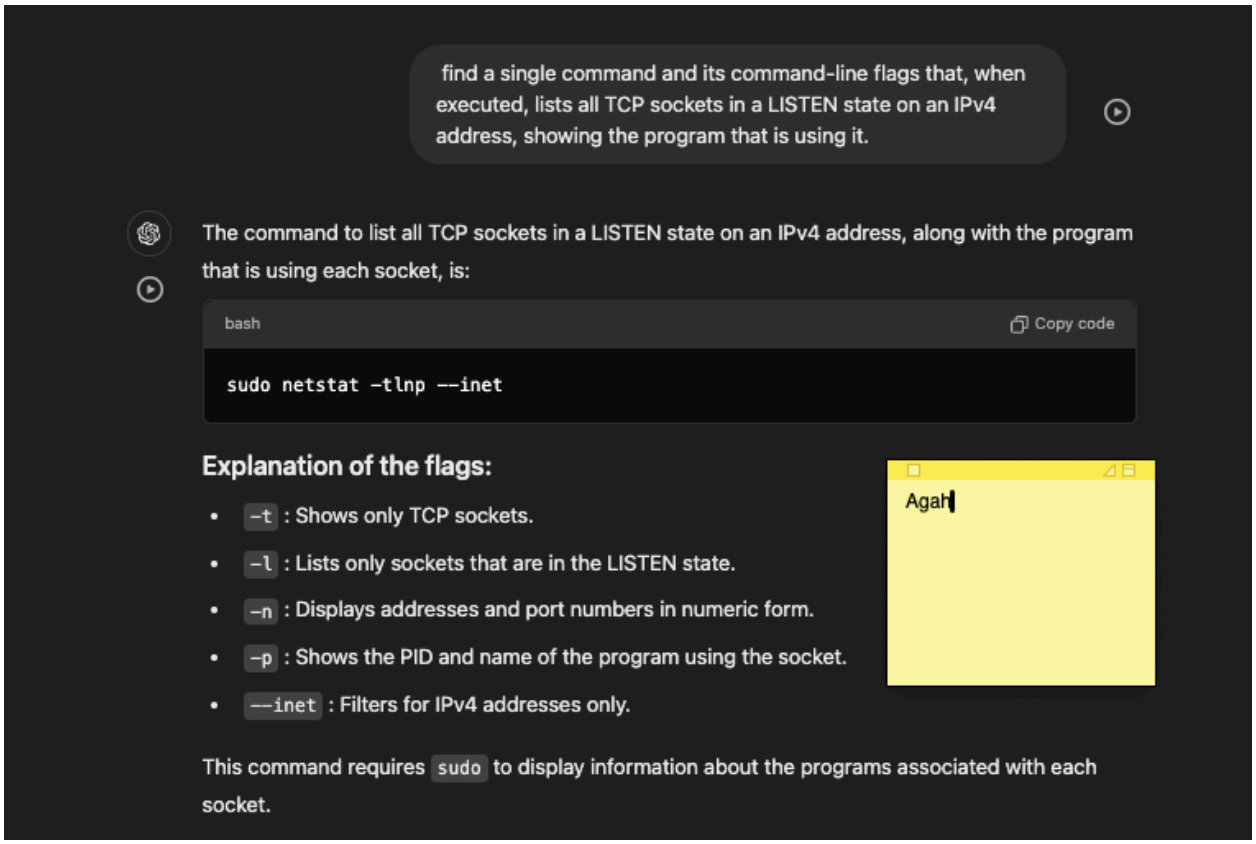
AHMAD ZAFAR AGAH WEEK2 10/9/24

2.1 TCP, HTTP

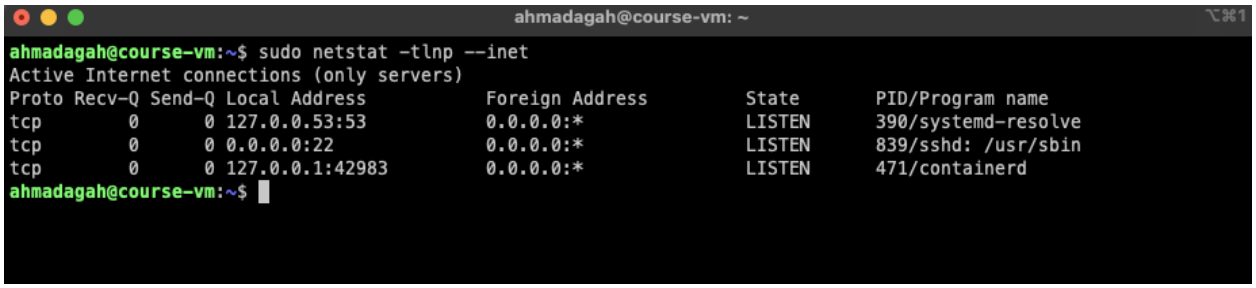
2.1.1 TCP #1: Sockets

ss, netstat

- **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.**



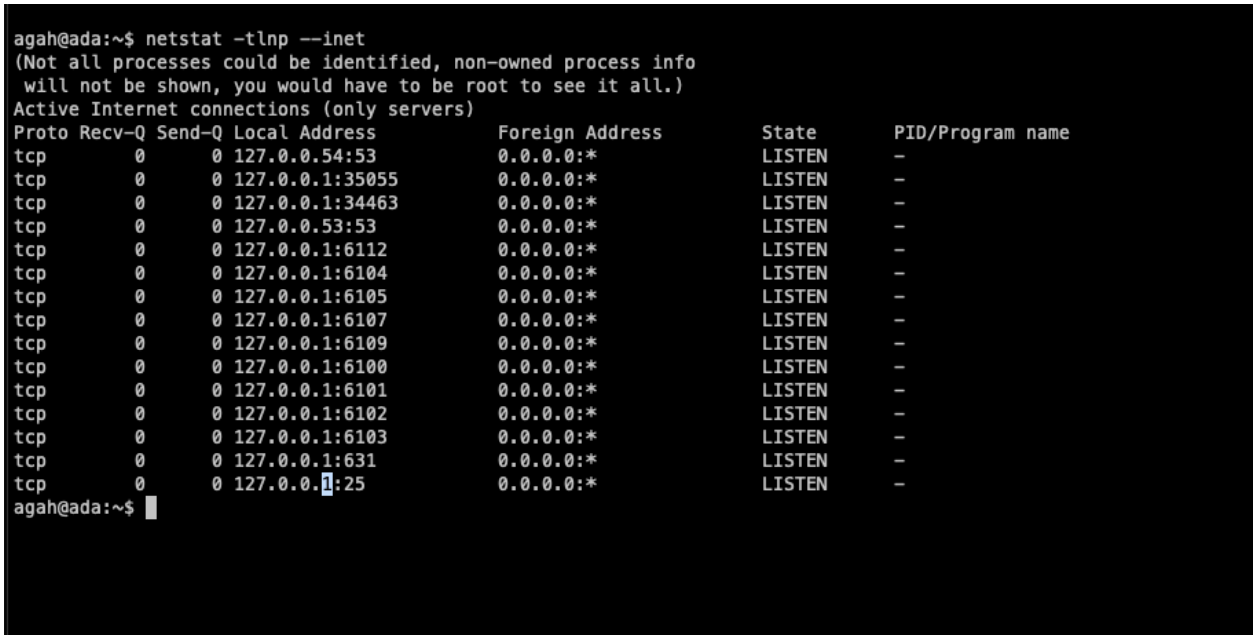
- **List a service that can be contacted from any interface on the machine. List a service that can only be contacted by local processes.**

The sshd service (Secure Shell Daemon) is listening on `0.0.0.0:22`. The IP `0.0.0.0` indicates that it is bound to all available network interfaces, meaning it can be contacted from any network interface on the machine, both local and external.

The systemd-resolve service is listening on `127.0.0.53:53`. The IP `127.0.0.53` is within the `127.0.0.0/8` loopback range, which means it is only accessible from local processes on the same machine and cannot be accessed externally.

Similarly, the containerd service is listening on `127.0.0.1:42983`, another loopback address, which also restricts it to local processes only.

- **Run the command again, but do not use `sudo` as this is a machine managed by CAT. Include a screenshot of the output.**

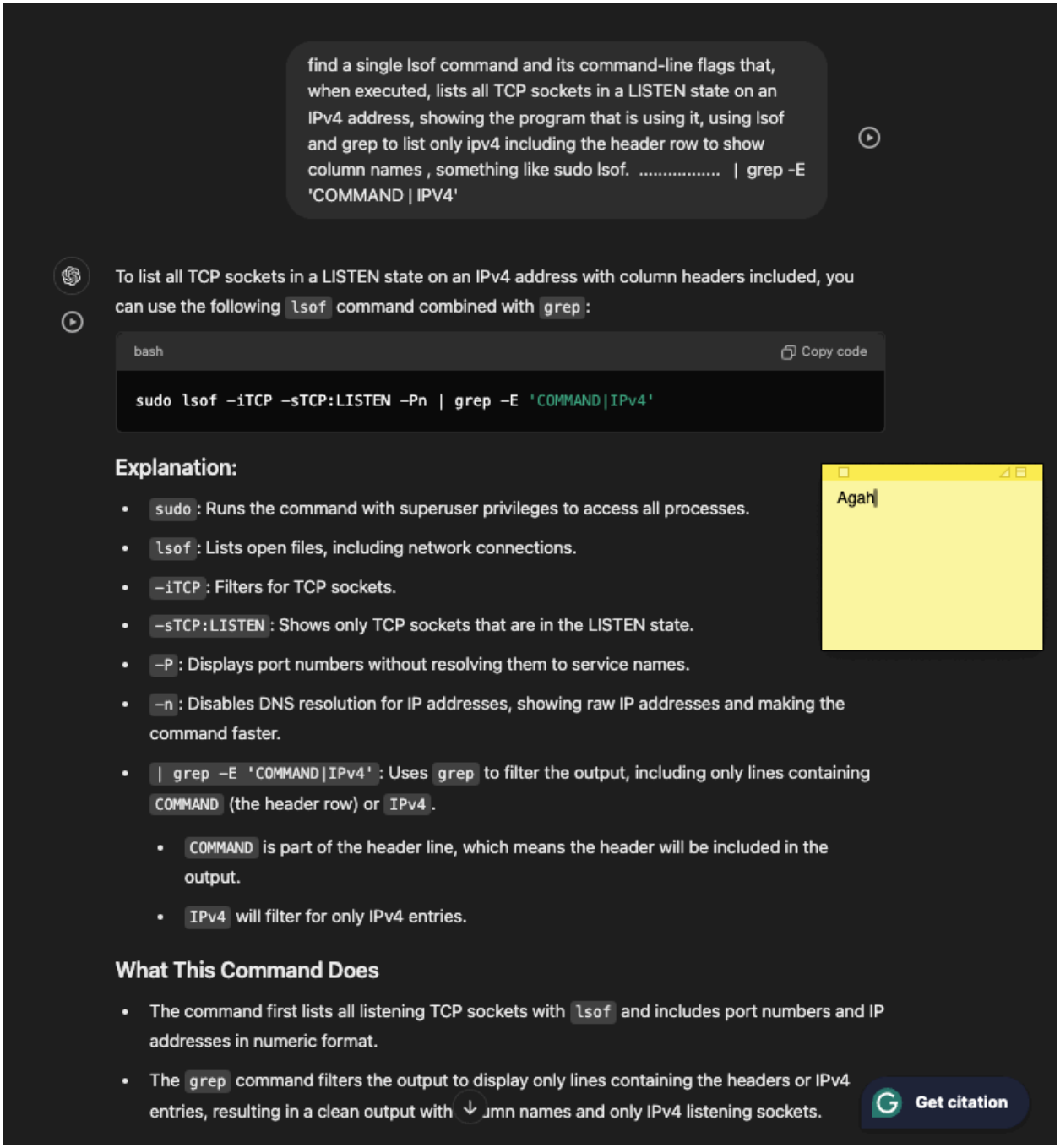


- **List the services that this machine provides for external access**

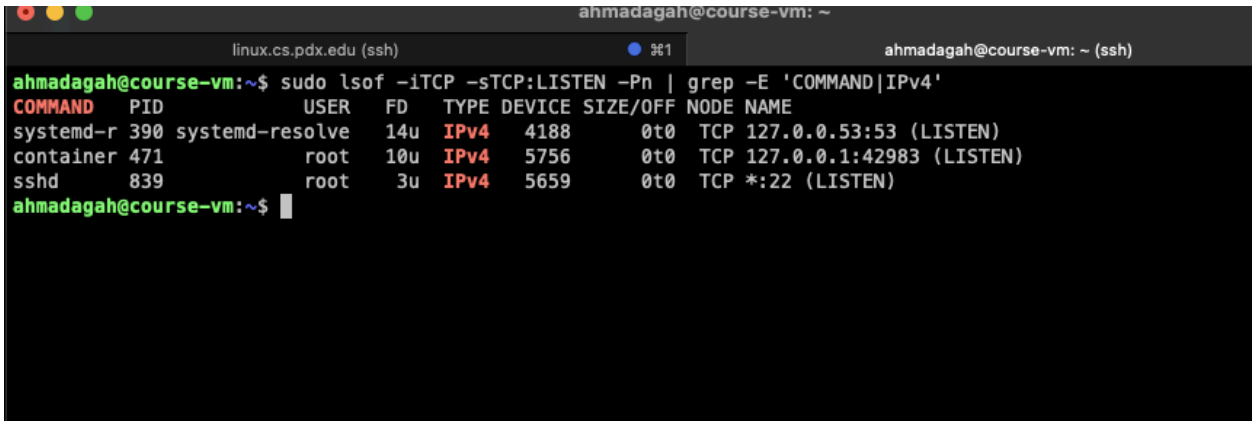
none of the listed services are available for external access, all services shown are bound to `127.0.0.1` (the loopback address), which means they are only accessible from the local machine.

2.1.2 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.

2.1.3 TCP #2: Throughput

2.1.4 iperf

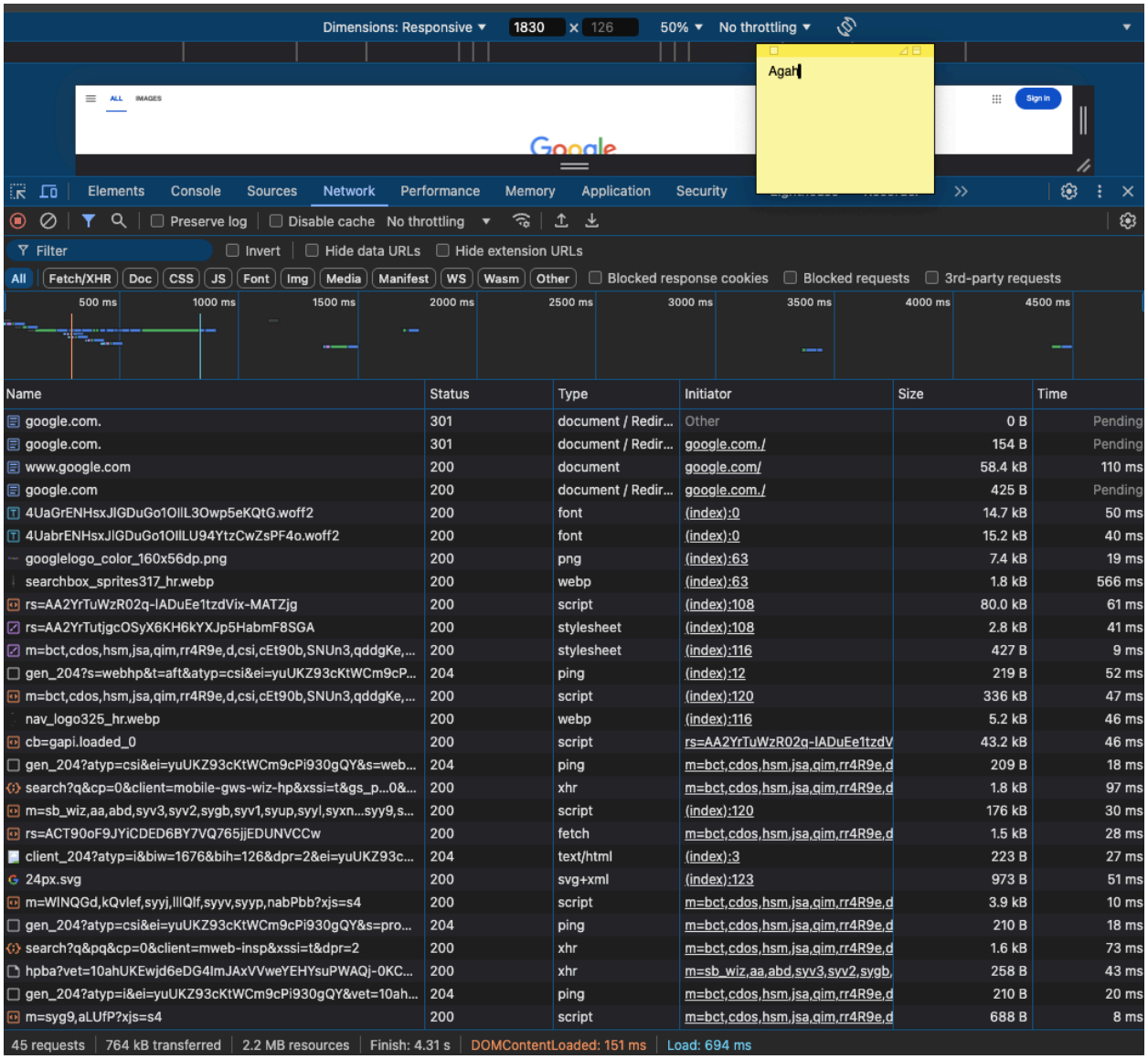
- **Show a screenshot of the measured bandwidth available between your us-west1-b VM and each of the other Compute Engine VMs. Explain the relative differences (or lack thereof) in your results.**

```
agah@west1-b:~$ sudo iperf -c 10.152.0.2 -p 80
------------------------------------------------------------
Client connecting to 10.152.0.2, TCP port 80
TCP window size: 16.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.7 port 52162 connected with 10.152.0.2 port 80 (icwnd/mss/irtt=13/1408/139085)
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.2317 sec   183 MBytes   150 Mbits/sec
agah@west1-b:~$ sudo iperf -c 10.142.0.3 -p 80
------------------------------------------------------------
Client connecting to 10.142.0.3, TCP port 80
TCP window size: 16.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.7 port 38528 connected with 10.142.0.3 port 80 (icwnd/mss/irtt=13/1408/66240)
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.1041 sec   412 MBytes   342 Mbits/sec
agah@west1-b:~$ sudo iperf -c 10.132.0.2 -p 80
------------------------------------------------------------
Client connecting to 10.132.0.2, TCP port 80
TCP window size: 16.0 KByte (default)
------------------------------------------------------------
[  1] local 10.138.0.7 port 59910 connected with 10.132.0.2 port 80 (icwnd/mss/irtt=13/1408/135867)
[ ID] Interval       Transfer     Bandwidth
[  1] 0.0000-10.2021 sec   195 MBytes   160 Mbits/sec
agah@west1-b:~$
```

VM at `10.142.0.3` has the highest bandwidth, possibly indicating that it is either on the same network segment or has fewer competing network resources compared to the others, VMs at `10.152.0.2` and `10.132.0.2` have lower bandwidths, potentially due to being located farther away, experiencing more network congestion, or having less optimal network paths.

2.1.5 HTTP #3: Requests

- **Take a screenshot of the initial requests for your lab notebook.**



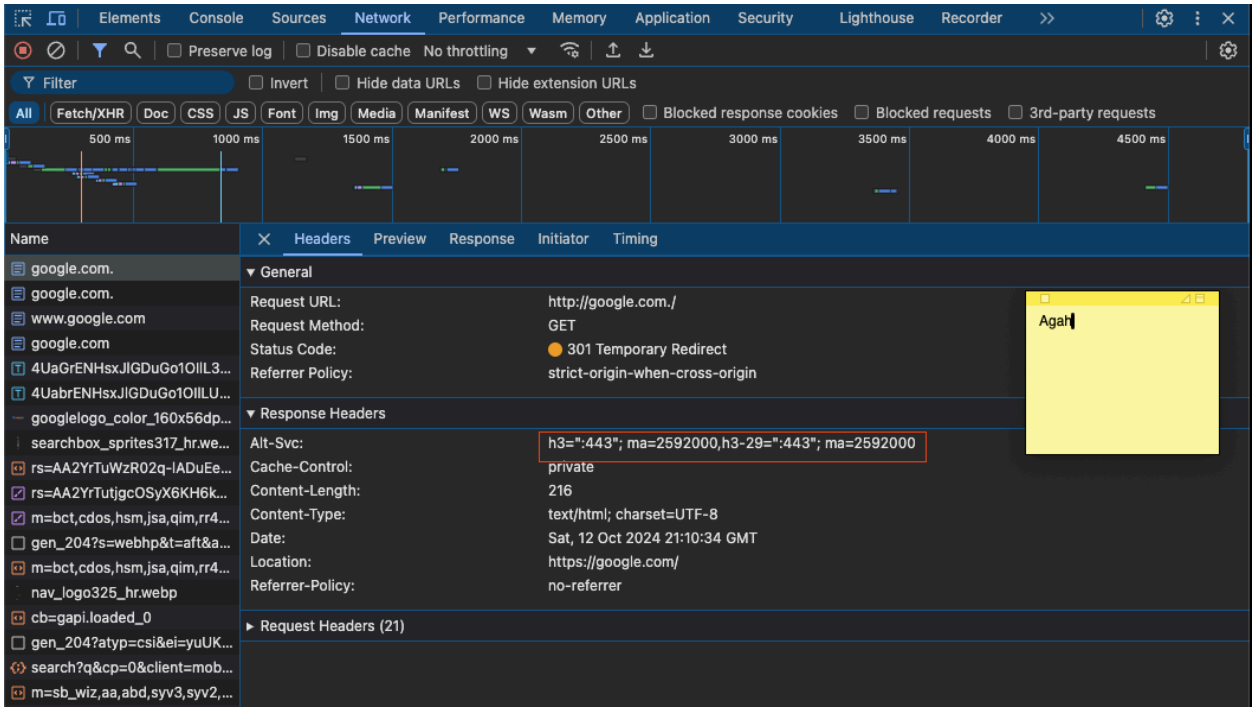
- **What is the URL being requested?**

  `http://google.com./`

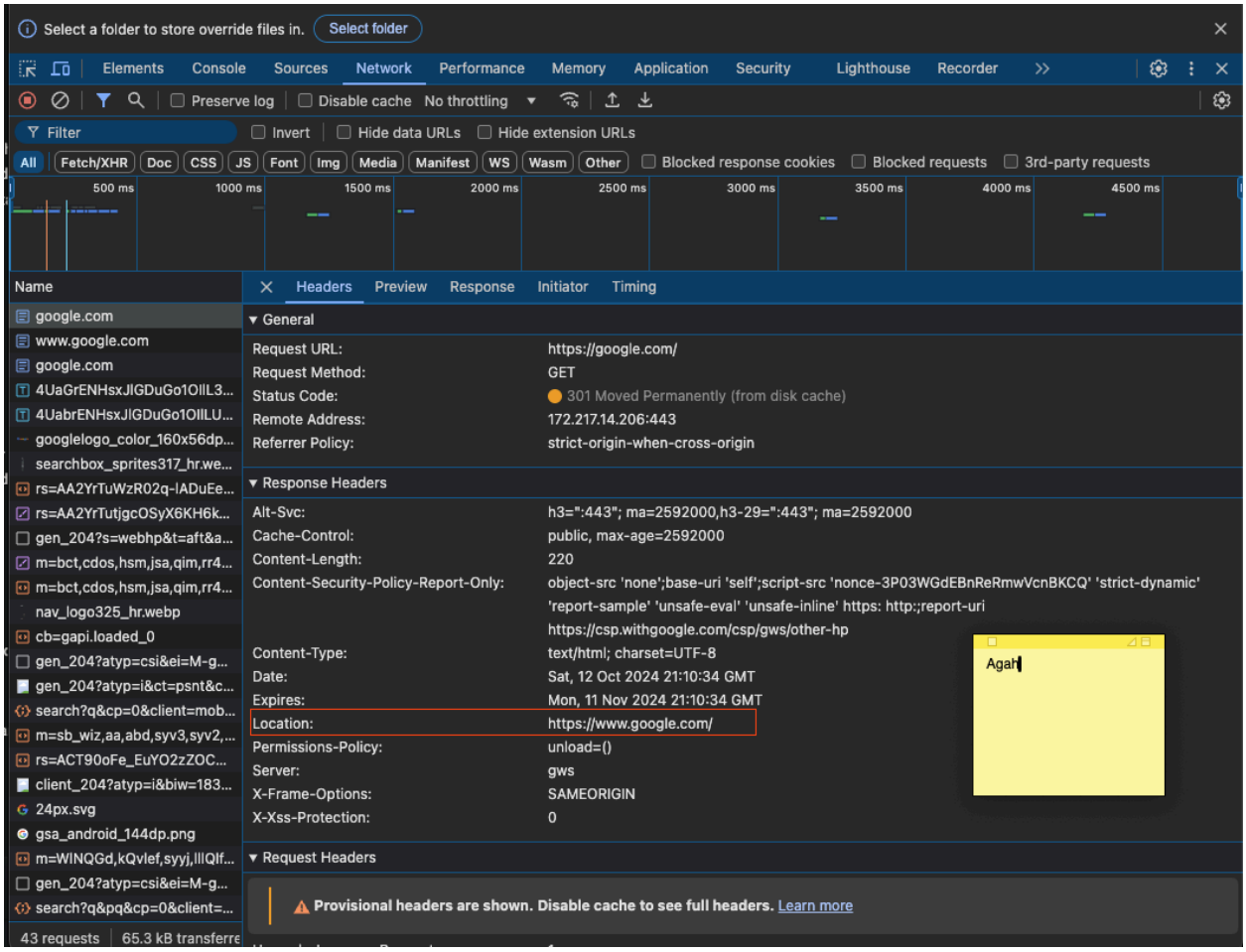
- **Explain the HTTP status code that is returned and what the code indicates**

  301 Moved Permanently: This status code indicates that the requested resource has been permanently moved to a different URL. For example, http://google.com redirects to https://www.google.com. The `301` status code instructs the browser to make future requests to the new URL.

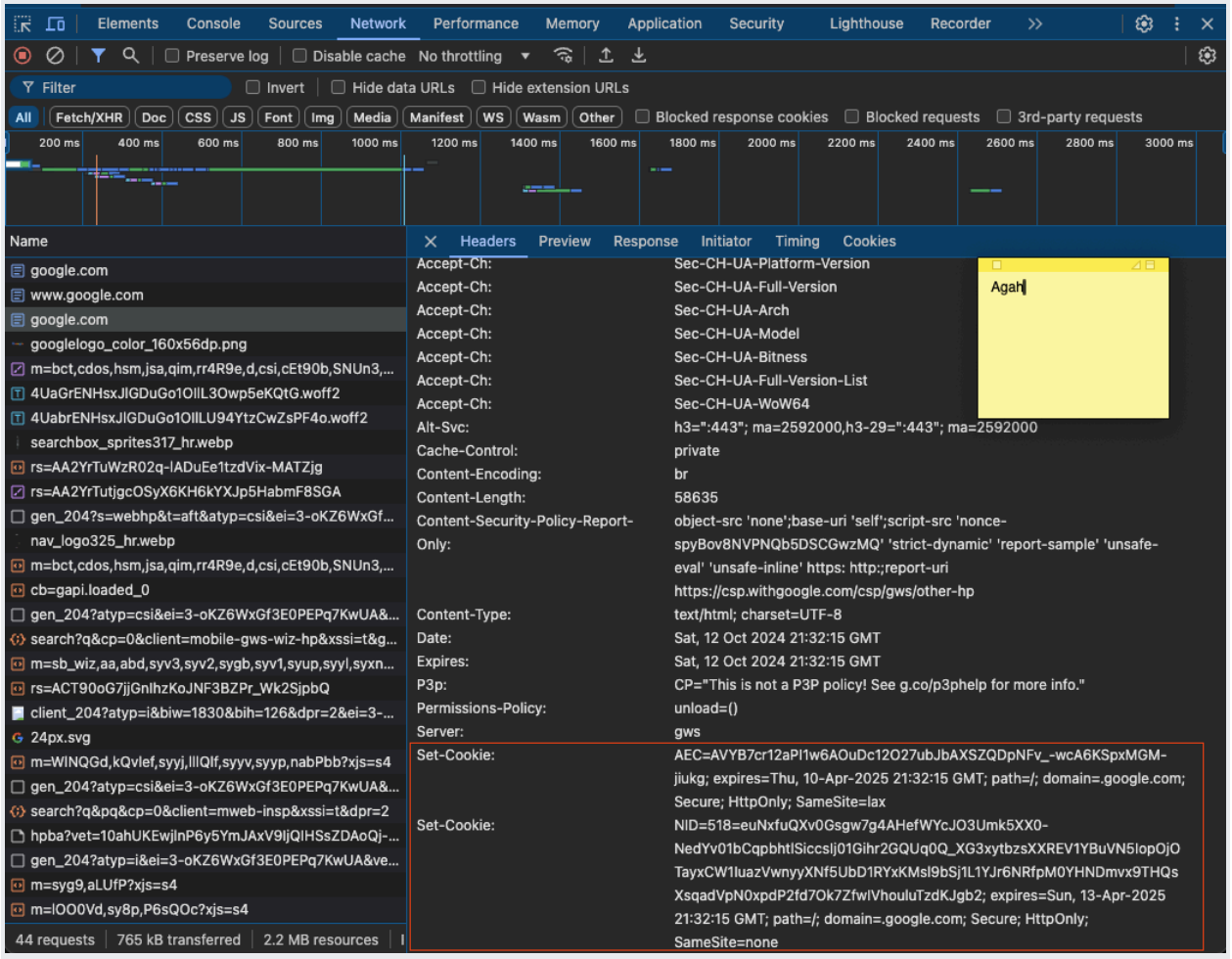
- **Take a screenshot indicating the version of the HTTP protocol that is used for each request. (Hint: look at the response status line and `alt-svc:` HTTP response headers indicating HTTP/2 or HTTP/3).**



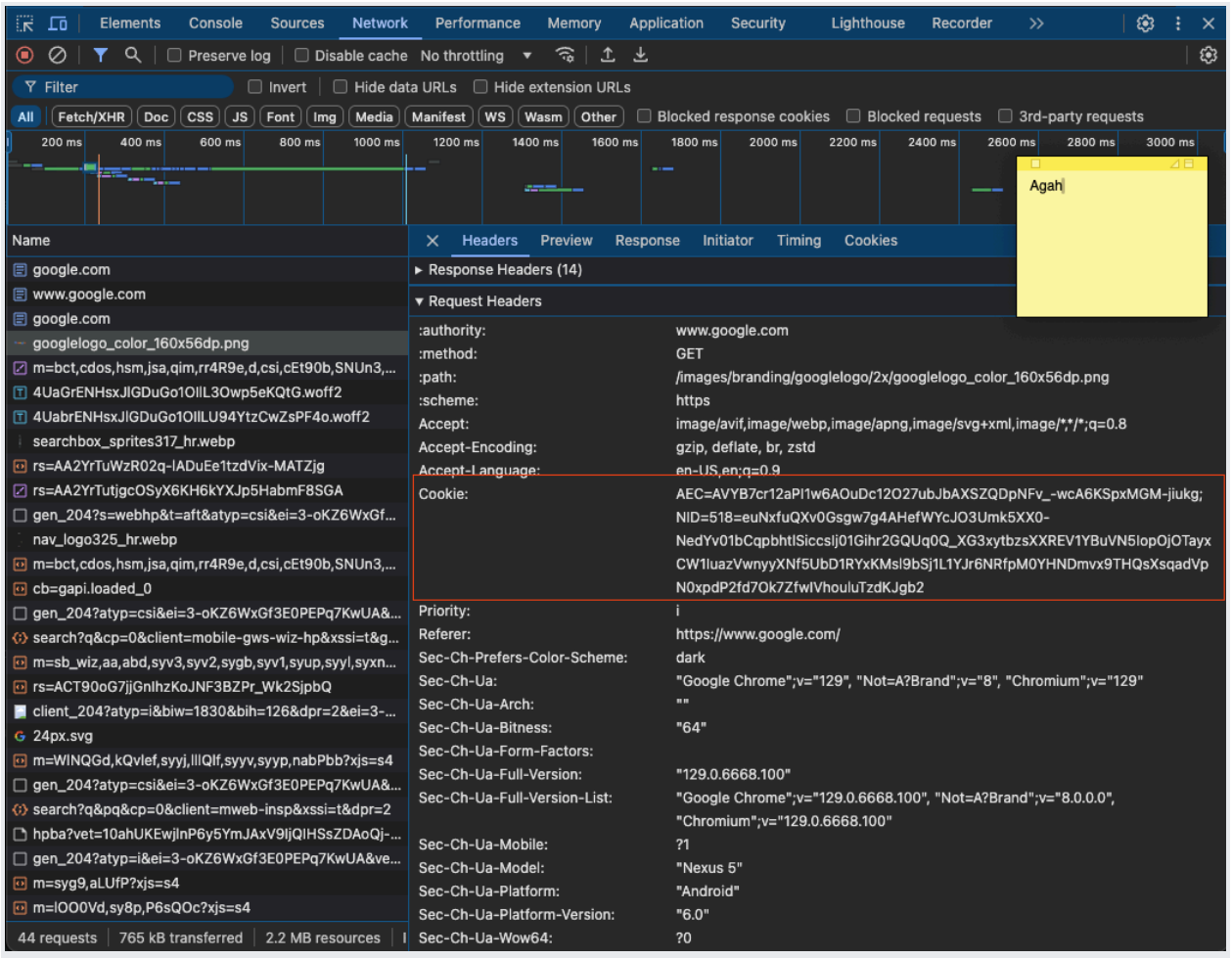
- **Show the URLs the browser is redirected to via this header.**



- **Take a screenshot of when cookies are set via Set-Cookie:**

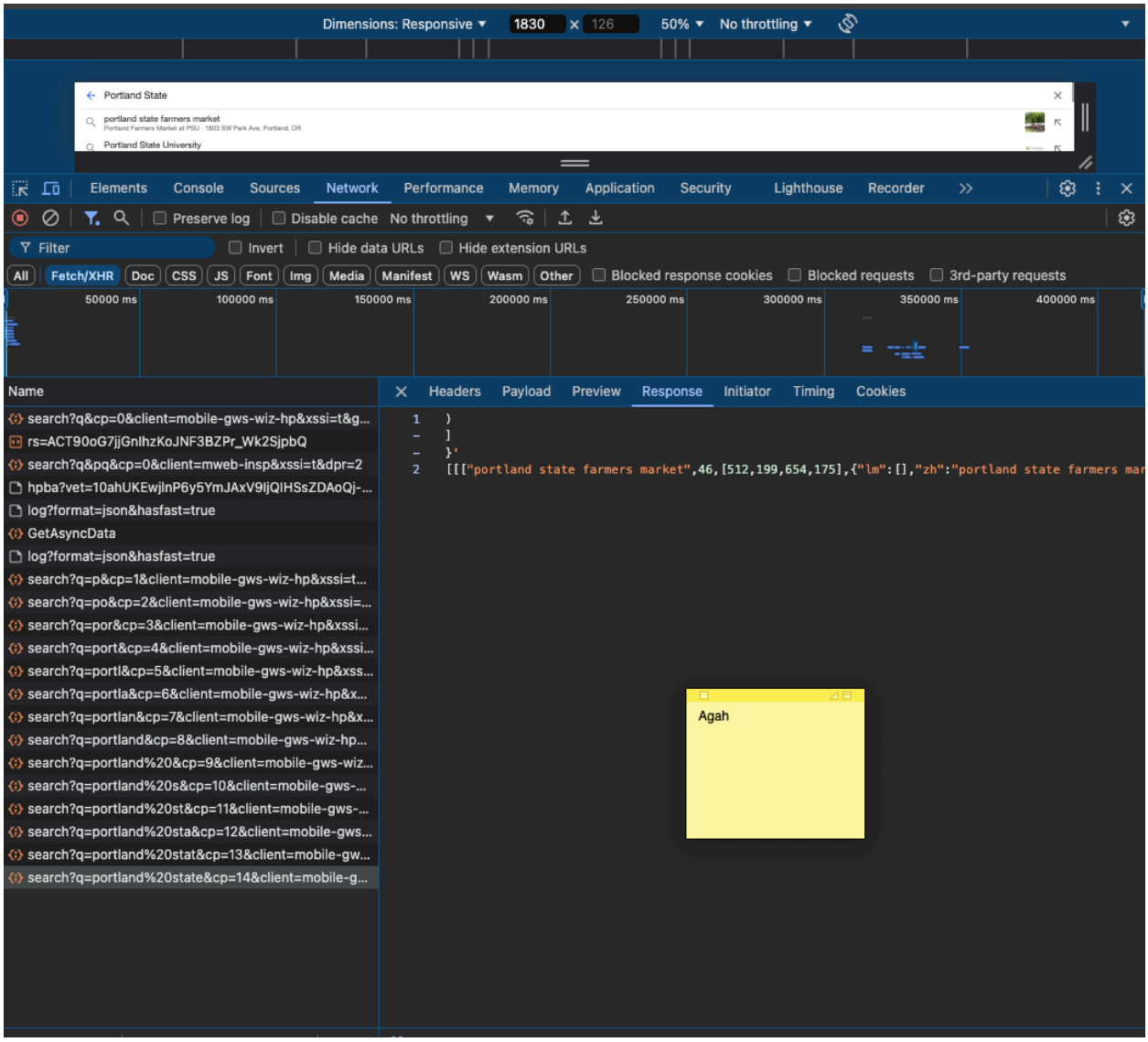


- **Take a screenshot of when cookies are attached via Cookie:**



2.1.6 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"**

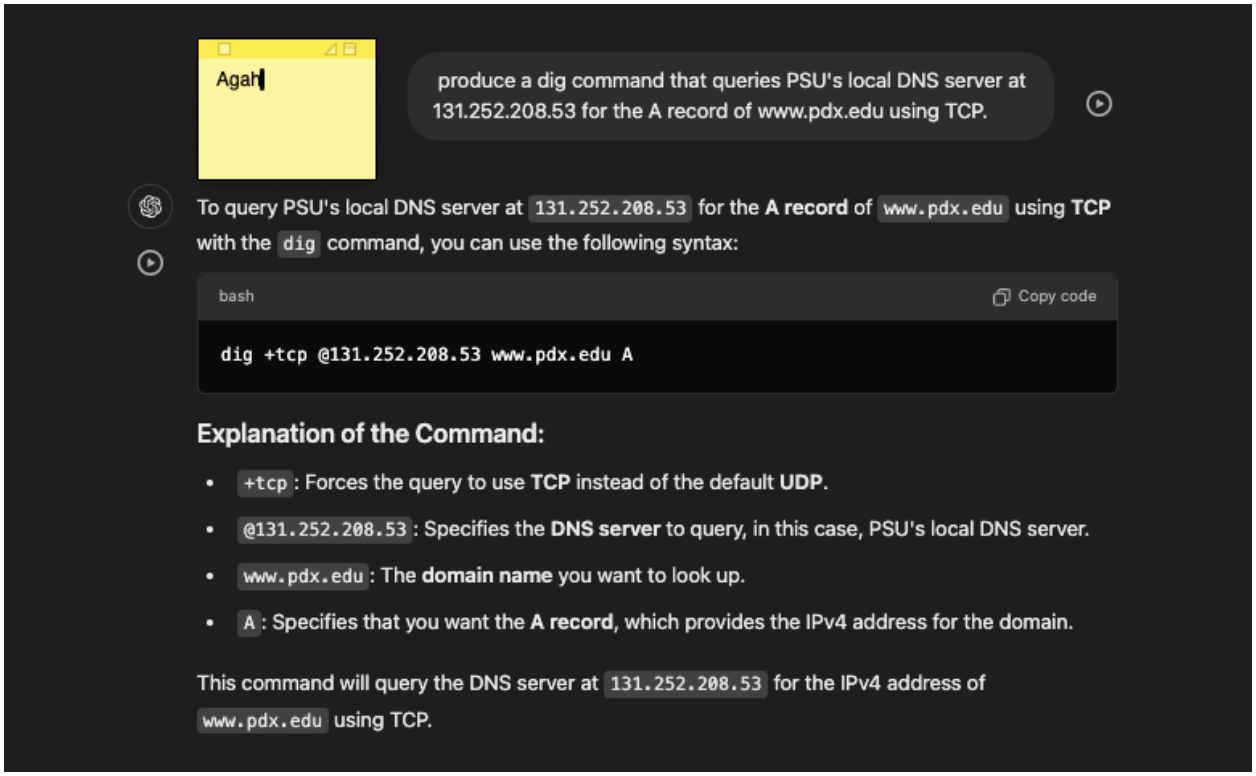


2.2 DNS, Recap

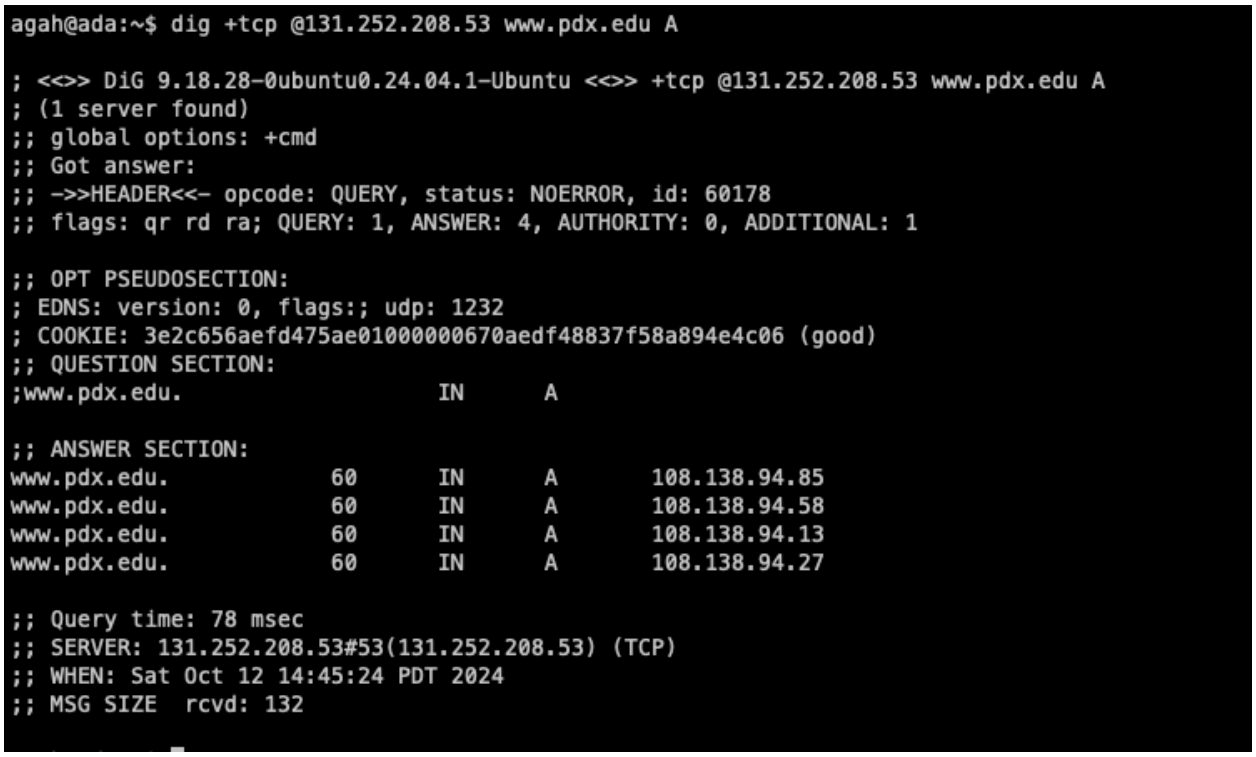
2.2.1 DNS reconnaissance #1 (dig)

pdx.edu

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



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



```
agah@ada:~$ dig @131.252.208.53 pdx.edu MX

; <<>> DiG 9.18.28-0ubuntu0.24.04.1-Ubuntu <<>> @131.252.208.53 pdx.edu MX
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27560
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 140d389359642ca701000000670af4c5fc79bd85048e212a (good)
;; QUESTION SECTION:
;pdx.edu.                       IN      MX

;; ANSWER SECTION:
pdx.edu.                2735    IN      MX      5 alt1.aspmx.l.google.com.
pdx.edu.                2735    IN      MX      10 alt4.aspmx.l.google.com.
pdx.edu.                2735    IN      MX      1 aspmx.l.google.com.
pdx.edu.                2735    IN      MX      5 alt2.aspmx.l.google.com.
pdx.edu.                2735    IN      MX      10 alt3.aspmx.l.google.com.

;; Query time: 2 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (UDP)
;; WHEN: Sat Oct 12 15:14:29 PDT 2024
;; MSG SIZE  rcvd: 182
```

- **What cloud provider hosts the web site for www.pdx.edu?**

  Amazon Web Services (AWS) hosts the website for `www.pdx.edu` through its CloudFront service.

```
agah@ada:~$ whois 108.138.94.85

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2024, American Registry for Internet Numbers, Ltd.
#



# start

NetRange:       108.128.0.0 - 108.139.255.255
CIDR:           108.136.0.0/14, 108.128.0.0/13
NetName:        AMAZO-4
NetHandle:      NET-108-128-0-0-1
Parent:         NET108 (NET-108-0-0-0-0)
NetType:        Direct Allocation
OriginAS:
Organization:   Amazon.com, Inc. (AMAZO-4)
RegDate:        2018-09-18
Updated:        2018-09-18
Ref:            https://rdap.arin.net/registry/ip/108.128.0.0



OrgName:        Amazon.com, Inc.
OrgId:          AMAZO-4
Address:        Amazon Web Services, Inc.
Address:        P.O. Box 81226
City:           Seattle
StateProv:      WA
PostalCode:     98108-1226
Country:        US
RegDate:        2005-09-29
Updated:        2022-09-30
Comment:        For details of this service please see
Comment:        http://ec2.amazonaws.com
Ref:            https://rdap.arin.net/registry/entity/AMAZO-4
```

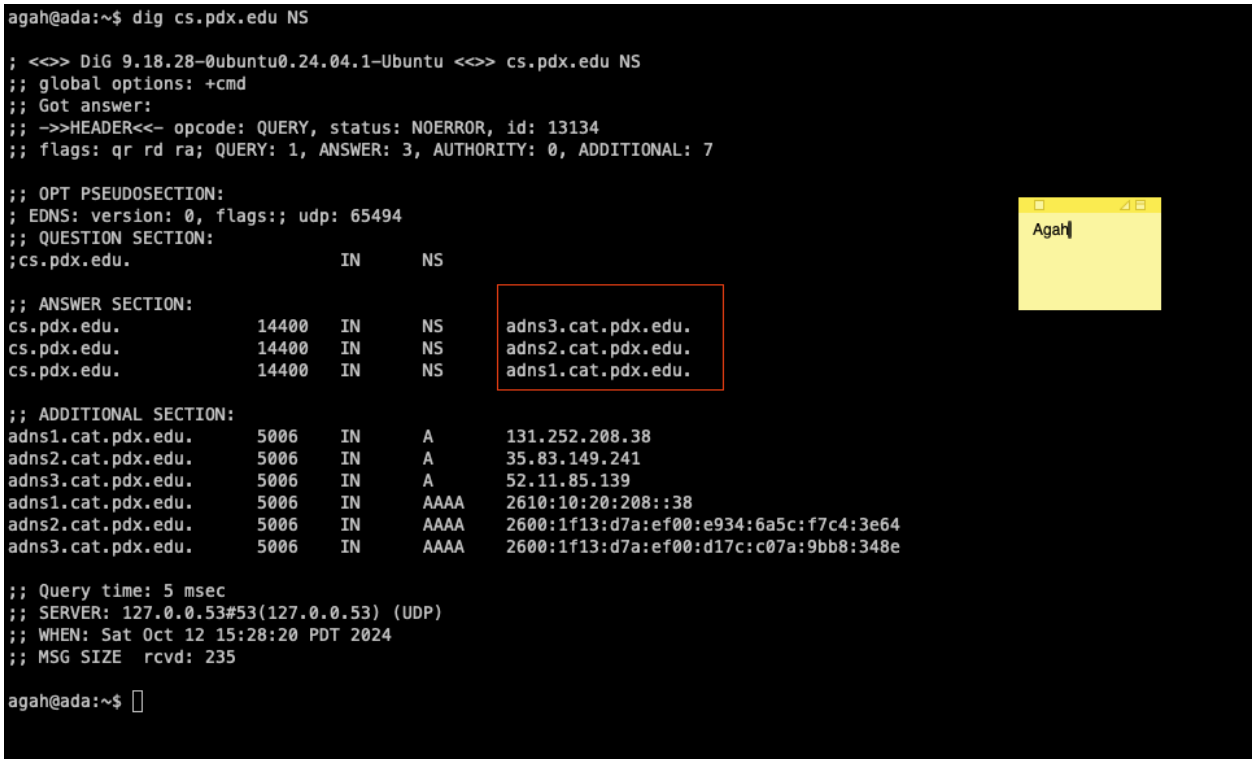
- **What cloud provider handles mail for pdx.edu?**

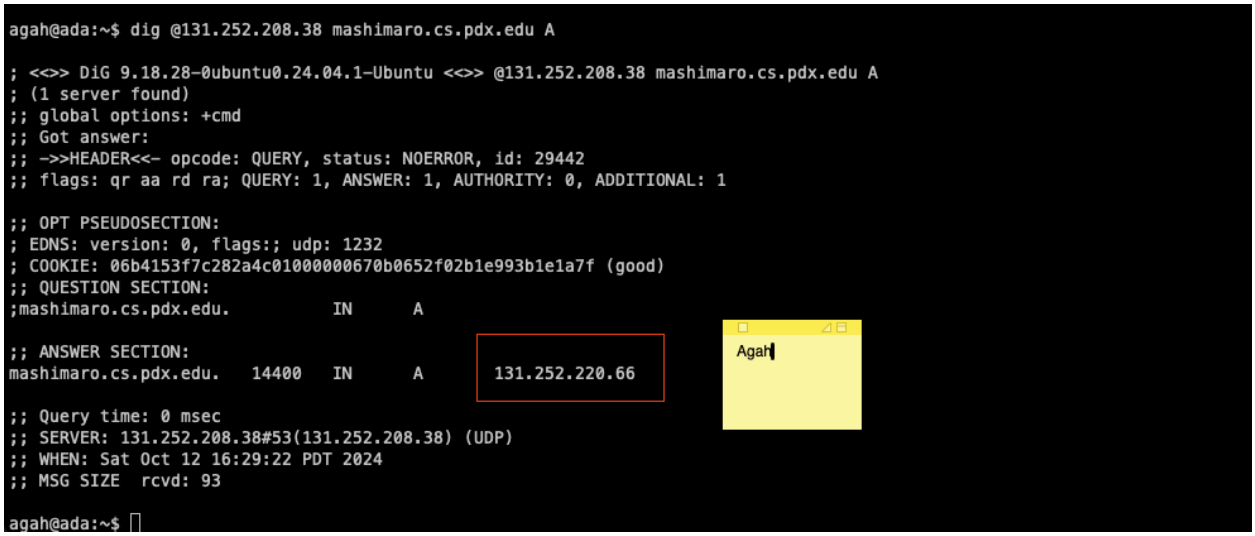
  The MX record query for `pdx.edu` returned mail servers such as `aspmx.l.google.com`, `alt1.aspmx.l.google.com`, etc.

These servers are part of Google Workspace (formerly G Suite), which indicates that **Google handles email for `pdx.edu`.**

mashimaro.cs.pdx.edu

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

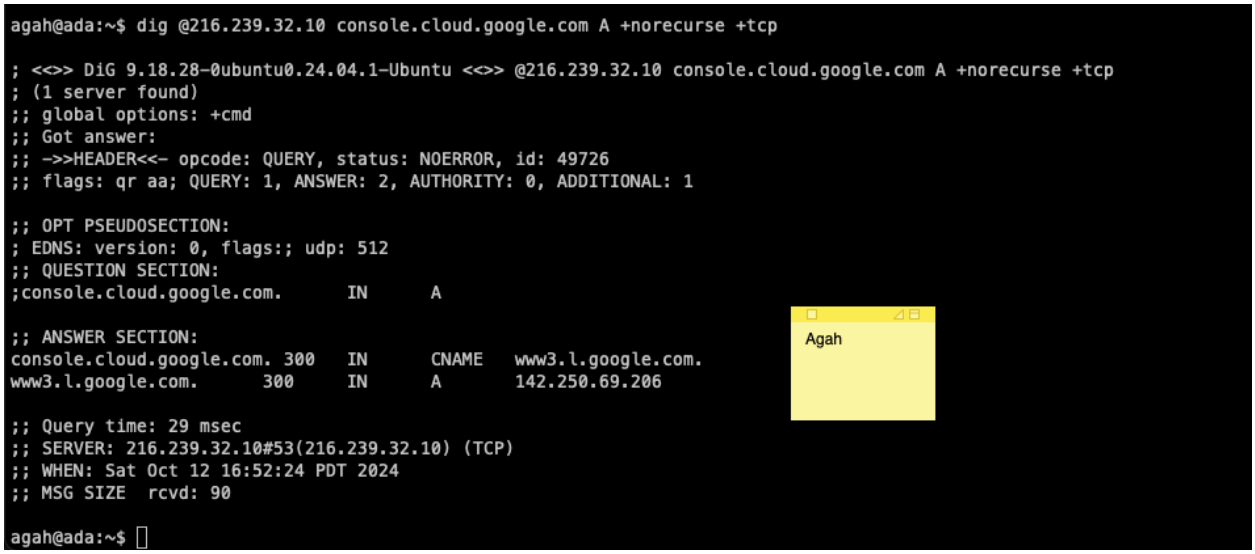




2.2.2 Iterative DNS lookups

- **List all of the iterative dig commands performed for the lookup**
    1. **Dig**
    2. **dig @192.5.5.241 cloud.google.com NS +norecurse +tcp**
    3. **dig @192.43.172.30 google.com NS +norecurse +tcp**
    4. **dig @216.239.32.10 cloud.google.com NS +norecurse +tcp**
    5. **dig @216.239.32.10 console.cloud.google.com A +norecurse +tcp**

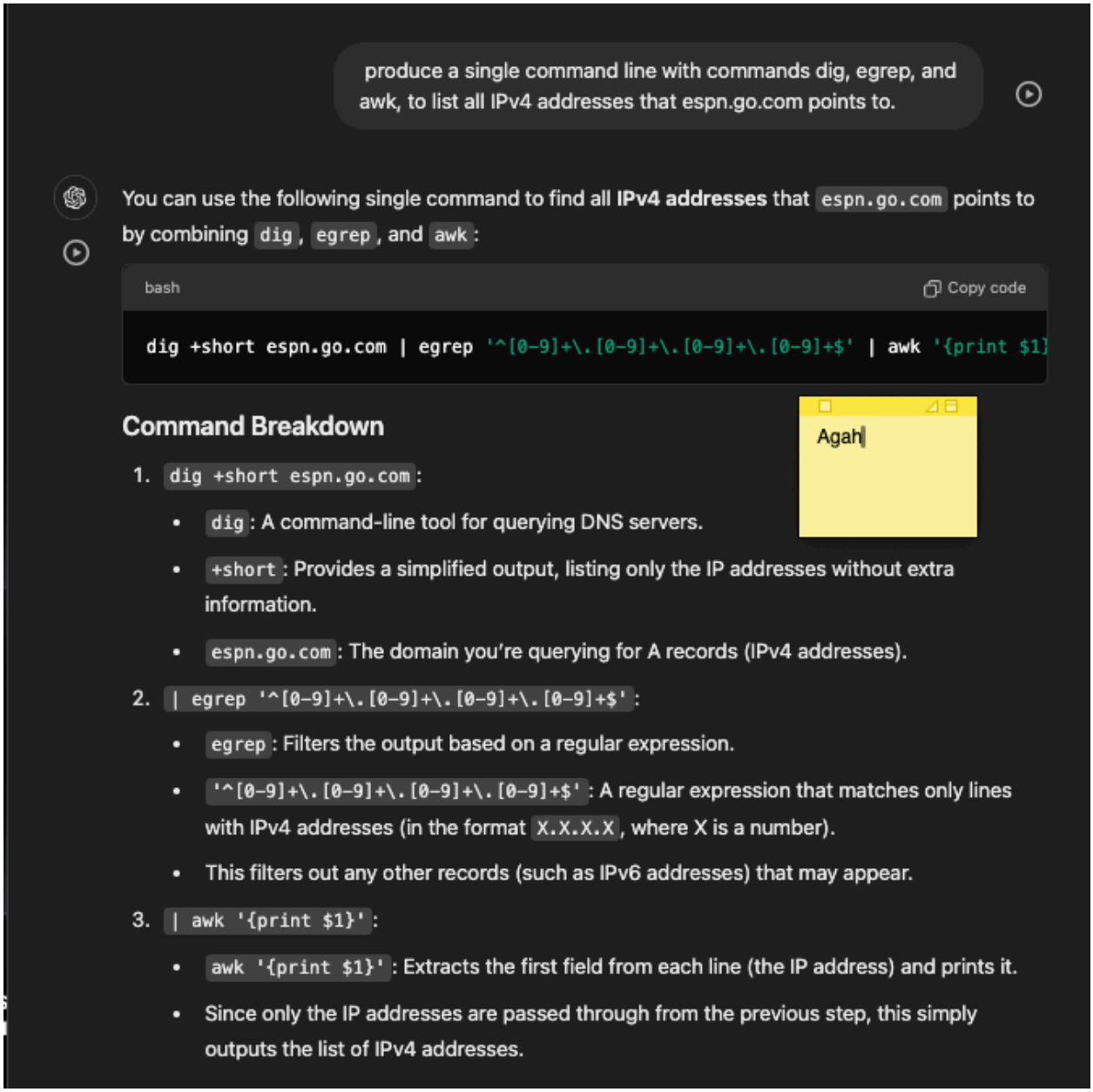
- **Take a screenshot of the results of the final query for your lab notebook.**



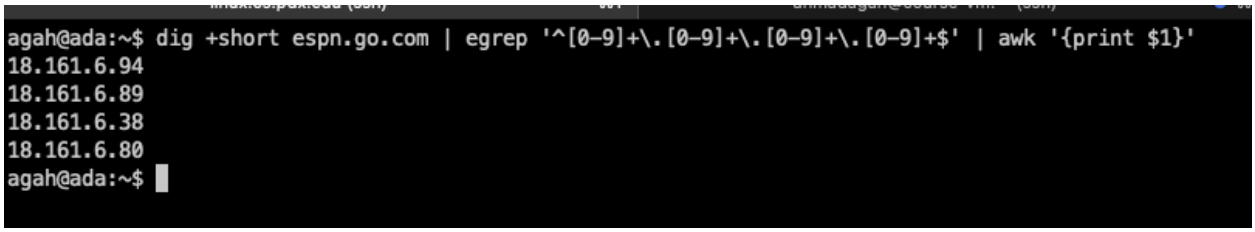
2.2.3 Reverse DNS lookups

Aliases and reverse lookups

- **Take a screenshot of the prompt and the command produced**



- **Take a screenshot of its results for your lab notebook**



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

```
agah@ada:~$ for ip in $(dig +short espn.go.com | egrep '^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$' | awk '{print $1}'); do
    dig +short -x "$ip"
done | egrep '^[a-zA-Z0-9.-]+$' | awk '{print $1}'
server-18-161-6-80.hio52.r.cloudfront.net.
server-18-161-6-89.hio52.r.cloudfront.net.
server-18-161-6-94.hio52.r.cloudfront.net.
server-18-161-6-38.hio52.r.cloudfront.net.
agah@ada:~$
```

2.2.4 Host enumeration

- **Take a screenshot of the results in your lab notebook**

```
agah@ada:~$
agah@ada:~$ for ip in 131.252.220.{0..255}; do     dig +short -x "$ip"; done | e
grep '^[a-zA-Z0-9.-]+$' | awk '{print $1}' > 220hosts.txt
agah@ada:~$ grep -i -E "ford|toyota|honda|bmw|aidu" 220hosts.txt
bmw.cs.pdx.edu.
ford.cs.pdx.edu.
honda.cs.pdx.edu.
toyota.cs.pdx.edu.
agah@ada:~$
```

2.2.5 Geographic DNS #2

- **What geographic locations do ipinfo.io and DB-IP return?**

Geolocation data from **ipinfo.io** Product: API, real-time


IP ADDRESS: 131.252.208.53


COUNTRY: United States


REGION: Oregon


CITY: Portland

Agah


ISP: Not available


ORGANIZATION: AS6366 Portland State University


LATITUDE: 45.5234


LONGITUDE: -122.6762

Incorrect location? Contact ipinfo.io


view map

Geolocation data from **DB-IP** Product: API, real-time


IP ADDRESS: 131.252.208.53


COUNTRY: United States


REGION: Oregon


CITY: Portland (North Portland)


ISP: Portland State University

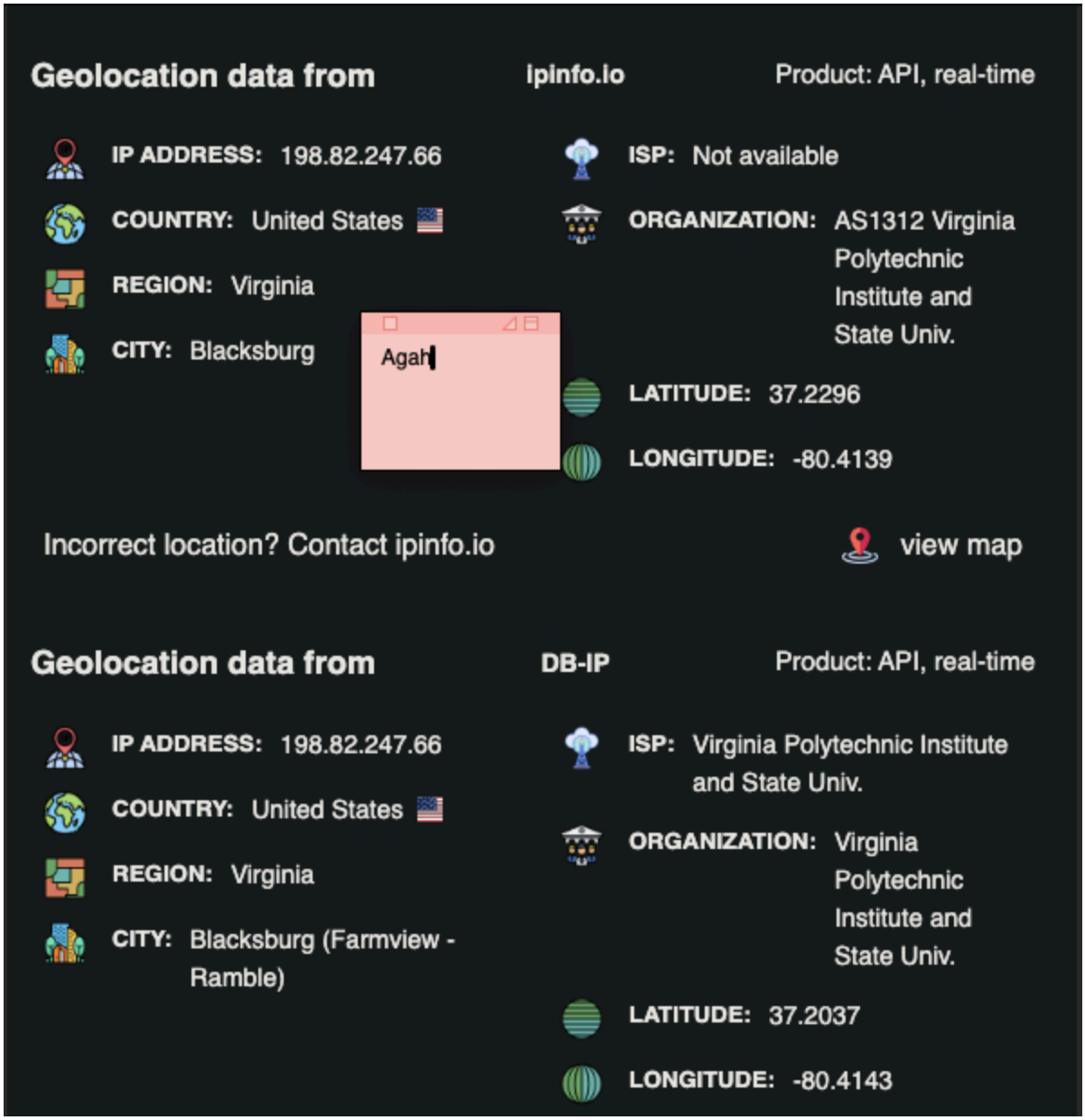

ORGANIZATION: Portland State University


LATITUDE: 45.584


LONGITUDE: -122.728

Incorrect location? Contact DB-IP


view map



- **Record one address for www.google.com from each result for your lab notebook.**

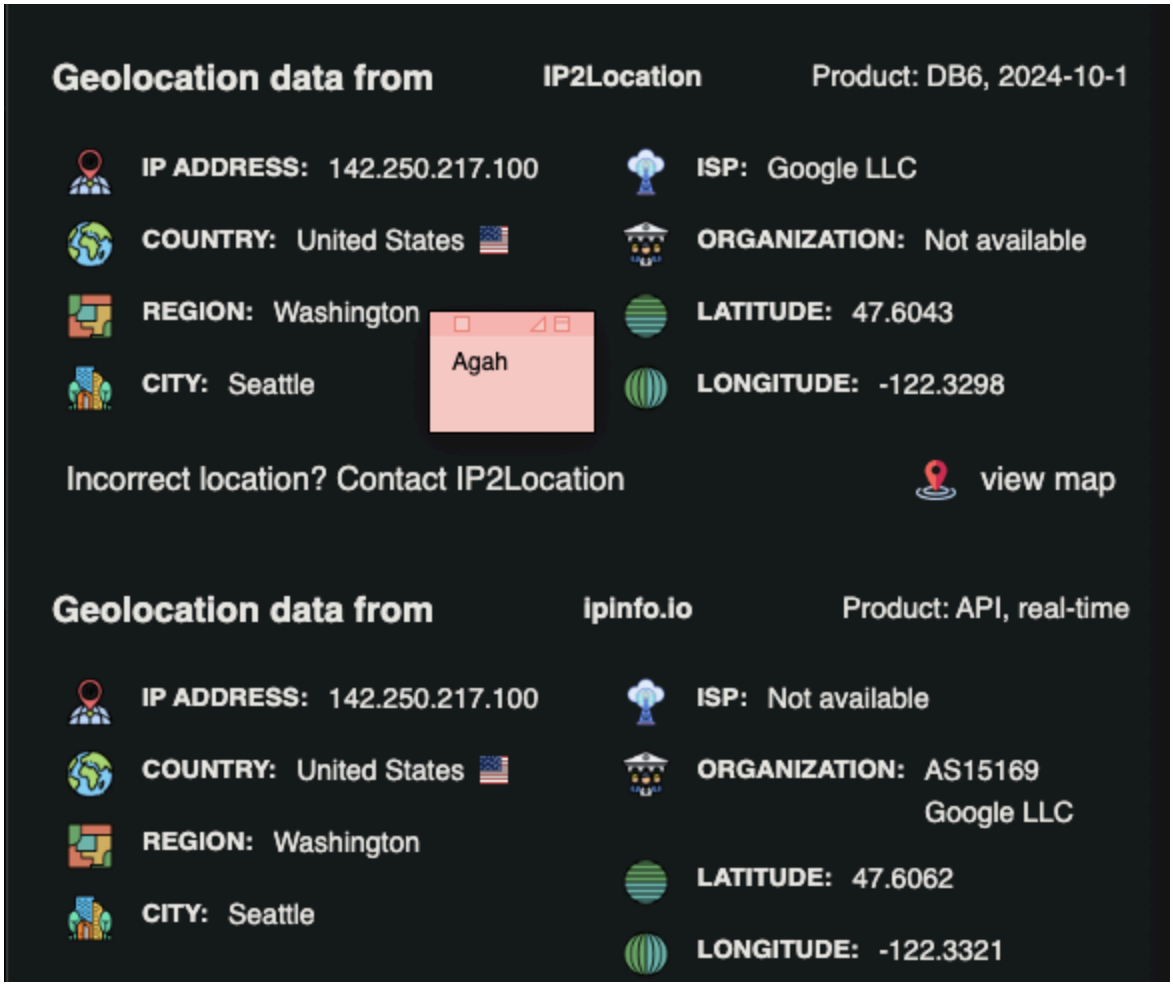
```
agah@ada:~$ dig @131.252.208.53 www.google.com

; <<>> DiG 9.18.28-0ubuntu0.24.04.1-Ubuntu <<>> @131.252.208.53 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45427
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: fec17b15411aa00d01000000670b6911b6e269247eb41ab2 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         219     IN      A       142.250.217.100

;; Query time: 1 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (UDP)
;; WHEN: Sat Oct 12 23:30:41 PDT 2024
;; MSG SIZE  rcvd: 87
```

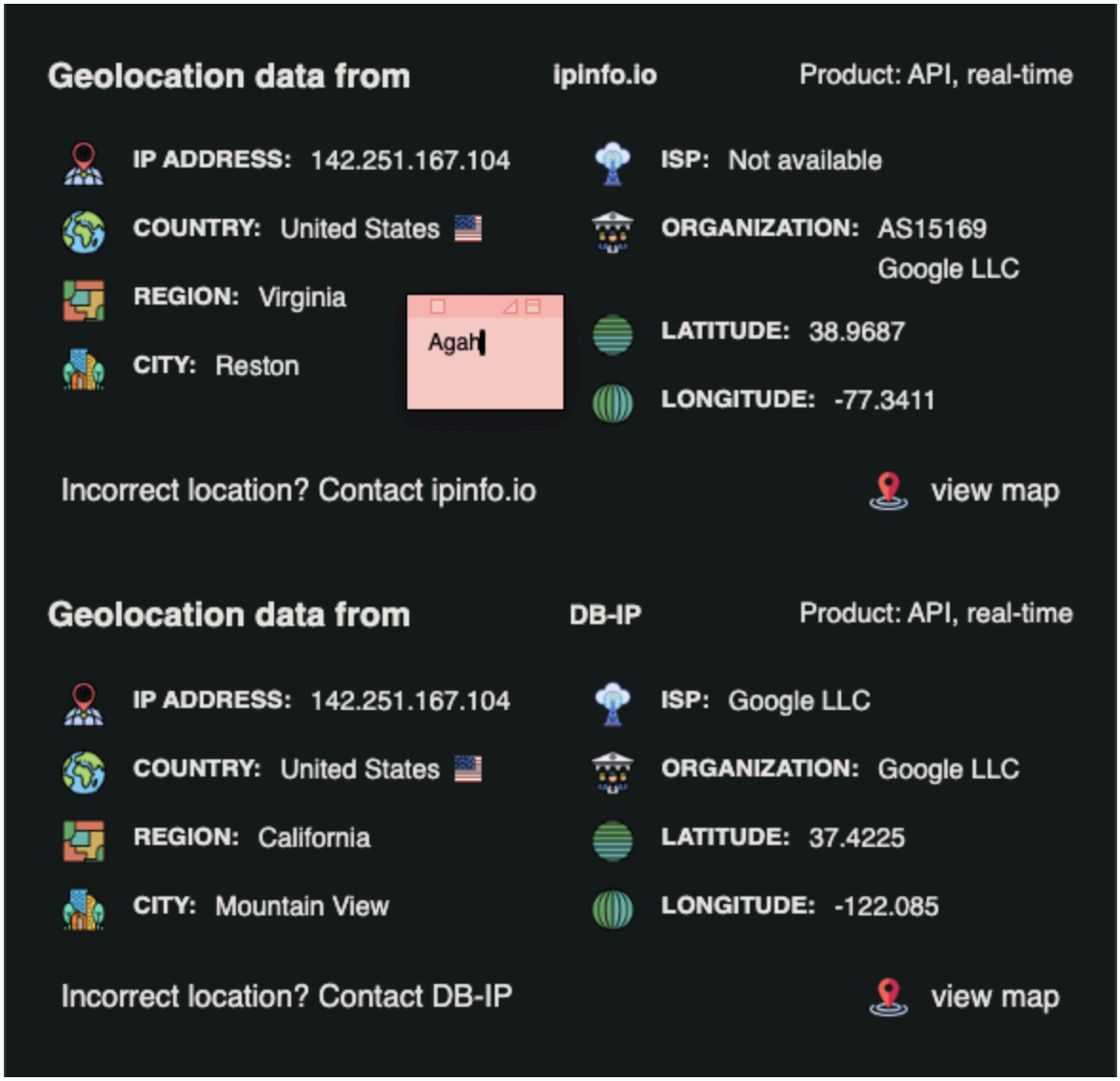
```
agah@ada:~$ dig @198.82.247.66 www.google.com

; <<>> DiG 9.18.28-0ubuntu0.24.04.1-Ubuntu <<>> @198.82.247.66 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50099
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 94c245fffaf9fa69793e08e0670b69793751ee299b4bb83e (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         24      IN      A       142.251.167.104
www.google.com.         24      IN      A       142.251.167.106
www.google.com.         24      IN      A       142.251.167.103
www.google.com.         24      IN      A       142.251.167.147
www.google.com.         24      IN      A       142.251.167.105
www.google.com.         24      IN      A       142.251.167.99

;; Query time: 70 msec
;; SERVER: 198.82.247.66#53(198.82.247.66) (UDP)
;; WHEN: Sat Oct 12 23:32:25 PDT 2024
;; MSG SIZE  rcvd: 167
```

- **What are the geographic coordinates of each DNS server and the IP address it resolves for www.google.com?**

  IP addresses are included in the screenshot below.

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


Geolocation data from IP2Location Product: DB6, 2024-10-1
IP ADDRESS: 142.250.217.100
COUNTRY: United States
REGION: Washington
CITY: Seattle
ISP: Google LLC
ORGANIZATION: Not available
LATITUDE: 47.6043
LONGITUDE: -122.3298
Agah
Incorrect location? Contact IP2Location
view map
Geolocation data from ipinfo.io Product: API, real-time
IP ADDRESS: 142.250.217.100
COUNTRY: United States
REGION: Washington
CITY: Seattle
ISP: Not available
ORGANIZATION: AS15169 Google LLC
LATITUDE: 47.6062
LONGITUDE: -122.3321



2.2.6 Wireshark Lab #3

```
ahmadagah@course-vm:~$ ip addr show
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 1500 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 71051sec preferred_lft 71051sec
    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:b0:f6:25:6f 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
```

```
ahmadagah@course-vm:~$ ip route show
default via 10.138.0.1 dev ens4 proto dhcp src 10.138.0.2 metric 100
10.138.0.1 dev ens4 proto dhcp scope link src 10.138.0.2 metric 100
169.254.169.254 via 10.138.0.1 dev ens4 proto dhcp src 10.138.0.2 metric 100
172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown
ahmadagah@course-vm:~$
```

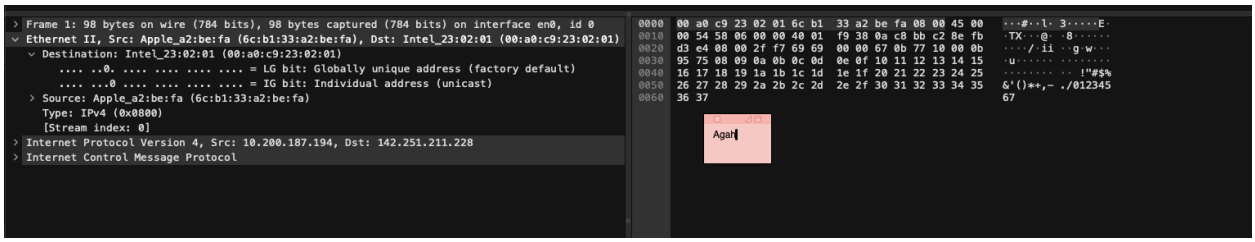
The IP address of the VM —---- src 10.138.0.2

The name of the local virtual ethernet interface —------- dev ens

The IP address of the default router —---------- default via 10.138.0.1

2.2.7 -

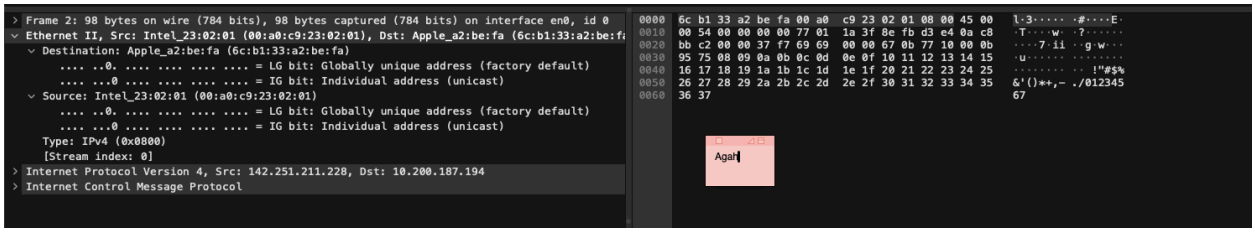
- **Take a screenshot of the bytes in the packet dump window as shown below**



- **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?**

  the destination MAC address (`00:a0:c9:23:02:01`) represents the MAC of my local router interface, which the VM is using to reach Google's servers on the internet.

- **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?**



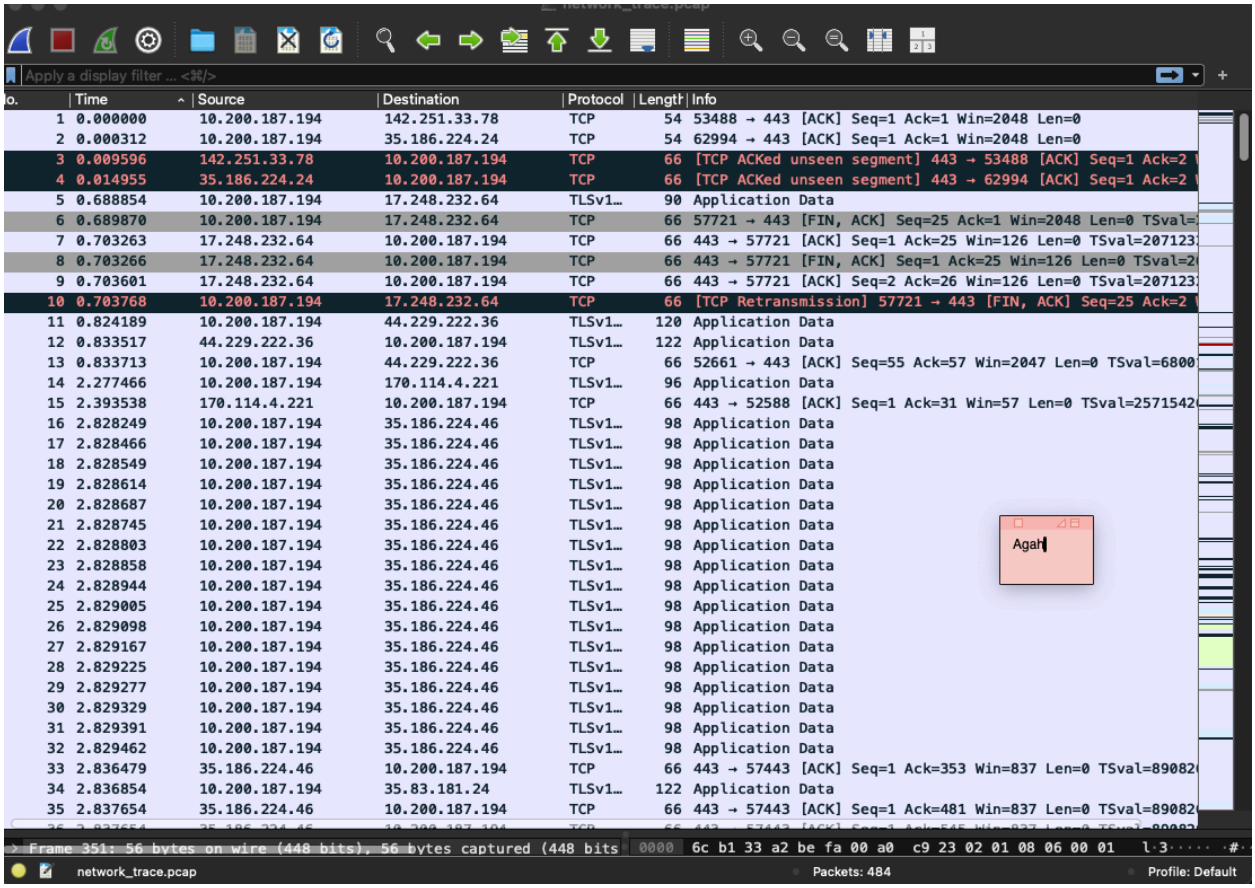
The destination MAC address (`6c:b1:33:a2:be:fa`) in this second packet corresponds to the VM, showing that this packet is directed back to your VM after being processed by the router and Google's servers.

2.2.8 Network Recap Lab #4

2.2.9 Collect trace

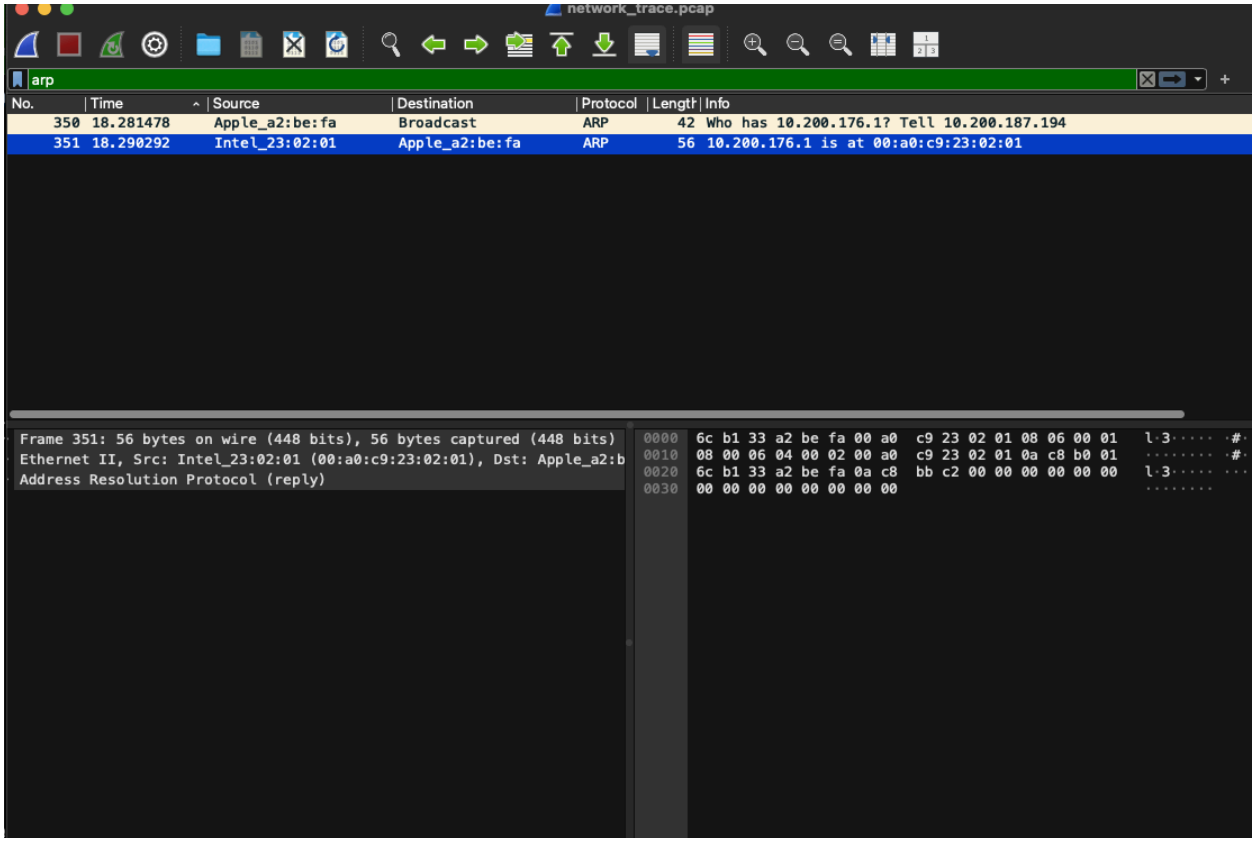
2.2.10 Analyze trace

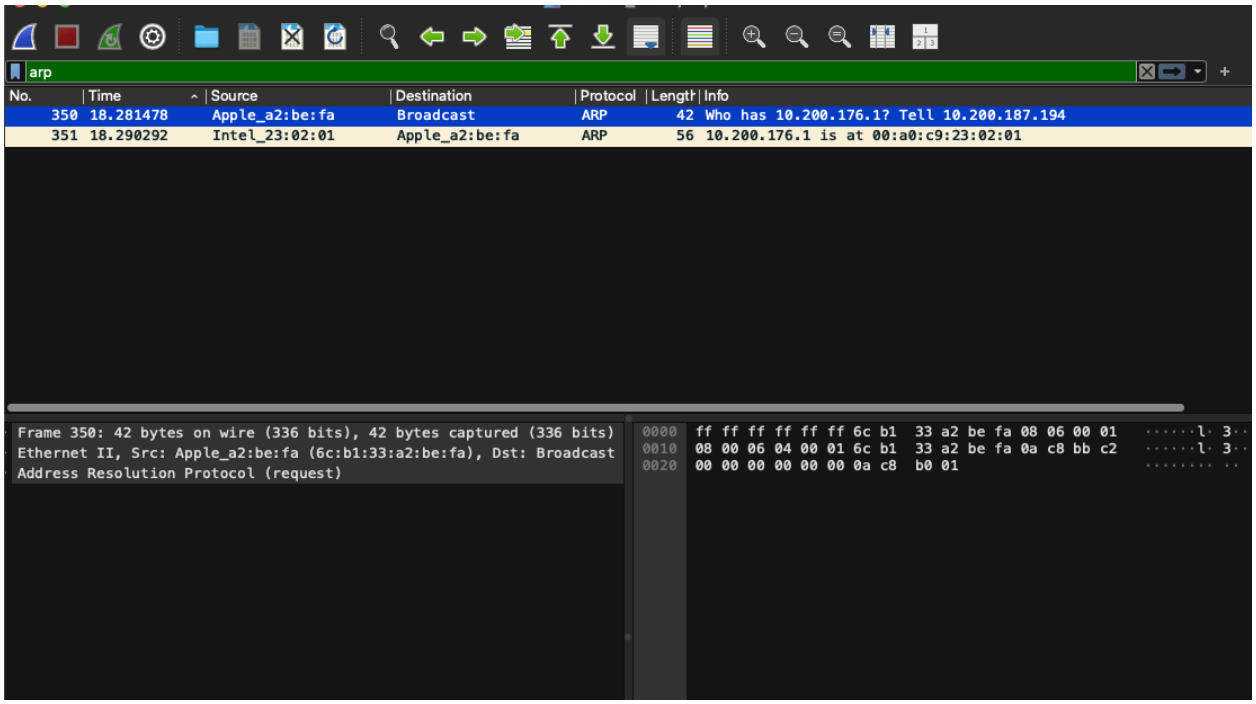
- **Take a screenshot of the all of the packets returned within Wireshark that includes their packet numbers**



ARP

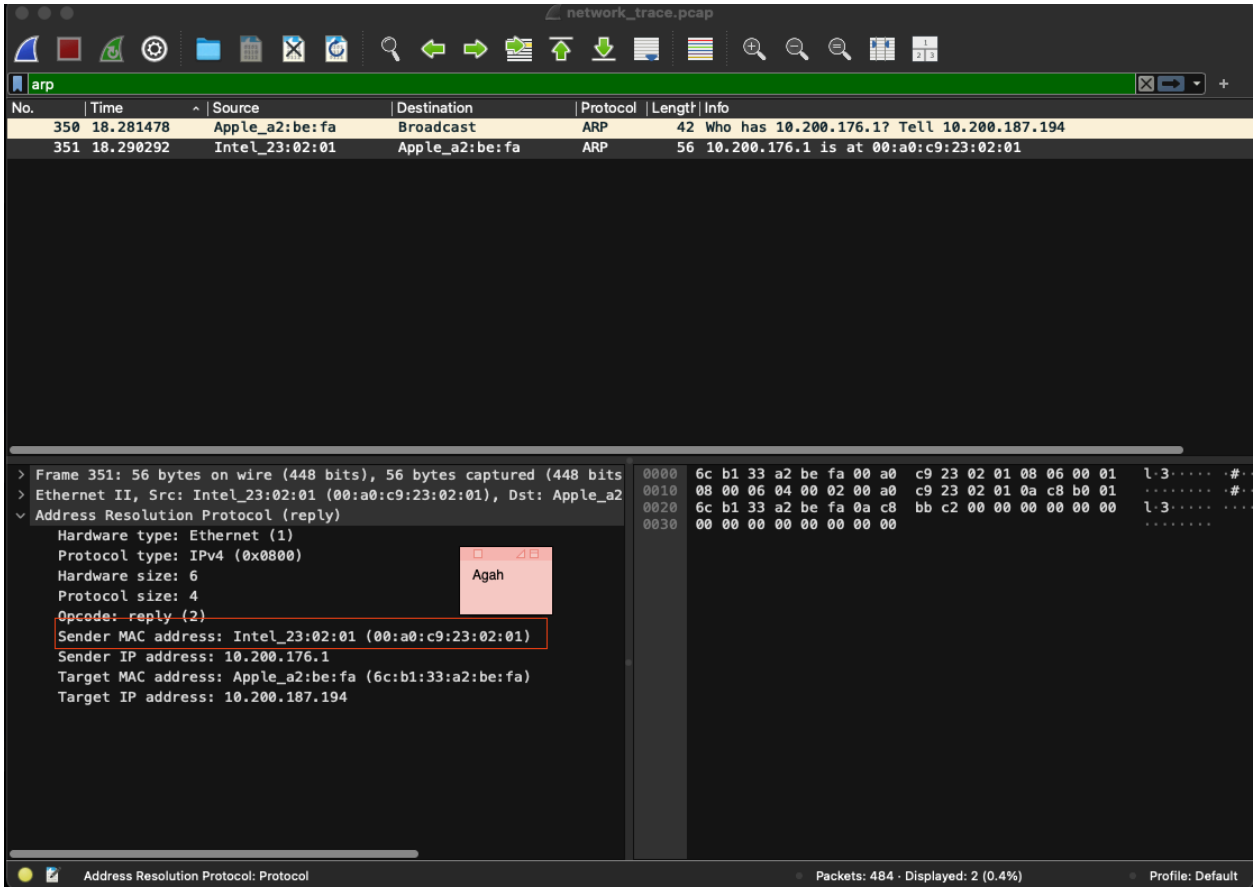
- **What packet numbers in the trace are the result of the VM attempting to get the hardware address of the default router?**


network_trace.pcap
arp
No. Time Source Destination Protocol Length Info
350 18.281478 Apple_a2:be:fa Broadcast ARP 42 Who has 10.200.176.1? Tell 10.200.187.194
351 18.290292 Intel_23:02:01 Apple_a2:be:fa ARP 56 10.200.176.1 is at 00:a0:c9:23:02:01
Frame 351: 56 bytes on wire (448 bits), 56 bytes captured (448 bits)
Ethernet II, Src: Intel_23:02:01 (00:a0:c9:23:02:01), Dst: Apple_a2:b
Address Resolution Protocol (reply)
0000 6c b1 33 a2 be fa 00 a0 c9 23 02 01 08 06 00 01
0010 08 00 06 04 00 02 00 a0 c9 23 02 01 0a c8 b0 01
0020 6c b1 33 a2 be fa 0a c8 bb c2 00 00 00 00 00 00
0030 00 00 00 00 00 00 00 00


arp
No. Time Source Destination Protocol Length Info
350 18.281478 Apple_a2:be:fa Broadcast ARP 42 Who has 10.200.176.1? Tell 10.200.187.194
351 18.290292 Intel_23:02:01 Apple_a2:be:fa ARP 56 10.200.176.1 is at 00:a0:c9:23:02:01
Frame 350: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
Ethernet II, Src: Apple_a2:be:fa (6c:b1:33:a2:be:fa), Dst: Broadcast
Address Resolution Protocol (request)
0000 ff ff ff ff ff ff 6c b1 33 a2 be fa 08 06 00 01
0010 08 00 06 04 00 01 6c b1 33 a2 be fa 0a c8 bb c2
0020 00 00 00 00 00 00 0a c8 b0 01

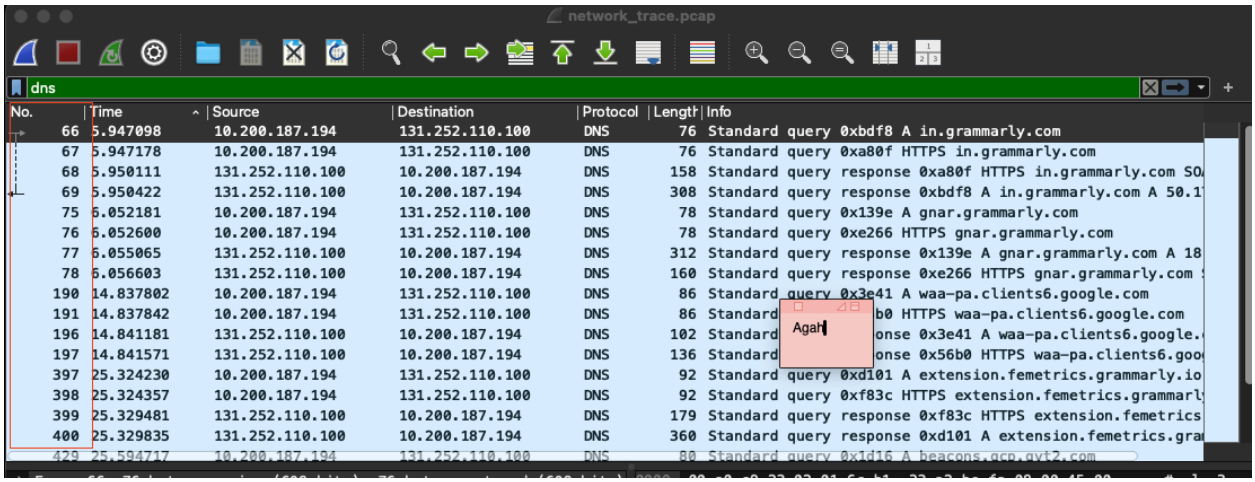
Packet 350 and 351

- **What is this hardware address?**

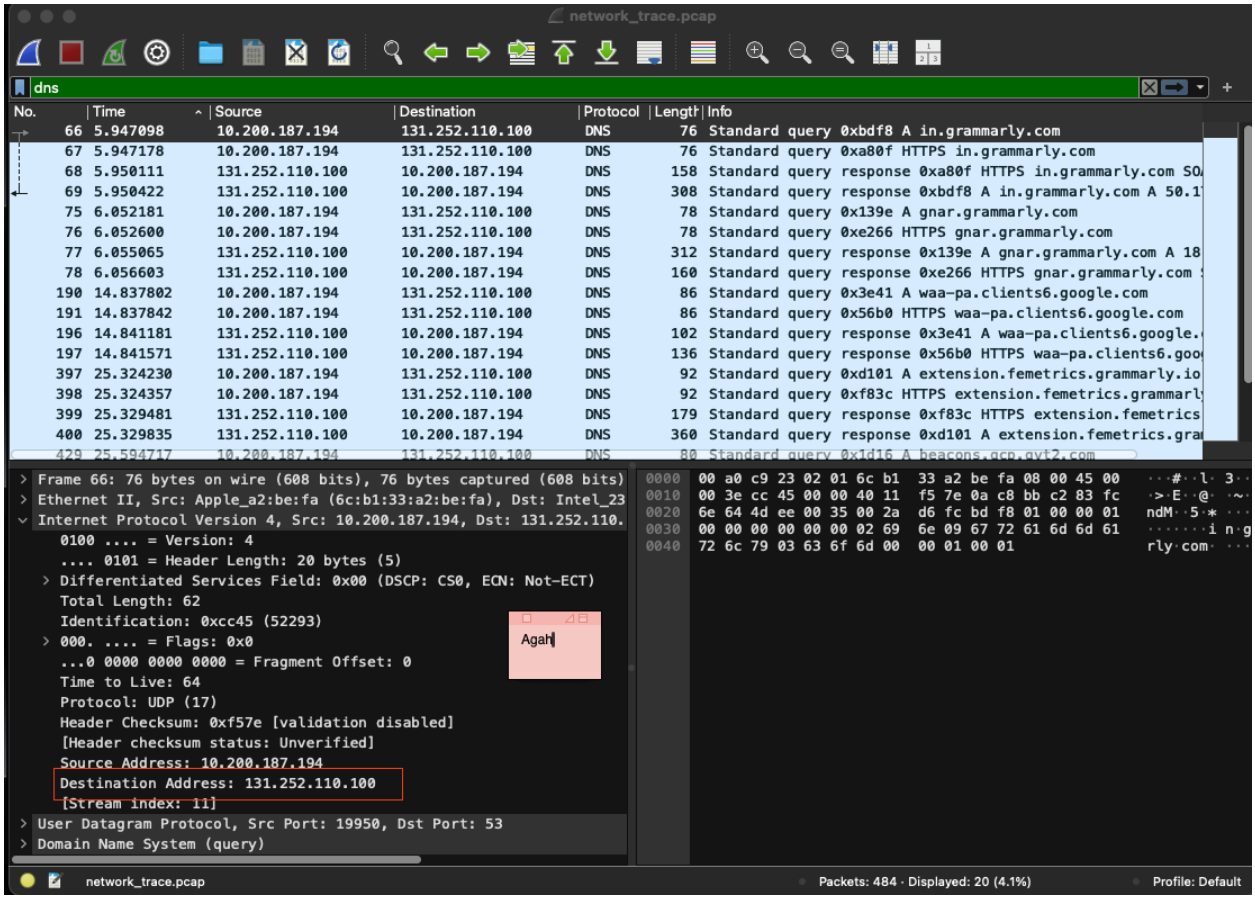


DNS

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

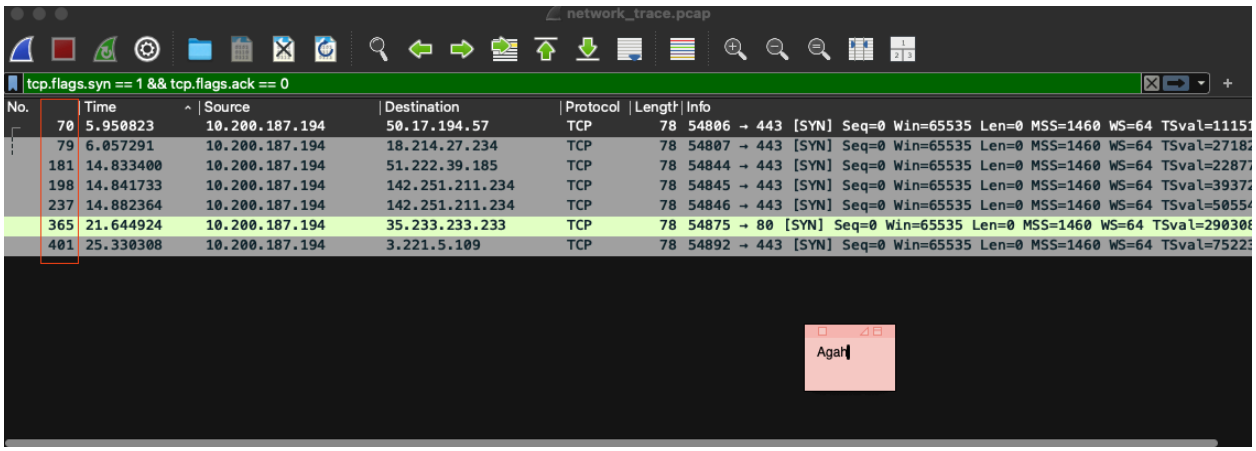


- **What is the IP address of the local DNS server being queried?**

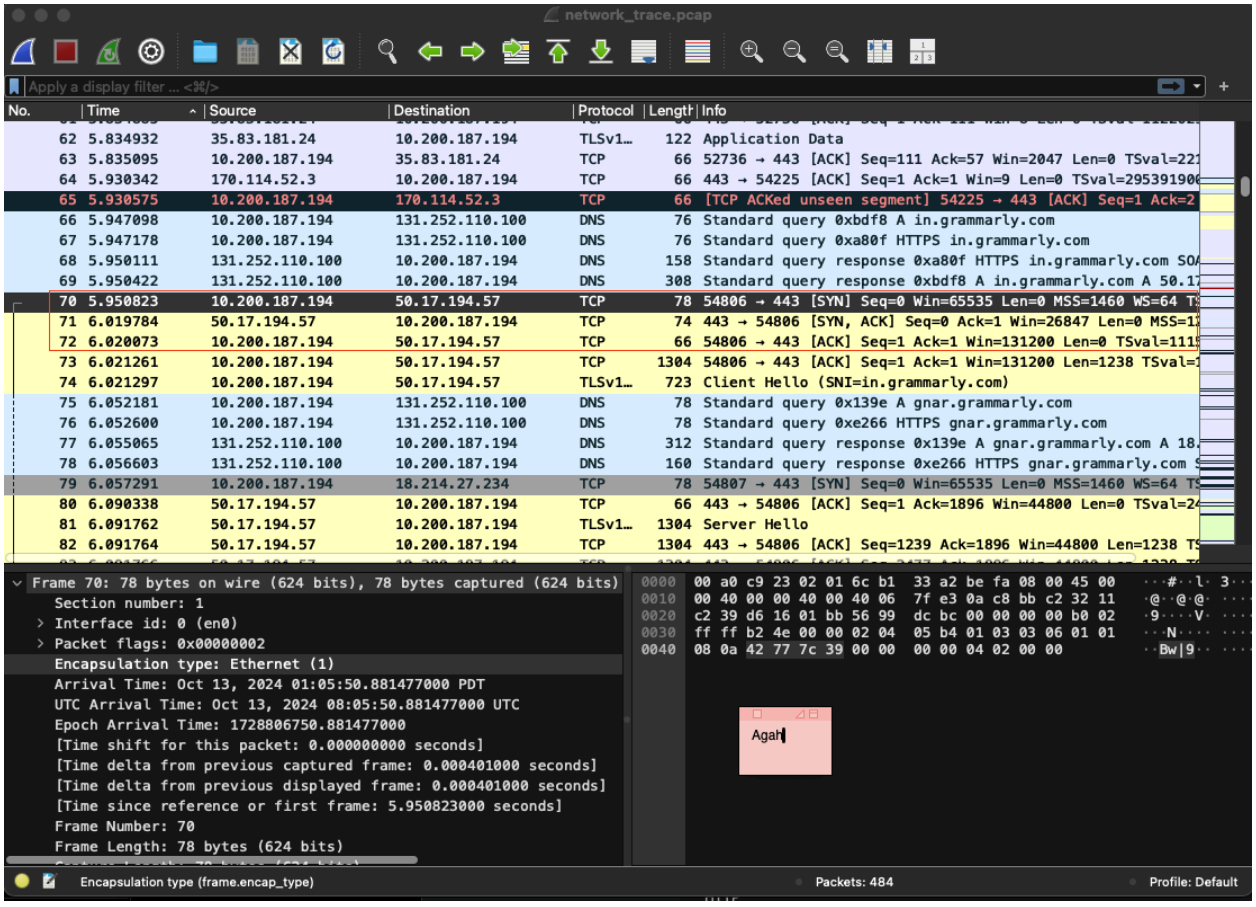


TCP

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



- **How long does it take to perform the initial TCP handshake?**

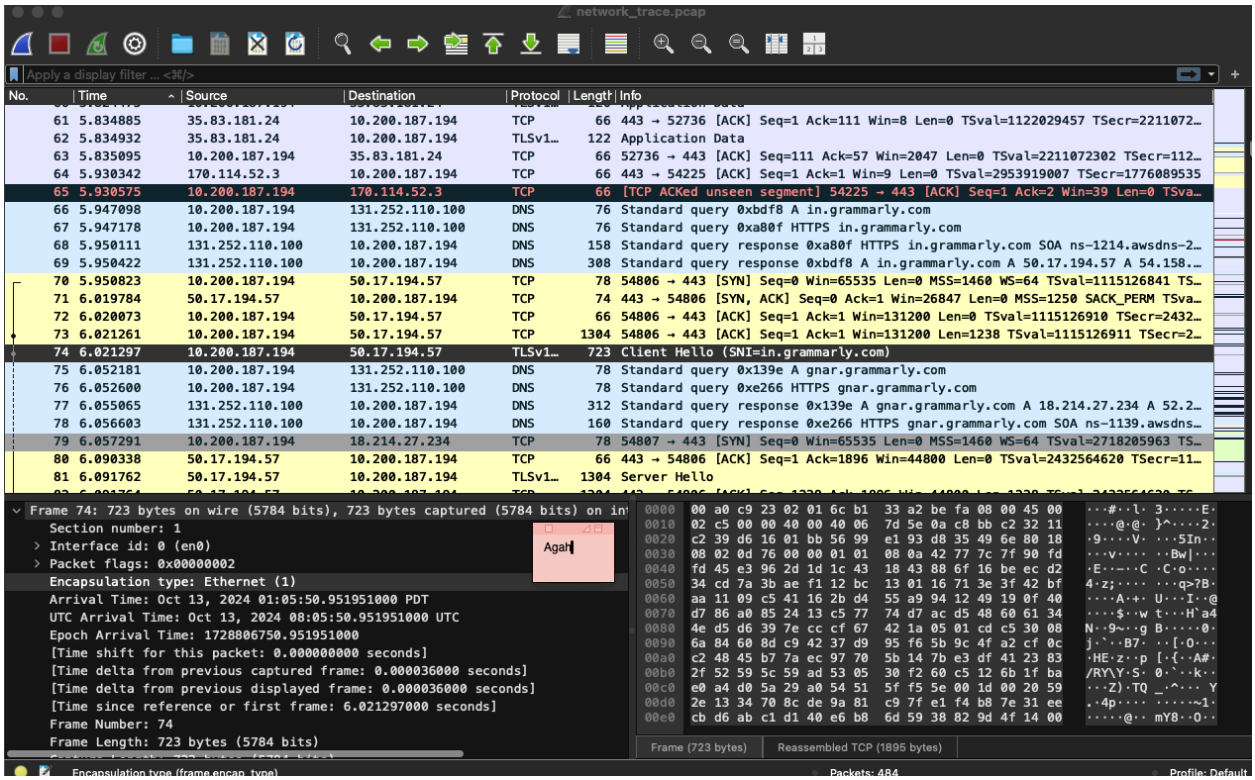


Subtract the timestamp of the SYN packet from the timestamp of the ACK packet: 6.020073−5.950823=0.06925 seconds6.020073 - 5.950823 = 0.06925

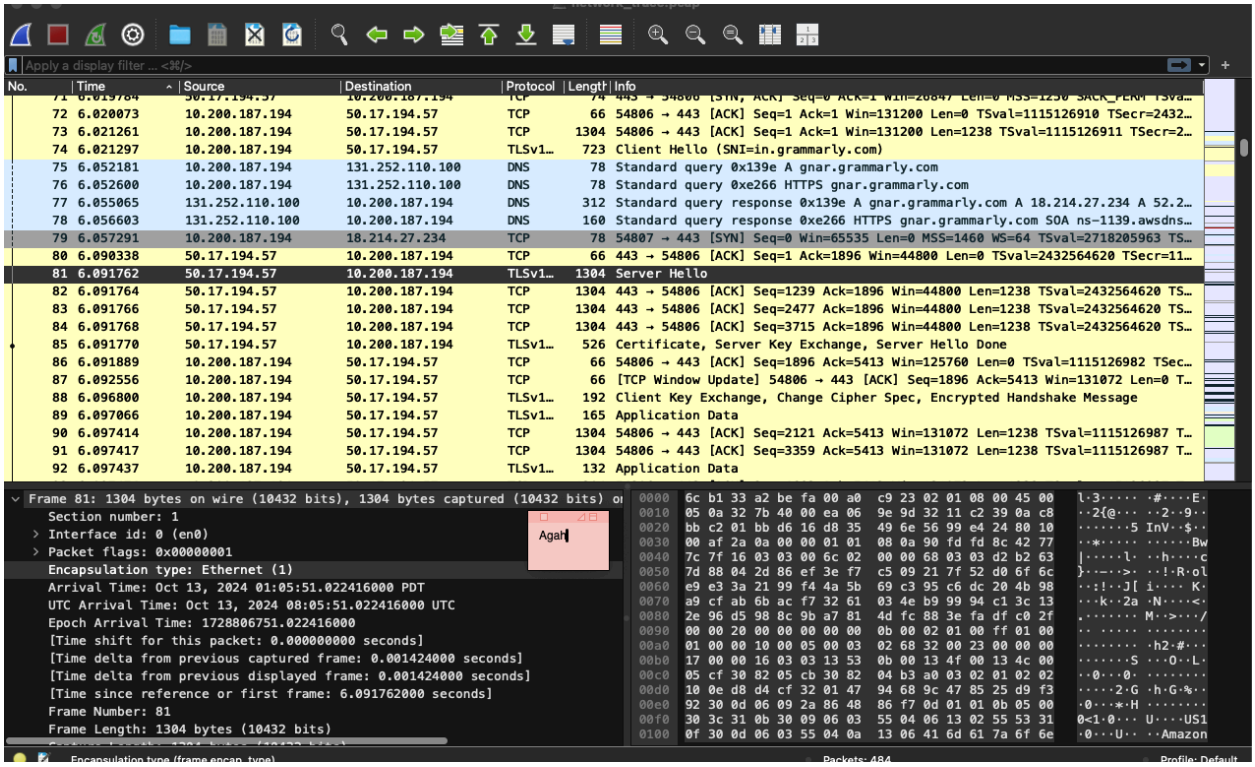
HTTP

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

  Request



  Response



- **How long does it take to process the HTTP request after the handshake?**

  6.091762 -6.021297 = 0.070465 seconds

2.2.11 Clean up