Week2

Lab2

Armant Touche

Class/Instructor: CS430P/ Dr. Wu-Chang
Date: 10/07/22

Table of Contents

TCP, HTTP (Link)

- [ ] 1. TCP #1 (`netstat`, `lsof`, `netcat`/`nc`)

A. Run the command using sudo and take a screenshot of the output to include in your lab notebook.

```
atouche@atouche:~$ sudo netstat -lptn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:33569         0.0.0.0:*               LISTEN      707/containerd
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      552/systemd-resolve
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      691/cupsd
tcp6       0      0 ::1:631                 :::*                    LISTEN      691/cupsd
```

B. For port numbers that are named, examine /etc/services and find the port number that corresponds to it. Include this mapping in your lab notebook.

552/systemd-resolve

```
atouche@atouche:~$ cat /etc/services |grep "53"
domain          53/tcp                          # Domain Name Server
domain          53/udp
```

691/cupsd

```
atouche@atouche:~$ cat /etc/services |grep 631
ipp             631/tcp                         # Internet Printing Protocol
```

707/containerd

Screenshot for containerd N/A is a user/registered port

C. For ports that only have a number, what service might it be providing based on the name of the program that is being run?

53 = DNS; 631 = Internet Printing Protocol; 33569 = Registered Port/Service

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

```
atouche@ada:~$ netstat -lnpt
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:45195         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:45467         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:46567         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:44065         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:44107         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:41103         0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:46485           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:42619         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:40327         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:40421         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:37523         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:37845         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6017          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6016          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6019          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6018          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6021          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6020          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6023          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6022          0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6011          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6010          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6013          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6012          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6015          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:6014          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:35263         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:32791         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:25            0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:34103         0.0.0.0:*               LISTEN      -
tcp6       0      0 ::1:6017                :::*                    LISTEN      -
tcp6       0      0 ::1:6016                :::*                    LISTEN      -
tcp6       0      0 ::1:6019                :::*                    LISTEN      -
tcp6       0      0 ::1:6018                :::*                    LISTEN      -
tcp6       0      0 ::1:6021                :::*                    LISTEN      -
tcp6       0      0 ::1:6020                :::*                    LISTEN      -
tcp6       0      0 ::1:6023                :::*                    LISTEN      -
tcp6       0      0 ::1:6022                :::*                    LISTEN      -
tcp6       0      0 ::1:6011                :::*                    LISTEN      -
tcp6       0      0 ::1:6010                :::*                    LISTEN      -
tcp6       0      0 ::1:6013                :::*                    LISTEN      -
tcp6       0      0 ::1:6012                :::*                    LISTEN      -
tcp6       0      0 ::1:6015                :::*                    LISTEN      -
tcp6       0      0 ::1:6014                :::*                    LISTEN      -
tcp6       0      0 ::1:25                  :::*                    LISTEN      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      -
tcp6       0      0 127.0.0.1:8212          :::*                    LISTEN      -
tcp6       0      0 :::46413                :::*                    LISTEN      -
tcp6       0      0 :::113                  :::*                    LISTEN      -
tcp6       0      0 :::111                  :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
```

E. What services does this machine provide for external access?

- 22 (`ssh`) is what was used for remote sessions
- 25 (`smtp`) is used for receiving emails over the internet
- 53 (`dns`) is used for domain name system
- 111 (`sunrpc`) is RPC 4.0 portmapper
- 113 (`auth`) is authentication tap ident

F. **Use the `-i` and the `-s` flag of `lsof` to generate a listing that is equivalent to the one generated with `netstat` previously and include it in your lab notebook.**

```
atouche@atouche:~$ sudo lsof -i4 -s
COMMAND   PID            USER   FD   TYPE DEVICE SIZE NODE NAME
systemd-r 548 systemd-resolve   12u  IPv4  20795       UDP localhost:domain
systemd-r 548 systemd-resolve   13u  IPv4  20796       TCP localhost:domain (LISTEN)
avahi-dae 584           avahi   12u  IPv4  24025       UDP *:mdns
avahi-dae 584           avahi   14u  IPv4  24027       UDP *:44962
cupsd     586            root    7u  IPv4  23315       TCP localhost:ipp (LISTEN)
NetworkMa 588            root   23u  IPv4  24966       UDP atouche:bootpc->_gateway:bootps
cups-brow 665            root    7u  IPv4  23452       UDP *:631
container 688            root   12u  IPv4  25390       TCP localhost:38009 (LISTEN)
atouche@atouche:~$ sudo lsof -i4 -s | wc -l
9
```

G. **Include for your lab notebook, the version of `ssh` that is being used. (Type Ctrl+c to exit)**

```
nc: getaddrinfo for host "atouche@linux.cs.pd
atouche@atouche:~$ nc linux.cs.pdx.edu 22
SSH-2.0-OpenSSH_8.9p1 Ubuntu-3
^C
```

☐ 2. TCP #2 (`iperf`)

A. Create 4 VMs: us-west1-b, us-east1, australia, europe
B. Install `iperf`

- [ ] 3. Throughput tests

    A. Start iperf server port(80)

    B. From us-west1-b connect:

        a. us-east1-b

```
atouche@cloudshell:~ (cloud-touche-atouche)$ iperf -c 35.185.43.18 -p 80
------------------------------------------------------------
Client connecting to 35.185.43.18, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 172.17.0.4 port 49766 connected with 35.185.43.18 port 80
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.0313 sec   293 MBytes   245 Mbits/sec
```

        b. australia-southeast1-b

```
atouche@cloudshell:~ (cloud-touche-atouche)$ iperf -c 35.189.58.142 -p 80
------------------------------------------------------------
Client connecting to 35.189.58.142, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 172.17.0.4 port 53882 connected with 35.189.58.142 port 80
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.0508 sec   120 MBytes  99.8 Mbits/sec
```

        c. europe-central2-a

```
atouche@cloudshell:~ (cloud-touche-atouche)$ iperf -c 34.116.228.5 -p 80
------------------------------------------------------------
Client connecting to 34.116.228.5, TCP port 80
TCP window size: 85.0 KByte (default)
------------------------------------------------------------
[  3] local 172.17.0.4 port 57190 connected with 34.116.228.5 port 80
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.1546 sec   115 MBytes  95.2 Mbits/sec
```

    C. Based on throughput (Mbits/sec) speed, ranked fast to slow:

        1. us-east1-b
        2. australia-southeast1-b
        3. Europe-central2-a

- Difference in throughput could be due to edge distance between instance1 (us-west1-b) and other instance(2,3,4). Number of hops probably increases drastically and is proportional to throughput. After a quick Internet search, there were some references that stated that the number of hops can potentially reduce network performance.

- [ ] 4. HTTP #3 (Browser tools)

    A. Enable quic

- [ ] 5. Developer tools

    A. Visit and inspect first query header requests

        a. What is the URL being requested?

            i. http://www.google.com

        b. What are the Host: (HTTP 1.1) or :authority: (HTTP 2.0) headers sent by the browser? What is the User-Agent: HTTP header that is sent?

            i. Assuming this question is asking about Request header:

                1. Location: www.google.com

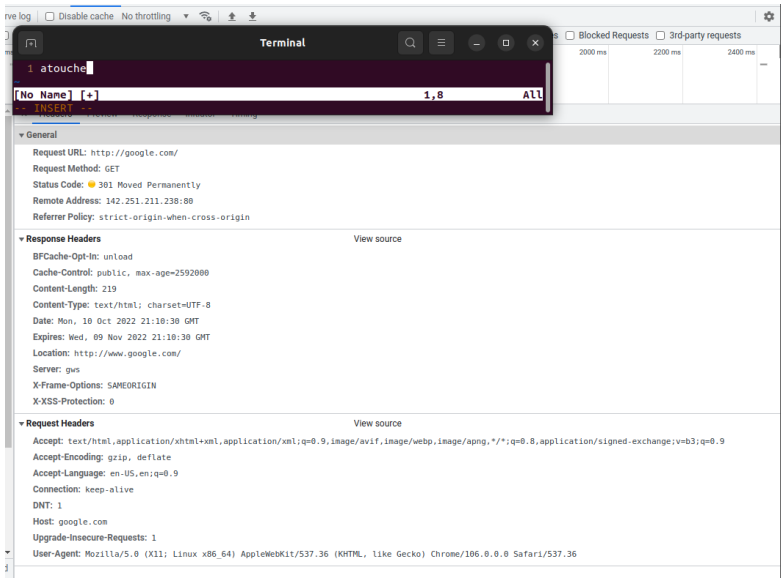
            ii. User agent

                1. Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/105.0.0.0 Safari/537.36

        c. What is the HTTP status code in the response and what does it mean?

            i. HTTP 301 **Moved Permanently** redirect status response code indicates that the requested resource has been definitively moved to the URL given by the Location headers. A browser redirects to the new URL and search engines update their links to the resource

d. **Look up the status code. Show the associated HTTP response header that is sent in conjunction with this status code for the request**



B. Visit and inspect second query header requests

a. **What is the URL being requested? Is it using HTTP or HTTPS?**

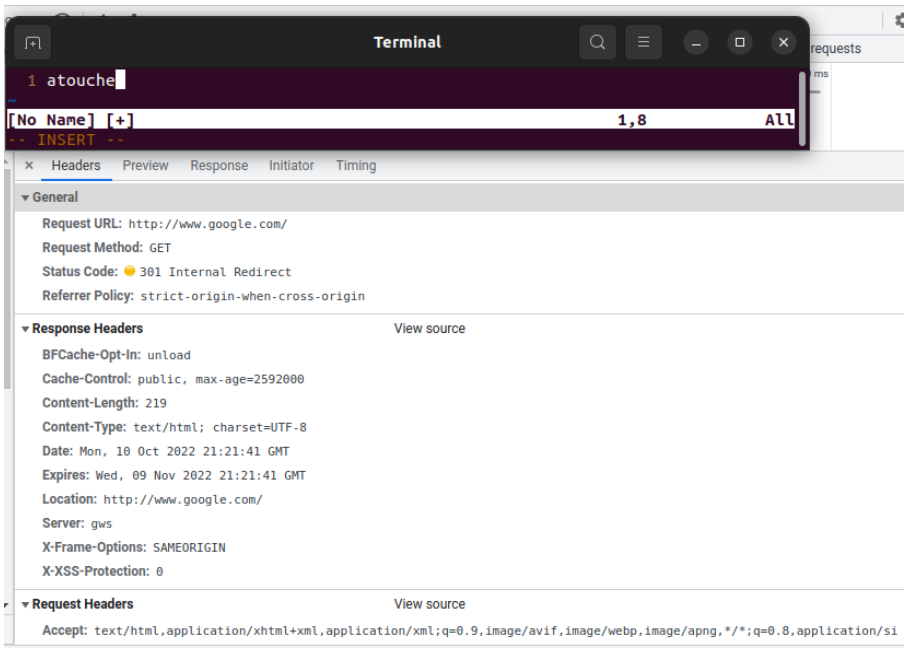
i. http://www.google.com

ii. http was used

b. **What is the HTTP status code in the response and what does it mean? Is it different from the first status code? If so, what is the semantic difference?**

i. HTTP 301 **Moved Permanently** redirect status response code indicates that the requested resource has been definitively moved to the URL given by the Location headers. A browser redirects to the new URL and search engines update their links to the resource

c. **Look up the status code. Show the associated HTTP response header that is sent in conjunction with this status code for the request.**



C. Visit and inspect third query header requests

a. **What is the URL being requested?**

i. https://www.google.com

b. **What is the HTTP status code in the response?**

i. 200

c. **Look for an alt-svc: HTTP response header. Does the server believe the client can use HTTP3/QUIC?**

i. With the quic=443, the server believes the client can support HTTPS which is why the redirect occurred when GET at http://google.com took place.

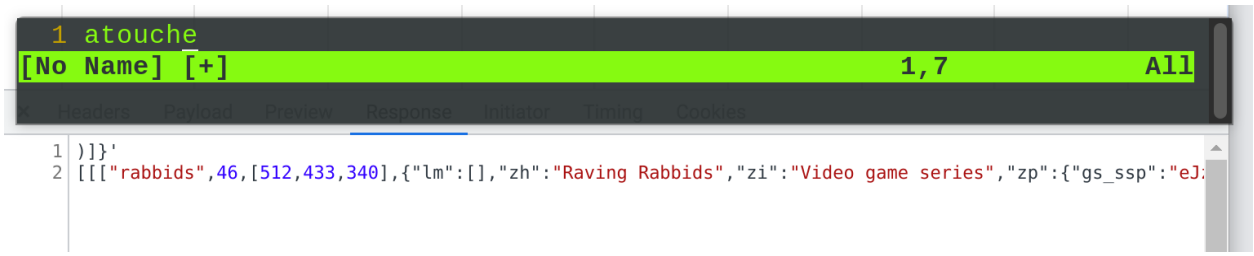
d. **Examine the HTTP response headers for cookies. Show the cookies that are set and which ones specify that no SameSite restrictions are in place. What does the setting indicate about the cookies that are set?**

   i. **set-cookie:** 1P_JAR=2022-10-06-18; expires=Sat, 05-Nov-2022 18:42:26 GMT; path=/; domain=.google.com; Secure; SameSite=none

   ii. The cookie set in third request will become invalid once you visit other url paths excluding '/'=root path.

☐ 6. Async HTTP Request

A. rabbid



DNS, Recap (Link)

- [ ] DNS #1 (dig)

A. Use dig to query the local DNS server for the A record of www.pdx.edu using TCP. Then, use dig to do the same for the MX record of pdx.edu. What do the ANSWER sections explain about where PSU's web/mail services are run from?

```
atouche@ada:~$ dig -t A www.pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> -t A www.pdx.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24163
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 320a6afa62ead5d90100000063448f09b26d74aef31a7f83 (good)
;; QUESTION SECTION:
;www.pdx.edu.                   IN      A

;; ANSWER SECTION:
www.pdx.edu.            736     IN      A       54.214.67.95

;; Query time: 0 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (UDP)
;; WHEN: Mon Oct 10 14:30:49 PDT 2022
;; MSG SIZE  rcvd: 84

atouche@ada:~$ dig -t MX pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> -t MX pdx.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43807
;; flags: qr rd ra; QUERY: 1, ANSWER: 5, AUTHORITY: 0, ADDITIONAL: 3

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 472c07694af9a7180100000063448f12f008f5e1196fab22 (good)
;; QUESTION SECTION:
;pdx.edu.                       IN      MX

;; ANSWER SECTION:
pdx.edu.                6139    IN      MX      5 alt1.aspmx.l.google.com.
pdx.edu.                6139    IN      MX      5 alt2.aspmx.l.google.com.
pdx.edu.                6139    IN      MX      10 alt4.aspmx.l.google.com.
pdx.edu.                6139    IN      MX      1 aspmx.l.google.com.
pdx.edu.                6139    IN      MX      10 alt3.aspmx.l.google.com.

;; ADDITIONAL SECTION:
aspmx.l.google.com.     183     IN      A       74.125.135.27
aspmx.l.google.com.     41      IN      AAAA    2607:f8b0:400e:c09::1a

;; Query time: 3 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (UDP)
;; WHEN: Mon Oct 10 14:30:58 PDT 2022
;; MSG SIZE  rcvd: 226
```

a. Difference is **(a)** is an explicit FQDN/A record and **(b)** is multiple mail exchange servers at various domains list **[i-v]**.

B. Find the authoritative server (NS record type, AUTHORITY section response) for mashimaro.cs.pdx.edu and then query that server for the A record of mashimaro.cs.pdx.edu. Show both.

a. NS

```
atouche@ada:~/Documents/cs430p/lab/lab2$ dig +tcp mashimaro.cs.pdx.edu NS

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> +tcp mashimaro.cs.pdx.edu NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26594
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
```

b. A

```
atouche@ada:~$ dig @walt.ee.pdx.edu +tcp -t A mashimaro.cs.pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @walt.ee.pdx.edu +tcp -t A mashimaro.cs.pdx.edu
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3419
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: f8c0f7c5526c7c0401000000063449150649bca4142989548 (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) (TCP)
;; WHEN: Mon Oct 10 14:40:32 PDT 2022
;; MSG SIZE  rcvd: 93
```

C. Find the authoritative server for thefengs.com and then query that server for the A record of thefengs.com

```
atouche@ada:~$ dig +tcp -t NS thefengs.com | egrep ns | awk '{print $1, $5}'
;;
;;
thefengs.com. ns-cloud2.googledomains.com.
thefengs.com. ns-cloud1.googledomains.com.
thefengs.com. ns-cloud4.googledomains.com.
thefengs.com. ns-cloud3.googledomains.com.
ns-cloud1.googledomains.com. 216.239.32.106
ns-cloud2.googledomains.com. 216.239.34.106
ns-cloud3.googledomains.com. 216.239.36.106
ns-cloud4.googledomains.com. 216.239.38.106
ns-cloud1.googledomains.com. 2001:4860:4802:32::6a
ns-cloud2.googledomains.com. 2001:4860:4802:34::6a
ns-cloud3.googledomains.com. 2001:4860:4802:36::6a
ns-cloud4.googledomains.com. 2001:4860:4802:38::6a
```

D. **When a web request hits port 80 of 131.252.220.66, how does the server know which site to serve from? (i.e. what protocol header)**
   a. If port(80), then **http** will serve the site host on a http/s server. The GET request will detail the http protocol.

```
atouche@ada:~$ dig @ns-cloud1.googledomains.com +tcp -t A thefengs.com

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @ns-cloud1.googledomains.com +tcp -t A thefengs.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21376
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; WARNING: recursion requested but not available

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

;; ANSWER SECTION:
thefengs.com.           3600    IN      A       131.252.220.66

;; Query time: 51 msec
;; SERVER: 216.239.32.106#53(ns-cloud1.googledomains.com) (TCP)
;; WHEN: Mon Oct 10 14:46:24 PDT 2022
;; MSG SIZE  rcvd: 57
```

E. **DNS iterative lookups**
   a. On linux.cs.pdx.edu, perform an DNS iterative lookup on:
      i. www.cs.pdx.edu

```
l.edu-servers.net.      172800  IN      A       192.41.162.30
l.edu-servers.net.      172800  IN      AAAA    2001:500:d937::30
b.edu-servers.net.      172800  IN      A       192.33.14.30
b.edu-servers.net.      172800  IN      AAAA    2001:503:231d::2:30
c.edu-servers.net.      172800  IN      A       192.26.92.30
c.edu-servers.net.      172800  IN      AAAA    2001:503:83eb::30
d.edu-servers.net.      172800  IN      A       192.31.80.30
d.edu-servers.net.      172800  IN      AAAA    2001:500:856e::30
e.edu-servers.net.      172800  IN      A       192.12.94.30
e.edu-servers.net.      172800  IN      AAAA    2001:502:1ca1::30
f.edu-servers.net.      172800  IN      A       192.35.51.30
f.edu-servers.net.      172800  IN      AAAA    2001:503:d414::30
g.edu-servers.net.      172800  IN      A       192.42.93.30
```

```
atouche@ada:~$ dig @192.35.51.30 +tcp +norecurse pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @192.35.51.30 +tcp +norecurse pdx.edu
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52872
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;pdx.edu.                       IN      A

;; AUTHORITY SECTION:
pdx.edu.                172800  IN      NS      ns-cloud-e1.googledomains.com.
pdx.edu.                172800  IN      NS      ns-cloud-e2.googledomains.com.
pdx.edu.                172800  IN      NS      ns-cloud-e3.googledomains.com.
pdx.edu.                172800  IN      NS      ns-cloud-e4.googledomains.com.

;; Query time: 63 msec
;; SERVER: 192.35.51.30#53(192.35.51.30) (TCP)
;; WHEN: Mon Oct 10 14:52:08 PDT 2022
;; MSG SIZE  rcvd: 157
```

```
atouche@ada:~$ dig @ns-cloud-e1.googledomains.com +tcp +norecurse cs.pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @ns-cloud-e1.googledomains.com +tcp +norecurse cs.pdx.edu
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9331
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 3, ADDITIONAL: 4

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;cs.pdx.edu.                    IN      A

;; AUTHORITY SECTION:
cs.pdx.edu.             14400   IN      NS      walt.ee.pdx.edu.
cs.pdx.edu.             14400   IN      NS      dns1.pdx.edu.
cs.pdx.edu.             14400   IN      NS      dns0.pdx.edu.

;; ADDITIONAL SECTION:
dns0.pdx.edu.           14400   IN      A       131.252.120.128
dns1.pdx.edu.           14400   IN      A       131.252.120.129
walt.ee.pdx.edu.        14400   IN      A       131.252.208.38

;; Query time: 59 msec
;; SERVER: 216.239.32.110#53(ns-cloud-e1.googledomains.com) (TCP)
;; WHEN: Mon Oct 10 14:52:54 PDT 2022
;; MSG SIZE  rcvd: 147
```

```
atouche@ada:~$ dig @walt.ee.pdx.edu +tcp +norecurse linux.cs.pdx.edu

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @walt.ee.pdx.edu +tcp +norecurse linux.cs.pdx.edu
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33290
;; flags: qr aa ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 1f11288a5a6d619701000000634494acd890dffa83bc0bf2 (good)
;; QUESTION SECTION:
;linux.cs.pdx.edu.               IN      A

;; ANSWER SECTION:
linux.cs.pdx.edu.       14400   IN      CNAME   ada.cs.pdx.edu.
ada.cs.pdx.edu.         14400   IN      A       131.252.208.103

;; AUTHORITY SECTION:
cs.pdx.edu.             14400   IN      NS      dns0.pdx.edu.
cs.pdx.edu.             14400   IN      NS      walt.ee.pdx.edu.
cs.pdx.edu.             14400   IN      NS      dns1.pdx.edu.

;; Query time: 0 msec
;; SERVER: 131.252.208.38#53(walt.ee.pdx.edu) (TCP)
;; WHEN: Mon Oct 10 14:54:52 PDT 2022
;; MSG SIZE  rcvd: 181
```

☐ Reverse DNS Lookup

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

```
atouche@ada:~/Documents/cs430p/lab/lab2$ dig @131.252.208.53 +tcp -t A espn.go.com | egrep espn | awk '{print $5}'
<<>>

99.84.66.108
99.84.66.17
99.84.66.55
99.84.66.98
```

B. Take that list and create a single for loop in the shell that iterates over the list and performs a reverse lookup of each IP address to find each address's associated DNS name. As with the previous step, pipe the output of the for loop to egrep and awk so that the output consists only of the DNS names.

```
atouche@ada:~/Documents/cs430p/lab/lab2$ cat alias_rev_dns.sh
X=`dig +tcp -t A espn.go.com | egrep espn | awk '{print $5}'`
for i in `echo $X`
do
  dig -x $i +short
done

atouche@ada:~/Documents/cs430p/lab/lab2$ ./alias_rev_dns.sh
server-99-84-66-17.hio50.r.cloudfront.net.
server-99-84-66-108.hio50.r.cloudfront.net.
server-99-84-66-98.hio50.r.cloudfront.net.
server-99-84-66-55.hio50.r.cloudfront.net.
```

- [ ] Host enumeration
    A. Perform a DNS reverse lookup on all IP addresses on 131.252.220.0/24 subnet, output host names to 220hosts.txt and concatenate host names of car manufacturers:

```
atouche@ada:~/Documents/cs430p/lab/lab2$ cat 220hosts.txt | head -184 | tail -29
acura.cs.pdx.edu.
astonmartin.cs.pdx.edu.
audi.cs.pdx.edu.
bentley.cs.pdx.edu.
bmw.cs.pdx.edu.
cadillac.cs.pdx.edu.
ferrari.cs.pdx.edu.
fiat.cs.pdx.edu.
ford.cs.pdx.edu.
honda.cs.pdx.edu.
hummer.cs.pdx.edu.
jaguar.cs.pdx.edu.
jeep.cs.pdx.edu.
lamborghini.cs.pdx.edu.
landrover.cs.pdx.edu.
lexus.cs.pdx.edu.
lotus.cs.pdx.edu.
maserati.cs.pdx.edu.
mazda.cs.pdx.edu.
mclaren.cs.pdx.edu.
mercedes.cs.pdx.edu.
nissan.cs.pdx.edu.
panoz.cs.pdx.edu.
porsche.cs.pdx.edu.
subaru.cs.pdx.edu.
toyota.cs.pdx.edu.
tvr.cs.pdx.edu.
ultima.cs.pdx.edu.
volvo.cs.pdx.edu.
```

- [ ] DNS #2 (geographic DNS)
    a. Lookup Geographical DNS for:
        - [ ] 131.252.208.53
            - [ ] **ipinfo.io** location is Portland State University, Portland, Oregon US
            - [ ] **DB-IP** location is Portland State University, Portland (North Portland), Oregon US
        - [ ] 198.82.247.66
            - [ ] **ipinfo.io** location is Virginia Polytechnic Institute and State Univ., Blacksburg, VA
            - [ ] **DB-IP** location is Virginia Polytechnic Institute and State Univ., Blacksburg (Farmview - Ramble), VA

b. Resolve www.google.com w/ each DNS server IP:

```
atouche@ada:~$  dig www.google.com

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2484
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 339fb501abc98fef010000006343e64a317d6740d6399498 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         50      IN      A       142.251.211.228

;; Query time: 0 msec
;; SERVER: 131.252.208.53#53(131.252.208.53) (UDP)
;; WHEN: Mon Oct 10 02:30:50 PDT 2022
;; MSG SIZE  rcvd: 87

atouche@ada:~$ dig @198.82.247.66 www.google.com

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> @198.82.247.66 www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22753
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 1e065801c10d2d2006322fc46343e64e2a11c71a401a3359 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         192     IN      A       172.217.13.228

;; Query time: 71 msec
;; SERVER: 198.82.247.66#53(198.82.247.66) (UDP)
;; WHEN: Mon Oct 10 02:30:54 PDT 2022
;; MSG SIZE  rcvd: 87
```

c. Go back to iplocation and lookup geo data of each IP address returned. What geo locations do ipinfo.io and DB-UP return?

- [ ] 142.251.211.228

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

IP Address	Country	Region	City
142.251.211.228	United States	Washington	Seattle
ISP	**Organization**	**Latitude**	**Longitude**
Google LLC	Google LLC (google.com)	47.6062	-122.3321

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

IP Address	Country	Region	City
142.251.211.228	United States	Washington	Seattle
ISP	**Organization**	**Latitude**	**Longitude**
Google LLC	Google LLC	47.6062	-122.332

- [ ] 172.217.13.228

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

IP Address	Country	Region	City
172.217.13.228	United States	Virginia	Alexandria
ISP	**Organization**	**Latitude**	**Longitude**
Google LLC	Google LLC (google.com)	38.8048	-77.0469

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

IP Address	Country	Region	City
172.217.13.228	United States	District of Columbia	Washington D.C.
ISP	**Organization**	**Latitude**	**Longitude**
Google LLC	Google LLC	38.9072	-77.0369

- [ ] **What is the geographic distance between each pair of DNS server and web server?**

- Distance between 142.251.211.228 and www.google.com (A CDN on the west coast) is about 175 miles
- Distance between 172.217.13.228 and www.google.com (A CDN on the east coast) is about 273 miles

d. Perform a traceroute to all 4 IP addresses from a PSU network

- [ ] **Do the routes reveal any information on the accuracy of the geographic locations given? (Answer might be no)**
  - [ ] 131.252.208

```
atouche@ada:~/Documents/cs430p/lab/lab2$ traceroute 131.252.208
traceroute to 131.252.208 (131.252.0.208), 30 hops max, 60 byte packets
 1  radiant.seas.pdx.edu (131.252.208.212)  1.191 ms  1.172 ms  1.262 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```

  - [ ] 198.82.247.66

```
atouche@ada:~/Documents/cs430p/lab/lab2$ traceroute 198.82.247.66
traceroute to 198.82.247.66 (198.82.247.66), 30 hops max, 60 byte packets
 1  radiant.seas.pdx.edu (131.252.208.212)  1.083 ms  1.583 ms  1.692 ms
 2  CORE1.net.pdx.edu (131.252.5.142)  0.566 ms  0.540 ms  0.506 ms
 3  131.252.5.213 (131.252.5.213)  1.656 ms  1.237 ms  0.947 ms
 4  port-psu-pe-01.net.linkoregon.org (199.165.177.48)  1.326 ms  0.689 ms  0.652 ms
 5  eugn-oh-vpn-01.net.linkoregon.org (207.98.126.3)  10.717 ms  10.531 ms  10.355 ms
 6  bois-gtwy-pe-01.net.linkoregon.org (207.98.126.135)  10.662 ms  10.654 ms  10.685 ms
 7  bois-gtwy-pe-01-loren.net.linkoregon.org (163.253.5.65)  10.270 ms  10.169 ms  10.145 ms
 8  hundredge-0-0-0-24.703.core1.bois.net.internet2.edu (163.253.5.64)  13.398 ms  13.041 ms  13.003 ms
 9  fourhundredge-0-0-0-0.4079.core2.salt.net.internet2.edu (163.253.1.249)  65.459 ms  65.433 ms  65.420 ms
10  fourhundredge-0-0-0-23.4079.core1.salt.net.internet2.edu (163.253.1.32)  64.901 ms  65.592 ms fourhundredge-0-0-0-22.4079.core1.salt
.net.internet2.edu (163.253.1.30)  65.472 ms
11  fourhundredge-0-0-0-0.4079.core1.denv.net.internet2.edu (163.253.1.170)  64.943 ms  66.107 ms  66.981 ms
12  fourhundredge-0-0-0-0.4079.core1.kans.net.internet2.edu (163.253.1.243)  65.883 ms  65.877 ms  65.754 ms
13  fourhundredge-0-0-0-3.4079.core2.chic.net.internet2.edu (163.253.1.244)  66.739 ms  64.676 ms  65.646 ms
14  fourhundredge-0-0-0-3.4079.core2.eqch.net.internet2.edu (163.253.2.19)  66.085 ms  64.704 ms  65.736 ms
15  fourhundredge-0-0-0-0.4079.core2.clev.net.internet2.edu (163.253.2.16)  65.434 ms  65.386 ms  64.784 ms
16  fourhundredge-0-0-0-3.4079.core2.ashb.net.internet2.edu (163.253.1.138)  65.050 ms  65.011 ms  65.068 ms
17  192.122.175.14 (192.122.175.14)  63.416 ms  64.803 ms  65.038 ms
18  vtacs-1.msap.cns.vt.edu (192.70.187.18)  69.195 ms  69.146 ms  69.441 ms
19  isb-core.et-5-1-0.0.cns.vt.edu (128.173.0.206)  69.573 ms  69.582 ms  69.648 ms
20  cas-core.lo0.2000.cns.vt.edu (198.82.1.143)  69.496 ms  69.455 ms  69.432 ms
21  jeru.cns.vt.edu (198.82.247.66)  69.761 ms  69.756 ms  69.518 ms
```

- [ ] 142.251.211.228

```
atouche@ada:~/Documents/cs430p/lab/lab2$ traceroute 142.251.211.228
traceroute to 142.251.211.228 (142.251.211.228), 30 hops max, 60 byte packets
 1  radiant.seas.pdx.edu (131.252.208.212)  1.289 ms  1.389 ms  1.484 ms
 2  CORE1.net.pdx.edu (131.252.5.142)  0.709 ms  0.686 ms  0.642 ms
 3  131.252.5.213 (131.252.5.213)  0.983 ms  0.960 ms  1.203 ms
 4  google.nwax.net (198.32.195.34)  4.161 ms  4.799 ms  4.733 ms
 5  74.125.243.177 (74.125.243.177)  5.477 ms  5.510 ms  5.406 ms
 6  216.239.43.231 (216.239.43.231)  4.476 ms 216.239.43.121 (216.239.43.121)  4.707 ms 216.239.43.231 (216.239.43.231)  4.651 ms
 7  sea30s13-in-f4.1e100.net (142.251.211.228)  5.984 ms  5.090 ms  4.406 ms
atouche@ada:~/Documents/cs430p/lab/lab2$
```

- [ ] 172.217.13.228

```
atouche@ada:~/Documents/cs430p/lab/lab2$ traceroute 172.217.13.228
traceroute to 172.217.13.228 (172.217.13.228), 30 hops max, 60 byte packets
 1  radiant.seas.pdx.edu (131.252.208.212)  1.306 ms  1.379 ms  1.475 ms
 2  CORE1.net.pdx.edu (131.252.5.142)  0.762 ms  0.727 ms  0.690 ms
 3  131.252.5.213 (131.252.5.213)  1.234 ms  1.078 ms  1.094 ms
 4  google.nwax.net (198.32.195.34)  3.788 ms  12.761 ms  3.917 ms
 5  108.170.245.123 (108.170.245.123)  4.825 ms 74.125.243.195 (74.125.243.195)  4.541 ms 74.125.243.194 (74.125.243.194)  4.794 ms
 6  216.239.57.194 (216.239.57.194)  11.277 ms 216.239.63.6 (216.239.63.6)  11.180 ms 216.239.41.34 (216.239.41.34)  18.491 ms
 7  142.250.213.69 (142.250.213.69)  52.903 ms * 142.250.213.61 (142.250.213.61)  59.142 ms
 8  * 142.251.64.248 (142.251.64.248)  67.792 ms 142.251.64.254 (142.251.64.254)  65.989 ms
 9  142.250.236.132 (142.250.236.132)  68.019 ms 172.253.74.192 (172.253.74.192)  68.148 ms 142.250.236.136 (142.250.236.136)  68.121 ms
10  142.251.49.72 (142.251.49.72)  66.653 ms 108.170.232.198 (108.170.232.198)  66.146 ms 216.239.49.196 (216.239.49.196)  75.204 ms
11  108.170.246.33 (108.170.246.33)  66.698 ms 108.170.246.65 (108.170.246.65)  66.695 ms 108.170.246.33 (108.170.246.33)  65.931 ms
12  108.170.232.213 (108.170.232.213)  66.305 ms  66.282 ms  65.594 ms
13  iad23s61-in-f4.1e100.net (172.217.13.228)  65.508 ms  65.400 ms  64.990 ms
```

- [ ] Since routes are not static and since Multipath Discovery Algo — although useful — may be limited in measuring accuracy in runtime in conjunction to route changes during traceroute operation.

- [ ] Network Recap Lab#3
  A. Use ip cmd

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

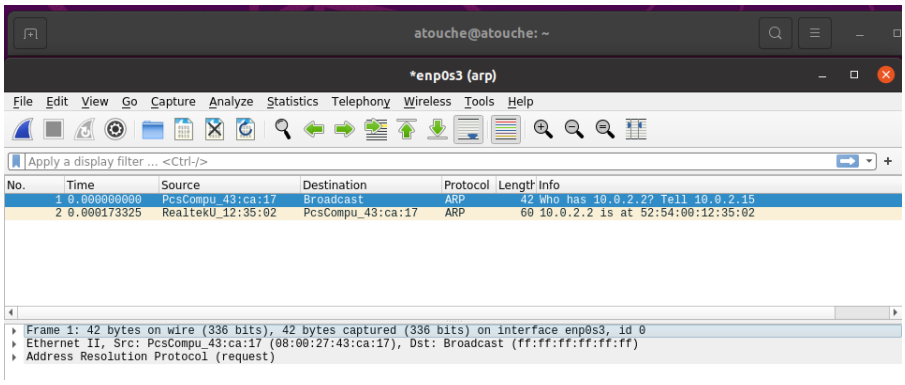
  B. Use netstat

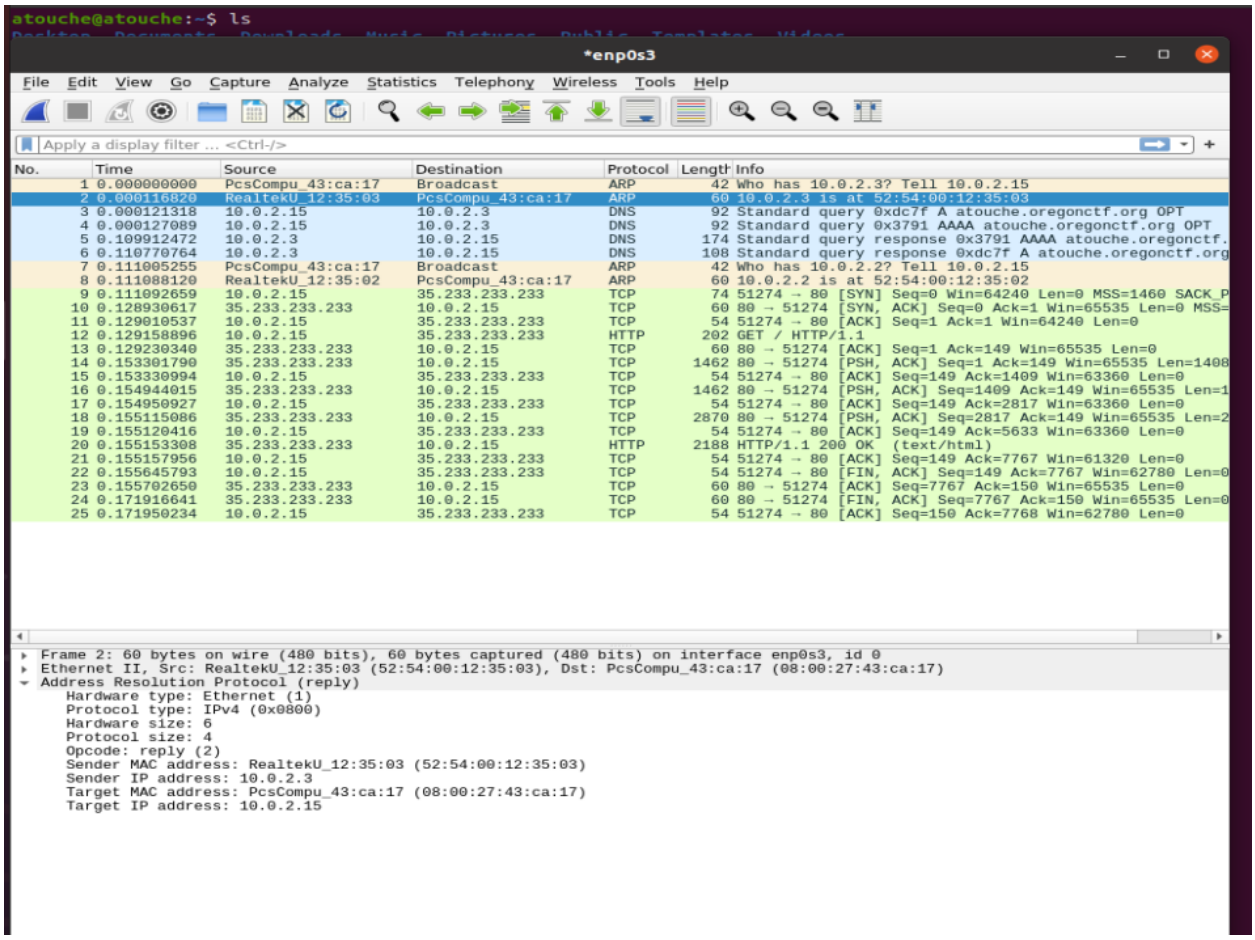
```
atouche@atouche:~$ netstat -nr
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         10.0.2.2        0.0.0.0         UG        0 0          0 enp0s3
10.0.2.0        0.0.0.0         255.255.255.0   U         0 0          0 enp0s3
169.254.0.0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

  C. Temp change DNS
  D. Dump ARP table
     a. $ arp -an | awk -F '[()]' '{print $2}' > arp_entrie

- [ ] Collect and analyze the network trace of connection
  A. Analyze trace

a. **Take a screenshot of the trace within Wireshark and include an annotation of the packets in the trace to explain the purpose of each of the packets being exchanged.**





b. No.1 packet is the ARP request packet that contains the source MAC address and the source IP address and the destination IP address. No.2 packet updates the requestor's ARP cache for future reference.

B. How many DNS request are made

a. 2 queries/requests and 2 replies receives

C. How many TCP connections does the browser initiate simultaneously to the site?

a. 2 connection

D. How many HTTP GET requests are there for embedded objects?

a. 12th packet is the only HTTP GET for embedded object