Nathaniel Davis-Perez
3/24/25
SER 321

Assignment 1

https://github.com/notnatedavis/ser321-spring25B-ndavispe)

1.x. Command Line Tasks Linux System :

```
1 : mkdir cli_assignment

2 : cd cli_assignment

3 : touch stuff.txt

4 : cat > stuff.txt <<EOF

> test

> text

> here

> EOF

5 : 3 words 4 lines (extra line added automatically at the end)

6 : cat >> stuff.txt <<EOF

> more

> text

> here

7 : xubuntu@xubuntu:~/Desktop/cli_assignment$ mkdir draft

8 : mv stuff.txt draft

9 : cd draft , touch .secret.txt

10 : cp -a ~/Desktop/cli_assignment/draft/. ~/Desktop/cli_assignment/final

11 : mv draft draft.remove

12 : mv ~/Desktop/cli_assignment/draft.remove ~/Desktop/cli_assignment/final

13 : find . -printf "%M %-20u %-20g %p\n"

drwxrwxr-x vboxuser            vboxuser            .
```

Nathaniel Davis-Perez
3/24/25
SER 321

```
drwxrwxr-x vboxuser          vboxuser          ./final

-rw-rw-r-- vboxuser          vboxuser          ./final/stuff.txt

drwxrwxr-x vboxuser          vboxuser          ./final/draft.remove

-rw-rw-r-- vboxuser          vboxuser          ./final/draft.remove/stuff.txt

-rw-rw-r-- vboxuser          vboxuser          ./final/draft.remove/.secret.txt

-rw-rw-r-- vboxuser          vboxuser          ./final/.secret.txt
```

```
14 : zcat NASA_access_log_Aug95.gz (or) gunzip -c NASA_access_log_Aug95.gz

15 : gunzip -k NASA_access_log_Aug95.gz

16 : mv NASA_access_log_Aug95 logs.txt

17 : mv logs.txt ~/Desktop/cli_assignment

18 : head -n 100 ~/Desktop/cli_assignment/logs.txt

19 : head -n 100 ~/Desktop/cli_assignment/logs.txt > ~/Desktop/cli_assignment/logs_top_100.txt

20 : tail -n 100 ~/Desktop/cli_assignment/logs.txt

21 : tail -n 100 ~/Desktop/cli_assignment/logs.txt > ~/Desktop/cli_assignment/logs_bottom_100.txt

22 : cat ~/Desktop/cli_assignment/logs_top_100.txt ~/Desktop/cli_assignment/logs_bottom_100.txt > ~/Desktop/cli_assignment/logs_snapshot.txt

23 : echo "ndavispe: This is a great assignment" >> logs_snapshot.txt

echo "3/24/2025" >> logs_snapshot.txt

24 : cd ~/Desktop/cli_assignment

less logs.txt

25 : tail -n +2 ~/Desktop/marks.csv

cut -d '%' -f 2 ~/Desktop/marks.csv | tail -n +2 > ~/Desktop/student_names.txt

26 : cut -d '%' -f 3 ~/Desktop/marks.csv | tail -n +2 | sort -n

27 : awk -F '%' 'NR>1 {sum += $3; count++} END {print "Average:", sum/count}' ~/Desktop/marks.csv
```

Nathaniel Davis-Perez
3/24/25
SER 321

28 : awk -F '%' 'NR>1 {sum += $3; count++} END {print "Average:", sum/count}' ~/Desktop/marks.csv > ~/Desktop/cli_assignment/done.txt

29 : cd cli_assignment

mv done.txt ~/Desktop/cli_assignment/final

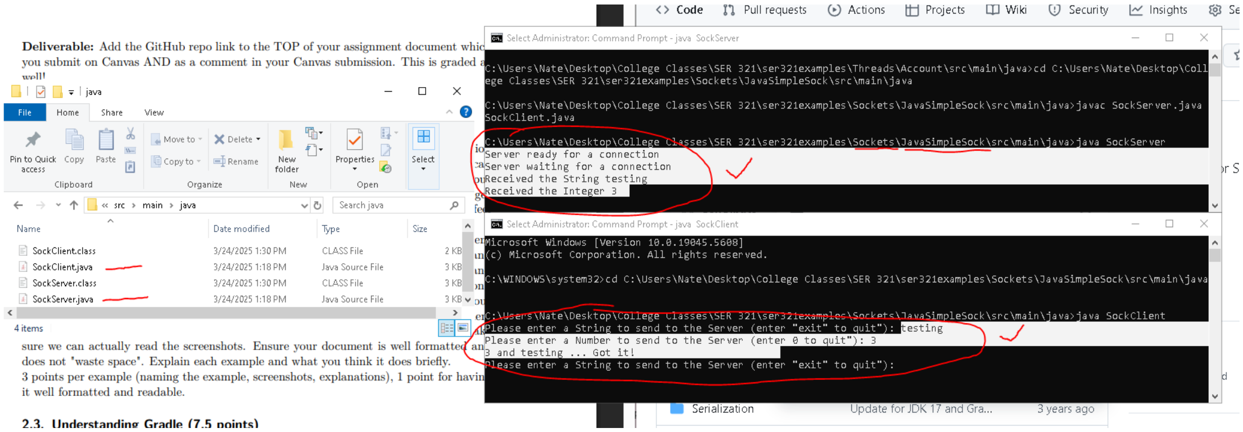
30 : cd final

mv done.txt average.txt

2.2 Examples :

2.2.1 : SockClient & SockServer

SockServer is responsible for attempting to establish a connection along with input/output streams and SockClient is responsible for maintaining a connection to established SockServer passing input/output streams.



Deliverable: Add the GitHub repo link to the TOP of your assignment document whic
you submit on Canvas AND as a comment in your Canvas submission. This is graded a
well!

Name	Date modified	Type	Size
SockClient.class	3/24/2025 1:30 PM	CLASS File	2 KB
SockClient.java	3/24/2025 1:18 PM	Java Source File	3 KB
SockServer.class	3/24/2025 1:30 PM	CLASS File	3 KB
SockServer.java	3/24/2025 1:18 PM	Java Source File	3 KB

sure we can actually read the screenshots. Ensure your document is well formatted an
does not "waste space". Explain each example and what you think it does briefly.
3 points per example (naming the example, screenshots, explanations), 1 point for havin
it well formatted and readable.

2.3. Understanding Gradle (7.5 points)

Select Administrator: Command Prompt - java SockServer

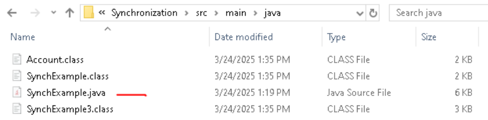
```
C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Threads\Account\src\main\java>cd C:\Users\Nate\Desktop\Coll
ege Classes\SER 321\ser321examples\Sockets\JavaSimpleSock\src\main\java

C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Sockets\JavaSimpleSock\src\main\java>javac SockServer.java
SockClient.java

C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Sockets\JavaSimpleSock\src\main\java>java SockServer
Server ready for a connection
Server waiting for a connection
Received the String testing
Received the Integer 3
```

Select Administrator: Command Prompt - java SockClient

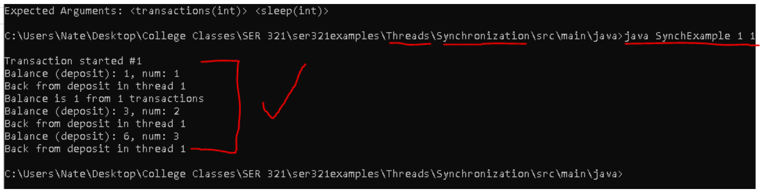
```
Microsoft Windows [Version 10.0.19045.5608]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Sockets\JavaSimpleSock\src\main\java


C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Sockets\JavaSimpleSock\src\main\java>java SockClient
Please enter a String to send to the Server (enter "exit" to quit"): testing
Please enter a Number to send to the Server (enter 0 to quit"): 3
3 and testing ... Got it!
Please enter a String to send to the Server (enter "exit" to quit"):
```

2.2.2 : SynchExample

Handles creating a thread for each ‘transaction’ and printing associated (passed) values for each thread.



Name	Date modified	Type	Size
Account.class	3/24/2025 1:35 PM	CLASS File	2 KB
SynchExample.class	3/24/2025 1:35 PM	CLASS File	2 KB
SynchExample.java	3/24/2025 1:19 PM	Java Source File	6 KB
SynchExample3.class	3/24/2025 1:35 PM	CLASS File	3 KB

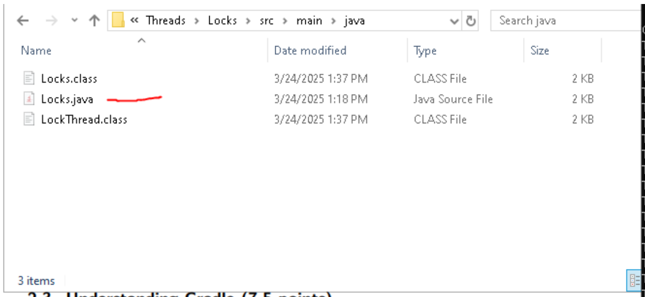


```
Expected Arguments: <transactions(int)> <sleep(int)>

C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Threads\Synchronization\src\main\java>java SynchExample 1 1

Transaction started #1
Balance (deposit): 1, num: 1
Back from deposit in thread 1
Balance is 1 from 1 transactions
Balance (deposit): 3, num: 2
Back from deposit in thread 1
Balance (deposit): 6, num: 3
Back from deposit in thread 1

C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Threads\Synchronization\src\main\java>
```

Nathaniel Davis-Perez
3/24/25
SER 321

2.2.3 : Locks

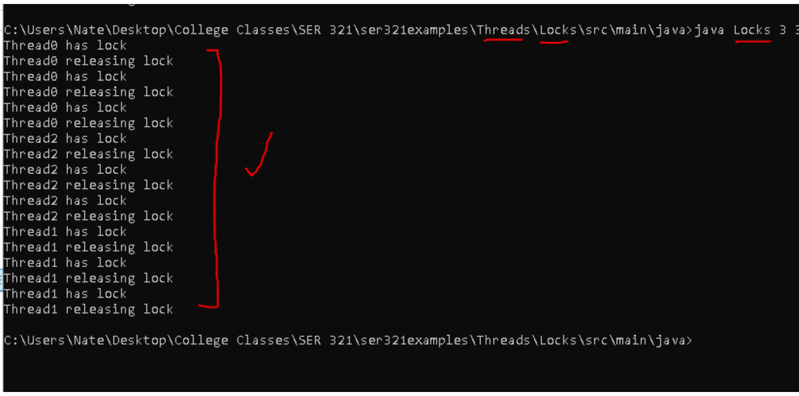
Handles creating new LockThread mutexs with passed parameters and printing status’



Name	Date modified	Type	Size
Locks.class	3/24/2025 1:37 PM	CLASS File	2 KB
Locks.java	3/24/2025 1:18 PM	Java Source File	2 KB
LockThread.class	3/24/2025 1:37 PM	CLASS File	2 KB

2.3. Understanding Gradle (7.5 points)

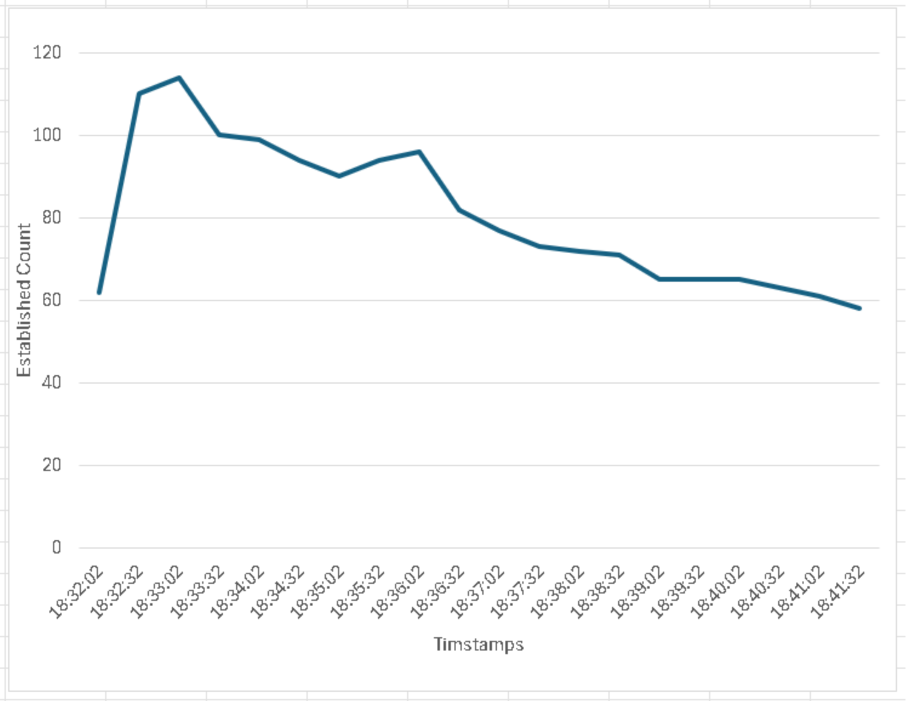
Go to the Gradle/Java Gradle example, in the Gradle file you will see a comment 'Try:' with some tasks following. Make the appropriate changes in the build.gradle file and Java file(s), so that the Gradle task works with different arguments.
Deliverable: Copy the Java Gradle folder into your assignment repo 'ser321-spring25B-



```
C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Threads\Locks\src\main\java>java Locks 3 3
Thread0 has lock
Thread0 releasing lock
Thread0 has lock
Thread0 releasing lock
Thread0 has lock
Thread0 releasing lock
Thread2 has lock
Thread2 releasing lock
Thread2 has lock
Thread2 releasing lock
Thread2 has lock
Thread2 releasing lock
Thread1 has lock
Thread1 releasing lock
Thread1 has lock
Thread1 releasing lock
Thread1 has lock
Thread1 releasing lock

C:\Users\Nate\Desktop\College Classes\SER 321\ser321examples\Threads\Locks\src\main\java>
```

2.4 : (youtube.com/watch?v=tvg6ELQT_Dg)

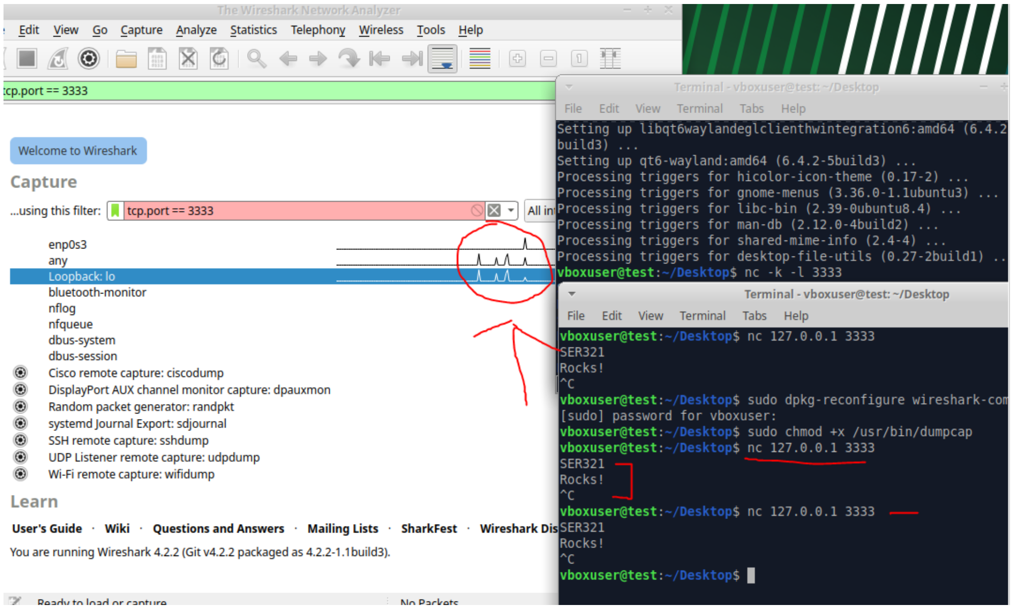
3.3 :



Nathaniel Davis-Perez
3/24/25
SER 321

```
# monitor_sockets.ps1 script for socket states of 10 min

$duration = 10 * 60  # 10 min in seconds

$interval = 30      # check every 30 seconds

$endTime = (Get-Date).AddSeconds($duration)

while ((Get-Date) -lt $endTime) {

   # get current timestamp

   $timestamp = Get-Date -Format "yyyy-MM-dd HH:mm:ss"

   # count ESTABLISHED and LISTEN connections

   $netstatOutput = netstat -ano | Select-String -Pattern "ESTABLISHED|LISTEN"

   $listenCount = ($netstatOutput | Select-String -Pattern "LISTEN").Count

   $establishedCount = ($netstatOutput | Select-String -Pattern "ESTABLISHED").Count

   # output to CSV

   "$timestamp, LISTEN=$listenCount, ESTABLISHED=$establishedCount" | Out-File -Append -FilePath "socket_data.csv"

   Start-Sleep -Seconds $interval

}
```

3.2.5 :


The Wireshark Network Analyzer
Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Welcome to Wireshark
Capture
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Dis
Terminal - vboxuser@test: ~/Desktop
File Edit View Terminal Tabs Help
Terminal - vboxuser@test: ~/Desktop
File Edit View Terminal Tabs Help

tcp.port == 3333

...using this filter: tcp.port == 3333 All in

enp0s3
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump

You are running Wireshark 4.2.2 (Git v4.2.2 packaged as 4.2.2-1.1build3).

Ready to load or capture No Packets

```
Setting up libqt6waylandeglclienthwintegration6:amd64 (6.4.2
build3) ...
Setting up qt6-wayland:amd64 (6.4.2-5build3) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1.1ubuntu3) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for shared-mime-info (2.4-4) ...
Processing triggers for desktop-file-utils (0.27-2build1) ..
vboxuser@test:~/Desktop$ nc -k -l 3333
```

```
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ sudo dpkg-reconfigure wireshark-com
[sudo] password for vboxuser:
vboxuser@test:~/Desktop$ sudo chmod +x /usr/bin/dumpcap
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ 
```

Nathaniel Davis-Perez
3/24/25
SER 321

a. commands used (nc -k -l 3333) and (nc 127.0.0.1 3333). The first command is responsible for -listening on port 3333 and maintain listening -k even after disconnection

b. 4 frames for 2 lines

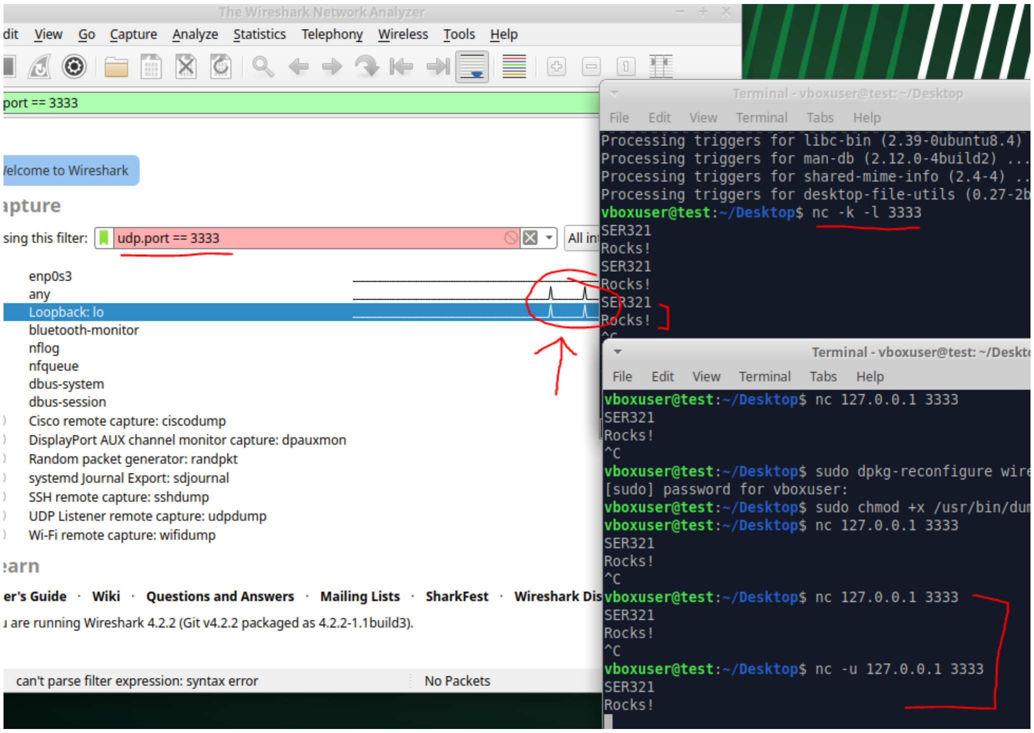
c. 2 Packets for 2 lines , 1 for SER321 and 1 for Rocks!

d. 9 Packets for whole process

e. 12 bytes sent (SER321+\n = 7 , Rocks!+\n = 6 , -1 for the header ?)

f. 498 bytes , 54 bytes for 9 packets + 12 from data

g. ((498-12) / 12) * 100 = ~97% overhead


The Wireshark Network Analyzer
dit
View
Go
Capture
Analyze
Statistics
Telephony
Wireless
Tools
Help
Welcome to Wireshark
apture
earn
er's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Dis
Terminal - vboxuser@test: ~/Desktop
File
Edit
View
Terminal
Tabs
Help
Terminal - vboxuser@test: ~/Deskto

port == 3333

sing this filter: udp.port == 3333

```
enp0s3
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump
```

ı are running Wireshark 4.2.2 (Git v4.2.2 packaged as 4.2.2-1.1build3).

can't parse filter expression: syntax error   No Packets

```
Processing triggers for libc-bin (2.39-0ubuntu8.4)
Processing triggers for man-db (2.12.0-4build2) ...
Processing triggers for shared-mime-info (2.4-4) ..
Processing triggers for desktop-file-utils (0.27-2b
vboxuser@test:~/Desktop$ nc -k -l 3333
SER321
Rocks!
SER321
Rocks!
SER321
Rocks!
^C
```

```
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ sudo dpkg-reconfigure wire
[sudo] password for vboxuser:
vboxuser@test:~/Desktop$ sudo chmod +x /usr/bin/dum
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ nc 127.0.0.1 3333
SER321
Rocks!
^C
vboxuser@test:~/Desktop$ nc -u 127.0.0.1 3333
SER321
Rocks!
```

a. commands used (nc -k -l -u 3333) and (nc -u 127.0.0.1 3333). The first command is responsible for -listening on port 3333 and maintain listening -k even after disconnection specifically using -u UDP

b. 2 frames for 2 lines

c. 2 Packets for 2 lines , 1 for SER321 and 1 for Rocks!

d. 2 Packets for whole process since UDP is connectionless

e. 70 bytes total sent (SER321+\n = 7 , Rocks!+\n = 7) , 20+8+7 = 35*2(data sent) = 70

f. 14 bytes (SER321+\n = 7 , Rocks!+\n = 7)

Nathaniel Davis-Perez
3/24/25
SER 321

g. 70 – 14 = 56 , 56 / 14 = 400% overhead

3.3.1 : (https://youtu.be/Uw18U0ofp1k)

3.3.2 : On wireshark the needed changes were updating the filter to isolate AWS traffic using ip.addr == AWS_IP. This was because previously loopback was used for local interface. For Gradle calls the changes were within the build.gradle file within the AWS ec2 client to set the server IP/port manually.

3.3.3 : This doesn't work without issues and can't be done the same way as in 3.3.2. This is because local servers hold a private IP and are typically hidden behind NAT.

3.3.4 : You can easily reach your server on AWS with a client running in your local network but not as easily in the other direction because of how dynamic IP's change in intervals. The 'issue' if you wanted to run your server locally and reach it remotely would be that common ports get blocked and the IP is constantly changing.