GitHub Repository Link: https://github.com/mrscot18/ser321-summer2022-C-mrscot-.git

2. Command line tasks (15 points)

System being used is a terminal on my MAC computer - maddyscott@Maddys-MacBook-Pro

```
1.  mkdir cli_assignment
2.  cd cli_assignment/
3.  touch stuff.txt
4.  cat>stuff.txt -→ CONTROL + d
5.  wc -w stuff.txt
6.  cat>>stuff.txt
7.  mkdir draft
8.  mv stuff.txt draft
9.  cd draft →touch .secret.txt
10. cd.. →cp -R draft final
11. mv draft draft.remove
12. mv draft.remove final
13. ls -a -l -R
14. vim NASA_access_log_Aug95.gz
15. gunzip NASA_access_log_Aug95.gz
16. mv NASA_access_log_Aug95 logs.txt
17. mv logs.txt cli_assignment
18. head -100 logs.txt | cat -n
19. head -100 logs.txt |cat>logs_top_100.txt
20. tail -100 logts.txt
21. tail -100 logs.txt |cat>logs_top_100.txt
22. cat logs_top_100.txt logs_bottom_100.txt>logs_snapshot.txt
23. echo “mrscot: This is a great assignment 05/19/2022”>>logs_snapshot.txt
24. less logs.txt
25. column -t -d marks.csv | cut -d ‘%’ -f1 | sed “1 d”
26. column -t -d marks.csv | cut -d ‘%’ -f4 | sort -g
27. $ column -td marks.csv | cut -d ‘%’ -f3 | sed “1 d” | awk ‘{total += $1} END
    {print(total/NR)}’
28. $ column -td marks.csv | cut -d ‘%’ -f3 | sed “1 d” | awk ‘{total += $1} END
    {print(total/NR)}’ > done.txt
29. mv done.txt cli_assignment/final
30. mv done.txt average.txt
```

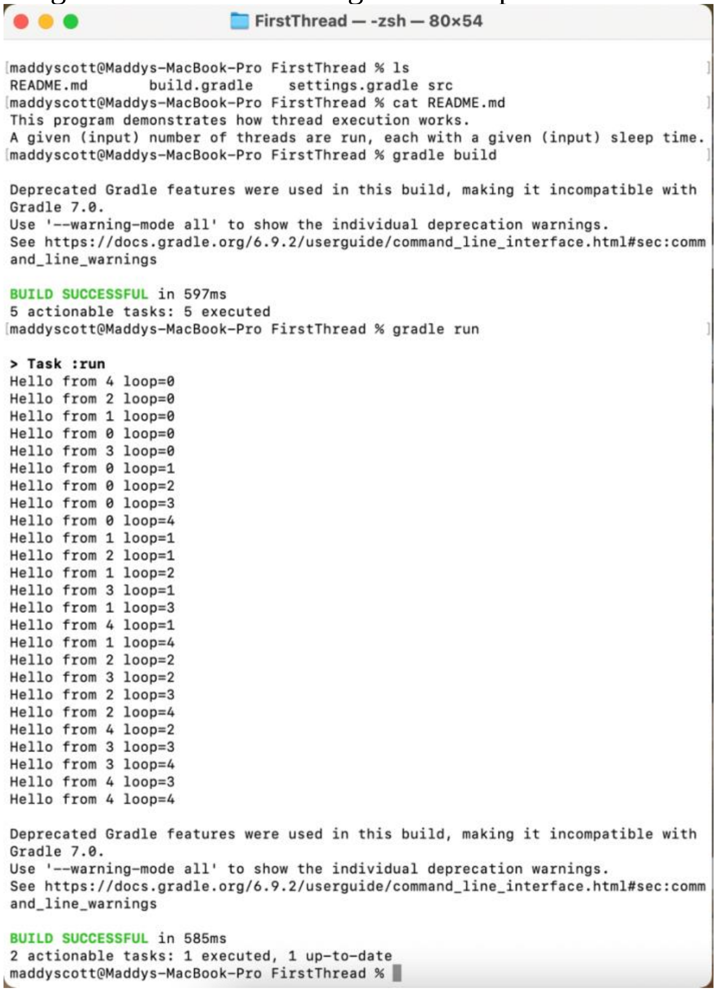
3. Some Setup and Examples (30 points)

3.1. Setup a GitHub repo to submit your assignments (5 points)

GitHub Repository Link: https://github.com/mrscot18/ser321-summer2022-C-mrscot-.git

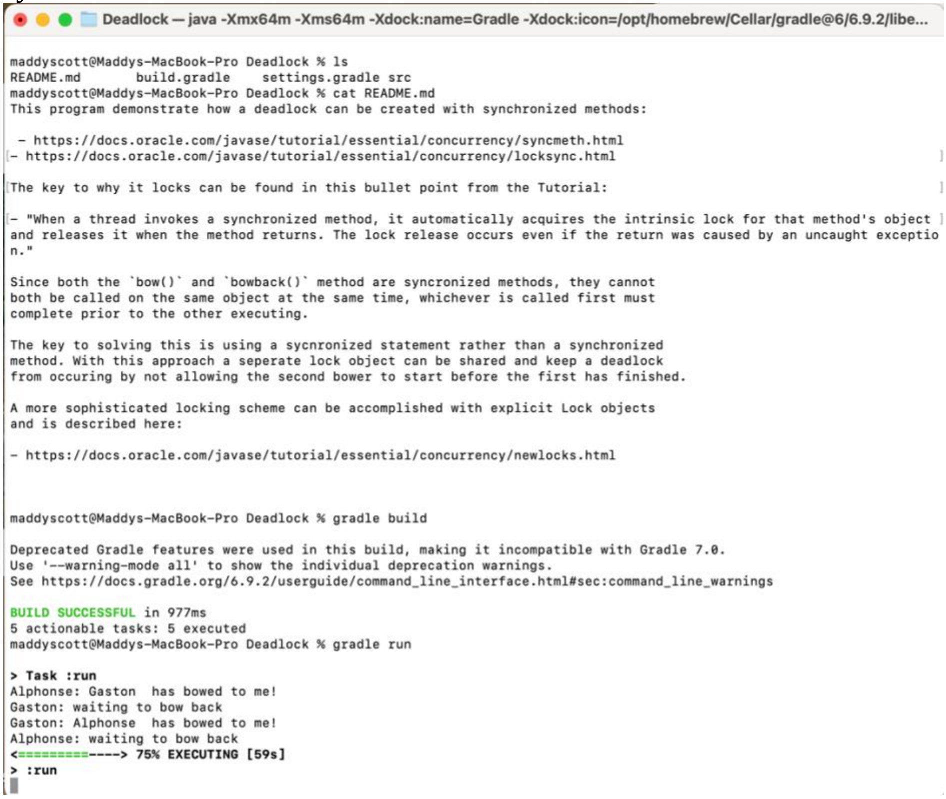
3.2. Running examples (10 points)

Example 1: First Thread. This example shows how thread execution works. A number of threads are given and run and also given a sleep time.



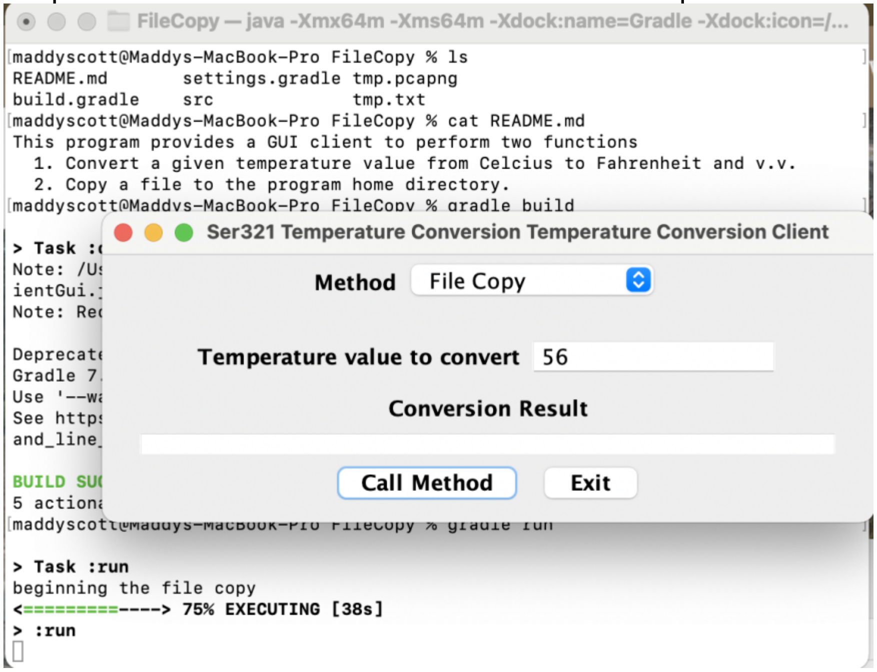
```
maddyscott@Maddys-MacBook-Pro FirstThread % ls
README.md       build.gradle    settings.gradle src
maddyscott@Maddys-MacBook-Pro FirstThread % cat README.md
This program demonstrates how thread execution works.
A given (input) number of threads are run, each with a given (input) sleep time.
maddyscott@Maddys-MacBook-Pro FirstThread % gradle build

Deprecated Gradle features were used in this build, making it incompatible with
Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.9.2/userguide/command_line_interface.html#sec:comm
and_line_warnings

BUILD SUCCESSFUL in 597ms
5 actionable tasks: 5 executed
maddyscott@Maddys-MacBook-Pro FirstThread % gradle run

> Task :run
Hello from 4 loop=0
Hello from 2 loop=0
Hello from 1 loop=0
Hello from 0 loop=0
Hello from 3 loop=0
Hello from 0 loop=1
Hello from 0 loop=2
Hello from 0 loop=3
Hello from 0 loop=4
Hello from 1 loop=1
Hello from 2 loop=1
Hello from 1 loop=2
Hello from 3 loop=1
Hello from 1 loop=3
Hello from 4 loop=1
Hello from 1 loop=4
Hello from 2 loop=2
Hello from 3 loop=2
Hello from 2 loop=3
Hello from 2 loop=4
Hello from 4 loop=2
Hello from 3 loop=3
Hello from 3 loop=4
Hello from 4 loop=3
Hello from 4 loop=4

Deprecated Gradle features were used in this build, making it incompatible with
Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.9.2/userguide/command_line_interface.html#sec:comm
and_line_warnings

BUILD SUCCESSFUL in 585ms
2 actionable tasks: 1 executed, 1 up-to-date
maddyscott@Maddys-MacBook-Pro FirstThread %
```

Example 2: Deadlock (within threads). This example shows how a deadlock can be created with synchronized methods.


Deadlock — java -Xmx64m -Xms64m -Xdock:name=Gradle -Xdock:icon=/opt/homebrew/Cellar/gradle@6/6.9.2/libe...

```
maddyscott@Maddys-MacBook-Pro Deadlock % ls
README.md       build.gradle    settings.gradle src
maddyscott@Maddys-MacBook-Pro Deadlock % cat README.md
This program demonstrate how a deadlock can be created with synchronized methods:

 - https://docs.oracle.com/javase/tutorial/essential/concurrency/syncmeth.html
- https://docs.oracle.com/javase/tutorial/essential/concurrency/locksync.html

The key to why it locks can be found in this bullet point from the Tutorial:

- "When a thread invokes a synchronized method, it automatically acquires the intrinsic lock for that method's object
and releases it when the method returns. The lock release occurs even if the return was caused by an uncaught exceptio
n."

Since both the `bow()` and `bowback()` method are syncronized methods, they cannot
both be called on the same object at the same time, whichever is called first must
complete prior to the other executing.

The key to solving this is using a sycnronized statement rather than a synchronized
method. With this approach a seperate lock object can be shared and keep a deadlock
from occuring by not allowing the second bower to start before the first has finished.

A more sophisticated locking scheme can be accomplished with explicit Lock objects
and is described here:

- https://docs.oracle.com/javase/tutorial/essential/concurrency/newlocks.html



maddyscott@Maddys-MacBook-Pro Deadlock % gradle build

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/6.9.2/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 977ms
5 actionable tasks: 5 executed
maddyscott@Maddys-MacBook-Pro Deadlock % gradle run

> Task :run
Alphonse: Gaston  has bowed to me!
Gaston: waiting to bow back
Gaston: Alphonse  has bowed to me!
Alphonse: waiting to bow back
<=========----> 75% EXECUTING [59s]
> :run
```

Example 3: File Copy (Threads). This example has two functions, it converts a given temperature from Celsius to Fahrenheit. And then copies a file to the program home directory.


FileCopy — java -Xmx64m -Xms64m -Xdock:name=Gradle -Xdock:icon=/...
Ser321 Temperature Conversion Temperature Conversion Client
Conversion Result
Call Method
Exit

```
maddyscott@Maddys-MacBook-Pro FileCopy % ls
README.md       settings.gradle tmp.pcapng
build.gradle    src             tmp.txt
maddyscott@Maddys-MacBook-Pro FileCopy % cat README.md
This program provides a GUI client to perform two functions
  1. Convert a given temperature value from Celcius to Fahrenheit and v.v.
  2. Copy a file to the program home directory.
maddyscott@Maddys-MacBook-Pro FileCopy % gradle build

> Task :c
Note: /Us
ientGui.j
Note: Rec

Deprecate
Gradle 7.
Use '--wa
See https
and_line_

BUILD SUC
5 actiona
maddyscott@Maddys-MacBook-Pro FileCopy % gradle run

> Task :run
beginning the file copy
<=========----> 75% EXECUTING [38s]
> :run
```

Method: File Copy

Temperature value to convert: 56

3.3. Understanding Gradle (7.5 points)

Java Gradle Folder within my assignment repo “ser321-summer2022-C-mrscot” in my Assignment 1 Directory.

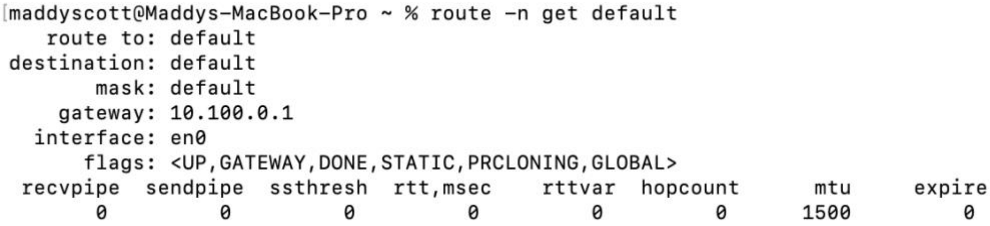
3.4. Set up your second system (7.5 points)

I will be using AWS as my second system.

https://drive.google.com/file/d/1rehn2iGsYTOVij21lbDyOJ2nAtl0n2ki/view?usp=sharing

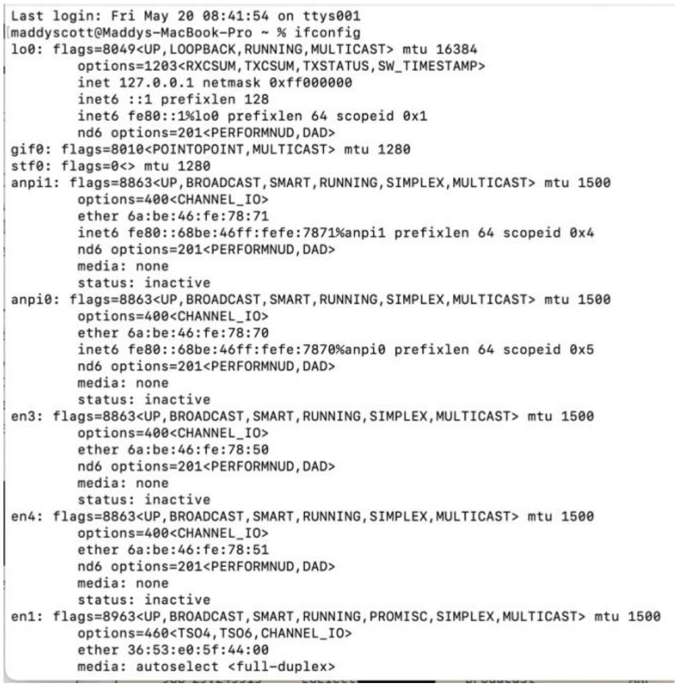
Part II. Networking

4.1 Explore the Data Link Layer with ARP



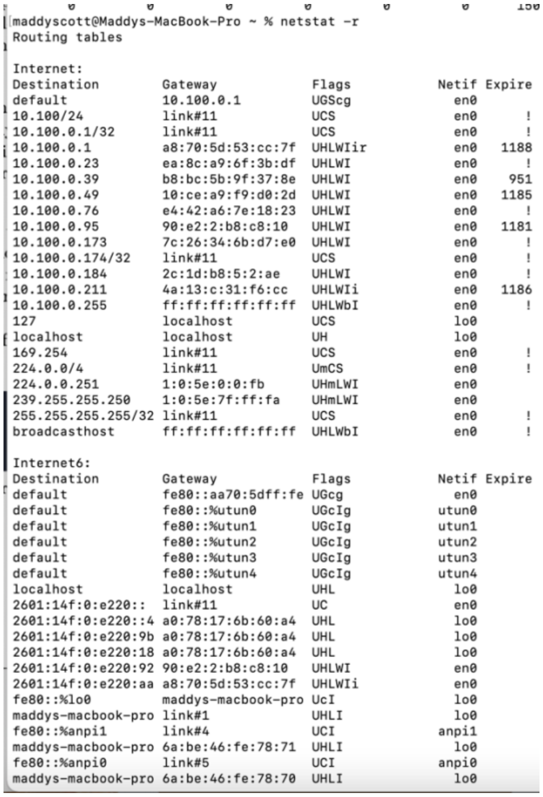
```
maddyscott@Maddys-MacBook-Pro ~ % route -n get default
   route to: default
destination: default
       mask: default
    gateway: 10.100.0.1
  interface: en0
      flags: <UP,GATEWAY,DONE,STATIC,PRCLONING,GLOBAL>
 recvpipe  sendpipe  ssthresh  rtt,msec    rttvar  hopcount      mtu     expire
       0         0         0         0         0         0      1500         0
```

Figure 1: route -n get default



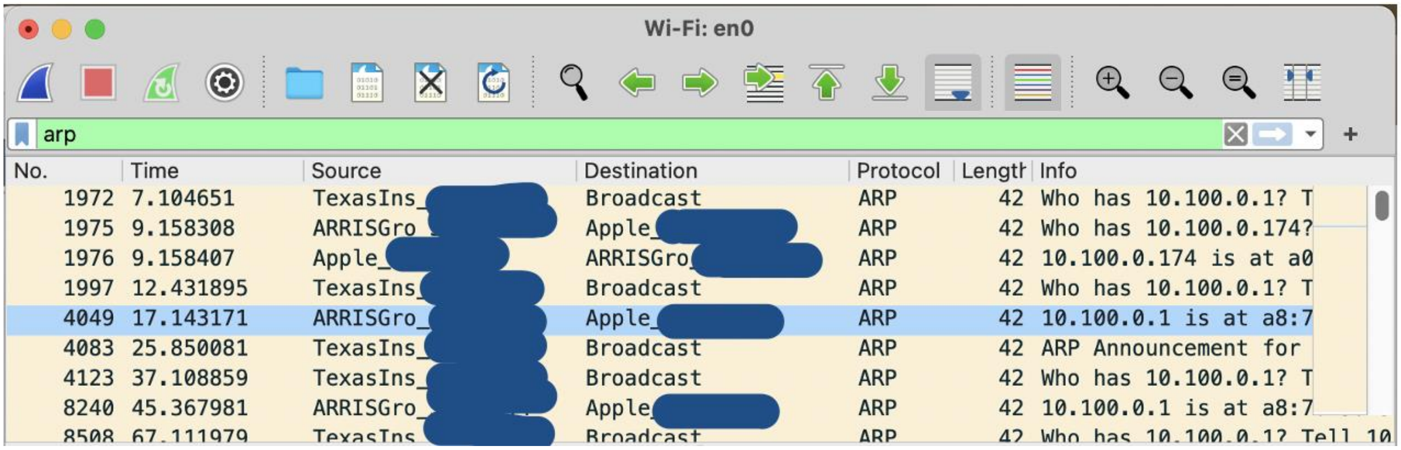
```
Last login: Fri May 20 08:41:54 on ttys001
maddyscott@Maddys-MacBook-Pro ~ % ifconfig
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
	options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
	inet 127.0.0.1 netmask 0xff000000
	inet6 ::1 prefixlen 128
	inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
	nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
anpi1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
	options=400<CHANNEL_IO>
	ether 6a:be:46:fe:78:71
	inet6 fe80::68be:46ff:fefe:7871%anpi1 prefixlen 64 scopeid 0x4
	nd6 options=201<PERFORMNUD,DAD>
	media: none
	status: inactive
anpi0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
	options=400<CHANNEL_IO>
	ether 6a:be:46:fe:78:70
	inet6 fe80::68be:46ff:fefe:7870%anpi0 prefixlen 64 scopeid 0x5
	nd6 options=201<PERFORMNUD,DAD>
	media: none
	status: inactive
en3: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
	options=400<CHANNEL_IO>
	ether 6a:be:46:fe:78:50
	nd6 options=201<PERFORMNUD,DAD>
	media: none
	status: inactive
en4: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
	options=400<CHANNEL_IO>
	ether 6a:be:46:fe:78:51
	nd6 options=201<PERFORMNUD,DAD>
	media: none
	status: inactive
en1: flags=8963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu 1500
	options=460<TSO4,TSO6,CHANNEL_IO>
	ether 36:53:e0:5f:44:00
	media: autoselect <full-duplex>
```

Figure 3: ifconfig



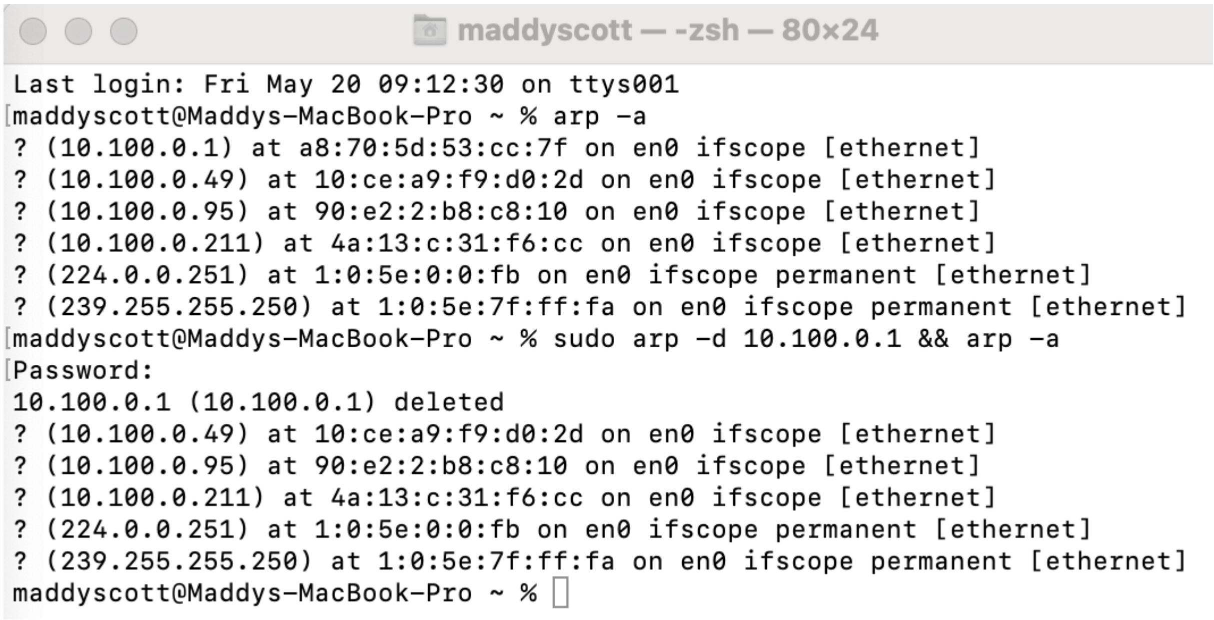
```
maddyscott@Maddys-MacBook-Pro ~ % netstat -r
Routing tables

Internet:
Destination        Gateway            Flags           Netif Expire
default            10.100.0.1         UGScg             en0
10.100/24          link#11            UCS               en0      !
10.100.0.1/32      link#11            UCS               en0      !
10.100.0.1         a8:70:5d:53:cc:7f  UHLWIir           en0   1188
10.100.0.23        ea:8c:a9:6f:3b:df  UHLWI             en0      !
10.100.0.39        b8:bc:5b:9f:37:8e  UHLWI             en0    951
10.100.0.49        10:ce:a9:f9:d0:2d  UHLWI             en0   1185
10.100.0.76        e4:42:a6:7e:18:23  UHLWI             en0      !
10.100.0.95        90:e2:2:b8:c8:10   UHLWI             en0   1181
10.100.0.173       7c:26:34:6b:d7:e0  UHLWI             en0      !
10.100.0.174/32    link#11            UCS               en0      !
10.100.0.184       2c:1d:b8:5:2:ae    UHLWI             en0      !
10.100.0.211       4a:13:c:31:f6:cc   UHLWIi            en0   1186
10.100.0.255       ff:ff:ff:ff:ff:ff  UHLWbI            en0      !
127                localhost          UCS               lo0
localhost          localhost          UH                lo0
169.254            link#11            UCS               en0      !
224.0.0/4          link#11            UmCS              en0      !
224.0.0.251        1:0:5e:0:0:fb      UHmLWI            en0
239.255.255.250    1:0:5e:7f:ff:fa    UHmLWI            en0
255.255.255.255/32 link#11            UCS               en0      !
broadcasthost      ff:ff:ff:ff:ff:ff  UHLWbI            en0      !

Internet6:
Destination        Gateway            Flags           Netif Expire
default            fe80::aa70:5dff:fe UGcg              en0
default            fe80::%utun0       UGcIg           utun0
default            fe80::%utun1       UGcIg           utun1
default            fe80::%utun2       UGcIg           utun2
default            fe80::%utun3       UGcIg           utun3
default            fe80::%utun4       UGcIg           utun4
localhost          localhost          UHL               lo0
2601:14f:0:e220::  link#11            UC                en0
2601:14f:0:e220::4 a0:78:17:6b:60:a4  UHL               lo0
2601:14f:0:e220:9b a0:78:17:6b:60:a4  UHL               lo0
2601:14f:0:e220:18 a0:78:17:6b:60:a4  UHL               lo0
2601:14f:0:e220:92 90:e2:2:b8:c8:10   UHLWI             en0
2601:14f:0:e220:aa a8:70:5d:53:cc:7f  UHLWIi            en0
fe80::%lo0         maddys-macbook-pro UcI               lo0
maddys-macbook-pro link#1             UHLI              lo0
fe80::%anpi1       link#4             UCI             anpi1
maddys-macbook-pro 6a:be:46:fe:78:71  UHLI              lo0
fe80::%anpi0       link#5             UCI             anpi0
maddys-macbook-pro 6a:be:46:fe:78:70  UHLI              lo0
```

Figure 2: netstat -r

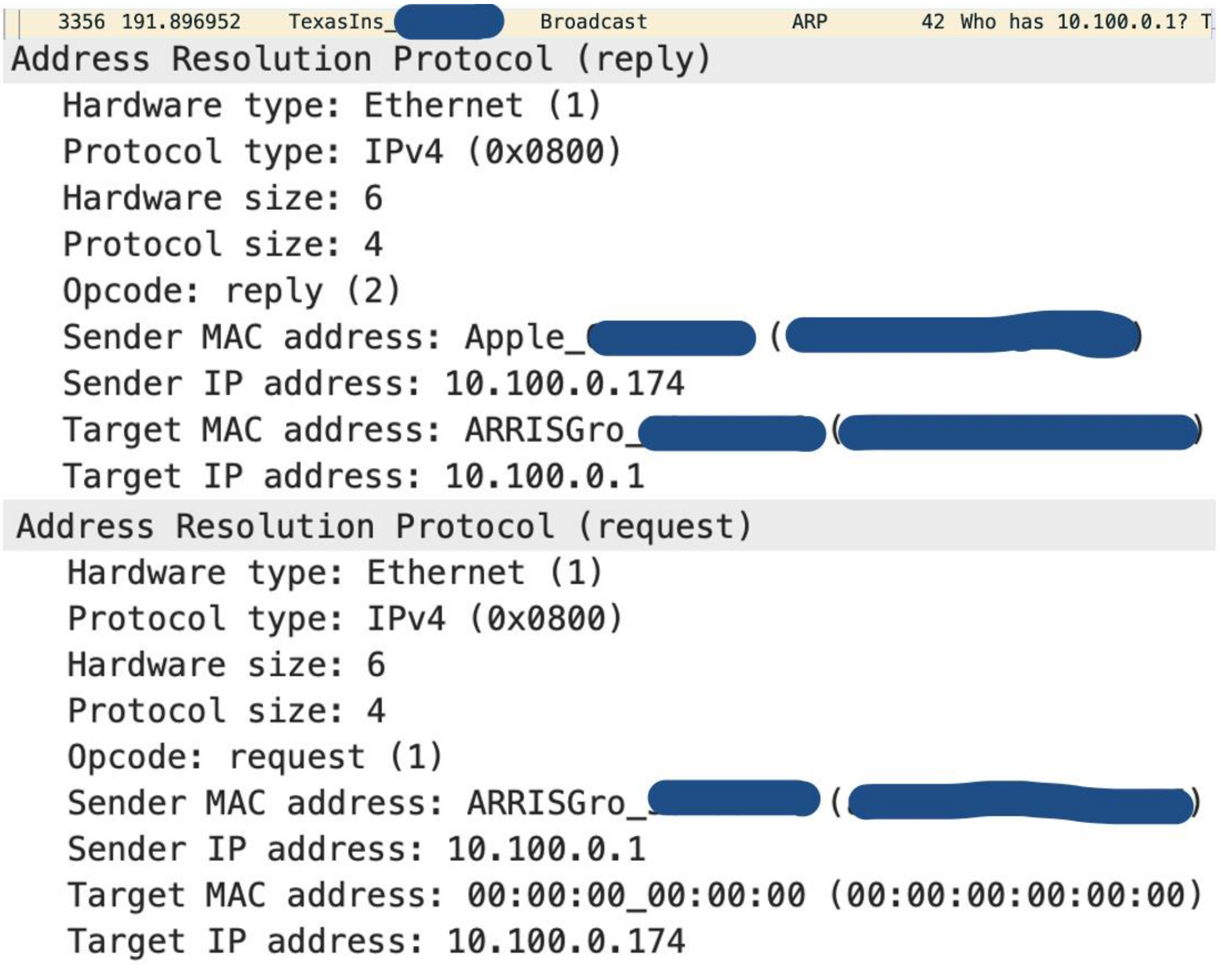


arp

No.	Time	Source	Destination	Protocol	Length	Info
1972	7.104651	TexasIns_ [redacted]	Broadcast	ARP	42	Who has 10.100.0.1? T
1975	9.158308	ARRISGro_ [redacted]	Apple_ [redacted]	ARP	42	Who has 10.100.0.174?
1976	9.158407	Apple_ [redacted]	ARRISGro [redacted]	ARP	42	10.100.0.174 is at a0
1997	12.431895	TexasIns_ [redacted]	Broadcast	ARP	42	Who has 10.100.0.1? T
4049	17.143171	ARRISGro_ [redacted]	Apple_ [redacted]	ARP	42	10.100.0.1 is at a8:7
4083	25.850081	TexasIns_ [redacted]	Broadcast	ARP	42	ARP Announcement for
4123	37.108859	TexasIns_ [redacted]	Broadcast	ARP	42	Who has 10.100.0.1? T
8240	45.367981	ARRISGro_ [redacted]	Apple [redacted]	ARP	42	10.100.0.1 is at a8:7
8508	67.111979	TexasIns [redacted]	Broadcast	ARP	42	Who has 10.100.0.1? Tell 10



```
Last login: Fri May 20 09:12:30 on ttys001
maddyscott@Maddys-MacBook-Pro ~ % arp -a
? (10.100.0.1) at a8:70:5d:53:cc:7f on en0 ifscope [ethernet]
? (10.100.0.49) at 10:ce:a9:f9:d0:2d on en0 ifscope [ethernet]
? (10.100.0.95) at 90:e2:2:b8:c8:10 on en0 ifscope [ethernet]
? (10.100.0.211) at 4a:13:c:31:f6:cc on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
? (239.255.255.250) at 1:0:5e:7f:ff:fa on en0 ifscope permanent [ethernet]
maddyscott@Maddys-MacBook-Pro ~ % sudo arp -d 10.100.0.1 && arp -a
Password:
10.100.0.1 (10.100.0.1) deleted
? (10.100.0.49) at 10:ce:a9:f9:d0:2d on en0 ifscope [ethernet]
? (10.100.0.95) at 90:e2:2:b8:c8:10 on en0 ifscope [ethernet]
? (10.100.0.211) at 4a:13:c:31:f6:cc on en0 ifscope [ethernet]
? (224.0.0.251) at 1:0:5e:0:0:fb on en0 ifscope permanent [ethernet]
? (239.255.255.250) at 1:0:5e:7f:ff:fa on en0 ifscope permanent [ethernet]
maddyscott@Maddys-MacBook-Pro ~ %
```



```
3356 191.896952    TexasIns_[redacted]    Broadcast          ARP       42 Who has 10.100.0.1? T
Address Resolution Protocol (reply)
   Hardware type: Ethernet (1)
   Protocol type: IPv4 (0x0800)
   Hardware size: 6
   Protocol size: 4
   Opcode: reply (2)
   Sender MAC address: Apple_[redacted] ([redacted])
   Sender IP address: 10.100.0.174
   Target MAC address: ARRISGro_[redacted] ([redacted])
   Target IP address: 10.100.0.1
Address Resolution Protocol (request)
   Hardware type: Ethernet (1)
   Protocol type: IPv4 (0x0800)
   Hardware size: 6
   Protocol size: 4
   Opcode: request (1)
   Sender MAC address: ARRISGro_[redacted] ([redacted])
   Sender IP address: 10.100.0.1
   Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
   Target IP address: 10.100.0.174
```

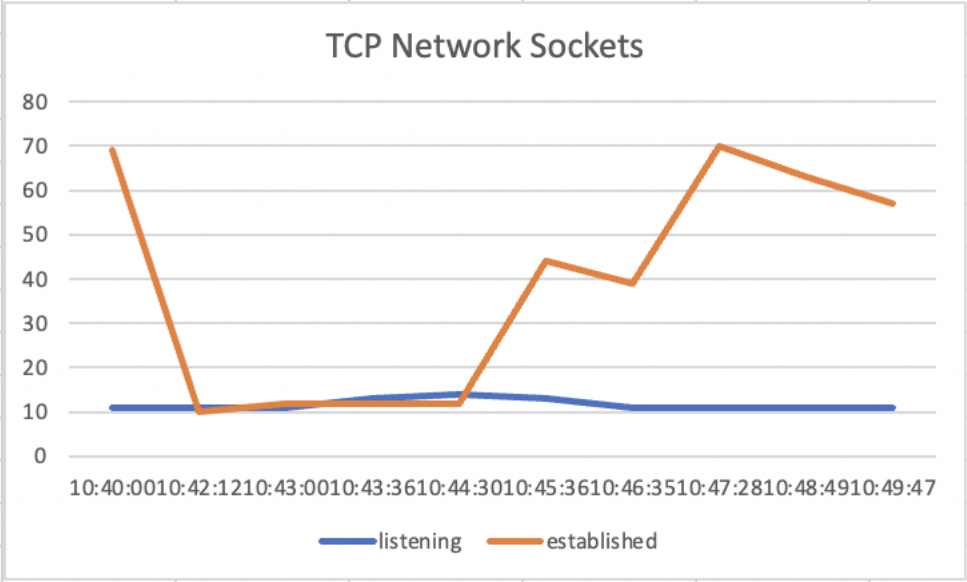
1. What opcode is used to indicate a request? What about a reply?
   a. Request opcode: request (1)
   b. Reply opcode: reply (2)

2. How large is the ARP header for a request? For a reply?
   a. For both of them, it is 28 bytes.

3. What value is carried on a request for the unknown target MAC address?
   a. ff:ff:ff:ff:ff:ff

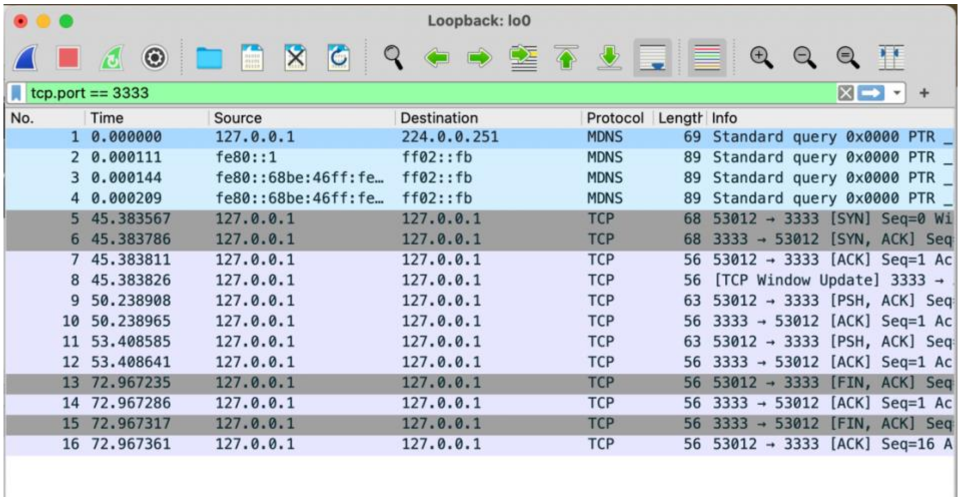
4. What Ethernet Type value indicates that ARP is the higher layer protocol?
   a. ARP (0x0806)

4.2 Understanding TCP network sockets



I used the command netstat -an in order to find the established and listening states.

4.3 Sniffing TCP/UDP traffic



tcp.port == 3333

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	224.0.0.251	MDNS	69	Standard query 0x0000 PTR _
2	0.000111	fe80::1	ff02::fb	MDNS	89	Standard query 0x0000 PTR _
3	0.000144	fe80::68be:46ff:fe…	ff02::fb	MDNS	89	Standard query 0x0000 PTR _
4	0.000209	fe80::68be:46ff:fe…	ff02::fb	MDNS	89	Standard query 0x0000 PTR _
5	45.383567	127.0.0.1	127.0.0.1	TCP	68	53012 → 3333 [SYN] Seq=0 Wi
6	45.383786	127.0.0.1	127.0.0.1	TCP	68	3333 → 53012 [SYN, ACK] Seq
7	45.383811	127.0.0.1	127.0.0.1	TCP	56	53012 → 3333 [ACK] Seq=1 Ac
8	45.383826	127.0.0.1	127.0.0.1	TCP	56	[TCP Window Update] 3333 →
9	50.238908	127.0.0.1	127.0.0.1	TCP	63	53012 → 3333 [PSH, ACK] Seq
10	50.238965	127.0.0.1	127.0.0.1	TCP	56	3333 → 53012 [ACK] Seq=1 Ac
11	53.408585	127.0.0.1	127.0.0.1	TCP	63	53012 → 3333 [PSH, ACK] Seq
12	53.408641	127.0.0.1	127.0.0.1	TCP	56	3333 → 53012 [ACK] Seq=1 Ac
13	72.967235	127.0.0.1	127.0.0.1	TCP	56	53012 → 3333 [FIN, ACK] Seq
14	72.967286	127.0.0.1	127.0.0.1	TCP	56	3333 → 53012 [ACK] Seq=1 Ac
15	72.967317	127.0.0.1	127.0.0.1	TCP	56	3333 → 53012 [FIN, ACK] Seq
16	72.967361	127.0.0.1	127.0.0.1	TCP	56	53012 → 3333 [ACK] Seq=16 A

a. Explain both the commands you used in detail. What did they actually do?

   a. The purpose of the command is to create a continuous listener on the port 3333. -k allows netcat to continuously listen for traffic over port 3333 while -l allows pot 3333 as a listener. This allowed netcat to run the connection as a TCP.

b. How many frames were sent back and forth to capture these 2 lines?

   a. 4 frames were needed to capture the lines.

c. How many packets were sent back and forth to capture the 2 lines?

   a. 4 packets were needed.

d. How many packets were needed to capture the whole process?

   a. 7 packets were needed.

e. How many bytes is the data that was sent?

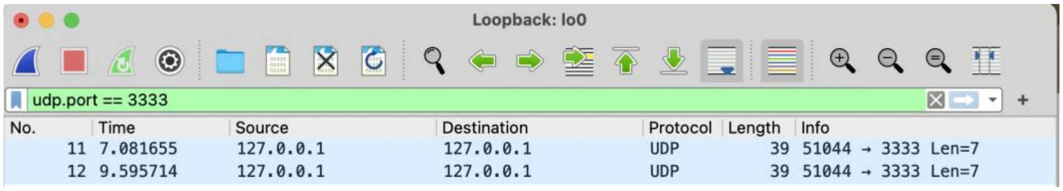
    a. 196

f. How many total bytes went over the wire for the whole process?

    a. 346 bytes

g. How much overhead was there?

    a. Around 55%



Loopback: lo0

udp.port == 3333

No.	Time	Source	Destination	Protocol	Length	Info
11	7.081655	127.0.0.1	127.0.0.1	UDP	39	51044 → 3333 Len=7
12	9.595714	127.0.0.1	127.0.0.1	UDP	39	51044 → 3333 Len=7

a. Explain both the commands you used in detail. What did they actually do?

    a. The purpose of the command is to create a continuous listener on the port 3333 via UDP. -k allows netcat to continuously listen for traffic over port 3333 while -l allows pot 3333 as a listener. This allowed netcat to run the connection as a UDP.

b. How many frames were sent back and forth to capture these 2 lines?

    a. 2 frames were needed to capture the lines.

c. How many packets were sent back and forth to capture the 2 lines?

    a. 2 packets were needed.

d. How many packets were needed to capture the whole process?

    a. 2 packets were needed.

e. How many total bytes went over the wire?

    a. 79 bytes. 100% of the overhead was a result of the two lines.

f. How many bytes is the data that was sent?

    a. 79 bytes

g. How much overhead was there?

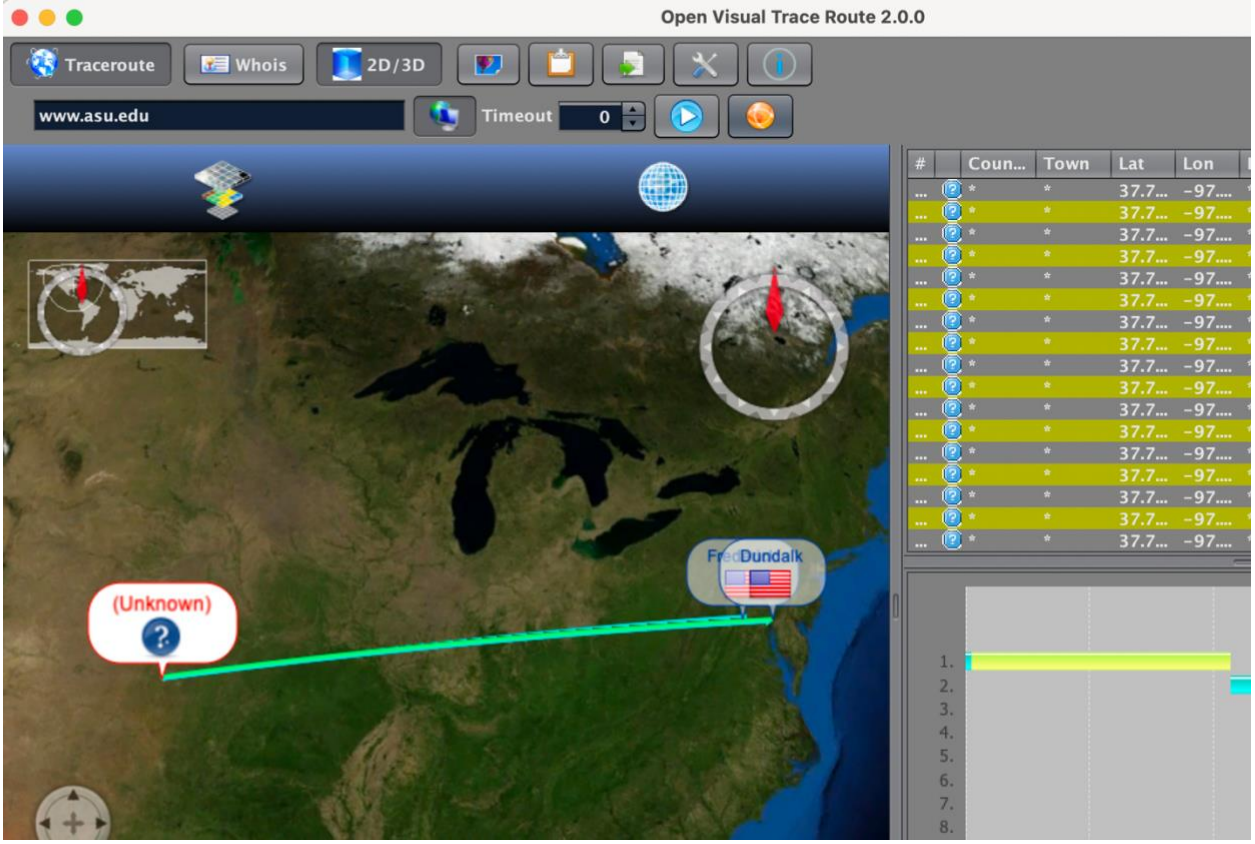
    a. 100%

h. What is the difference in relative overhead between UDP and TCP and why?

    a. UDP is connectionless while TCP is built on connection. The TCP header has a greater number of fields than the UDP header. Sequence number, TCP data offset, reserved data, acknowledgment number, control flags, urgent pointer, TCP optional data, and the window size are all TCP exchange information that is not included in the UDP exchange. The relative parts of the packet trace include the source, checksum field, and the destination port number.

4.4 Internet Protocol (IP) Routing

First Trace:

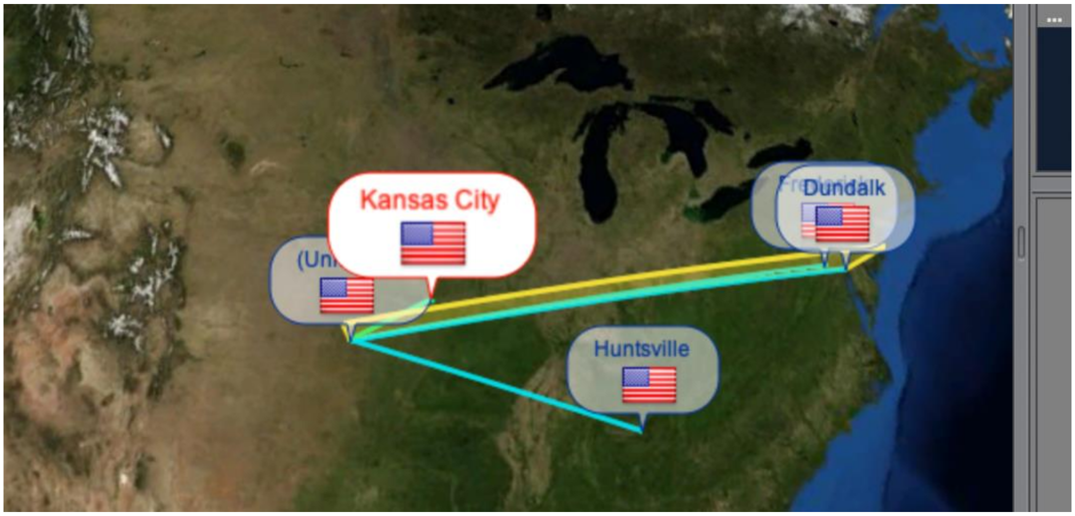


www.asu.edu

Timeout 0

#		Coun...	Town	Lat	Lon
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....
...	?	*	*	37.7...	-97....

Second Trace:



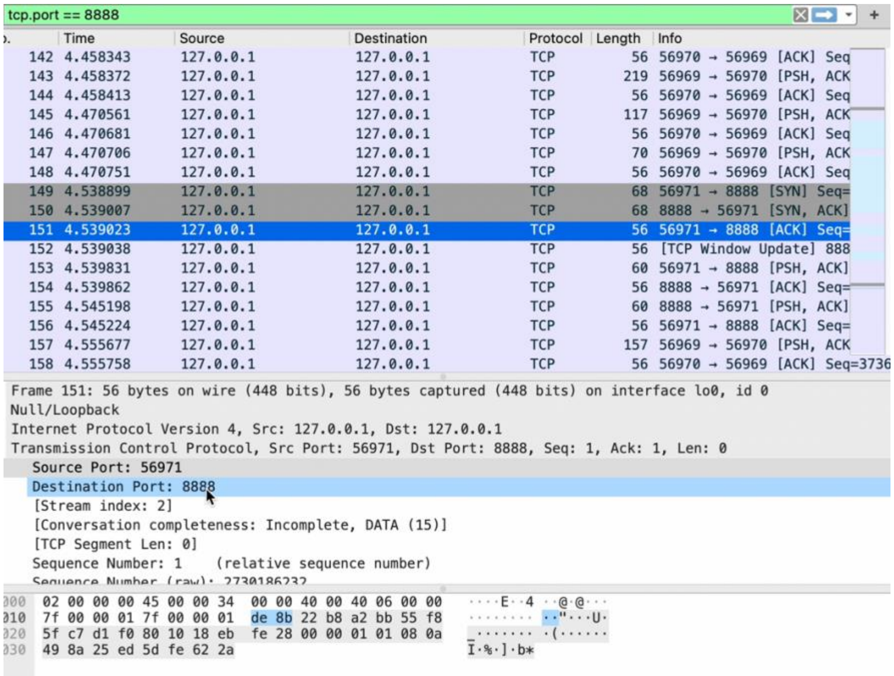
1. Which is fastest?
    a. The first route was much faster

2. Which has the fewest hops?
    a. The second route had the fewest hops

4.5 Running Client servers in different ways

4.5.1

https://drive.google.com/file/d/10focprqZGIuSt2sLhakfJ_1mwv5Ik_p9/view?usp=sharing

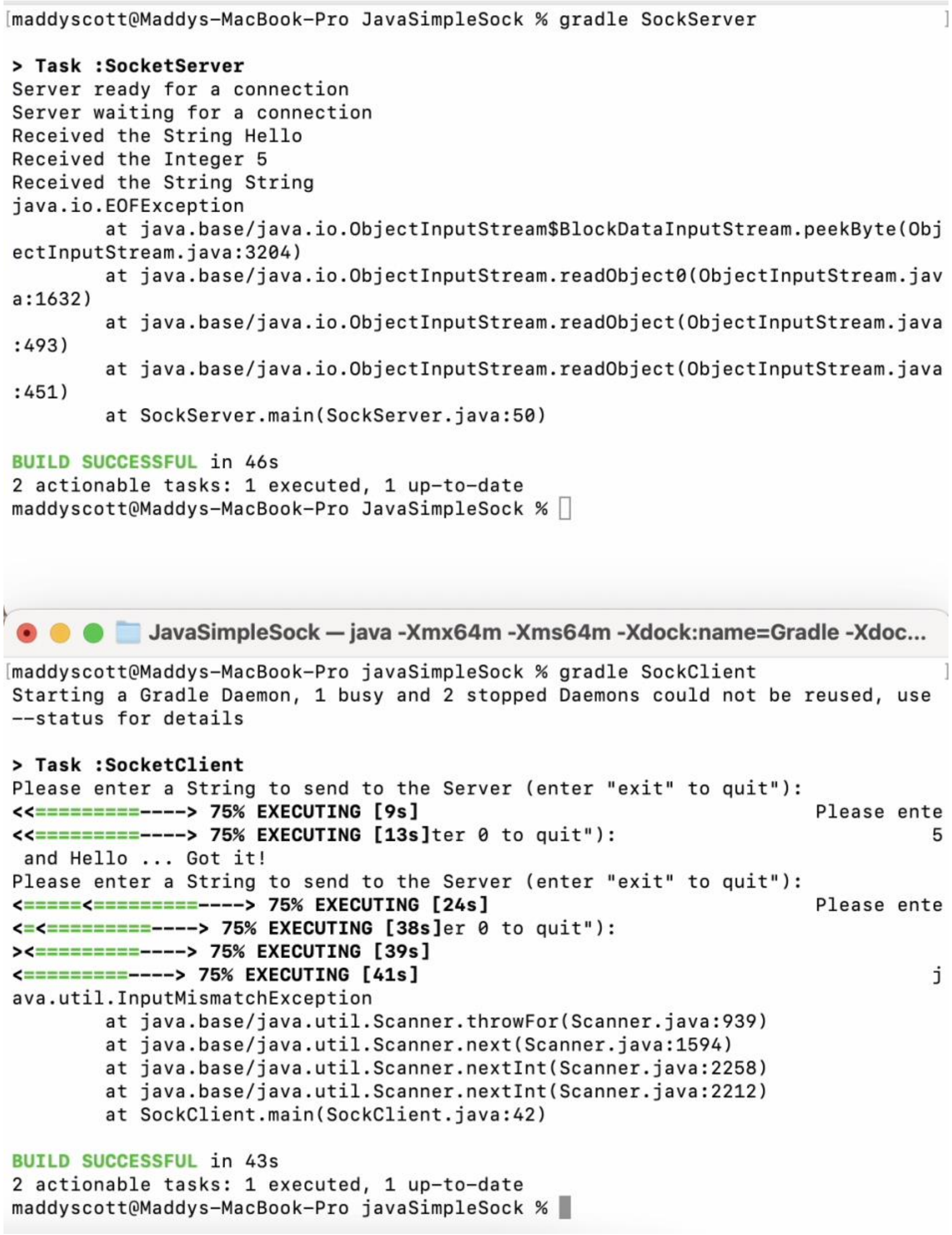


```
tcp.port == 8888
).   Time      Source      Destination   Protocol Length Info
 142 4.458343  127.0.0.1   127.0.0.1     TCP        56 56970 → 56969 [ACK] Seq
 143 4.458372  127.0.0.1   127.0.0.1     TCP       219 56969 → 56970 [PSH, ACK
 144 4.458413  127.0.0.1   127.0.0.1     TCP        56 56970 → 56969 [ACK] Seq
 145 4.470561  127.0.0.1   127.0.0.1     TCP       117 56969 → 56970 [PSH, ACK
 146 4.470681  127.0.0.1   127.0.0.1     TCP        56 56970 → 56969 [ACK] Seq
 147 4.470706  127.0.0.1   127.0.0.1     TCP        70 56969 → 56970 [PSH, ACK
 148 4.470751  127.0.0.1   127.0.0.1     TCP        56 56970 → 56969 [ACK] Seq
 149 4.538899  127.0.0.1   127.0.0.1     TCP        68 56971 → 8888 [SYN] Seq=
 150 4.539007  127.0.0.1   127.0.0.1     TCP        68 8888 → 56971 [SYN, ACK]
 151 4.539023  127.0.0.1   127.0.0.1     TCP        56 56971 → 8888 [ACK] Seq=
 152 4.539038  127.0.0.1   127.0.0.1     TCP        56 [TCP Window Update] 888
 153 4.539831  127.0.0.1   127.0.0.1     TCP        60 56971 → 8888 [PSH, ACK]
 154 4.539862  127.0.0.1   127.0.0.1     TCP        56 8888 → 56971 [ACK] Seq=
 155 4.545198  127.0.0.1   127.0.0.1     TCP        60 8888 → 56971 [PSH, ACK]
 156 4.545224  127.0.0.1   127.0.0.1     TCP        56 56971 → 8888 [ACK] Seq=
 157 4.555677  127.0.0.1   127.0.0.1     TCP       157 56969 → 56970 [PSH, ACK
 158 4.555758  127.0.0.1   127.0.0.1     TCP        56 56970 → 56969 [ACK] Seq=3736
```

```
Frame 151: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface lo0, id 0
Null/Loopback
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
Transmission Control Protocol, Src Port: 56971, Dst Port: 8888, Seq: 1, Ack: 1, Len: 0
   Source Port: 56971
   Destination Port: 8888
   [Stream index: 2]
   [Conversation completeness: Incomplete, DATA (15)]
   [TCP Segment Len: 0]
   Sequence Number: 1    (relative sequence number)
   Sequence Number (raw): 2730186232
```

```
000  02 00 00 00 45 00 00 34  00 00 40 00 40 06 00 00   ····E··4 ··@·@···
010  7f 00 00 01 7f 00 00 01  de 8b 22 b8 a2 bb 55 f8   ········ ··"···U·
020  5f c7 d1 f0 80 10 18 eb  fe 28 00 00 01 01 08 0a   _······· ·(······
030  49 8a 25 ed 5d fe 62 2a                            I·%·]·b*
```

Figure 4: screenshot of the Wireshark after sending commands



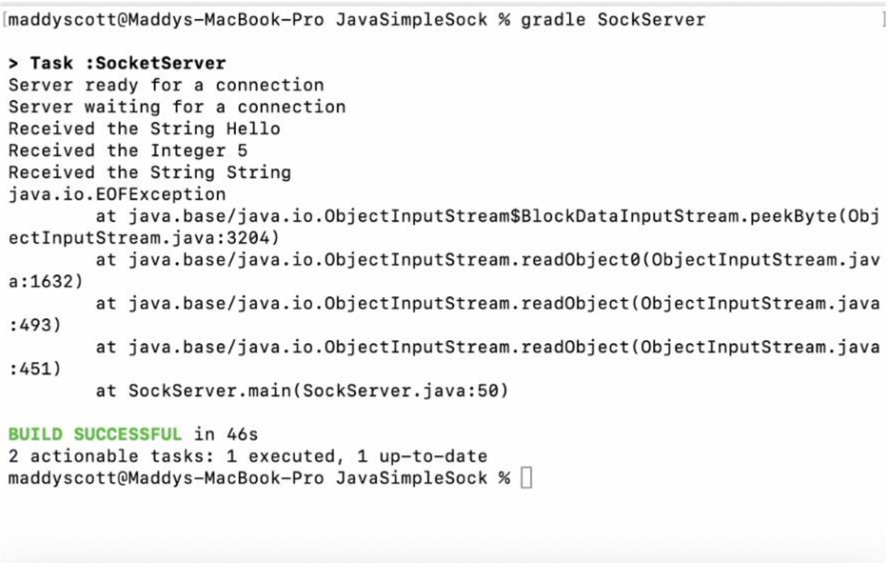
```
maddyscott@Maddys-MacBook-Pro JavaSimpleSock % gradle SockServer

> Task :SocketServer
Server ready for a connection
Server waiting for a connection
Received the String Hello
Received the Integer 5
Received the String String
java.io.EOFException
        at java.base/java.io.ObjectInputStream$BlockDataInputStream.peekByte(Obj
ectInputStream.java:3204)
        at java.base/java.io.ObjectInputStream.readObject0(ObjectInputStream.jav
a:1632)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java
:493)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java
:451)
        at SockServer.main(SockServer.java:50)

BUILD SUCCESSFUL in 46s
2 actionable tasks: 1 executed, 1 up-to-date
maddyscott@Maddys-MacBook-Pro JavaSimpleSock %
```

JavaSimpleSock — java -Xmx64m -Xms64m -Xdock:name=Gradle -Xdoc...

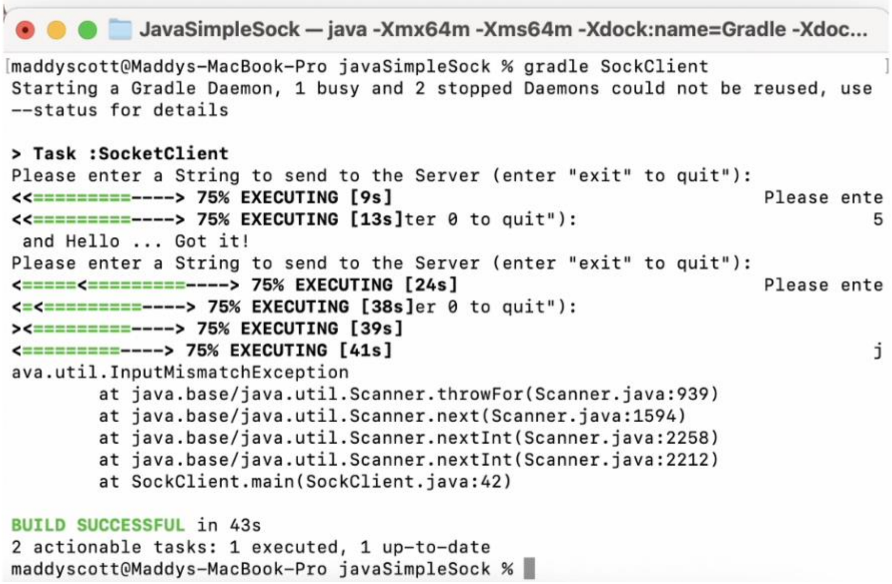
```
maddyscott@Maddys-MacBook-Pro javaSimpleSock % gradle SockClient
Starting a Gradle Daemon, 1 busy and 2 stopped Daemons could not be reused, use
--status for details

> Task :SocketClient
Please enter a String to send to the Server (enter "exit" to quit"):
<<=========----> 75% EXECUTING [9s]                                     Please ente
<<=========----> 75% EXECUTING [13s]ter 0 to quit"):                            5
 and Hello ... Got it!
Please enter a String to send to the Server (enter "exit" to quit"):
<=====<=========----> 75% EXECUTING [24s]                               Please ente
<=<=========----> 75% EXECUTING [38s]er 0 to quit"):
><=========----> 75% EXECUTING [39s]
<=========----> 75% EXECUTING [41s]                                             j
ava.util.InputMismatchException
        at java.base/java.util.Scanner.throwFor(Scanner.java:939)
        at java.base/java.util.Scanner.next(Scanner.java:1594)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2258)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2212)
        at SockClient.main(SockClient.java:42)

BUILD SUCCESSFUL in 43s
2 actionable tasks: 1 executed, 1 up-to-date
maddyscott@Maddys-MacBook-Pro javaSimpleSock %
```

Figure 5: screenshot of the commands used

4.5.2



```
maddyscott@Maddys-MacBook-Pro JavaSimpleSock % gradle SockServer

> Task :SocketServer
Server ready for a connection
Server waiting for a connection
Received the String Hello
Received the Integer 5
Received the String String
java.io.EOFException
        at java.base/java.io.ObjectInputStream$BlockDataInputStream.peekByte(Obj
ectInputStream.java:3204)
        at java.base/java.io.ObjectInputStream.readObject0(ObjectInputStream.jav
a:1632)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java
:493)
        at java.base/java.io.ObjectInputStream.readObject(ObjectInputStream.java
:451)
        at SockServer.main(SockServer.java:50)

BUILD SUCCESSFUL in 46s
2 actionable tasks: 1 executed, 1 up-to-date
maddyscott@Maddys-MacBook-Pro JavaSimpleSock % 
```


JavaSimpleSock — java -Xmx64m -Xms64m -Xdock:name=Gradle -Xdoc...

```
maddyscott@Maddys-MacBook-Pro javaSimpleSock % gradle SockClient
Starting a Gradle Daemon, 1 busy and 2 stopped Daemons could not be reused, use
--status for details

> Task :SocketClient
Please enter a String to send to the Server (enter "exit" to quit"):
<<=========----> 75% EXECUTING [9s]                                   Please ente
<<=========----> 75% EXECUTING [13s]ter 0 to quit"):                            5
 and Hello ... Got it!
Please enter a String to send to the Server (enter "exit" to quit"):
<=====<=========----> 75% EXECUTING [24s]                             Please ente
<=<=========----> 75% EXECUTING [38s]er 0 to quit"):
><=========----> 75% EXECUTING [39s]
<=========----> 75% EXECUTING [41s]                                             j
ava.util.InputMismatchException
        at java.base/java.util.Scanner.throwFor(Scanner.java:939)
        at java.base/java.util.Scanner.next(Scanner.java:1594)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2258)
        at java.base/java.util.Scanner.nextInt(Scanner.java:2212)
        at SockClient.main(SockClient.java:42)

BUILD SUCCESSFUL in 43s
2 actionable tasks: 1 executed, 1 up-to-date
maddyscott@Maddys-MacBook-Pro javaSimpleSock % 
```

It is difficult reaching the server on the local network from outside of the network because there are a lot of signals being transmitted and the client and the server must be able to have all of the necessary information in order to communicate with one another. It makes it difficult doing so when you aren’t on the same network for both client and server. There are more steps involved such as changing the client to the IP address of the AWS.