_lqcsih3SAyH6cAwhX9RFD48TE63OOYKtrw2IJllpyk.woff2
KFOmCnqEu92Fr1Mu4mxK.woff2
gen_204?use_corp=on&atyp=i&zx=1705792731018&ogsr=1...50&loc=webhp&prid=538&ogd=com&ogprm=up&ap=1&vis=1
log?format=json&hasfast=true
```

49 requests | 946 kB transferred | 2.7 MB resources | Finish: 7.3 min | DOMContentLoaded: 4.36 s | Load: 8.70 s

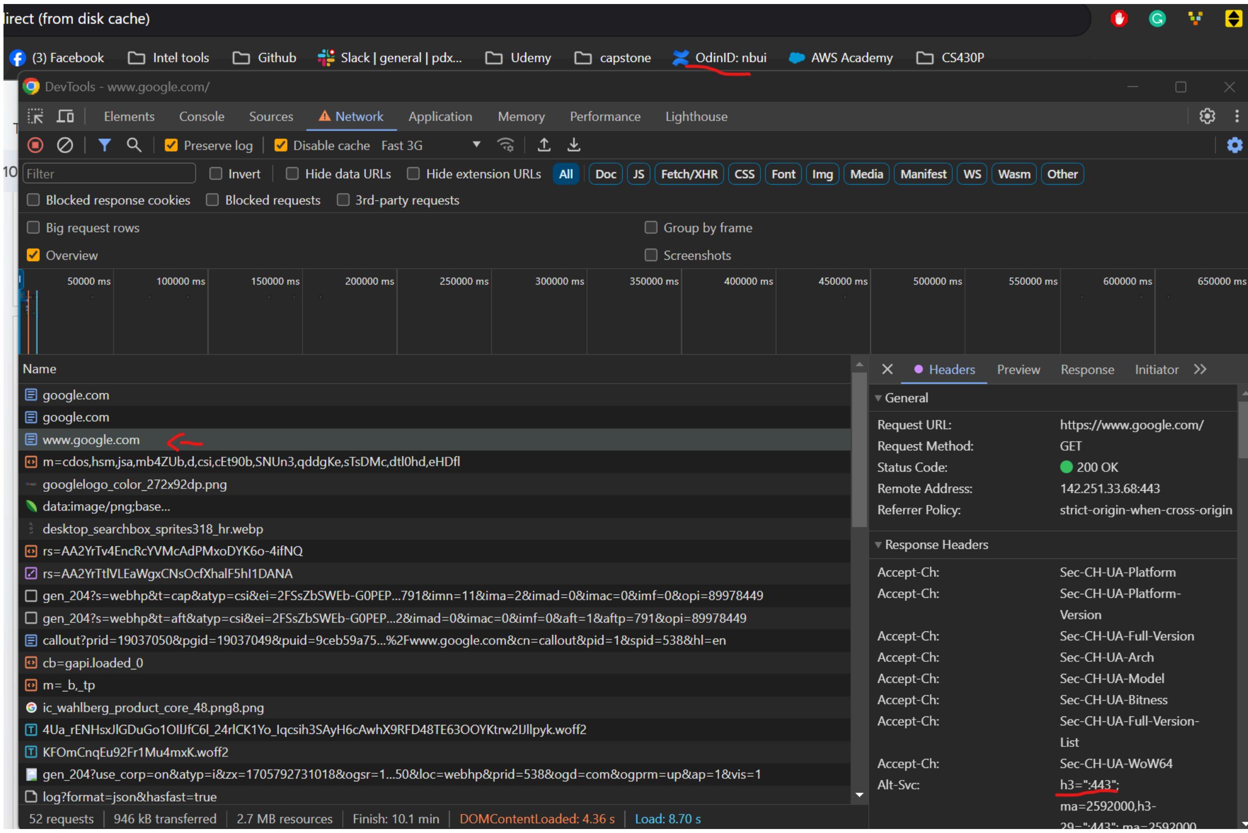
General

Request URL: https://google.com/
Request Method: GET
Status Code: ● 301 Moved Permanently
Remote Address: 142.251.33.110:443
Referrer Policy: strict-origin-when-cross-origin

Response Headers

Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
Cache-Control: public, max-age=2592000
Content-Length: 220
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-u5Enrs5AsQR4nzdHMw56jg' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline'

- ○



☑ Preserve log ☑ Disable cache Fast 3G

☐ Invert ☐ Hide data URLs ☐ Hide extension URLs

☐ Blocked response cookies ☐ Blocked requests ☐ 3rd-party requests

☐ Big request rows ☐ Group by frame

☑ Overview ☐ Screenshots

```
google.com
google.com
www.google.com
m=cdos,hsm,jsa,mb4ZUb,d,csi,cEt90b,SNUn3,qddgKe,sTsDMc,dtl0hd,eHDfl
googlelogo_color_272x92dp.png
data:image/png;base...
desktop_searchbox_sprites318_hr.webp
rs=AA2YrTv4EncRcYVMcAdPMxoDYK6o-4ifNQ
rs=AA2YrTtlVLEaWgxCNsOcfXhalF5hl1DANA
gen_204?s=webhp&t=cap&atyp=csi&ei=2FSsZbSWEb-G0PEP...791&imn=11&ima=2&imad=0&imac=0&imf=0&opi=89978449
gen_204?s=webhp&t=aft&atyp=csi&ei=2FSsZbSWEb-G0PEP...2&imad=0&imac=0&imf=0&aft=1&aftp=791&opi=89978449
callout?prid=19037050&pgid=19037049&puid=9ceb59a75...%2Fwww.google.com&cn=callout&pid=1&spid=538&hl=en
cb=gapi.loaded_0
m=_b,_tp
ic_wahlberg_product_core_48.png8.png
4Ua_rENHsxJlGDuGo1OIlJfC6l_24rlCK1Yo_lqcsih3SAyH6cAwhX9RFD48TE63OOYKtrw2IJllpyk.woff2
KFOmCnqEu92Fr1Mu4mxK.woff2
gen_204?use_corp=on&atyp=i&zx=1705792731018&ogsr=1...50&loc=webhp&prid=538&ogd=com&ogprm=up&ap=1&vis=1
log?format=json&hasfast=true
```

52 requests | 946 kB transferred | 2.7 MB resources | Finish: 10.1 min | DOMContentLoaded: 4.36 s | Load: 8.70 s

General

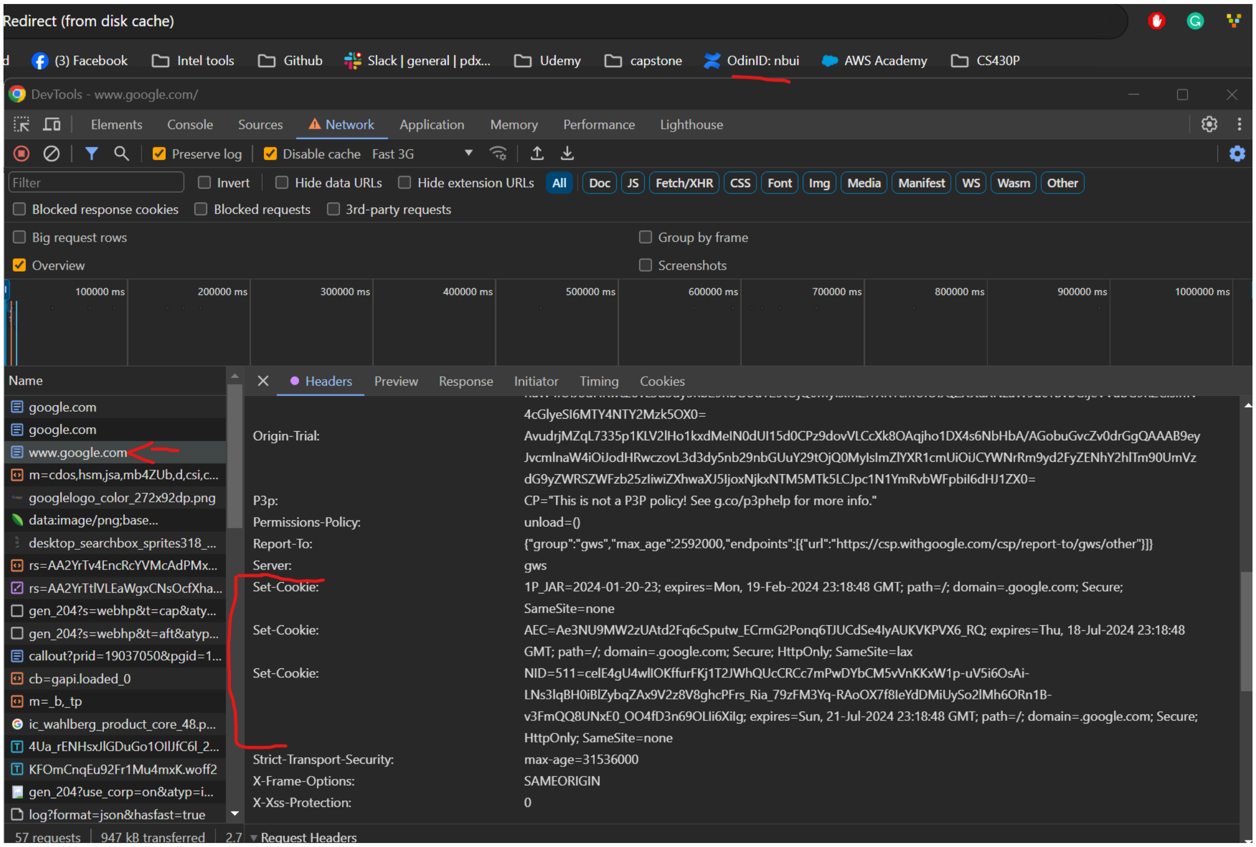
Request URL: https://www.google.com/
Request Method: GET
Status Code: ● 200 OK
Remote Address: 142.251.33.68:443
Referrer Policy: strict-origin-when-cross-origin

Response Headers

Accept-Ch: Sec-CH-UA-Platform
Accept-Ch: Sec-CH-UA-Platform-Version
Accept-Ch: Sec-CH-UA-Full-Version
Accept-Ch: Sec-CH-UA-Arch
Accept-Ch: Sec-CH-UA-Model
Accept-Ch: Sec-CH-UA-Bitness
Accept-Ch: Sec-CH-UA-Full-Version-List
Accept-Ch: Sec-CH-UA-WoW64
Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000

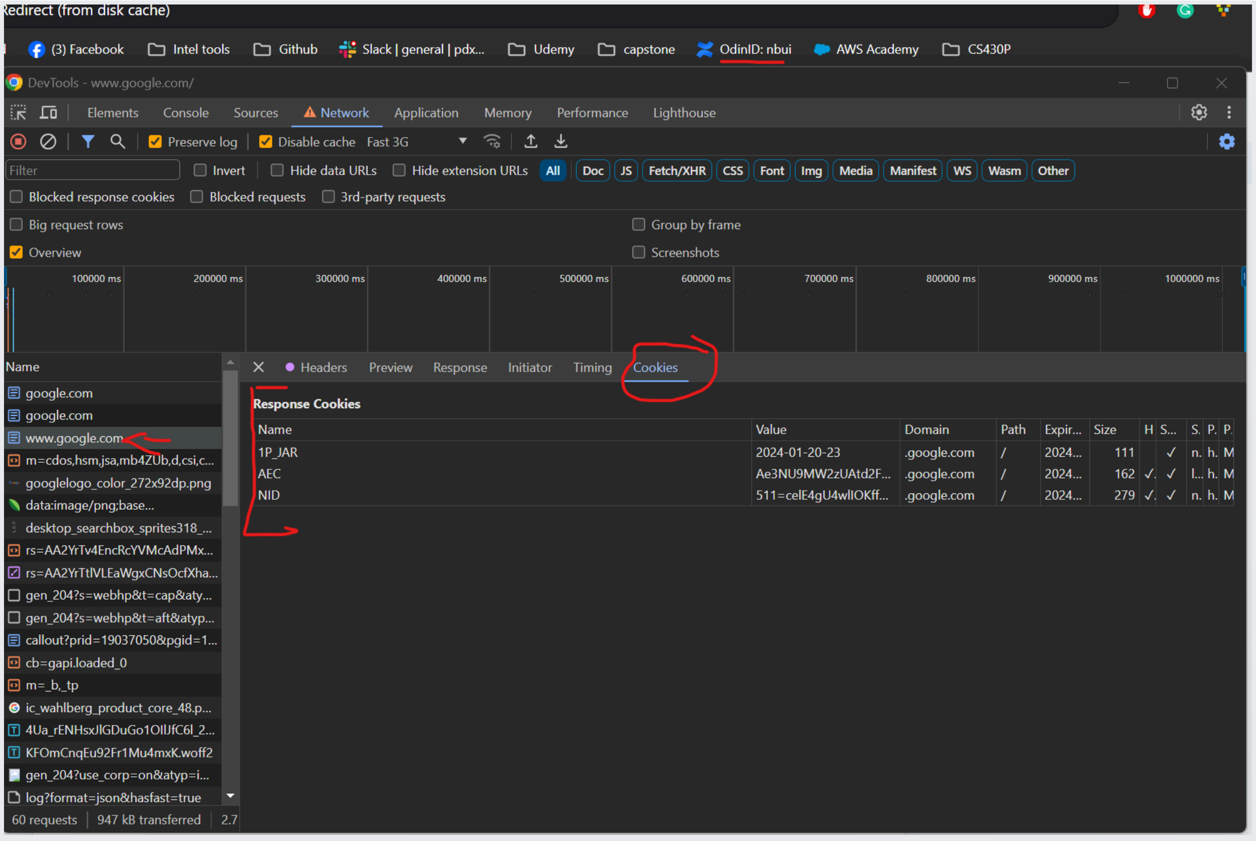
- **What URL does the first redirection send the browser to?** https://google.com/

- **What URL does the second redirection send the browser to?** https://www.google.com/
- **Take a screenshot of when cookies are set via Set-Cookie:**



	4cGlyeSI6MTY4NTY2Mzk5OX0=
Origin-Trial:	AvudrjMZqL7335p1KLV2lHo1kxdMeIN0dUI15d0CPz9dovVLCcXk8OAqjho1DX4s6NbHbA/AGobuGvcZv0drGgQAAAB9eyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlYXR1cmUiOiJCYWNrRm9yd2FyZENhY2hlTm90UmVzdG9yZWRSZWFzb25zIiwiZXhwaXJ5IjoxNjkxNTM5MTk5LCJpc1N1YmRvbWFpbiI6dHJ1ZX0=
P3p:	CP="This is not a P3P policy! See g.co/p3phelp for more info."
Permissions-Policy:	unload=()
Report-To:	{"group":"gws","max_age":2592000,"endpoints":[{"url":"https://csp.withgoogle.com/csp/report-to/gws/other"}]}
Server:	gws
Set-Cookie:	1P_JAR=2024-01-20-23; expires=Mon, 19-Feb-2024 23:18:48 GMT; path=/; domain=.google.com; Secure; SameSite=none
Set-Cookie:	AEC=Ae3NU9MW2zUAtd2Fq6cSputw_ECrmG2Ponq6TJUCdSe4lyAUKVKPVX6_RQ; expires=Thu, 18-Jul-2024 23:18:48 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=lax
Set-Cookie:	NID=511=celE4gU4wlIOKffurFKj1T2JWhQUcCRCc7mPwDYbCM5vVnKKxW1p-uV5i6OsAi-LNs3lqBH0iBlZybqZAx9V2z8V8ghcPFrs_Ria_79zFM3Yq-RAoOX7f8IeYdDMiUySo2lMh6ORn1B-v3FmQQ8UNxE0_OO4fD3n69OLli6Xilg; expires=Sun, 21-Jul-2024 23:18:48 GMT; path=/; domain=.google.com; Secure; HttpOnly; SameSite=none
Strict-Transport-Security:	max-age=31536000
X-Frame-Options:	SAMEORIGIN
X-Xss-Protection:	0

- Take a screenshot of when cookies are attached via Cookie:



☑ Preserve log ☑ Disable cache Fast 3G

☐ Invert ☐ Hide data URLs ☐ Hide extension URLs

☐ Blocked response cookies ☐ Blocked requests ☐ 3rd-party requests

☐ Big request rows ☐ Group by frame

☑ Overview ☐ Screenshots

Name
google.com
google.com
www.google.com
m=cdos,hsm,jsa,mb4ZUb,d,csi,c...
googlelogo_color_272x92dp.png
data:image/png;base...
desktop_searchbox_sprites318_...
rs=AA2YrTv4EncRcYVMcAdPMx...
rs=AA2YrTtlVLEaWgxCNsOcfXha...
gen_204?s=webhp&t=cap&aty...
gen_204?s=webhp&t=aft&atyp...
callout?prid=19037050&pgid=1...
cb=gapi.loaded_0
m=_b,_tp
ic_wahlberg_product_core_48.p...
4Ua_rENHsxJlGDuGo1OIlJfC6l_2...
KFOmCnqEu92Fr1Mu4mxK.woff2
gen_204?use_corp=on&atyp=i...
log?format=json&hasfast=true

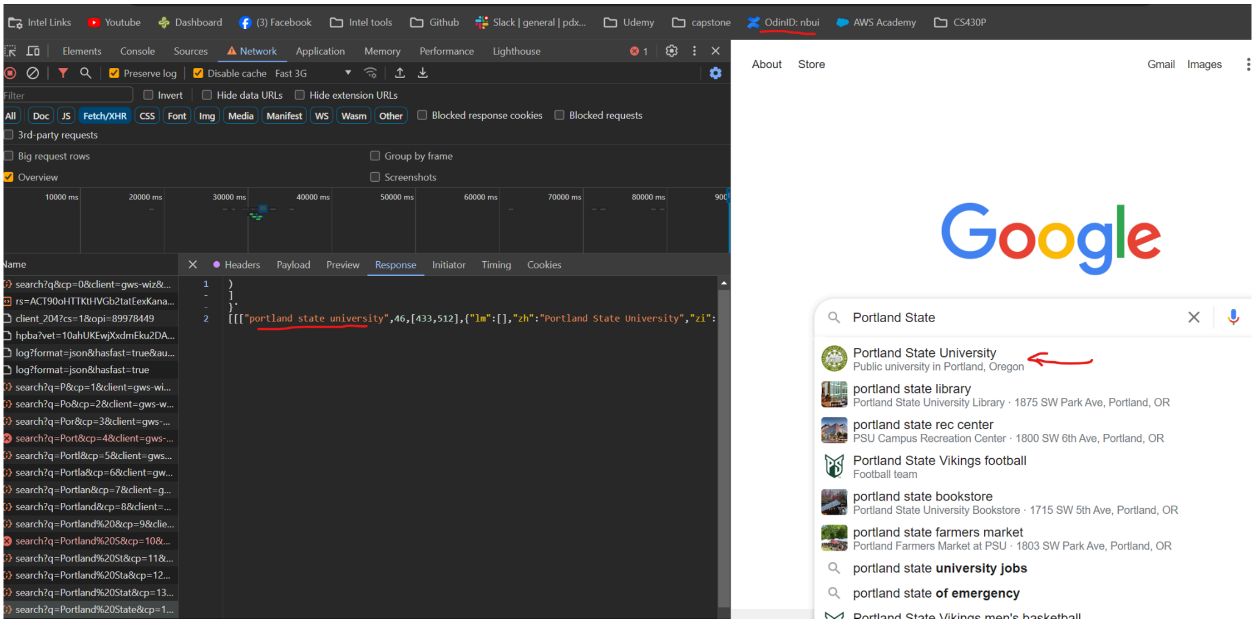
60 requests | 947 kB transferred | 2.7

Response Cookies

Name	Value	Domain	Path	Expir...	Size	H	S...	S.	P.	P.
1P_JAR	2024-01-20-23	.google.com	/	2024...	111		✓	n.	h.	M
AEC	Ae3NU9MW2zUAtd2F...	.google.com	/	2024...	162	✓.	✓	l...	h.	M
NID	511=celE4gU4wlIOKff...	.google.com	/	2024...	279	✓.	✓	n.	h.	M

6. Asynchronous HTTP requests

- Show the requests and responses in the listing. Click on the last request sent, then click on the response to see that its payload has returned the data that is then rendered on the search page similar to what is shown below for "rabbid"



```
search?q&cp=0&client=gws-wiz&...
rs=ACT90oHTTKtHVGb2tatEexKana...
client_204?cs=1&opi=89978449
hpba?vet=10ahUKEwjXxdmEku2DA...
log?format=json&hasfast=true&au...
log?format=json&hasfast=true
search?q=P&cp=1&client=gws-wi...
search?q=Po&cp=2&client=gws-w...
search?q=Por&cp=3&client=gws-...
search?q=Port&cp=4&client=gws-...
search?q=Portl&cp=5&client=gws...
search?q=Portla&cp=6&client=gw...
search?q=Portlan&cp=7&client=g...
search?q=Portland&cp=8&client=...
search?q=Portland%20&cp=9&clie...
search?q=Portland%20S&cp=10&...
search?q=Portland%20St&cp=11&...
search?q=Portland%20Sta&cp=12...
search?q=Portland%20Stat&cp=13...
search?q=Portland%20State&cp=1...
```

```
)
]
}'
[[["portland state university",46,[433,512],{"lm":[],"zh":"Portland State University","zi":
```

Portland State

Portland State University
Public university in Portland, Oregon

portland state library
Portland State University Library · 1875 SW Park Ave, Portland, OR

portland state rec center
PSU Campus Recreation Center · 1800 SW 6th Ave, Portland, OR

Portland State Vikings football
Football team

portland state bookstore
Portland State University Bookstore · 1715 SW 5th Ave, Portland, OR

portland state farmers market
Portland Farmers Market at PSU · 1803 SW Park Ave, Portland, OR

portland state **university jobs**

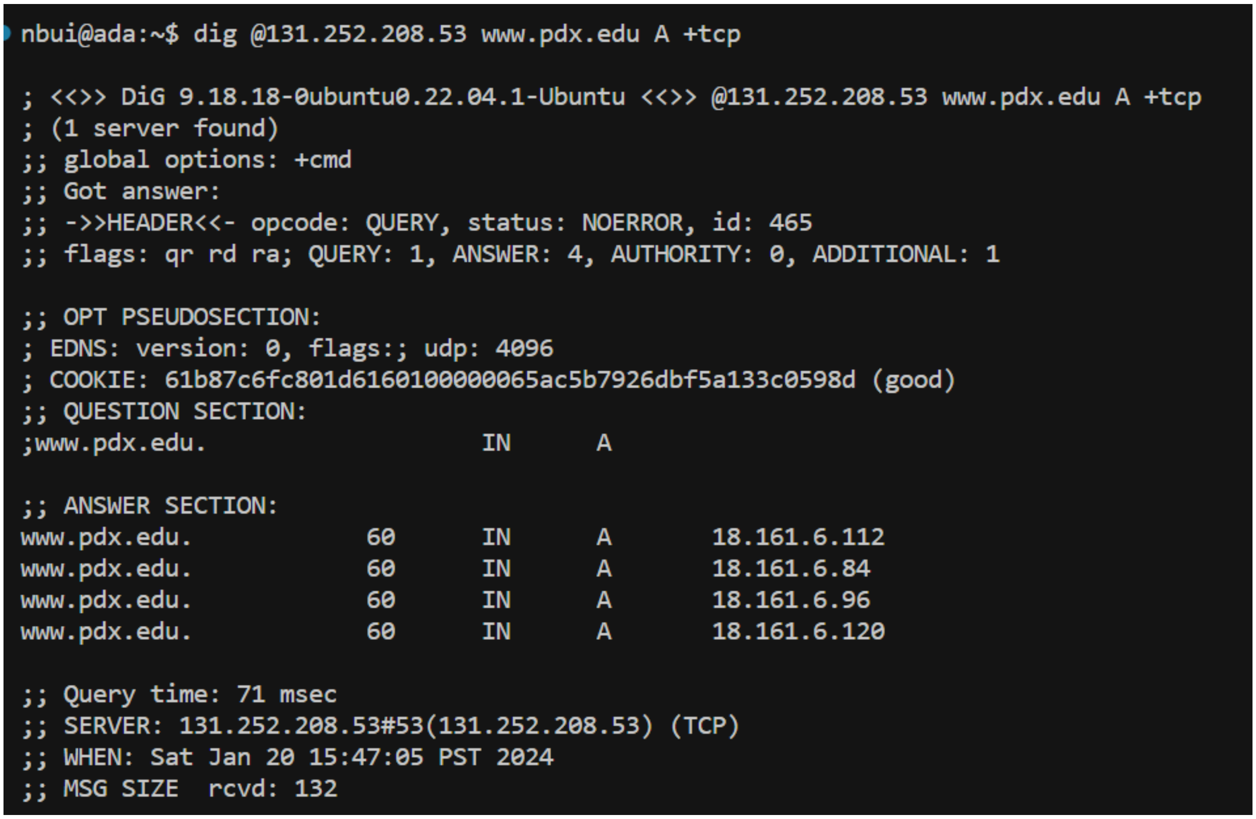
portland state **of emergency**

Portland State Vikings men's basketball

Lab 02.2

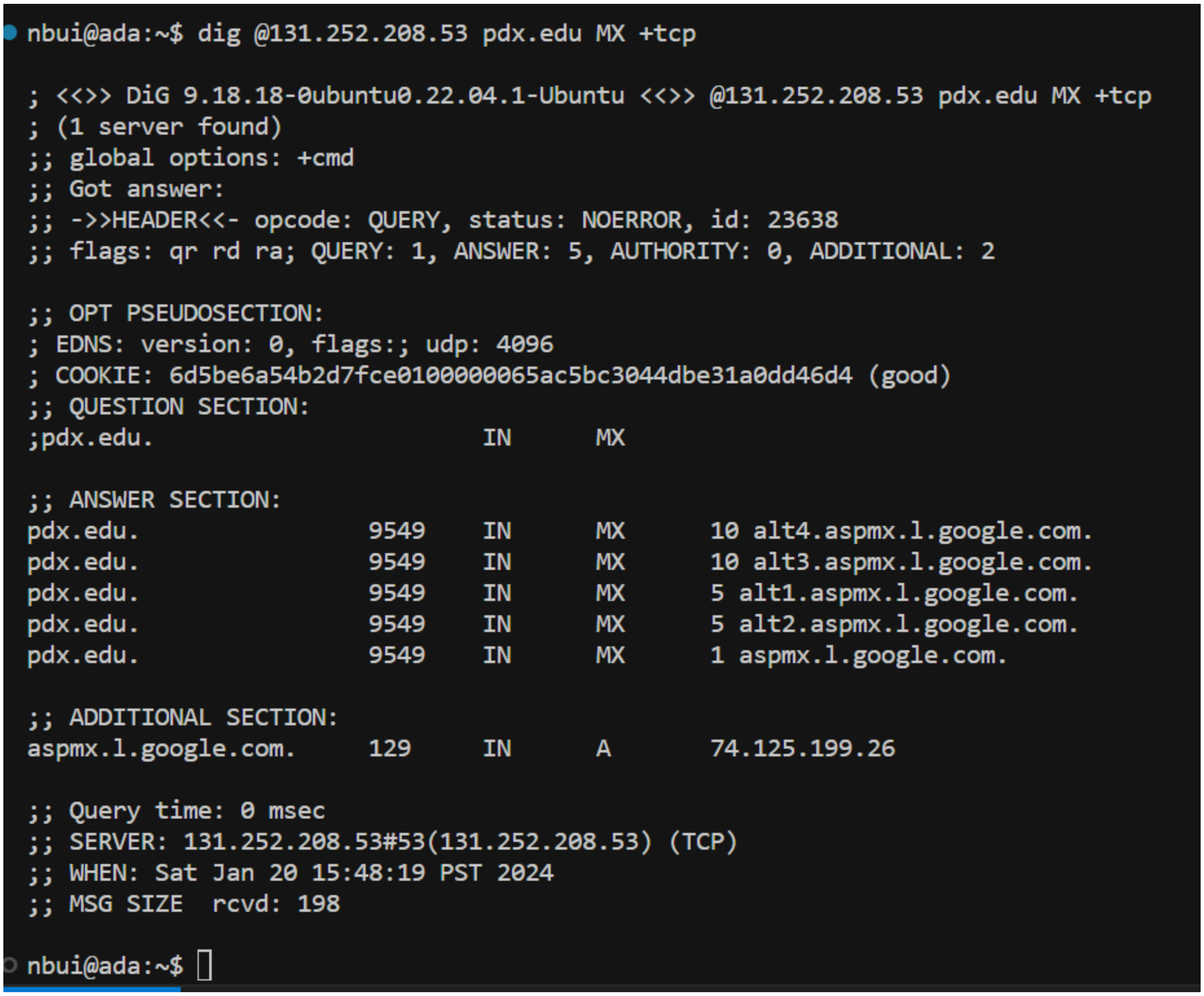
1. DNS reconnaissance #1 (dig)

- Take a screenshot of the prompt and the dig command produced.



```
nbui@ada:~$ dig @131.252.208.53 www.pdx.edu A +tcp

; <<>> DiG 9.18.18-0ubuntu0.22.04.1-Ubuntu <<>> @131.252.208.53 www.pdx.edu A +tcp
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 465
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 61b87c6fc801d6160100000065ac5b7926dbf5a133c0598d (good)
;; QUESTION SECTION:
;www.pdx.edu.                   IN      A

;; ANSWER SECTION:
www.pdx.edu.            60      IN      A       18.161.6.112
www.pdx.edu.            60      IN      A       18.161.6.84
www.pdx.edu.            60      IN      A       18.161.6.96
www.pdx.edu.            60      IN      A       18.161.6.120

;; Query time: 71 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (TCP)
;; WHEN: Sat Jan 20 15:47:05 PST 2024
;; MSG SIZE  rcvd: 132
```

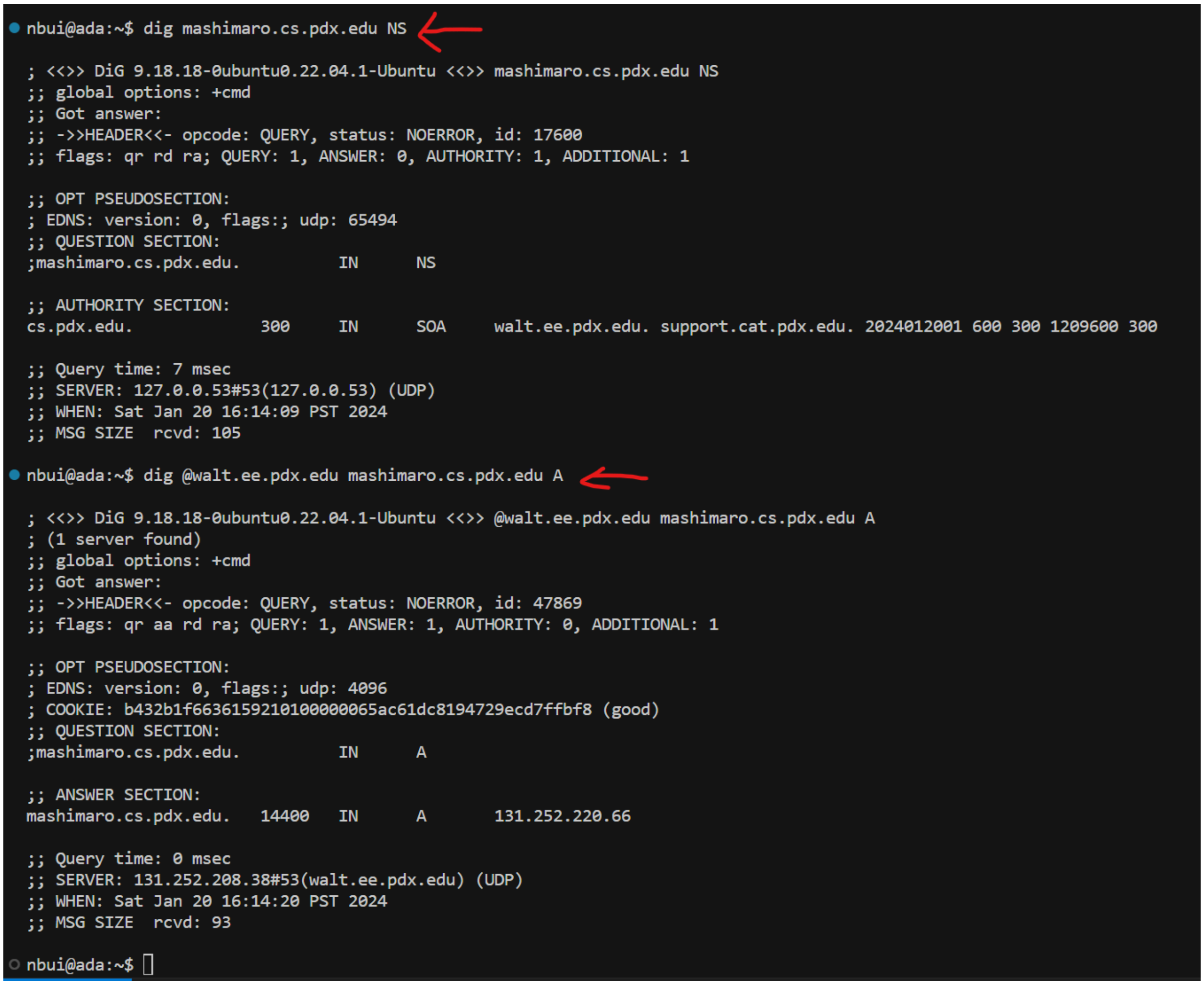
- Take a screenshot of the records returned for your lab notebook.



```
nbui@ada:~$ dig @131.252.208.53 pdx.edu MX +tcp

; <<>> DiG 9.18.18-0ubuntu0.22.04.1-Ubuntu <<>> @131.252.208.53 pdx.edu MX +tcp
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 23638
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 6d5be6a54b2d7fce0100000065ac5bc3044dbe31a0dd46d4 (good)
;; QUESTION SECTION:
;pdx.edu.                       IN      MX

;; ANSWER SECTION:
pdx.edu.                9549    IN      MX      10 alt4.aspmx.l.google.com.
pdx.edu.                9549    IN      MX      10 alt3.aspmx.l.google.com.
pdx.edu.                9549    IN      MX      5 alt1.aspmx.l.google.com.
pdx.edu.                9549    IN      MX      5 alt2.aspmx.l.google.com.
pdx.edu.                9549    IN      MX      1 aspmx.l.google.com.

;; ADDITIONAL SECTION:
aspmx.l.google.com.     129     IN      A       74.125.199.26

;; Query time: 0 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (TCP)
;; WHEN: Sat Jan 20 15:48:19 PST 2024
;; MSG SIZE  rcvd: 198

nbui@ada:~$ 
```

- What cloud provider hosts the web site for www.pdx.edu? Amazon CloudFront
- What cloud provider handles mail for pdx.edu? Google mail

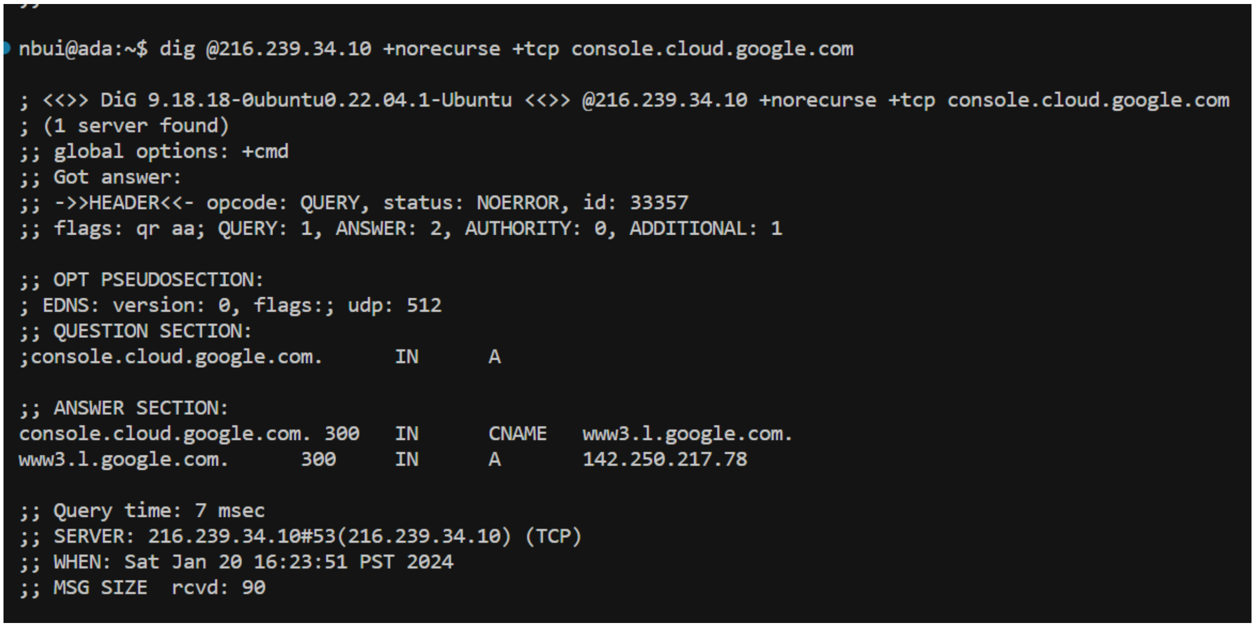
- Take a screenshot of the results for both records for your lab notebook.



```
nbui@ada:~$ dig mashimaro.cs.pdx.edu NS

; <<>> DiG 9.18.18-0ubuntu0.22.04.1-Ubuntu <<>> mashimaro.cs.pdx.edu NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17600
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;mashimaro.cs.pdx.edu.          IN      NS

;; AUTHORITY SECTION:
cs.pdx.edu.             300     IN      SOA     walt.ee.pdx.edu. support.cat.pdx.edu. 2024012001 600 300 1209600 300

;; Query time: 7 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Sat Jan 20 16:14:09 PST 2024
;; MSG SIZE  rcvd: 105

nbui@ada:~$ dig @walt.ee.pdx.edu mashimaro.cs.pdx.edu A

; <<>> DiG 9.18.18-0ubuntu0.22.04.1-Ubuntu <<>> @walt.ee.pdx.edu mashimaro.cs.pdx.edu A
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47869
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: b432b1f6636159210100000065ac61dc8194729ecd7ffbf8 (good)
;; QUESTION SECTION:
;mashimaro.cs.pdx.edu.          IN      A

;; ANSWER SECTION:
mashimaro.cs.pdx.edu.   14400   IN      A       131.252.220.66

;; Query time: 0 msec
;; SERVER: 131.252.208.38#53(walt.ee.pdx.edu) (UDP)
;; WHEN: Sat Jan 20 16:14:20 PST 2024
;; MSG SIZE  rcvd: 93

nbui@ada:~$ 
```

2. Iterative DNS lookups

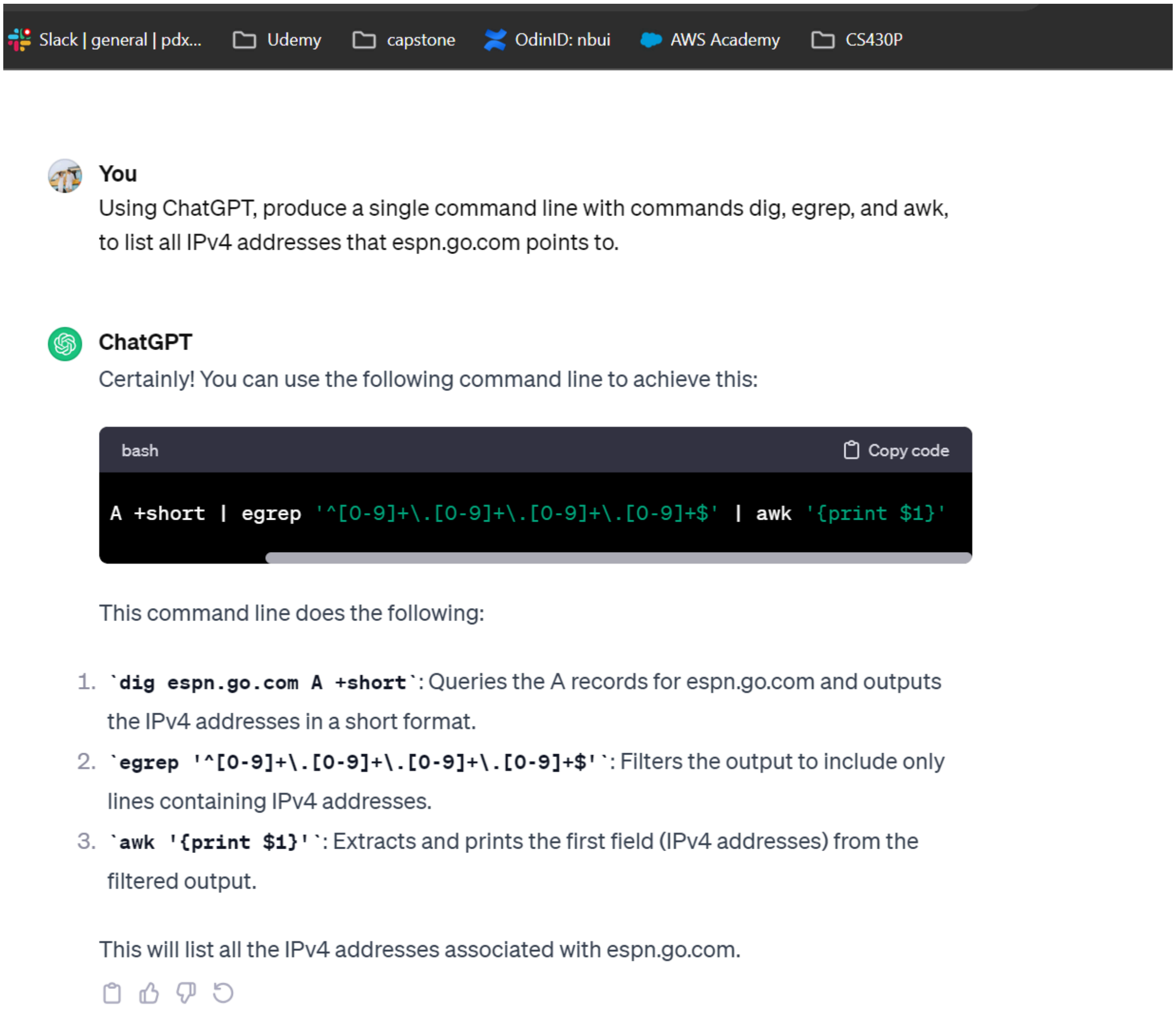
- List all of the iterative dig commands performed for the lookup
    - dig | grep root-servers
    - dig f.root-servers.net.
    - dig @192.5.5.241 +norecurse +tcp console.cloud.google.com
    - dig @192.43.172.30 +norecurse +tcp console.cloud.google.com
    - dig @216.239.34.10 +norecurse +tcp console.cloud.google.com
- Take a screenshot of the results of the final query for your lab notebook.



```
nbui@ada:~$ dig @216.239.34.10 +norecurse +tcp console.cloud.google.com

; <<>> DiG 9.18.18-0ubuntu0.22.04.1-Ubuntu <<>> @216.239.34.10 +norecurse +tcp console.cloud.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33357
;; flags: qr aa; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;console.cloud.google.com.      IN      A

;; ANSWER SECTION:
console.cloud.google.com. 300   IN      CNAME   www3.l.google.com.
www3.l.google.com.      300     IN      A       142.250.217.78

;; Query time: 7 msec
;; SERVER: 216.239.34.10#53(216.239.34.10) (TCP)
;; WHEN: Sat Jan 20 16:23:51 PST 2024
;; MSG SIZE  rcvd: 90
```

3. Reverse DNS lookups

- Take a screenshot of the prompt and the command produced



You

Using ChatGPT, produce a single command line with commands dig, egrep, and awk, to list all IPv4 addresses that espn.go.com points to.

ChatGPT

Certainly! You can use the following command line to achieve this:

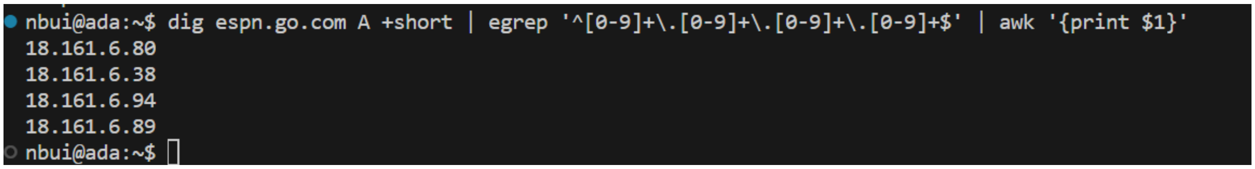
```bash
A +short | egrep '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$' | awk '{print $1}'
```

This command line does the following:

1. `dig espn.go.com A +short`: Queries the A records for espn.go.com and outputs the IPv4 addresses in a short format.
2. `egrep '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$'`: Filters the output to include only lines containing IPv4 addresses.
3. `awk '{print $1}'`: Extracts and prints the first field (IPv4 addresses) from the filtered output.

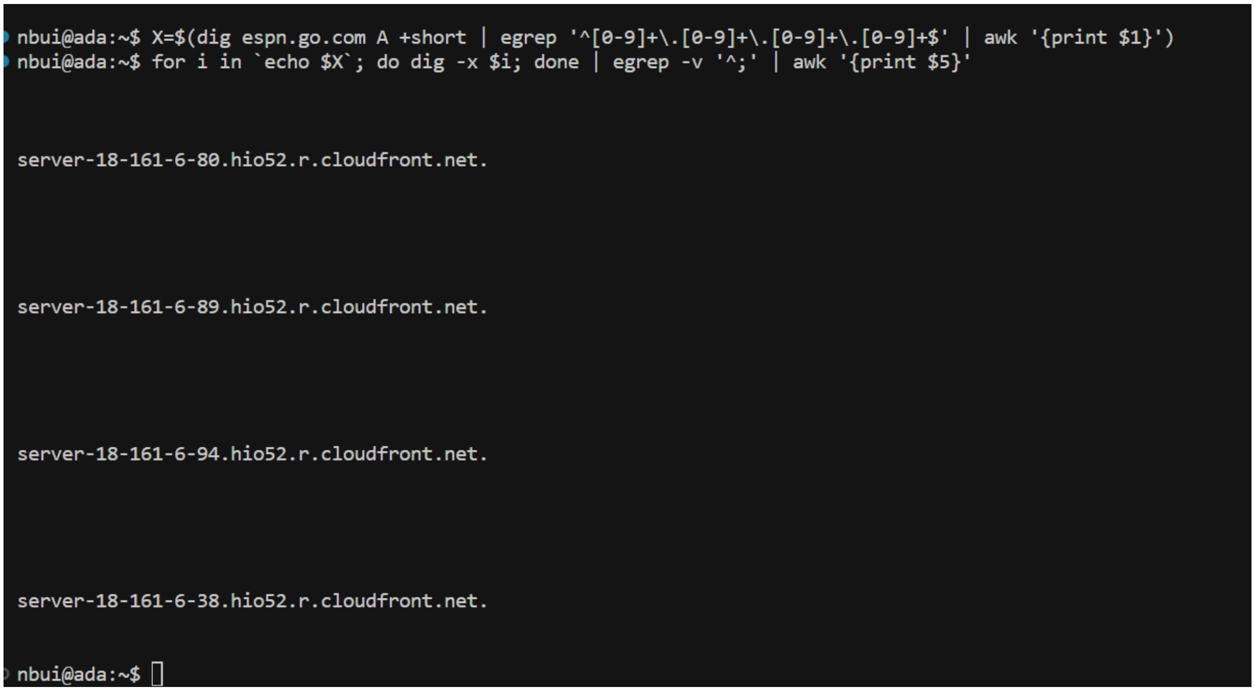
This will list all the IPv4 addresses associated with espn.go.com.

- Take a screenshot of its results for your lab notebook



```
nbui@ada:~$ dig espn.go.com A +short | egrep '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$' | awk '{print $1}'
18.161.6.80
18.161.6.38
18.161.6.94
18.161.6.89
nbui@ada:~$ 
```

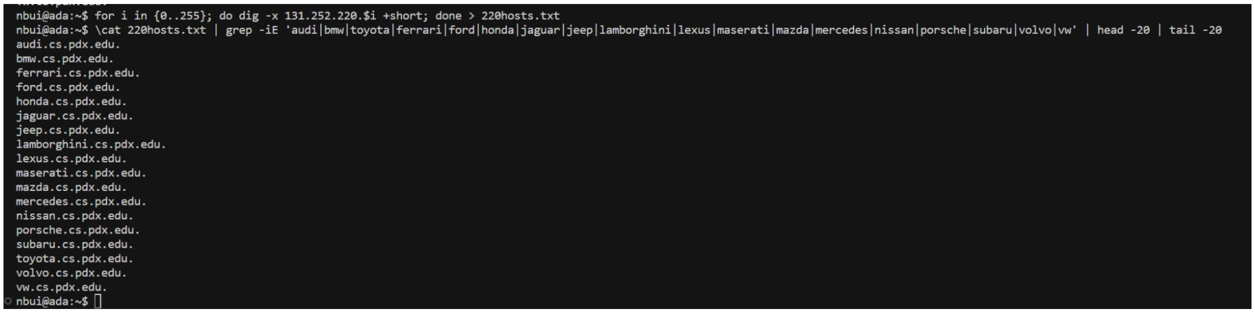
- Take a screenshot of the command and its results for your lab notebook



```
nbui@ada:~$ X=$(dig espn.go.com A +short | egrep '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$' | awk '{print $1}')
nbui@ada:~$ for i in `echo $X`; do dig -x $i; done | egrep -v '^;' | awk '{print $5}'



server-18-161-6-80.hio52.r.cloudfront.net.




server-18-161-6-89.hio52.r.cloudfront.net.




server-18-161-6-94.hio52.r.cloudfront.net.




server-18-161-6-38.hio52.r.cloudfront.net.

nbui@ada:~$ 
```

4. Host enumeration

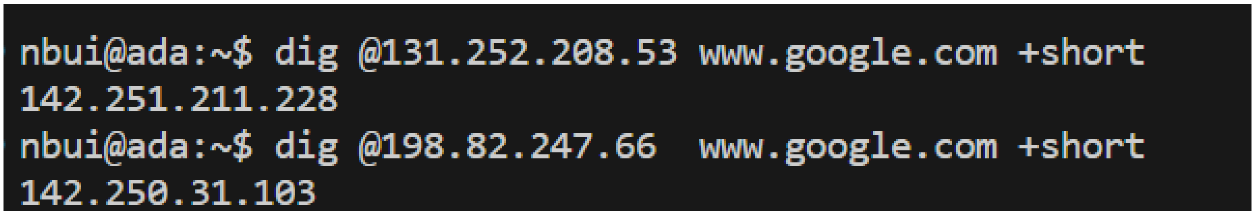
- Take a screenshot of the results in your lab notebook



```
nbui@ada:~$ for i in {0..255}; do dig -x 131.252.220.$i +short; done > 220hosts.txt
nbui@ada:~$ \cat 220hosts.txt | grep -iE 'audi|bmw|toyota|ferrari|ford|honda|jaguar|jeep|lamborghini|lexus|maserati|mazda|mercedes|nissan|porsche|subaru|volvo|vw' | head -20 | tail -20
audi.cs.pdx.edu.
bmw.cs.pdx.edu.
ferrari.cs.pdx.edu.
ford.cs.pdx.edu.
honda.cs.pdx.edu.
jaguar.cs.pdx.edu.
jeep.cs.pdx.edu.
lamborghini.cs.pdx.edu.
lexus.cs.pdx.edu.
maserati.cs.pdx.edu.
mazda.cs.pdx.edu.
mercedes.cs.pdx.edu.
nissan.cs.pdx.edu.
porsche.cs.pdx.edu.
subaru.cs.pdx.edu.
toyota.cs.pdx.edu.
volvo.cs.pdx.edu.
vw.cs.pdx.edu.
nbui@ada:~$ 
```

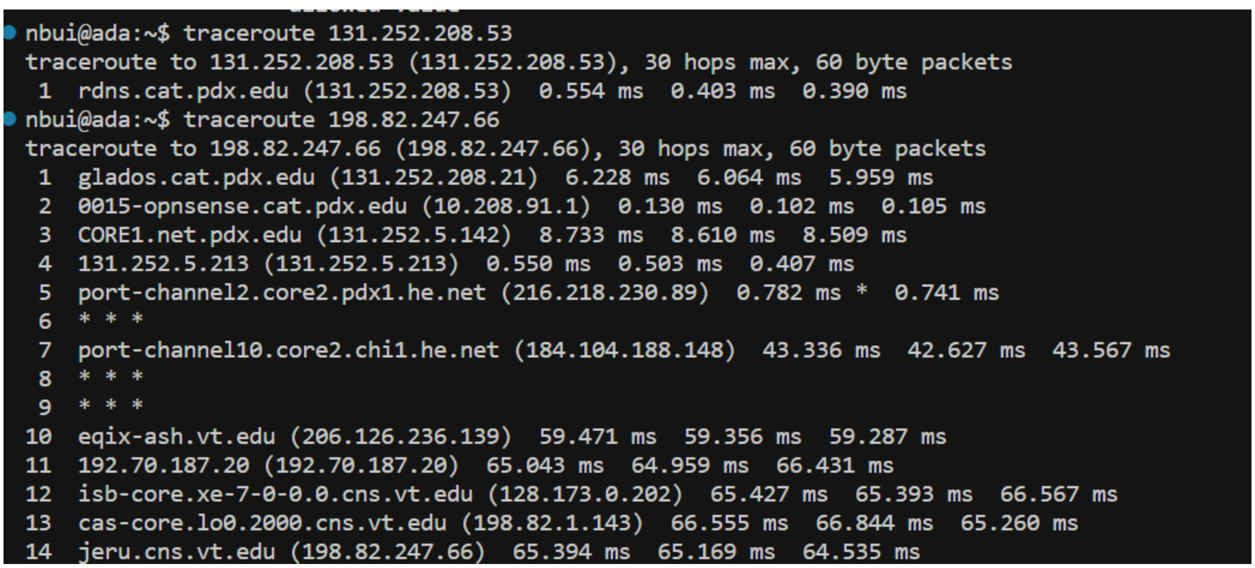
5. Geographic DNS #2

- What geographic locations do ipinfo.io and DB-IP return?
    - 131.252.208.53: Portland (North), Oregon
    - 198.82.247.66: Blacksburg, Virginia
- Record one address for www.google.com from each result for your lab notebook.

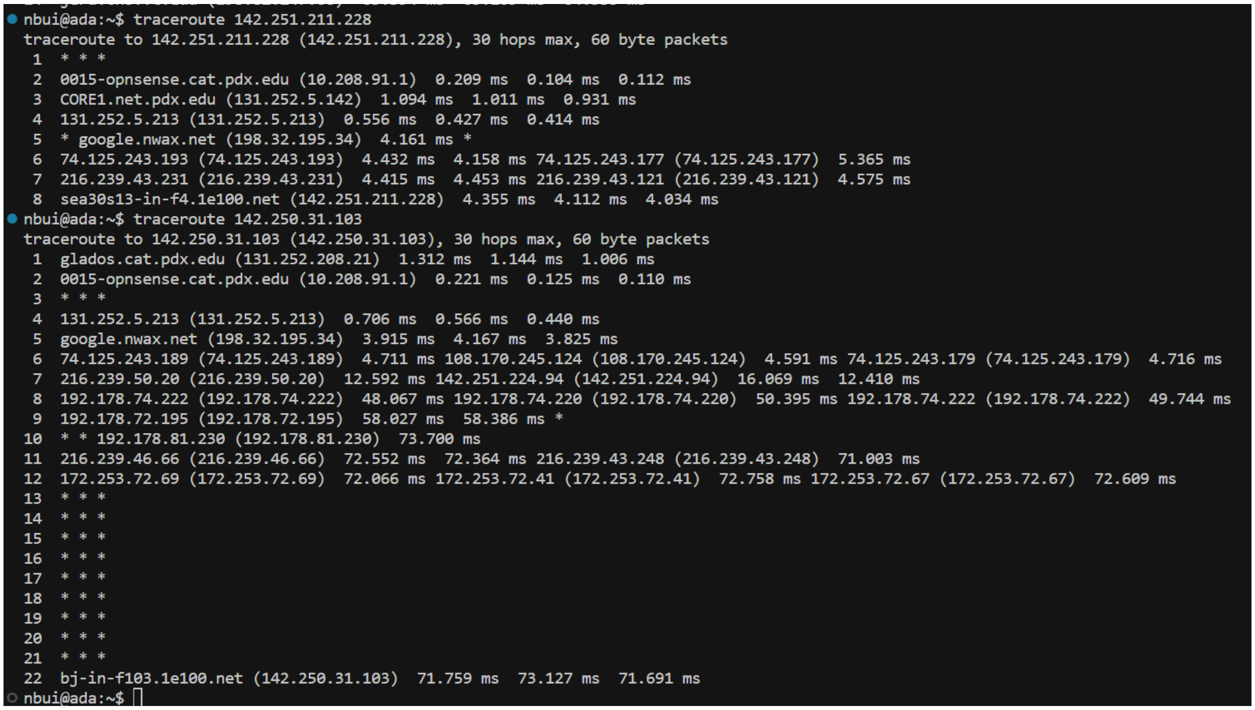


```
nbui@ada:~$ dig @131.252.208.53 www.google.com +short
142.251.211.228
nbui@ada:~$ dig @198.82.247.66  www.google.com +short
142.250.31.103
```

- What are the geographic coordinates of each DNS server and the IP address it resolves for www.google.com?
    - 142.251.211.228: Seattle, Washington
    - 142.250.31.103: Mountain View, California
- Take a screenshot of the results for your lab notebook.



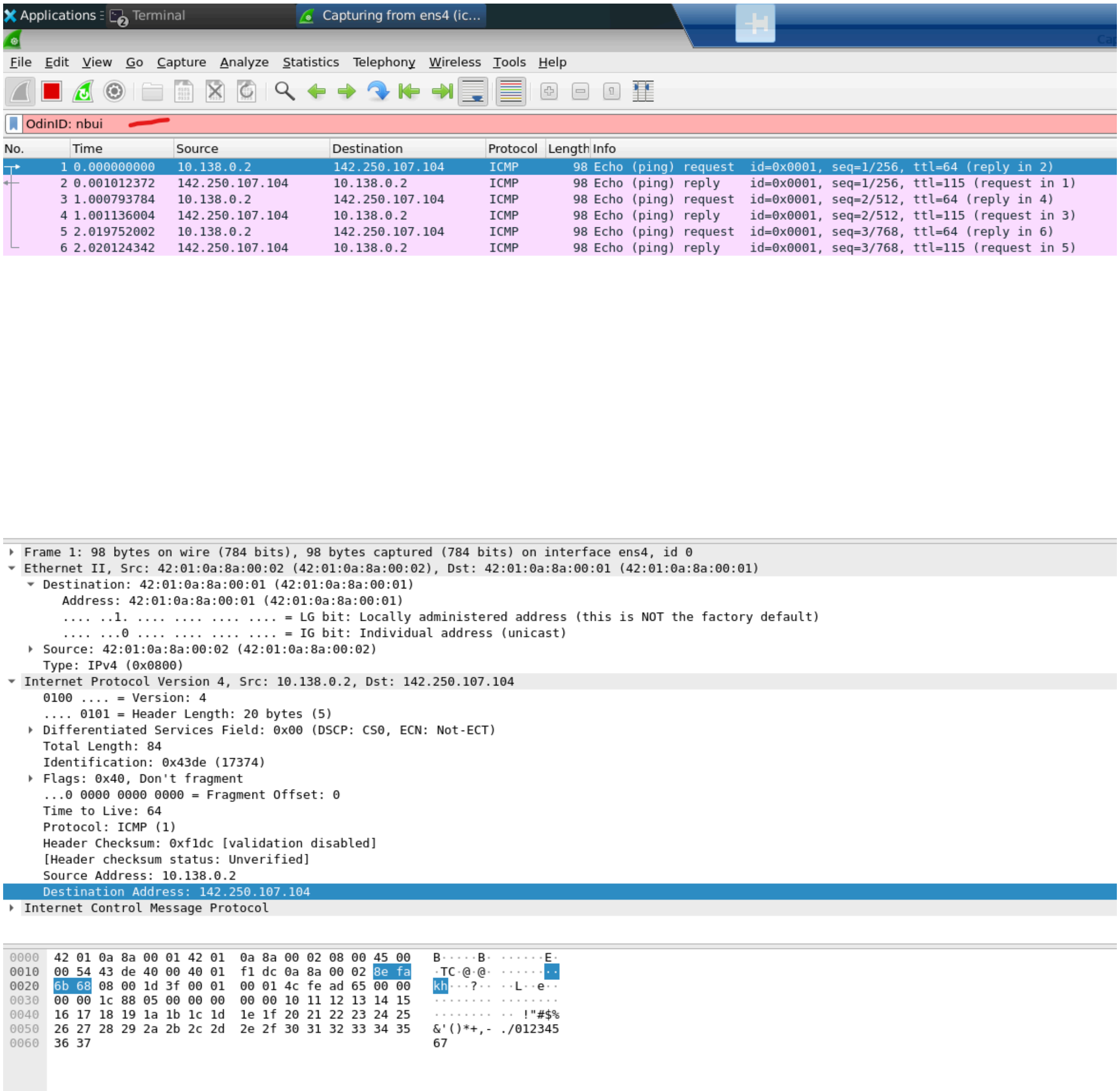
```
nbui@ada:~$ traceroute 131.252.208.53
traceroute to 131.252.208.53 (131.252.208.53), 30 hops max, 60 byte packets
 1  rdns.cat.pdx.edu (131.252.208.53)  0.554 ms  0.403 ms  0.390 ms
nbui@ada:~$ traceroute 198.82.247.66
traceroute to 198.82.247.66 (198.82.247.66), 30 hops max, 60 byte packets
 1  glados.cat.pdx.edu (131.252.208.21)  6.228 ms  6.064 ms  5.959 ms
 2  0015-opnsense.cat.pdx.edu (10.208.91.1)  0.130 ms  0.102 ms  0.105 ms
 3  CORE1.net.pdx.edu (131.252.5.142)  8.733 ms  8.610 ms  8.509 ms
 4  131.252.5.213 (131.252.5.213)  0.550 ms  0.503 ms  0.407 ms
 5  port-channel2.core2.pdx1.he.net (216.218.230.89)  0.782 ms *  0.741 ms
 6  * * *
 7  port-channel10.core2.chi1.he.net (184.104.188.148)  43.336 ms  42.627 ms  43.567 ms
 8  * * *
 9  * * *
10  eqix-ash.vt.edu (206.126.236.139)  59.471 ms  59.356 ms  59.287 ms
11  192.70.187.20 (192.70.187.20)  65.043 ms  64.959 ms  66.431 ms
12  isb-core.xe-7-0-0.0.cns.vt.edu (128.173.0.202)  65.427 ms  65.393 ms  66.567 ms
13  cas-core.lo0.2000.cns.vt.edu (198.82.1.143)  66.555 ms  66.844 ms  65.260 ms
14  jeru.cns.vt.edu (198.82.247.66)  65.394 ms  65.169 ms  64.535 ms
```



```
nbui@ada:~$ traceroute 142.251.211.228
traceroute to 142.251.211.228 (142.251.211.228), 30 hops max, 60 byte packets
 1  * * *
 2  0015-opnsense.cat.pdx.edu (10.208.91.1)  0.209 ms  0.104 ms  0.112 ms
 3  CORE1.net.pdx.edu (131.252.5.142)  1.094 ms  1.011 ms  0.931 ms
 4  131.252.5.213 (131.252.5.213)  0.556 ms  0.427 ms  0.414 ms
 5  * google.nwax.net (198.32.195.34)  4.161 ms *
 6  74.125.243.193 (74.125.243.193)  4.432 ms  4.158 ms 74.125.243.177 (74.125.243.177)  5.365 ms
 7  216.239.43.231 (216.239.43.231)  4.415 ms  4.453 ms 216.239.43.121 (216.239.43.121)  4.575 ms
 8  sea30s13-in-f4.1e100.net (142.251.211.228)  4.355 ms  4.112 ms  4.034 ms
nbui@ada:~$ traceroute 142.250.31.103
traceroute to 142.250.31.103 (142.250.31.103), 30 hops max, 60 byte packets
 1  glados.cat.pdx.edu (131.252.208.21)  1.312 ms  1.144 ms  1.006 ms
 2  0015-opnsense.cat.pdx.edu (10.208.91.1)  0.221 ms  0.125 ms  0.110 ms
 3  * * *
 4  131.252.5.213 (131.252.5.213)  0.706 ms  0.566 ms  0.440 ms
 5  google.nwax.net (198.32.195.34)  3.915 ms  4.167 ms  3.825 ms
 6  74.125.243.189 (74.125.243.189)  4.711 ms 108.170.245.124 (108.170.245.124)  4.591 ms 74.125.243.179 (74.125.243.179)  4.716 ms
 7  216.239.50.20 (216.239.50.20)  12.592 ms 142.251.224.94 (142.251.224.94)  16.069 ms  12.410 ms
 8  192.178.74.222 (192.178.74.222)  48.067 ms 192.178.74.220 (192.178.74.220)  50.395 ms 192.178.74.222 (192.178.74.222)  49.744 ms
 9  192.178.72.195 (192.178.72.195)  58.027 ms  58.386 ms *
10  * * 192.178.81.230 (192.178.81.230)  73.700 ms
11  216.239.46.66 (216.239.46.66)  72.552 ms  72.364 ms 216.239.43.248 (216.239.43.248)  71.003 ms
12  172.253.72.69 (172.253.72.69)  72.066 ms 172.253.72.41 (172.253.72.41)  72.758 ms 172.253.72.67 (172.253.72.67)  72.609 ms
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  bj-in-f103.1e100.net (142.250.31.103)  71.759 ms  73.127 ms  71.691 ms
nbui@ada:~$
```

7.

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



OdinID: nbui

```
No.   Time         Source           Destination      Protocol Length Info
    1 0.000000000  10.138.0.2       142.250.107.104  ICMP        98 Echo (ping) request  id=0x0001, seq=1/256, ttl=64 (reply in 2)
    2 0.001012372  142.250.107.104  10.138.0.2       ICMP        98 Echo (ping) reply    id=0x0001, seq=1/256, ttl=115 (request in 1)
    3 1.000793784  10.138.0.2       142.250.107.104  ICMP        98 Echo (ping) request  id=0x0001, seq=2/512, ttl=64 (reply in 4)
    4 1.001136004  142.250.107.104  10.138.0.2       ICMP        98 Echo (ping) reply    id=0x0001, seq=2/512, ttl=115 (request in 3)
    5 2.019752002  10.138.0.2       142.250.107.104  ICMP        98 Echo (ping) request  id=0x0001, seq=3/768, ttl=64 (reply in 6)
    6 2.020124342  142.250.107.104  10.138.0.2       ICMP        98 Echo (ping) reply    id=0x0001, seq=3/768, ttl=115 (request in 5)
```

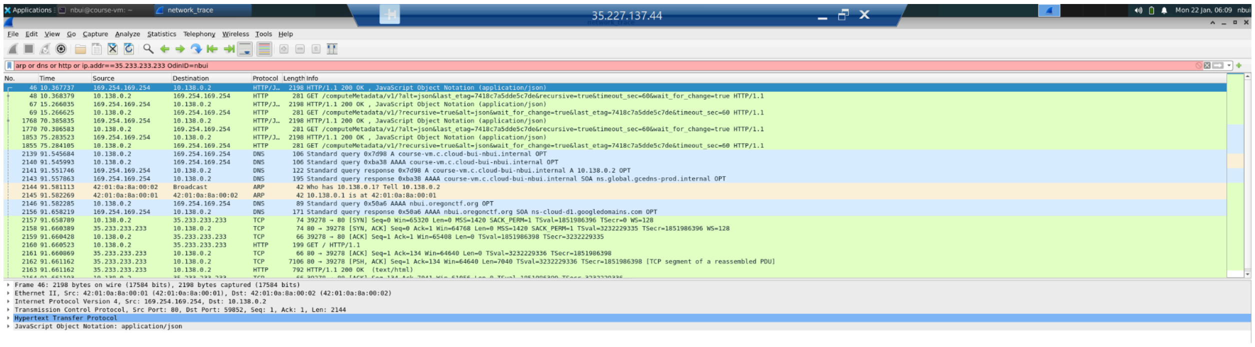
```
▸ Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface ens4, id 0
▾ Ethernet II, Src: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02), Dst: 42:01:0a:8a:00:01 (42:01:0a:8a:00:01)
  ▾ Destination: 42:01:0a:8a:00:01 (42:01:0a:8a:00:01)
      Address: 42:01:0a:8a:00:01 (42:01:0a:8a:00:01)
      .... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  ▸ Source: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02)
    Type: IPv4 (0x0800)
▾ Internet Protocol Version 4, Src: 10.138.0.2, Dst: 142.250.107.104
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 84
    Identification: 0x43de (17374)
  ▸ Flags: 0x40, Don't fragment
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 64
    Protocol: ICMP (1)
    Header Checksum: 0xf1dc [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.138.0.2
    Destination Address: 142.250.107.104
▸ Internet Control Message Protocol
```

```
0000  42 01 0a 8a 00 01 42 01  0a 8a 00 02 08 00 45 00   B·····B· ······E·
0010  00 54 43 de 40 00 40 01  f1 dc 0a 8a 00 02 8e fa   ·TC·@·@· ········
0020  6b 68 08 00 1d 3f 00 01  00 01 4c fe ad 65 00 00   kh···?·· ··L··e··
0030  00 00 1c 88 05 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
```

- **Does the destination MAC address correspond to an interface on the VM, an interface on the default router or an interface on Google's web site?** It corresponds to the default router HW address (42:01:0a:8a:00:01)
- **Does the destination MAC address correspond to an interface on the VM, an interface on the default router or an interface on Google's web site?** It corresponds to the VM's hardware address(42:01:0a:8a:00:02)

10. Analyze trace

- Take a screenshot of the all of the packets returned within Wireshark that includes their packet numbers



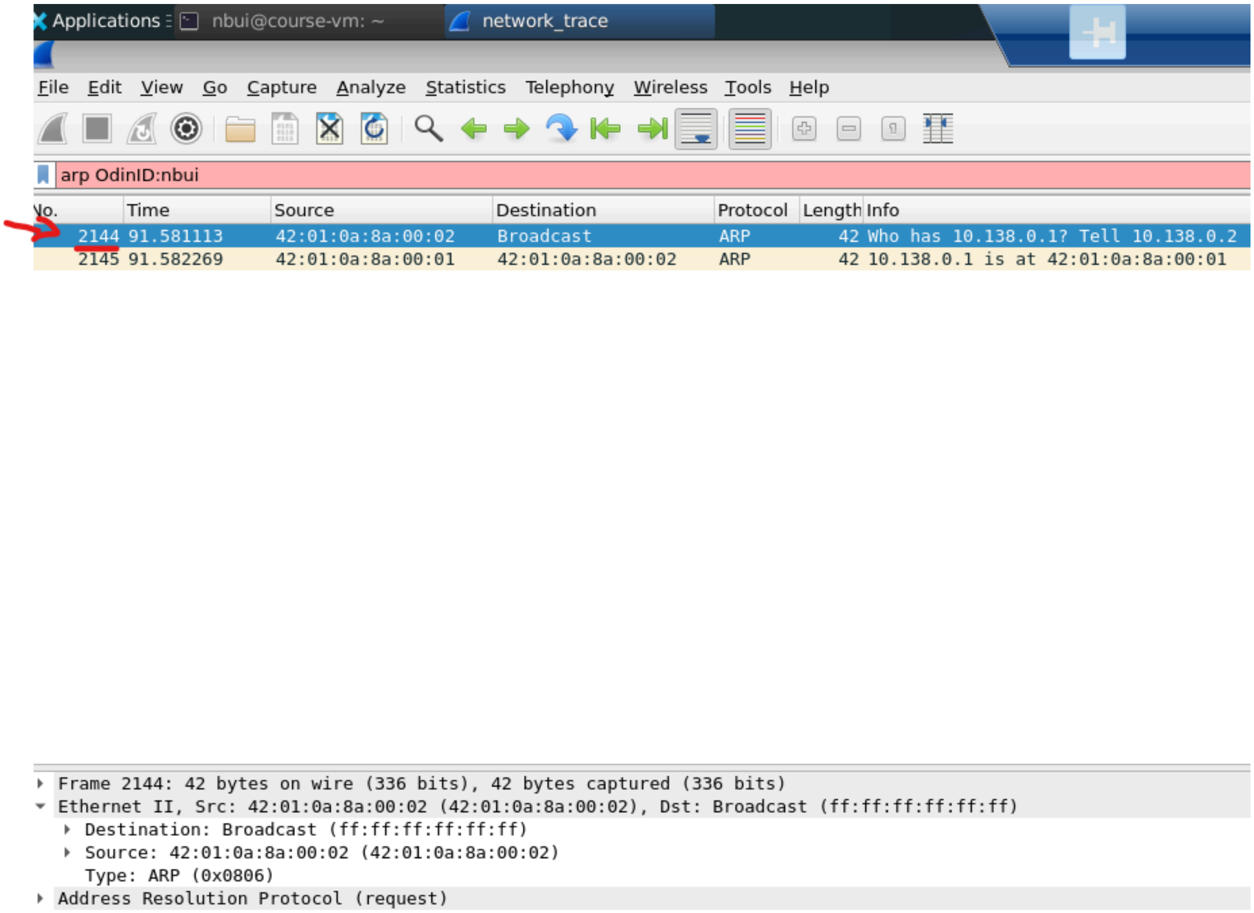
```
arp or dns or http or ip.addr==35.233.233.233 OdinID=nbui
```

```
No.   Time      Source             Destination        Protocol Length Info
  46 10.367737  169.254.169.254    10.138.0.2         HTTP/J…   2198 HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
  48 10.368379  10.138.0.2         169.254.169.254    HTTP       281 GET /computeMetadata/v1/?alt=json&last_etag=7418c7a5dde5c7de&recursive=true&timeout_sec=60&wait_for_change=true HTTP/1.1
  67 15.266035  169.254.169.254    10.138.0.2         HTTP/J…   2198 HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
  69 15.266625  10.138.0.2         169.254.169.254    HTTP       281 GET /computeMetadata/v1/?recursive=true&alt=json&wait_for_change=true&last_etag=7418c7a5dde5c7de&timeout_sec=60 HTTP/1.1
1768 70.385835  169.254.169.254    10.138.0.2         HTTP/J…   2198 HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
1770 70.386583  10.138.0.2         169.254.169.254    HTTP       281 GET /computeMetadata/v1/?alt=json&last_etag=7418c7a5dde5c7de&recursive=true&timeout_sec=60&wait_for_change=true HTTP/1.1
1853 75.283523  169.254.169.254    10.138.0.2         HTTP/J…   2198 HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
1855 75.284105  10.138.0.2         169.254.169.254    HTTP       281 GET /computeMetadata/v1/?recursive=true&alt=json&wait_for_change=true&last_etag=7418c7a5dde5c7de&timeout_sec=60 HTTP/1.1
2139 91.545684  10.138.0.2         169.254.169.254    DNS        106 Standard query 0x7d98 A course-vm.c.cloud-bui-nbui.internal OPT
2140 91.545993  10.138.0.2         169.254.169.254    DNS        106 Standard query 0xba38 AAAA course-vm.c.cloud-bui-nbui.internal OPT
2141 91.551746  169.254.169.254    10.138.0.2         DNS        122 Standard query response 0x7d98 A course-vm.c.cloud-bui-nbui.internal A 10.138.0.2 OPT
2143 91.557863  169.254.169.254    10.138.0.2         DNS        195 Standard query response 0xba38 AAAA course-vm.c.cloud-bui-nbui.internal SOA ns.global.gcedns-prod.internal OPT
2144 91.581113  42:01:0a:8a:00:02  Broadcast          ARP         42 Who has 10.138.0.1? Tell 10.138.0.2
2145 91.582269  42:01:0a:8a:00:01  42:01:0a:8a:00:02  ARP         42 10.138.0.1 is at 42:01:0a:8a:00:01
2146 91.582285  10.138.0.2         169.254.169.254    DNS         89 Standard query 0x50a6 AAAA nbui.oregonctf.org OPT
2156 91.658219  169.254.169.254    10.138.0.2         DNS        171 Standard query response 0x50a6 AAAA nbui.oregonctf.org SOA ns-cloud-d1.googledomains.com OPT
2157 91.658789  10.138.0.2         35.233.233.233     TCP         74 39278 → 80 [SYN] Seq=0 Win=65320 Len=0 MSS=1420 SACK_PERM=1 TSval=1851986396 TSecr=0 WS=128
2158 91.660389  35.233.233.233     10.138.0.2         TCP         74 80 → 39278 [SYN, ACK] Seq=0 Ack=1 Win=64768 Len=0 MSS=1420 SACK_PERM=1 TSval=3232229335 TSecr=1851986396 WS=128
2159 91.660428  10.138.0.2         35.233.233.233     TCP         66 39278 → 80 [ACK] Seq=1 Ack=1 Win=65408 Len=0 TSval=1851986398 TSecr=3232229335
2160 91.660523  10.138.0.2         35.233.233.233     HTTP       199 GET / HTTP/1.1
2161 91.660869  35.233.233.233     10.138.0.2         TCP         66 80 → 39278 [ACK] Seq=1 Ack=134 Win=64640 Len=0 TSval=3232229336 TSecr=1851986398
2162 91.661162  35.233.233.233     10.138.0.2         TCP       7106 80 → 39278 [PSH, ACK] Seq=1 Ack=134 Win=64640 Len=7040 TSval=3232229336 TSecr=1851986398 [TCP segment of a reassembled PDU]
2163 91.661162  35.233.233.233     10.138.0.2         HTTP       792 HTTP/1.1 200 OK  (text/html)
```

```
▸ Frame 46: 2198 bytes on wire (17584 bits), 2198 bytes captured (17584 bits)
▸ Ethernet II, Src: 42:01:0a:8a:00:01 (42:01:0a:8a:00:01), Dst: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02)
▸ Internet Protocol Version 4, Src: 169.254.169.254, Dst: 10.138.0.2
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 59852, Seq: 1, Ack: 1, Len: 2144
▸ Hypertext Transfer Protocol
▸ JavaScript Object Notation: application/json
```

ARP

- What packet numbers in the trace are the result of the VM attempting to get the hardware address of the default router?

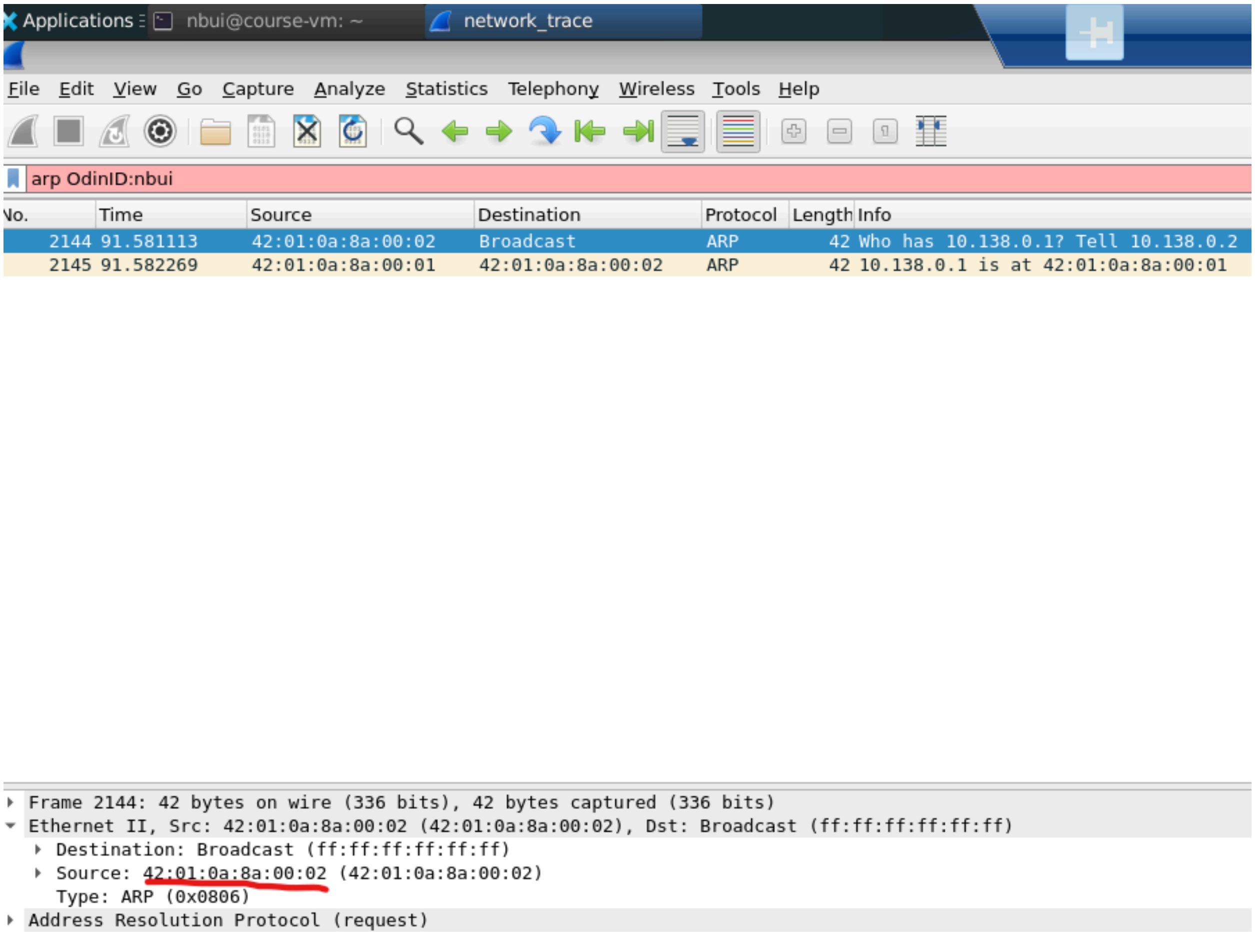


arp OdinID:nbui

```
No.   Time       Source             Destination        Protocol Length Info
2144 91.581113  42:01:0a:8a:00:02  Broadcast          ARP        42 Who has 10.138.0.1? Tell 10.138.0.2
2145 91.582269  42:01:0a:8a:00:01  42:01:0a:8a:00:02  ARP        42 10.138.0.1 is at 42:01:0a:8a:00:01
```

```
▸ Frame 2144: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
▾ Ethernet II, Src: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  ▸ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  ▸ Source: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02)
    Type: ARP (0x0806)
▸ Address Resolution Protocol (request)
```

- What is this hardware address?


Applications
nbui@course-vm: ~
network_trace
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

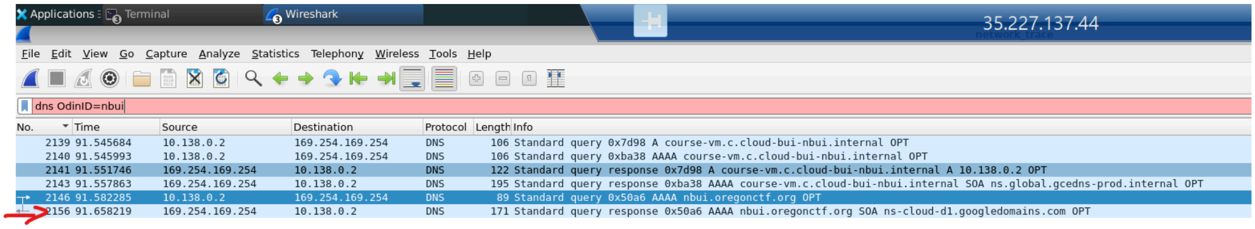
arp OdinID:nbui

```
No.   Time       Source             Destination        Protocol Length Info
 2144 91.581113  42:01:0a:8a:00:02  Broadcast          ARP      42 Who has 10.138.0.1? Tell 10.138.0.2
 2145 91.582269  42:01:0a:8a:00:01  42:01:0a:8a:00:02  ARP      42 10.138.0.1 is at 42:01:0a:8a:00:01
```

```
▸ Frame 2144: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
▾ Ethernet II, Src: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  ▸ Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  ▸ Source: 42:01:0a:8a:00:02 (42:01:0a:8a:00:02)
    Type: ARP (0x0806)
▸ Address Resolution Protocol (request)
```

DNS

- What packet numbers in the trace correspond to the DNS request for the web site?


Applications
Terminal
Wireshark
35.227.137.44
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

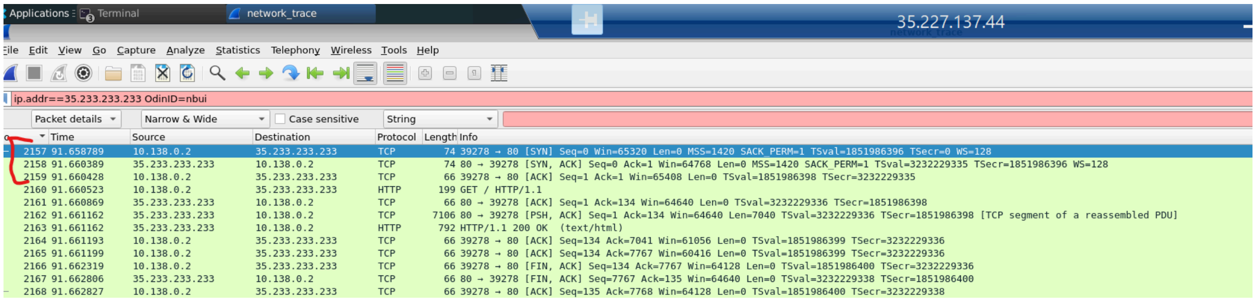
dns OdinID=nbui

```
No.   Time       Source           Destination      Protocol Length Info
 2139 91.545684  10.138.0.2       169.254.169.254  DNS      106 Standard query 0x7d98 A course-vm.c.cloud-bui-nbui.internal OPT
 2140 91.545993  10.138.0.2       169.254.169.254  DNS      106 Standard query 0xba38 AAAA course-vm.c.cloud-bui-nbui.internal OPT
 2141 91.551746  169.254.169.254  10.138.0.2       DNS      122 Standard query response 0x7d98 A course-vm.c.cloud-bui-nbui.internal A 10.138.0.2 OPT
 2143 91.557863  169.254.169.254  10.138.0.2       DNS      195 Standard query response 0xba38 AAAA course-vm.c.cloud-bui-nbui.internal SOA ns.global.gcedns-prod.internal OPT
 2146 91.582285  10.138.0.2       169.254.169.254  DNS       89 Standard query 0x50a6 AAAA nbui.oregonctf.org OPT
 2156 91.658219  169.254.169.254  10.138.0.2       DNS      171 Standard query response 0x50a6 AAAA nbui.oregonctf.org SOA ns-cloud-d1.googledomains.com OPT
```

-
- What is the IP address of the local DNS server being queried? 127.0.0.53

TCP

- What packet numbers in the trace correspond to the initial TCP handshake for the web request?



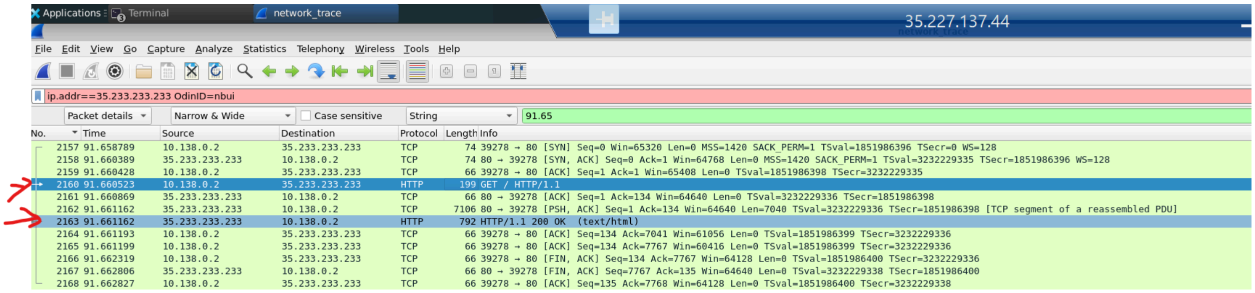
```
ip.addr==35.233.233.233 OdinID=nbui
```

```
No.  Time      Source          Destination     Protocol Length Info
2157 91.658789 10.138.0.2      35.233.233.233  TCP      74 39278 → 80 [SYN] Seq=0 Win=65320 Len=0 MSS=1420 SACK_PERM=1 TSval=1851986396 TSecr=0 WS=128
2158 91.660389 35.233.233.233  10.138.0.2      TCP      74 80 → 39278 [SYN, ACK] Seq=0 Ack=1 Win=64768 Len=0 MSS=1420 SACK_PERM=1 TSval=3232229335 TSecr=1851986396 WS=128
2159 91.660428 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=1 Ack=1 Win=65408 Len=0 TSval=1851986398 TSecr=3232229335
2160 91.660523 10.138.0.2      35.233.233.233  HTTP    199 GET / HTTP/1.1
2161 91.660869 35.233.233.233  10.138.0.2      TCP      66 80 → 39278 [ACK] Seq=1 Ack=134 Win=64640 Len=0 TSval=3232229336 TSecr=1851986398
2162 91.661162 35.233.233.233  10.138.0.2      TCP    7106 80 → 39278 [PSH, ACK] Seq=1 Ack=134 Win=64640 Len=7040 TSval=3232229336 TSecr=1851986398 [TCP segment of a reassembled PDU]
2163 91.661162 35.233.233.233  10.138.0.2      HTTP    792 HTTP/1.1 200 OK  (text/html)
2164 91.661193 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=134 Ack=7041 Win=61056 Len=0 TSval=1851986399 TSecr=3232229336
2165 91.661199 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=134 Ack=7767 Win=60416 Len=0 TSval=1851986399 TSecr=3232229336
2166 91.662319 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [FIN, ACK] Seq=134 Ack=7767 Win=64128 Len=0 TSval=1851986400 TSecr=3232229336
2167 91.662806 35.233.233.233  10.138.0.2      TCP      66 80 → 39278 [FIN, ACK] Seq=7767 Ack=135 Win=64640 Len=0 TSval=3232229338 TSecr=1851986400
2168 91.662827 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=135 Ack=7768 Win=64128 Len=0 TSval=1851986400 TSecr=3232229338
```

- How long does it take to perform the initial TCP handshake? Time=91.660389−91.658789 = 0.0016

HTTP

- What packet numbers in the trace correspond to the actual HTTP request and response?



```
ip.addr==35.233.233.233 OdinID=nbui
```

```
91.65
```

```
No.  Time      Source          Destination     Protocol Length Info
2157 91.658789 10.138.0.2      35.233.233.233  TCP      74 39278 → 80 [SYN] Seq=0 Win=65320 Len=0 MSS=1420 SACK_PERM=1 TSval=1851986396 TSecr=0 WS=128
2158 91.660389 35.233.233.233  10.138.0.2      TCP      74 80 → 39278 [SYN, ACK] Seq=0 Ack=1 Win=64768 Len=0 MSS=1420 SACK_PERM=1 TSval=3232229335 TSecr=1851986396 WS=128
2159 91.660428 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=1 Ack=1 Win=65408 Len=0 TSval=1851986398 TSecr=3232229335
2160 91.660523 10.138.0.2      35.233.233.233  HTTP    199 GET / HTTP/1.1
2161 91.660869 35.233.233.233  10.138.0.2      TCP      66 80 → 39278 [ACK] Seq=1 Ack=134 Win=64640 Len=0 TSval=3232229336 TSecr=1851986398
2162 91.661162 35.233.233.233  10.138.0.2      TCP    7106 80 → 39278 [PSH, ACK] Seq=1 Ack=134 Win=64640 Len=7040 TSval=3232229336 TSecr=1851986398 [TCP segment of a reassembled PDU]
2163 91.661162 35.233.233.233  10.138.0.2      HTTP    792 HTTP/1.1 200 OK  (text/html)
2164 91.661193 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=134 Ack=7041 Win=61056 Len=0 TSval=1851986399 TSecr=3232229336
2165 91.661199 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=134 Ack=7767 Win=60416 Len=0 TSval=1851986399 TSecr=3232229336
2166 91.662319 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [FIN, ACK] Seq=134 Ack=7767 Win=64128 Len=0 TSval=1851986400 TSecr=3232229336
2167 91.662806 35.233.233.233  10.138.0.2      TCP      66 80 → 39278 [FIN, ACK] Seq=7767 Ack=135 Win=64640 Len=0 TSval=3232229338 TSecr=1851986400
2168 91.662827 10.138.0.2      35.233.233.233  TCP      66 39278 → 80 [ACK] Seq=135 Ack=7768 Win=64128 Len=0 TSval=1851986400 TSecr=3232229338
```

- How long does it take to process the HTTP request after the handshake? Time=91.661162−91.660523 = 0.000639