Put Student Name(s) ↓	Put Student IDs ↓	Due Date	Grade Weight
Ishan Aakash Patel	146151238	As Posted	6%

Name	Lab5: Using TCPDump
Objective	By the end of this lab, students will be able to use `tcpdump` to capture and analyze network traffic on their local machine.
Prerequisites	• Basic knowledge of command-line interface (CLI). • Administrative (root) access to the system. • `tcpdump` installed on the system.

Instructions	• It is an Individual assignment. Put your name + Student ID in the empty spaces above.
	• Submit via the BB relevant link ONLY. NO submission via email please. Be sure to submit the final version file ONLY.
	• Show your genuine signs of your work is done on your machine. This includes:
	  ○ Screenshots that show your desktop background with Date/Time.
	  ○ Show a pop-up bx that shows "your name + IP".
	  ○ Show your logged account when applicable. Optional: Your photo.
	• Submit your report name: CYT215-Lab5-Student Name & ID
	
	Steps:
	
	Step 1: Installing `tcpdump`
	
	First, ensure that `tcpdump` is installed on your system.
	
	For Ubuntu/Debian:
	
	sh
	Copy code
	sudo apt-get install tcpdump
	
	For Red Hat/CentOS:
	
	sh
	Copy code
	sudo yum install tcpdump

Step 2: Basic Packet Capture

Start by capturing network packets on the default network interface.

1. **Open a terminal.**
2. **Run the following command to capture packets:**

```
sh
Copy code
sudo tcpdump
```

Explanation: This command starts capturing packets on the default network interface. `sudo` is required because capturing packets usually requires root privileges.

3. **Stop the capture after a few seconds by pressing `Ctrl+C`.**
    - Observe the output. It shows captured packets with details like source and destination IP addresses, ports, and protocols.

Step 3: Capture Packets on a Specific Interface

Identify the network interfaces available on your system.

1. **List network interfaces:**

```
sh
Copy code
sudo tcpdump -D
```

Explanation: This command lists all network interfaces that `tcpdump` can capture from.

2. **Capture packets on a specific interface (e.g., `eth0`):**

```
sh
Copy code
```

```
sudo tcpdump -i eth0
```

3. **Stop the capture after a few seconds by pressing `Ctrl+C`.**
    - Observe the output. This time, only packets from the specified interface are captured.

Step 4: Save Captured Packets to a File

Save the captured packets to a file for later analysis.

1. **Capture packets and save to a file:**

```
sh
Copy code
sudo tcpdump -i eth0 -w capture.pcap
```

Explanation: The `-w` option writes the captured packets to a file named `capture.pcap`.

2. **Stop the capture after a few seconds by pressing `Ctrl+C`.**

Step 5: Read Captured Packets from a File

Read and analyze the packets from the saved file.

1. **Read packets from the file:**

```
sh
Copy code
sudo tcpdump -r capture.pcap
```

Explanation: The `-r` option reads packets from a file instead of capturing live traffic.

2. **Observe the output.** This shows the packets captured previously.

Step 6: Apply Basic Filters

Filter the captured packets to display only specific types of traffic.

1. **Capture only TCP packets:**

   ```
   sh
   Copy code
   sudo tcpdump -i eth0 tcp
   ```

2. **Capture packets from/to a specific host (e.g., `192.168.1.1`):**

   ```
   sh
   Copy code
   sudo tcpdump -i eth0 host 192.168.1.1
   ```

3. **Capture packets on a specific port (e.g., HTTP port 80):**

   ```
   sh
   Copy code
   sudo tcpdump -i eth0 port 80
   ```

Step 7: Advanced Filtering

Use more advanced filters to capture specific types of traffic.

1. **Capture only HTTP GET requests:**

   ```
   sh
   Copy code
   sudo tcpdump -i eth0 'tcp port 80 and (((ip[2:2] - ((ip[0] & 0xf) << 2)) - ((tcp[12] & 0xf0) >>
   2)) = 0x47455420)'
   ```

   Explanation: This filter captures HTTP GET requests by looking for the "GET" method in the packet data.

2. **Capture DNS queries:**

```
sh
Copy code
sudo tcpdump -i eth0 port 53
```

Step 8: Verbose and Timestamp Options

Enhance the output with more detailed information and human-readable timestamps.

1. **Capture with verbose output:**

```
sh
Copy code
sudo tcpdump -i eth0 -v
```

2. **Capture with human-readable timestamps:**

```
sh
Copy code
sudo tcpdump -i eth0 -tttt
```

Step 9: Limit the Number of Packets

Capture only a specific number of packets.

1. **Capture only 10 packets:**

```
sh
Copy code
sudo tcpdump -i eth0 -c 10
```

Explanation: The `-c` option limits the capture to a specified number of packets.

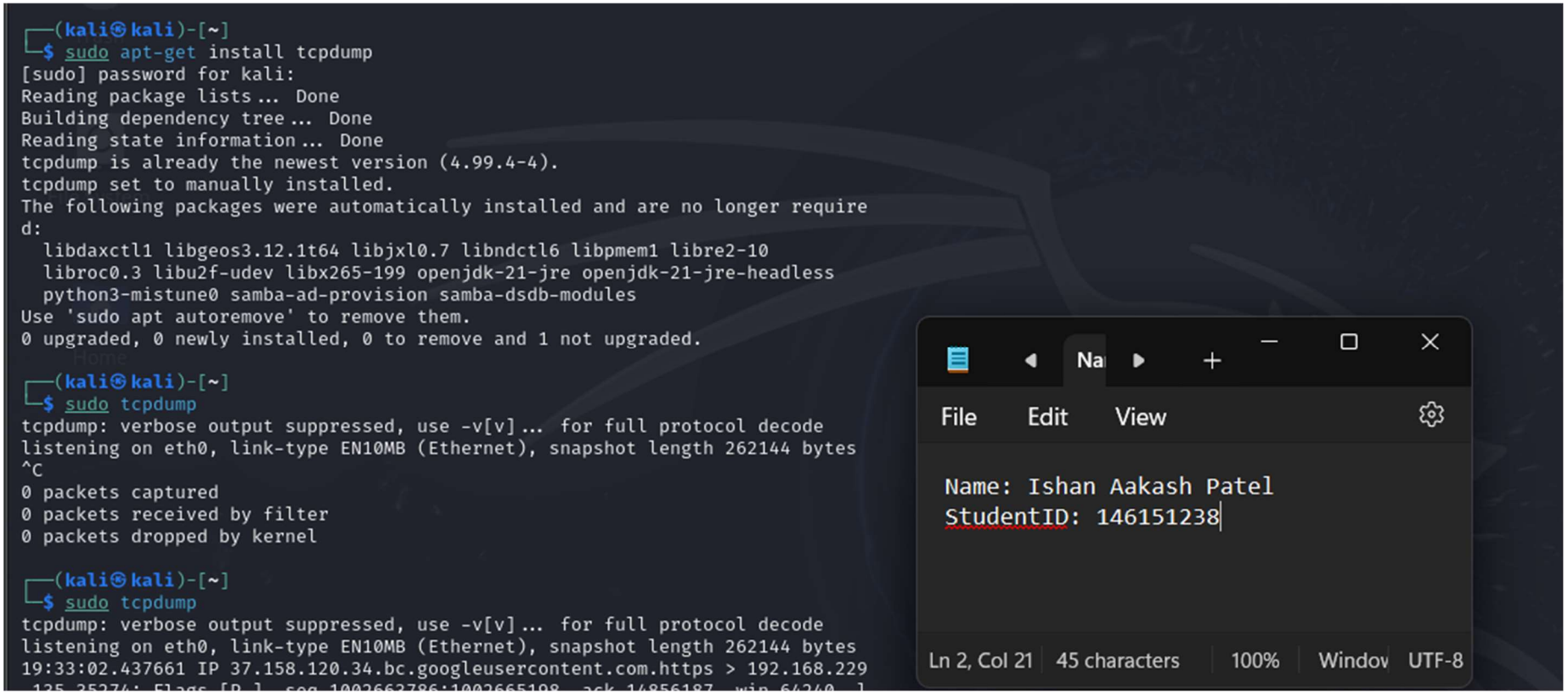
	Step 10: Analyzing the Captured Data Use Wireshark to analyze the captured data. 1. **Install Wireshark if not already installed.** ``` sh Copy code sudo apt-get install wireshark # For Ubuntu/Debian sudo yum install wireshark # For Red Hat/CentOS ``` 2. **Open the capture file in Wireshark:** o Launch Wireshark. o Open the `capture.pcap` file created earlier. o Use Wireshark's GUI to analyze the packets in detail. o Provide a summary of the packet you generated (example follow the TCP stream and describe with you see) o You may also choose to open the PCAP in Network Miner, or use AI to generate a script to analyze it as shown in class.
Students Work required for this activity	• For each step above, follow the instructions and provide screenshots of the steps followed.
Grading Alerts	• If you do NOT use this template or delete any part of it or use any other template, you will be degraded. • If you do NOT follow the fie naming convention, you will be degraded. • If you do NOT submit your file in PDF; you will be degraded. • If you do NOT show your account real name (when applicable); you will be degraded. • If you do NOT show your machine desktop background (with date & time) and IP, you will be degraded. If you do NOT write (in your own words) your learning experience for the activity practices, you will be degraded.

Step 1: Installing `tcpdump`

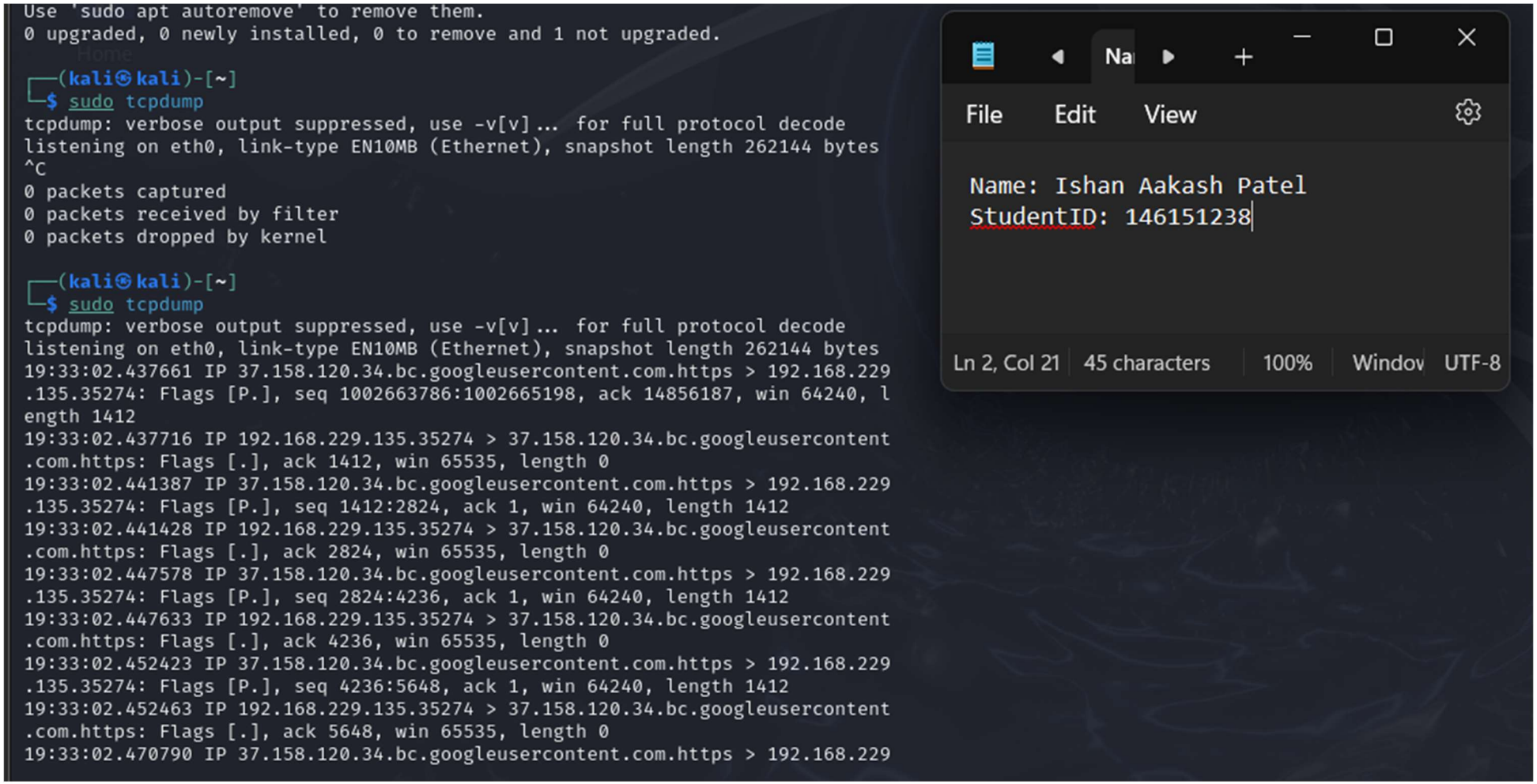
First, ensure that `tcpdump` is installed on your system.

For Ubuntu/Debian:

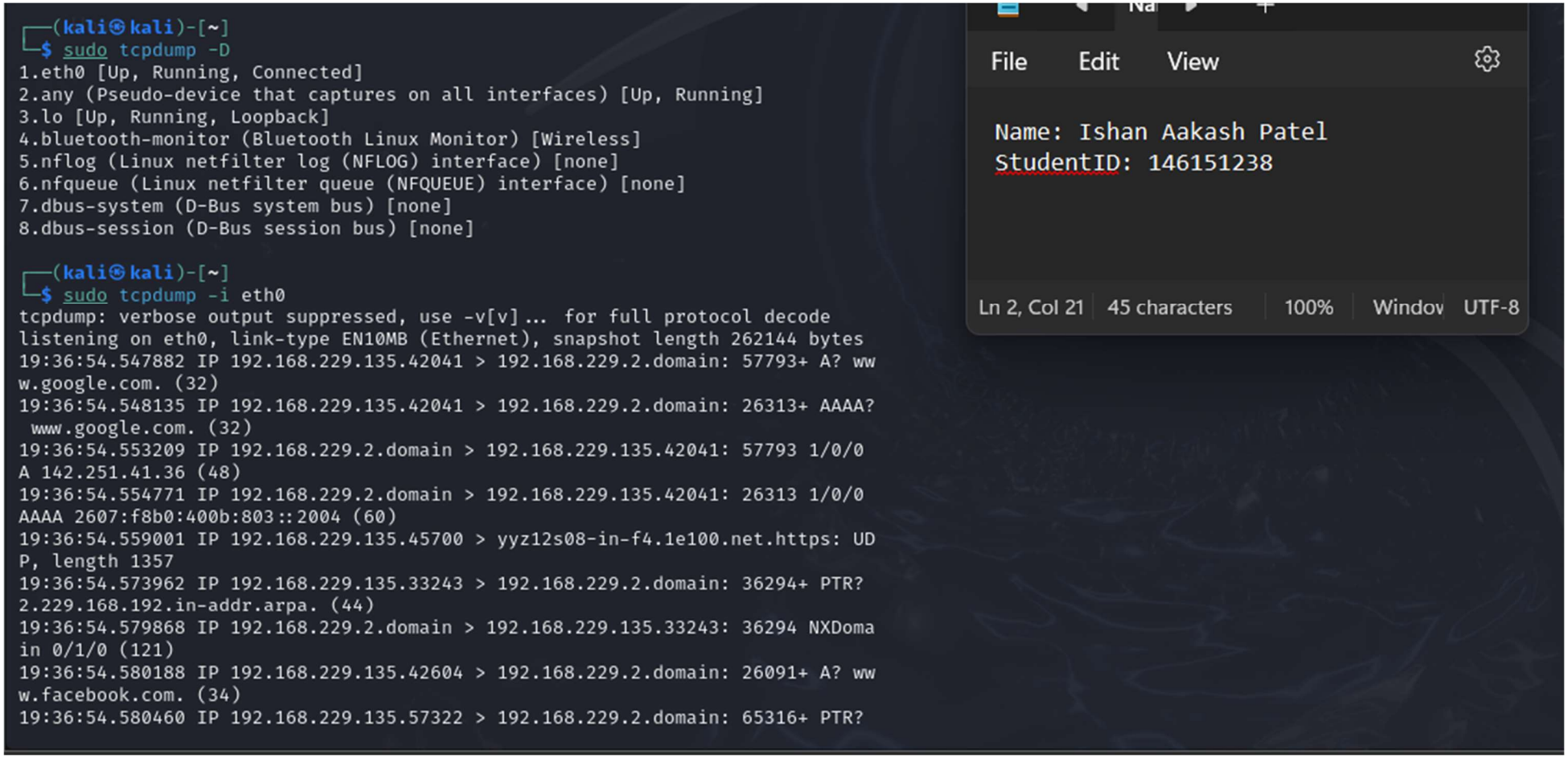
```
sudo apt-get install tcpdump
```



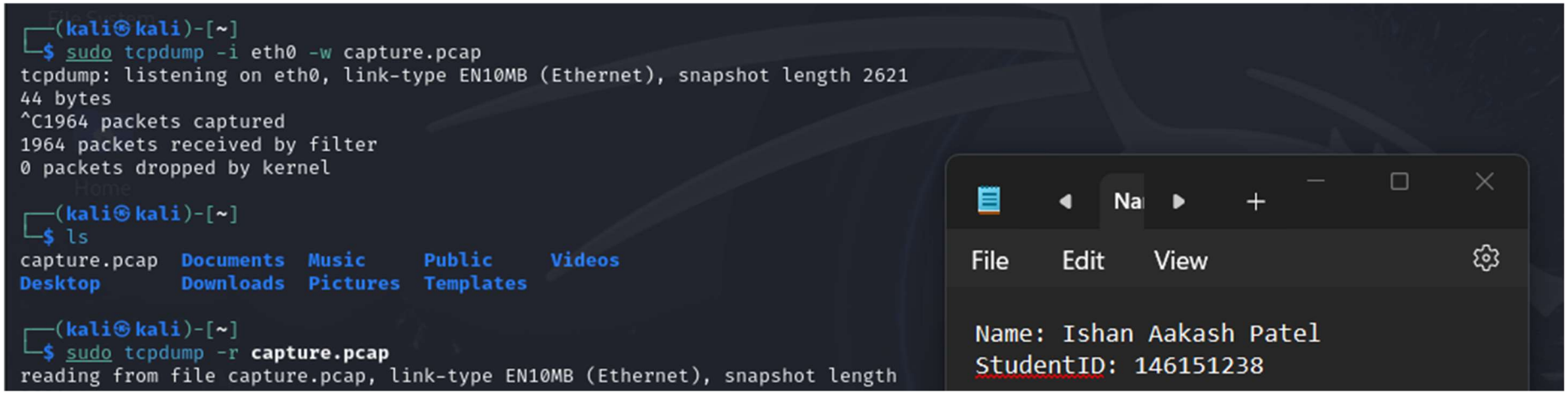
Step 2: Basic Packet Capture



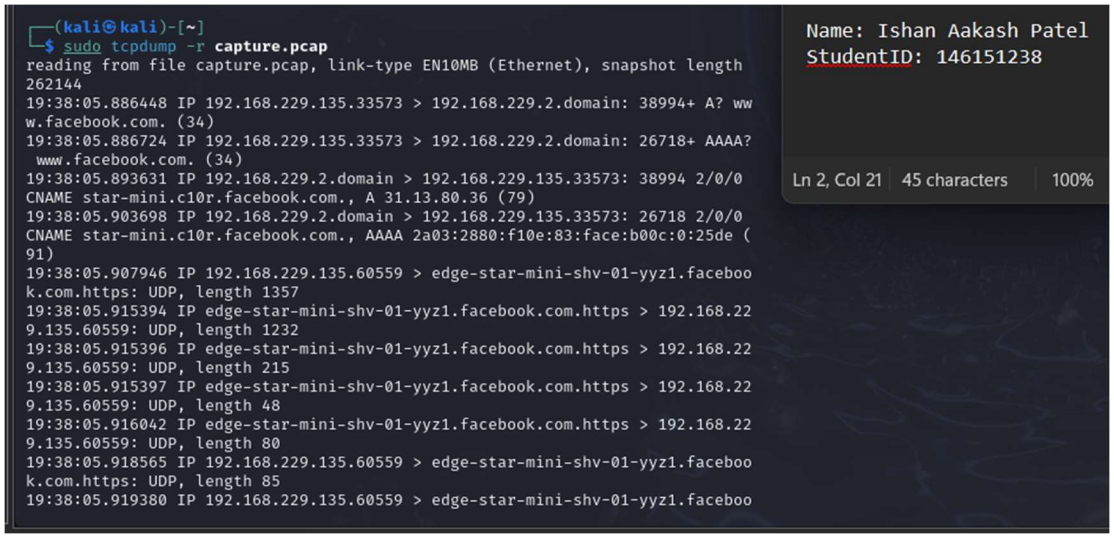
Step 3 : Capture Packets on a Specific Interface



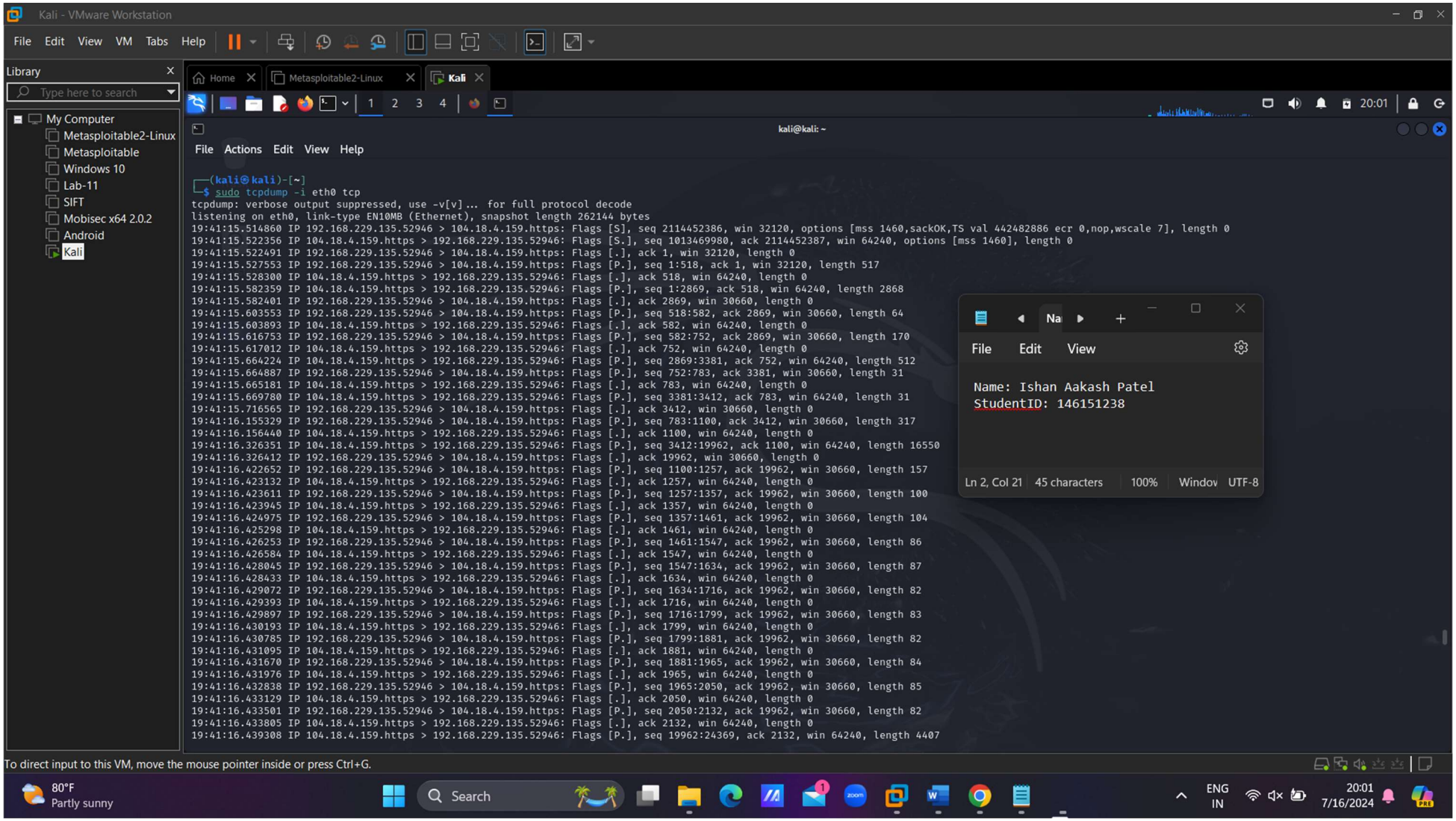
Step 4: Save Captured Packets to a File

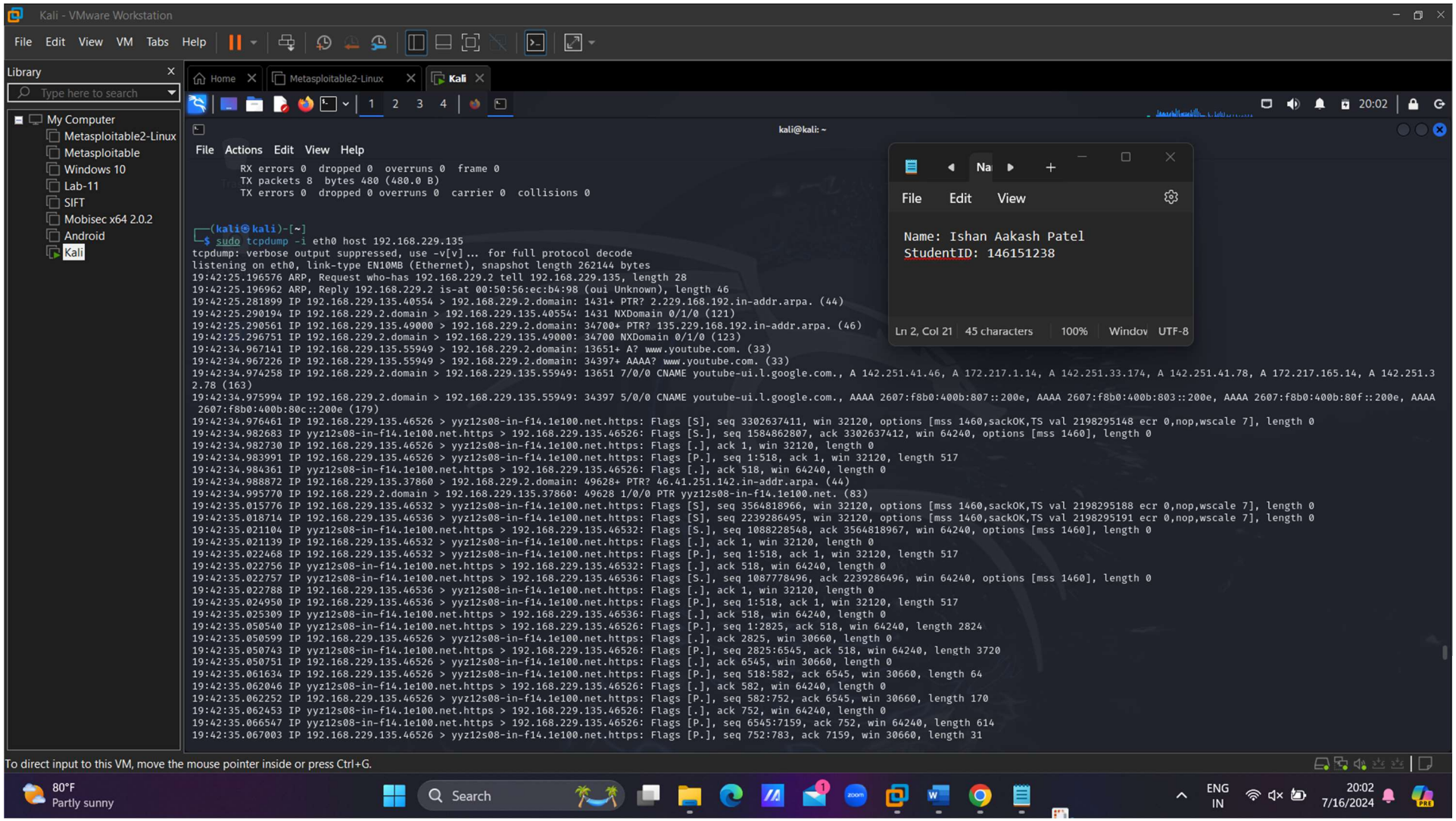


Step 5: Read Captured Packets from a File



Step 6: Apply Basic Filters




Kali - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Metasploitable2-Linux
Metasploitable
Windows 10
Lab-11
SIFT
Mobisec x64 2.0.2
Android
Kali
Home
Metasploitable2-Linux
Kali
kali@kali: ~
File Actions Edit View Help
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 8 bytes 480 (480.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
(kali㉿kali)-[~]
$ sudo tcpdump -i eth0 host 192.168.229.135
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:42:25.196576 ARP, Request who-has 192.168.229.2 tell 192.168.229.135, length 28
19:42:25.196962 ARP, Reply 192.168.229.2 is-at 00:50:56:ec:b4:98 (oui Unknown), length 46
19:42:25.281899 IP 192.168.229.135.40554 > 192.168.229.2.domain: 1431+ PTR? 2.229.168.192.in-addr.arpa. (44)
19:42:25.290194 IP 192.168.229.2.domain > 192.168.229.135.40554: 1431 NXDomain 0/1/0 (121)
19:42:25.290561 IP 192.168.229.135.49000 > 192.168.229.2.domain: 34700+ PTR? 135.229.168.192.in-addr.arpa. (46)
19:42:25.296751 IP 192.168.229.2.domain > 192.168.229.135.49000: 34700 NXDomain 0/1/0 (123)
19:42:34.967141 IP 192.168.229.135.55949 > 192.168.229.2.domain: 13651+ A? www.youtube.com. (33)
19:42:34.967226 IP 192.168.229.135.55949 > 192.168.229.2.domain: 34397+ AAAA? www.youtube.com. (33)
19:42:34.974258 IP 192.168.229.2.domain > 192.168.229.135.55949: 13651 7/0/0 CNAME youtube-ui.l.google.com., A 142.251.41.46, A 172.217.1.14, A 142.251.33.174, A 142.251.41.78, A 172.217.165.14, A 142.251.32.78 (163)
19:42:34.975994 IP 192.168.229.2.domain > 192.168.229.135.55949: 34397 5/0/0 CNAME youtube-ui.l.google.com., AAAA 2607:f8b0:400b:807::200e, AAAA 2607:f8b0:400b:803::200e, AAAA 2607:f8b0:400b:80f::200e, AAAA 2607:f8b0:400b:80c::200e (179)
19:42:34.976461 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [S], seq 3302637411, win 32120, options [mss 1460,sackOK,TS val 2198295148 ecr 0,nop,wscale 7], length 0
19:42:34.982683 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [S.], seq 1584862807, ack 3302637412, win 64240, options [mss 1460], length 0
19:42:34.982730 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [.], ack 1, win 32120, length 0
19:42:34.983991 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 1:518, ack 1, win 32120, length 517
19:42:34.984361 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [.], ack 518, win 64240, length 0
19:42:34.988872 IP 192.168.229.135.37860 > 192.168.229.2.domain: 49628+ PTR? 46.41.251.142.in-addr.arpa. (44)
19:42:34.995770 IP 192.168.229.2.domain > 192.168.229.135.37860: 49628 1/0/0 PTR yyz12s08-in-f14.1e100.net. (83)
19:42:35.015776 IP 192.168.229.135.46532 > yyz12s08-in-f14.1e100.net.https: Flags [S], seq 3564818966, win 32120, options [mss 1460,sackOK,TS val 2198295188 ecr 0,nop,wscale 7], length 0
19:42:35.018714 IP 192.168.229.135.46536 > yyz12s08-in-f14.1e100.net.https: Flags [S], seq 2239286495, win 32120, options [mss 1460,sackOK,TS val 2198295191 ecr 0,nop,wscale 7], length 0
19:42:35.021104 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46532: Flags [S.], seq 1088228548, ack 3564818967, win 64240, options [mss 1460], length 0
19:42:35.021139 IP 192.168.229.135.46532 > yyz12s08-in-f14.1e100.net.https: Flags [.], ack 1, win 32120, length 0
19:42:35.022468 IP 192.168.229.135.46532 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 1:518, ack 1, win 32120, length 517
19:42:35.022756 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46532: Flags [.], ack 518, win 64240, length 0
19:42:35.022757 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46536: Flags [S.], seq 1087778496, ack 2239286496, win 64240, options [mss 1460], length 0
19:42:35.022788 IP 192.168.229.135.46536 > yyz12s08-in-f14.1e100.net.https: Flags [.], ack 1, win 32120, length 0
19:42:35.024950 IP 192.168.229.135.46536 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 1:518, ack 1, win 32120, length 517
19:42:35.025309 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46536: Flags [.], ack 518, win 64240, length 0
19:42:35.050540 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [P.], seq 1:2825, ack 518, win 64240, length 2824
19:42:35.050599 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [.], ack 2825, win 30660, length 0
19:42:35.050743 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [P.], seq 2825:6545, ack 518, win 64240, length 3720
19:42:35.050751 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [.], ack 6545, win 30660, length 0
19:42:35.061634 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 518:582, ack 6545, win 30660, length 64
19:42:35.062046 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [.], ack 582, win 64240, length 0
19:42:35.062252 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 582:752, ack 6545, win 30660, length 170
19:42:35.062453 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [.], ack 752, win 64240, length 0
19:42:35.066547 IP yyz12s08-in-f14.1e100.net.https > 192.168.229.135.46526: Flags [P.], seq 6545:7159, ack 752, win 64240, length 614
19:42:35.067003 IP 192.168.229.135.46526 > yyz12s08-in-f14.1e100.net.https: Flags [P.], seq 752:783, ack 7159, win 30660, length 31
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% Windov UTF-8
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
80°F
Partly sunny
Search
ENG
IN
20:02
7/16/2024

```
┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 port 80
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:43:58.384623 IP 192.168.229.135.60344 > yyz12s07-in-f3.1e100.net.http: Flags [.], ack 1072840992, win 31545, length 0
19:43:58.384706 IP 192.168.229.135.60334 > yyz12s07-in-f3.1e100.net.http: Flags [.], ack 73242109, win 31545, length 0
19:43:58.384804 IP yyz12s07-in-f3.1e100.net.http > 192.168.229.135.60344: Flags [.], ack 1, win 64240, length 0
19:43:58.384805 IP yyz12s07-in-f3.1e100.net.http > 192.168.229.135.60334: Flags [.], ack 1, win 64240, length 0
19:43:58.636625 IP 192.168.229.135.54478 > yyz12s07-in-f3.1e100.net.http: Flags [.], ack 987241028, win 31545, length 0
19:43:58.636884 IP yyz12s07-in-f3.1e100.net.http > 192.168.229.135.54478: Flags [.], ack 1, win 64240, length 0
19:43:59.916595 IP 192.168.229.135.37614 > server-18-67-33-215.yto50.r.cloudfront.net.http: Flags [.], ack 1118835498, win 31177, length 0
19:43:59.917000 IP server-18-67-33-215.yto50.r.cloudfront.net.http > 192.168.229.135.37614: Flags [.], ack 1, win 64240, length 0
19:44:00.428664 IP 192.168.229.135.33786 > server-18-67-33-215.yto50.r.cloudfront.net.http: Flags [.], ack 73287235, win 31176, length 0
19:44:00.429492 IP server-18-67-33-215.yto50.r.cloudfront.net.http > 192.168.229.135.33786: Flags [.], ack 1, win 64240, length 0
19:44:01.200558 IP 192.168.229.135.42224 > cloudproxy10041.sucuri.net.http: Flags [.], ack 354226527, win 30660, length 0
19:44:01.200708 IP cloudproxy10041.sucuri.net.http > 192.168.229.135.42224: Flags [.], ack 1, win 64240, length 0
19:44:03.923917 IP cloudproxy10041.sucuri.net.http > 192.168.229.135.42224: Flags [FP.], seq 1, ack 1, win 64240, length 0
19:44:03.924211 IP 192.168.229.135.42224 > cloudproxy10041.sucuri.net.http: Flags [F.], seq 1, ack 2, win 30660, length 0
19:44:03.924787 IP cloudproxy10041.sucuri.net.http > 192.168.229.135.42224: Flags [.], ack 2, win 64239, length 0
^C
15 packets captured
15 packets received by filter
0 packets dropped by kernel

┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 'tcp port 80 and (((ip[2:2] - ((ip[0] & 0×f) << 2)) - ((tcp[12] & 0×f0) >> 2)) = 0×47455420)'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
```

Na

File Edit View

Name: Ishan Aakash Patel
StudentID: 146151238

Step 7: Advanced Filtering

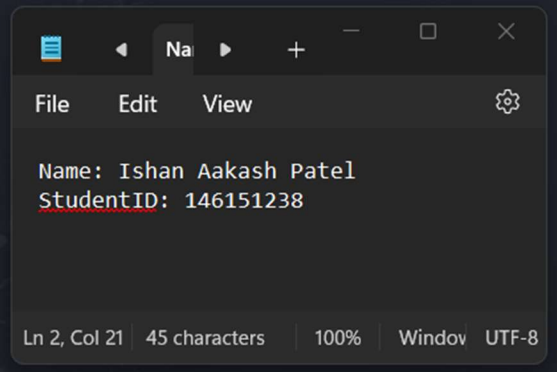
```
┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 'tcp port 80 and (((ip[2:2] - ((ip[0] & 0×f) << 2)) - ((tcp[12] & 0×f0) >> 2)) = 0×47455420)'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel

┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 'tcp port 80 and (((ip[2:2] - ((ip[0] & 0×f) << 2)) - ((tcp[12] & 0×f0) >> 2)) = 0×47455420)'
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C
0 packets captured
```

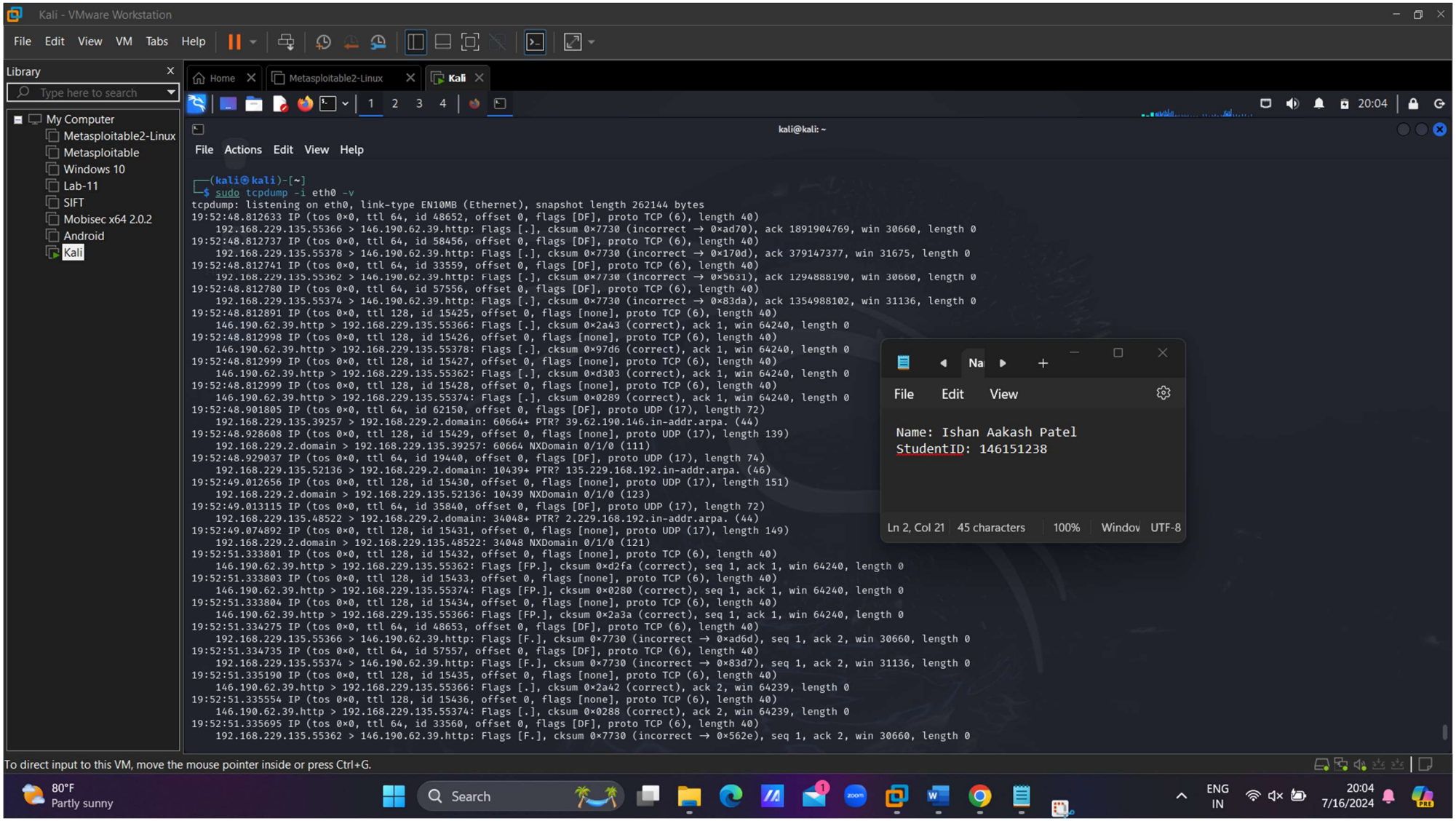
Name: Ishan Aakash Patel
StudentID: 146151238

Ln 2, Col 21 | 45 characters | 100%

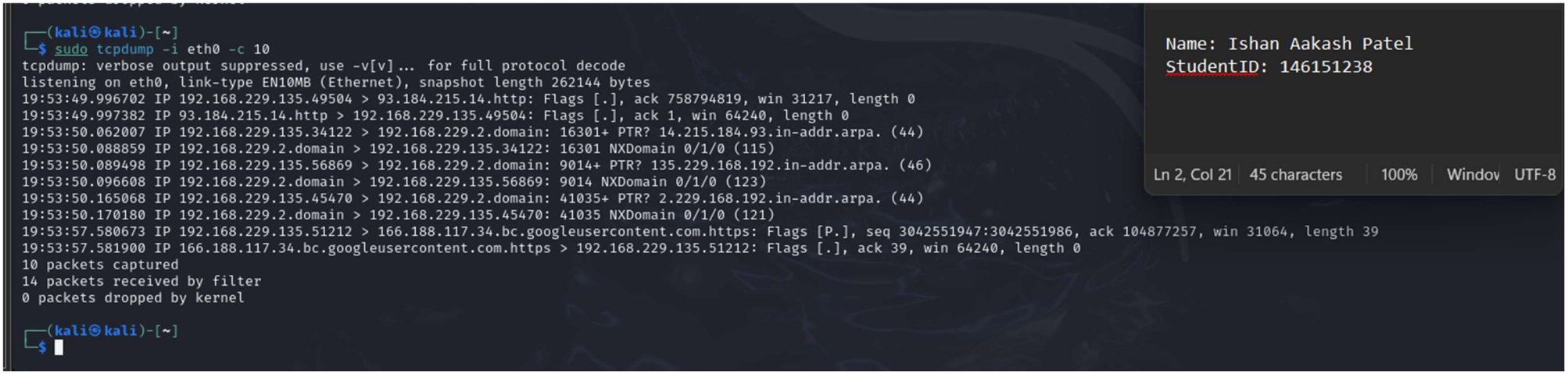
```
┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 port 53
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:52:07.857835 IP 192.168.229.135.57708 > 192.168.229.2.domain: 62414+ A? forums.kali.org. (33)
19:52:07.858351 IP 192.168.229.135.57708 > 192.168.229.2.domain: 48820+ AAAA? forums.kali.org. (33)
19:52:07.909590 IP 192.168.229.135.49772 > 192.168.229.2.domain: 56224+ PTR? 2.229.168.192.in-addr.arpa. (44)
19:52:07.922956 IP 192.168.229.2.domain > 192.168.229.135.57708: 62414 2/0/0 A 104.18.5.159, A 104.18.4.159 (65)
19:52:07.924184 IP 192.168.229.2.domain > 192.168.229.135.57708: 48820 2/0/0 AAAA 2606:4700::6812:49f, AAAA 2606:4700::6812:59f (89)
19:52:07.940958 IP 192.168.229.2.domain > 192.168.229.135.49772: 56224 NXDomain 0/1/0 (121)
19:52:07.942248 IP 192.168.229.135.54971 > 192.168.229.2.domain: 4045+ PTR? 135.229.168.192.in-addr.arpa. (46)
19:52:08.066441 IP 192.168.229.2.domain > 192.168.229.135.54971: 4045 NXDomain 0/1/0 (123)
19:52:12.329629 IP 192.168.229.135.41062 > 192.168.229.2.domain: 47943+ A? www.facebook.com. (34)
19:52:12.614761 IP 192.168.229.135.40902 > 192.168.229.2.domain: 40966+ A? static.xx.fbcdn.net. (37)
19:52:12.671984 IP 192.168.229.135.50839 > 192.168.229.2.domain: 51836+ A? scontent-yyz1-1.xx.fbcdn.net. (46)
19:52:12.672231 IP 192.168.229.135.50839 > 192.168.229.2.domain: 19783+ AAAA? scontent-yyz1-1.xx.fbcdn.net. (46)
19:52:13.435725 IP 192.168.229.2.domain > 192.168.229.135.41062: 47943 2/0/0 CNAME star-mini.c10r.facebook.com., A 31.13.80.36 (79)
19:52:13.436215 IP 192.168.229.135.51845 > 192.168.229.2.domain: 24920+ A? www.facebook.com. (34)
19:52:13.436346 IP 192.168.229.135.51845 > 192.168.229.2.domain: 36698+ AAAA? www.facebook.com. (34)
19:52:13.534848 IP 192.168.229.2.domain > 192.168.229.135.40902: 40966 2/0/0 CNAME scontent.xx.fbcdn.net., A 31.13.80.12 (76)
19:52:13.534848 IP 192.168.229.2.domain > 192.168.229.135.50839: 51836 1/0/0 A 31.13.80.12 (62)
19:52:13.534848 IP 192.168.229.2.domain > 192.168.229.135.50839: 19783 1/0/0 AAAA 2a03:2880:f00e:13:face:b00c:0:3 (74)
19:52:13.536872 IP 192.168.229.2.domain > 192.168.229.135.51845: 24920 2/0/0 CNAME star-mini.c10r.facebook.com., A 31.13.80.36 (79)
19:52:13.536873 IP 192.168.229.2.domain > 192.168.229.135.51845: 36698 2/0/0 CNAME star-mini.c10r.facebook.com., AAAA 2a03:2880:f10e:83:face:b00c:0:25de (91)
19:52:14.072209 IP 192.168.229.135.51771 > 192.168.229.2.domain: 37831+ A? static.xx.fbcdn.net. (37)
19:52:14.072290 IP 192.168.229.135.51771 > 192.168.229.2.domain: 43713+ AAAA? static.xx.fbcdn.net. (37)
19:52:14.319119 IP 192.168.229.2.domain > 192.168.229.135.51771: 37831 2/0/0 CNAME scontent.xx.fbcdn.net., A 31.13.80.12 (76)
19:52:14.319120 IP 192.168.229.2.domain > 192.168.229.135.51771: 43713 2/0/0 CNAME scontent.xx.fbcdn.net., AAAA 2a03:2880:f00e:13:face:b00c:0:3 (88)
^C
24 packets captured
24 packets received by filter
0 packets dropped by kernel
```



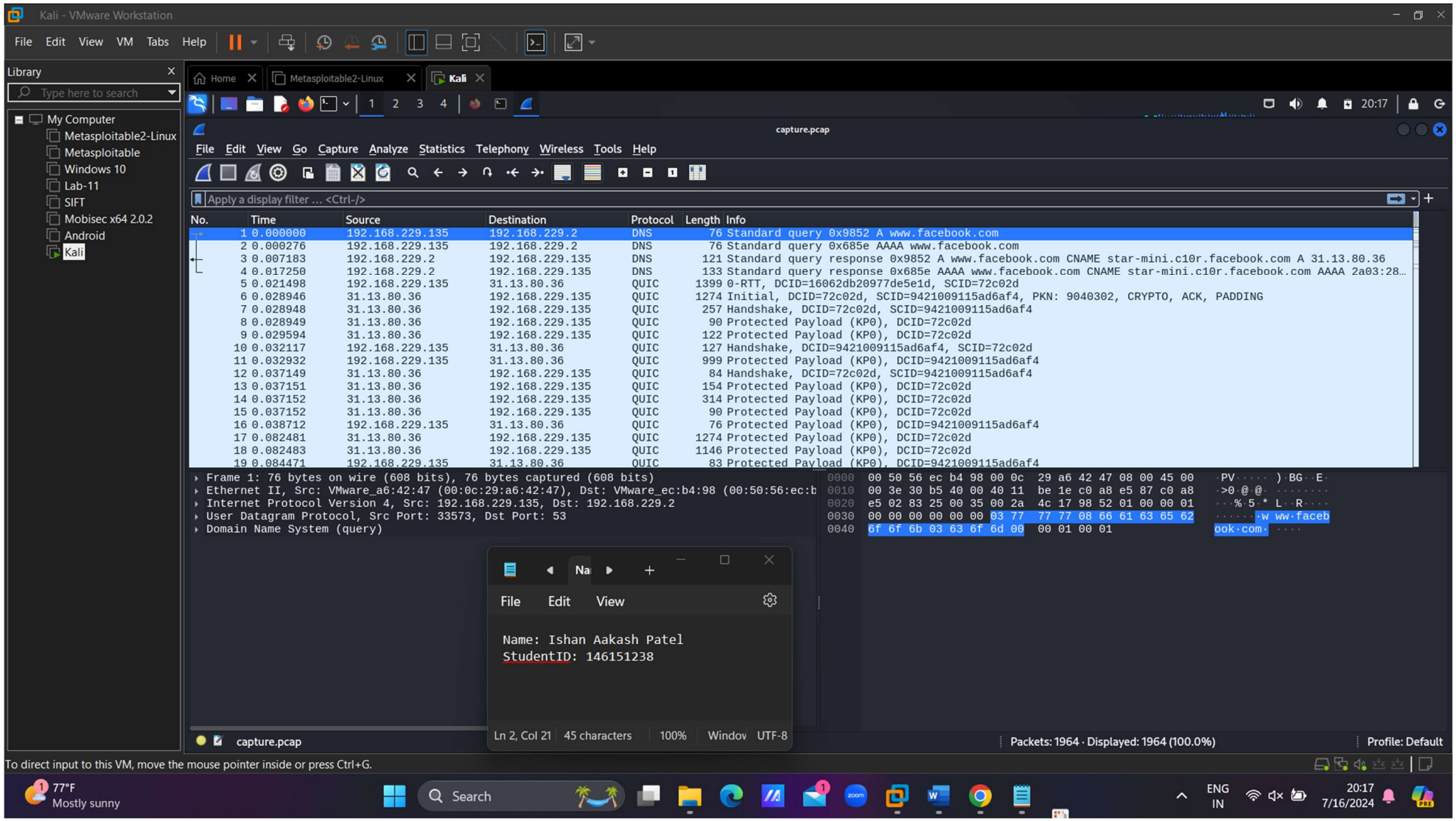
Step 8: Verbose and Timestamp Options

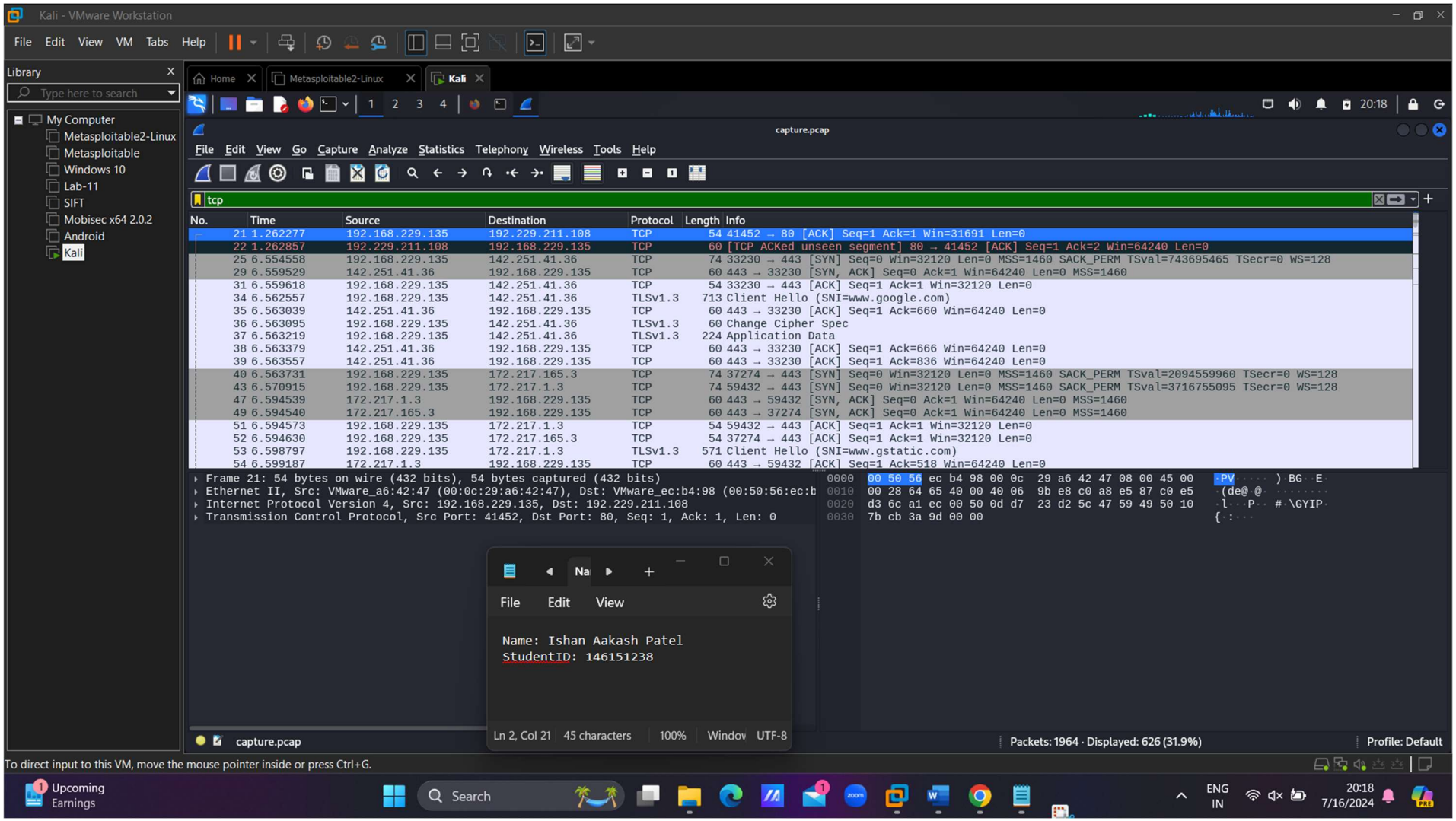


Step 9: Limit the Number of Packets

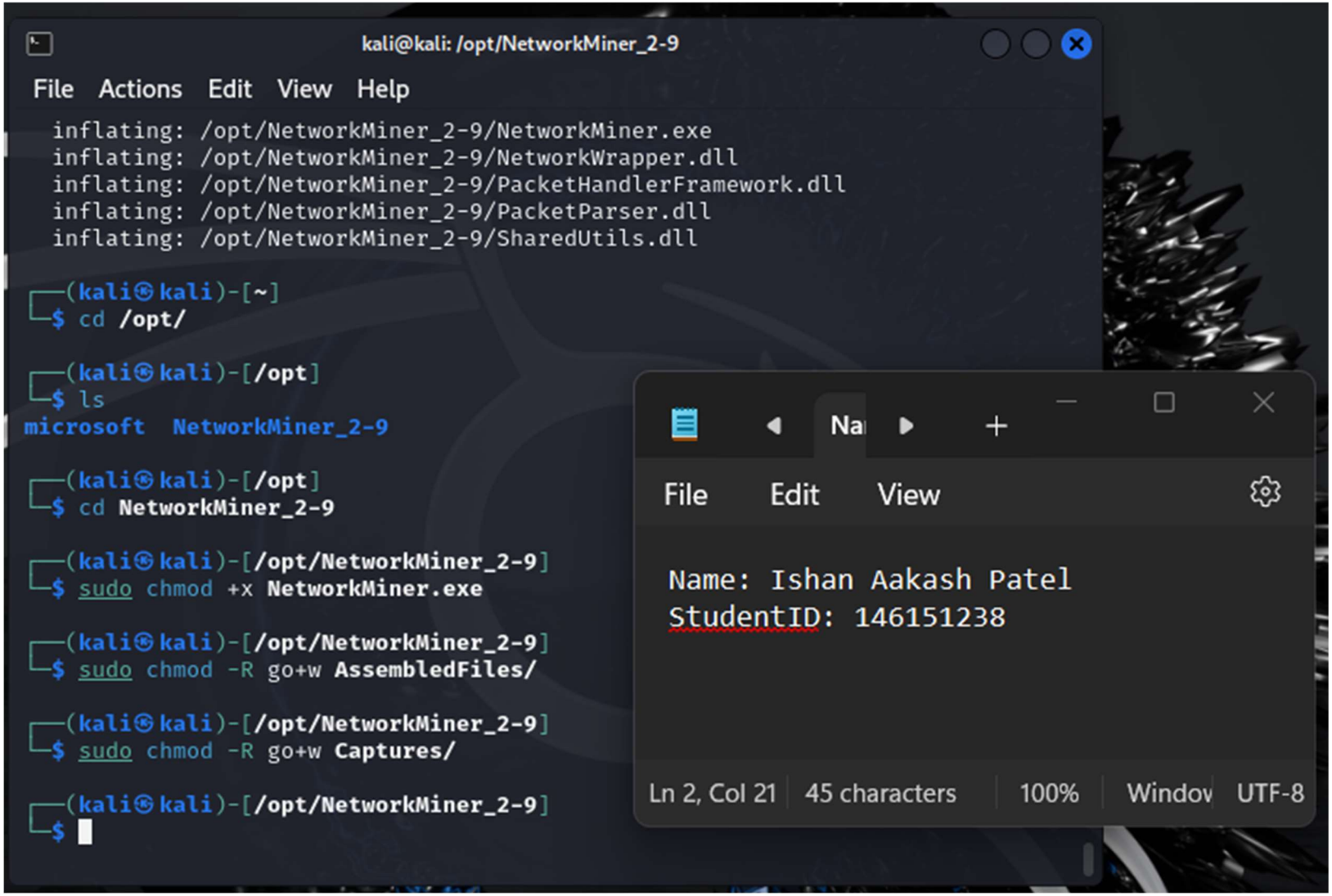


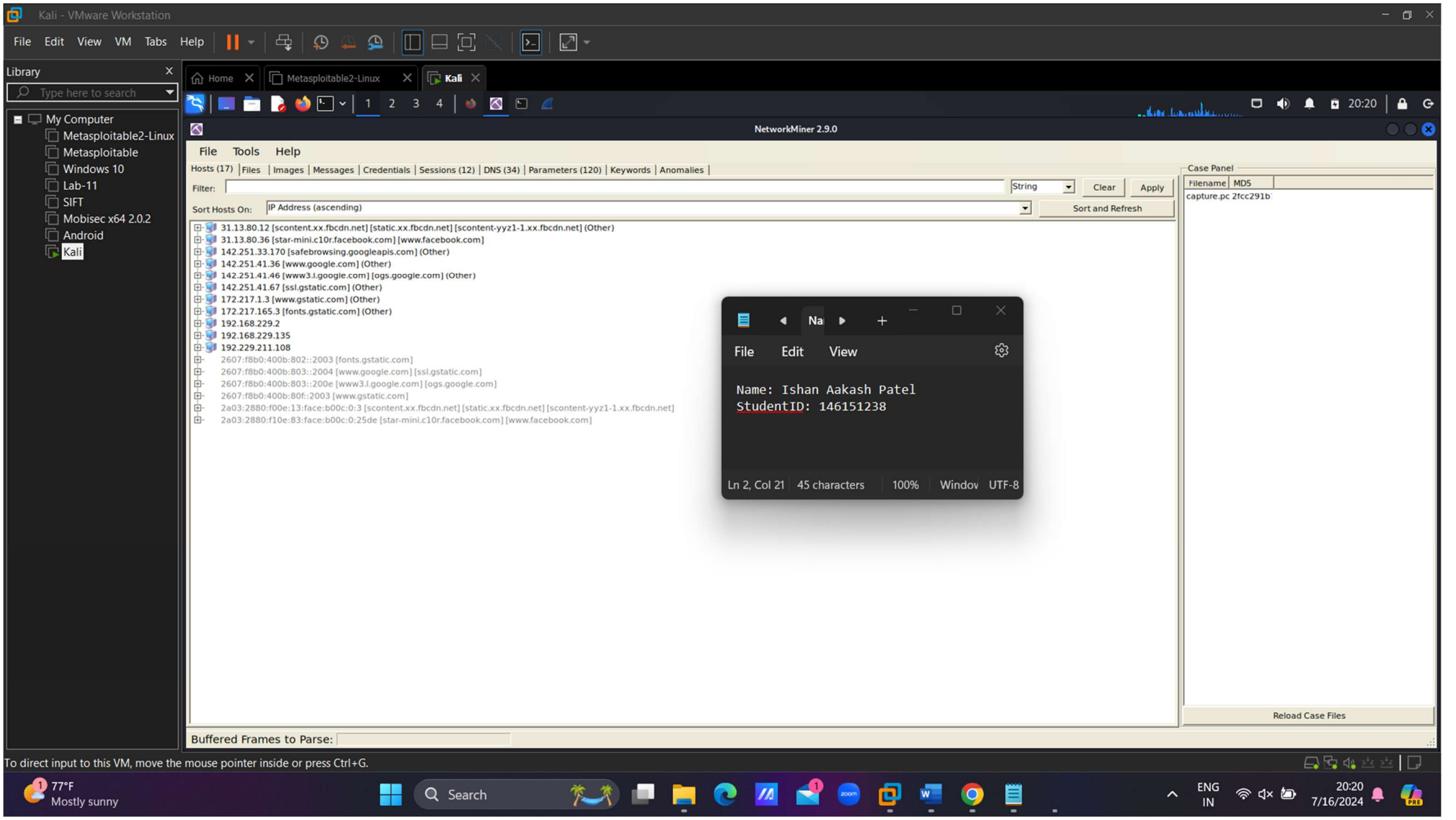
Step 10: Analyzing the Captured Data

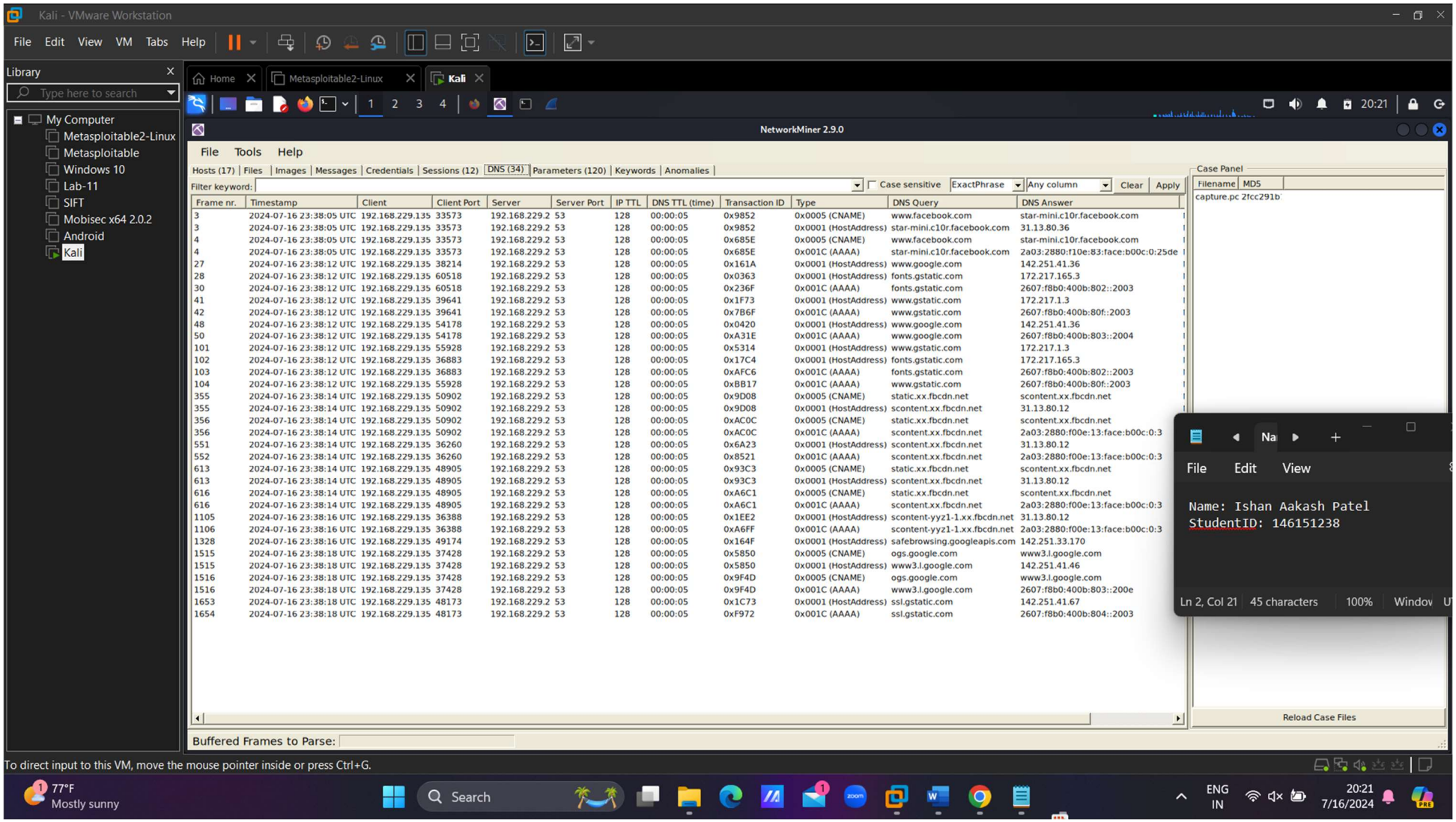


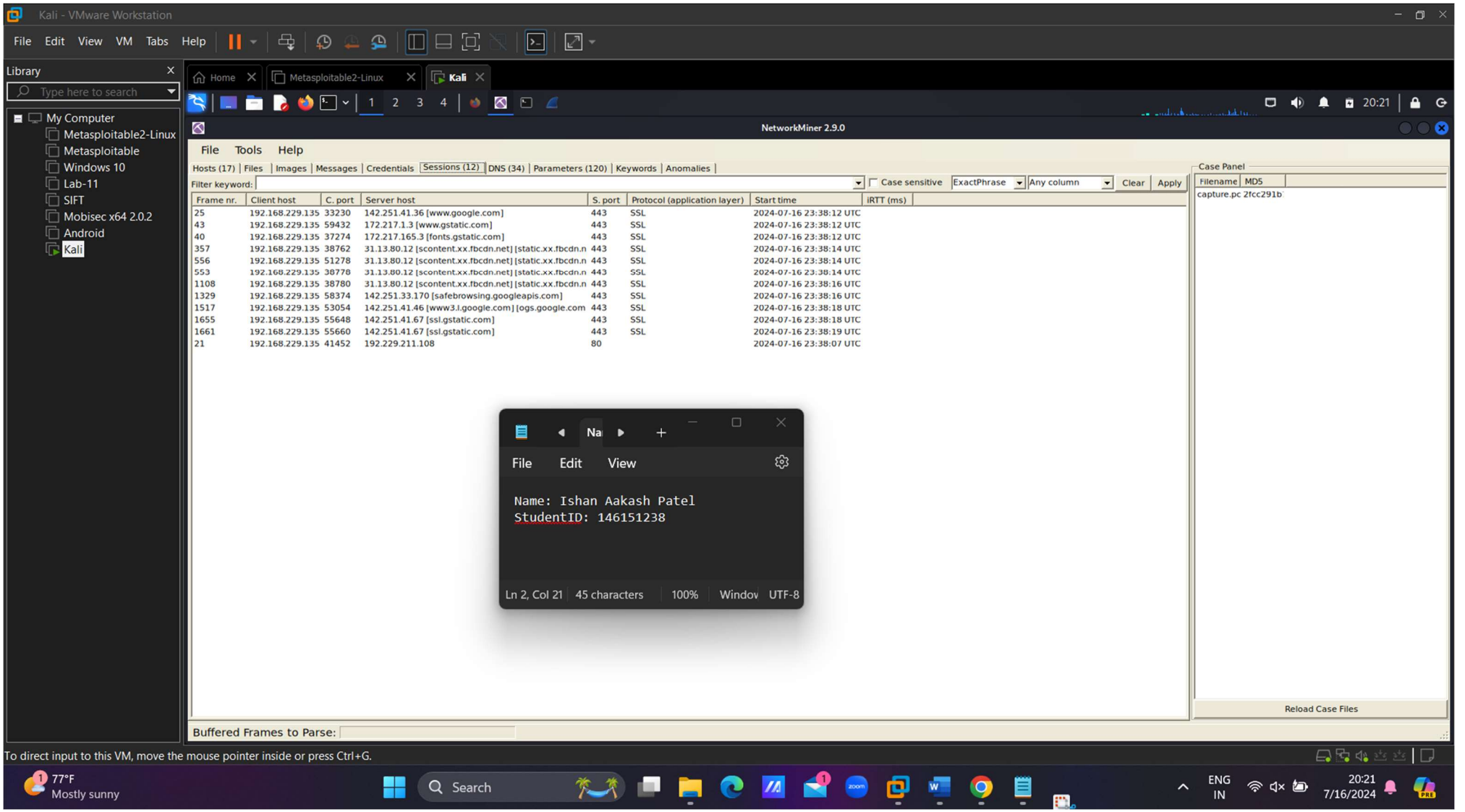

Kali - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Metasploitable2-Linux
Metasploitable
Windows 10
Lab-11
SIFT
Mobisec x64 2.0.2
Android
Kali
Home
Metasploitable2-Linux
Kali
capture.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp
No. Time Source Destination Protocol Length Info
21 1.262277 192.168.229.135 192.229.211.108 TCP 54 41452 → 80 [ACK] Seq=1 Ack=1 Win=31691 Len=0
22 1.262857 192.229.211.108 192.168.229.135 TCP 60 [TCP ACKed unseen segment] 80 → 41452 [ACK] Seq=1 Ack=2 Win=64240 Len=0
25 6.554558 192.168.229.135 142.251.41.36 TCP 74 33230 → 443 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=743695465 TSecr=0 WS=128
29 6.559529 142.251.41.36 192.168.229.135 TCP 60 443 → 33230 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
31 6.559618 192.168.229.135 142.251.41.36 TCP 54 33230 → 443 [ACK] Seq=1 Ack=1 Win=32120 Len=0
34 6.562557 192.168.229.135 142.251.41.36 TLSv1.3 713 Client Hello (SNI=www.google.com)
35 6.563039 142.251.41.36 192.168.229.135 TCP 60 443 → 33230 [ACK] Seq=1 Ack=660 Win=64240 Len=0
36 6.563095 192.168.229.135 142.251.41.36 TLSv1.3 60 Change Cipher Spec
37 6.563219 192.168.229.135 142.251.41.36 TLSv1.3 224 Application Data
38 6.563379 142.251.41.36 192.168.229.135 TCP 60 443 → 33230 [ACK] Seq=1 Ack=666 Win=64240 Len=0
39 6.563557 142.251.41.36 192.168.229.135 TCP 60 443 → 33230 [ACK] Seq=1 Ack=836 Win=64240 Len=0
40 6.563731 192.168.229.135 172.217.165.3 TCP 74 37274 → 443 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=2094559960 TSecr=0 WS=128
43 6.570915 192.168.229.135 172.217.1.3 TCP 74 59432 → 443 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PERM TSval=3716755095 TSecr=0 WS=128
47 6.594539 172.217.1.3 192.168.229.135 TCP 60 443 → 59432 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
49 6.594540 172.217.165.3 192.168.229.135 TCP 60 443 → 37274 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
51 6.594573 192.168.229.135 172.217.1.3 TCP 54 59432 → 443 [ACK] Seq=1 Ack=1 Win=32120 Len=0
52 6.594630 192.168.229.135 172.217.165.3 TCP 54 37274 → 443 [ACK] Seq=1 Ack=1 Win=32120 Len=0
53 6.598797 192.168.229.135 172.217.1.3 TLSv1.3 571 Client Hello (SNI=www.gstatic.com)
54 6.599187 172.217.1.3 192.168.229.135 TCP 60 443 → 59432 [ACK] Seq=1 Ack=518 Win=64240 Len=0
Frame 21: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
Ethernet II, Src: VMware_a6:42:47 (00:0c:29:a6:42:47), Dst: VMware_ec:b4:98 (00:50:56:ec:b
Internet Protocol Version 4, Src: 192.168.229.135, Dst: 192.229.211.108
Transmission Control Protocol, Src Port: 41452, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
0000 00 50 56 ec b4 98 00 0c 29 a6 42 47 08 00 45 00
0010 00 28 64 65 40 00 40 06 9b e8 c0 a8 e5 87 c0 e5
0020 d3 6c a1 ec 00 50 0d d7 23 d2 5c 47 59 49 50 10
0030 7b cb 3a 9d 00 00
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% Windov UTF-8
capture.pcap
Packets: 1964 · Displayed: 626 (31.9%)
Profile: Default
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
Upcoming Earnings
Search
ENG IN
20:18 7/16/2024

Installing NetworkMiner




Kali - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Metasploitable2-Linux
Metasploitable
Windows 10
Lab-11
SIFT
Mobisec x64 2.0.2
Android
Kali
Home
Metasploitable2-Linux
Kali
NetworkMiner 2.9.0
File Tools Help
Hosts (17) | Files | Images | Messages | Credentials | Sessions (12) | DNS (34) | Parameters (120) | Keywords | Anomalies
Filter:
String
Clear
Apply
Sort Hosts On:
IP Address (ascending)
Sort and Refresh
31.13.80.12 [scontent.xx.fbcdn.net] [static.xx.fbcdn.net] [scontent-yyz1-1.xx.fbcdn.net] (Other)
31.13.80.36 [star-mini.c10r.facebook.com] [www.facebook.com]
142.251.33.170 [safebrowsing.googleapis.com] (Other)
142.251.41.36 [www.google.com] (Other)
142.251.41.46 [www3.l.google.com] [ogs.google.com] (Other)
142.251.41.67 [ssl.gstatic.com] (Other)
172.217.1.3 [www.gstatic.com] (Other)
172.217.165.3 [fonts.gstatic.com] (Other)
192.168.229.2
192.168.229.135
192.229.211.108
2607:f8b0:400b:802::2003 [fonts.gstatic.com]
2607:f8b0:400b:803::2004 [www.google.com] [ssl.gstatic.com]
2607:f8b0:400b:803::200e [www3.l.google.com] [ogs.google.com]
2607:f8b0:400b:80f::2003 [www.gstatic.com]
2a03:2880:f00e:13:face:b00c:0:3 [scontent.xx.fbcdn.net] [static.xx.fbcdn.net] [scontent-yyz1-1.xx.fbcdn.net]
2a03:2880:f10e:83:face:b00c:0:25de [star-mini.c10r.facebook.com] [www.facebook.com]
Case Panel
Filename MD5
capture.pc 2fcc291b
Reload Case Files
Buffered Frames to Parse:
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% UTF-8
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
77°F
Mostly sunny
Search
ENG
IN
20:20
7/16/2024


Kali - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Metasploitable2-Linux
Metasploitable
Windows 10
Lab-11
SIFT
Mobisec x64 2.0.2
Android
Kali
Home
Metasploitable2-Linux
Kali
NetworkMiner 2.9.0
File Tools Help
Hosts (17) | Files | Images | Messages | Credentials | Sessions (12) | DNS (34) | Parameters (120) | Keywords | Anomalies
Filter keyword:
Case sensitive
ExactPhrase
Any column
Clear
Apply
Frame nr. | Timestamp | Client | Client Port | Server | Server Port | IP TTL | DNS TTL (time) | Transaction ID | Type | DNS Query | DNS Answer
3 | 2024-07-16 23:38:05 UTC | 192.168.229.135 | 33573 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9852 | 0x0005 (CNAME) | www.facebook.com | star-mini.c10r.facebook.com
3 | 2024-07-16 23:38:05 UTC | 192.168.229.135 | 33573 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9852 | 0x0001 (HostAddress) | star-mini.c10r.facebook.com | 31.13.80.36
4 | 2024-07-16 23:38:05 UTC | 192.168.229.135 | 33573 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x685E | 0x0005 (CNAME) | www.facebook.com | star-mini.c10r.facebook.com
4 | 2024-07-16 23:38:05 UTC | 192.168.229.135 | 33573 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x685E | 0x001C (AAAA) | star-mini.c10r.facebook.com | 2a03:2880:f10e:83:face:b00c:0:25de
27 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 38214 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x161A | 0x0001 (HostAddress) | www.google.com | 142.251.41.36
28 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 60518 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x0363 | 0x0001 (HostAddress) | fonts.gstatic.com | 172.217.165.3
30 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 60518 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x236F | 0x001C (AAAA) | fonts.gstatic.com | 2607:f8b0:400b:802::2003
41 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 39641 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x1F73 | 0x0001 (HostAddress) | www.gstatic.com | 172.217.1.3
42 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 39641 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x7B6F | 0x001C (AAAA) | www.gstatic.com | 2607:f8b0:400b:80f::2003
48 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 54178 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x0420 | 0x0001 (HostAddress) | www.google.com | 142.251.41.36
50 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 54178 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xA31E | 0x001C (AAAA) | www.google.com | 2607:f8b0:400b:803::2004
101 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 55928 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x5314 | 0x0001 (HostAddress) | www.gstatic.com | 172.217.1.3
102 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 36883 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x17C4 | 0x0001 (HostAddress) | fonts.gstatic.com | 172.217.165.3
103 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 36883 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xAFC6 | 0x001C (AAAA) | fonts.gstatic.com | 2607:f8b0:400b:802::2003
104 | 2024-07-16 23:38:12 UTC | 192.168.229.135 | 55928 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xBB17 | 0x001C (AAAA) | www.gstatic.com | 2607:f8b0:400b:80f::2003
355 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 50902 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9D08 | 0x0005 (CNAME) | static.xx.fbcdn.net | scontent.xx.fbcdn.net
355 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 50902 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9D08 | 0x0001 (HostAddress) | scontent.xx.fbcdn.net | 31.13.80.12
356 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 50902 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xAC0C | 0x0005 (CNAME) | static.xx.fbcdn.net | scontent.xx.fbcdn.net
356 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 50902 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xAC0C | 0x001C (AAAA) | scontent.xx.fbcdn.net | 2a03:2880:f00e:13:face:b00c:0:3
551 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 36260 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x6A23 | 0x0001 (HostAddress) | scontent.xx.fbcdn.net | 31.13.80.12
552 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 36260 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x8521 | 0x001C (AAAA) | scontent.xx.fbcdn.net | 2a03:2880:f00e:13:face:b00c:0:3
613 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 48905 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x93C3 | 0x0005 (CNAME) | static.xx.fbcdn.net | scontent.xx.fbcdn.net
613 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 48905 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x93C3 | 0x0001 (HostAddress) | scontent.xx.fbcdn.net | 31.13.80.12
616 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 48905 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xA6C1 | 0x0005 (CNAME) | static.xx.fbcdn.net | scontent.xx.fbcdn.net
616 | 2024-07-16 23:38:14 UTC | 192.168.229.135 | 48905 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xA6C1 | 0x001C (AAAA) | scontent.xx.fbcdn.net | 2a03:2880:f00e:13:face:b00c:0:3
1105 | 2024-07-16 23:38:16 UTC | 192.168.229.135 | 36388 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x1EE2 | 0x0001 (HostAddress) | scontent-yyz1-1.xx.fbcdn.net | 31.13.80.12
1106 | 2024-07-16 23:38:16 UTC | 192.168.229.135 | 36388 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xA6FF | 0x001C (AAAA) | scontent-yyz1-1.xx.fbcdn.net | 2a03:2880:f00e:13:face:b00c:0:3
1328 | 2024-07-16 23:38:16 UTC | 192.168.229.135 | 49174 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x164F | 0x0001 (HostAddress) | safebrowsing.googleapis.com | 142.251.33.170
1515 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 37428 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x5850 | 0x0005 (CNAME) | ogs.google.com | www3.l.google.com
1515 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 37428 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x5850 | 0x0001 (HostAddress) | www3.l.google.com | 142.251.41.46
1516 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 37428 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9F4D | 0x0005 (CNAME) | ogs.google.com | www3.l.google.com
1516 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 37428 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x9F4D | 0x001C (AAAA) | www3.l.google.com | 2607:f8b0:400b:803::200e
1653 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 48173 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0x1C73 | 0x0001 (HostAddress) | ssl.gstatic.com | 142.251.41.67
1654 | 2024-07-16 23:38:18 UTC | 192.168.229.135 | 48173 | 192.168.229.2 | 53 | 128 | 00:00:05 | 0xF972 | 0x001C (AAAA) | ssl.gstatic.com | 2607:f8b0:400b:804::2003
Case Panel
Filename | MD5
capture.pc 2fcc291b
Reload Case Files
Buffered Frames to Parse:
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 | 45 characters | 100% | Windov
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
77°F
Mostly sunny
Search
ENG
IN
20:21
7/16/2024


Kali - VMware Workstation
File Edit View VM Tabs Help
Library
Type here to search
My Computer
Metasploitable2-Linux
Metasploitable
Windows 10
Lab-11
SIFT
Mobisec x64 2.0.2
Android
Kali
Home
Metasploitable2-Linux
Kali
NetworkMiner 2.9.0
File Tools Help
Hosts (17) | Files | Images | Messages | Credentials | Sessions (12) | DNS (34) | Parameters (120) | Keywords | Anomalies
Filter keyword:
Case sensitive
ExactPhrase
Any column
Clear
Apply
Frame nr. | Client host | C. port | Server host | S. port | Protocol (application layer) | Start time | iRTT (ms)
25 | 192.168.229.135 | 33230 | 142.251.41.36 [www.google.com] | 443 | SSL | 2024-07-16 23:38:12 UTC
43 | 192.168.229.135 | 59432 | 172.217.1.3 [www.gstatic.com] | 443 | SSL | 2024-07-16 23:38:12 UTC
40 | 192.168.229.135 | 37274 | 172.217.165.3 [fonts.gstatic.com] | 443 | SSL | 2024-07-16 23:38:12 UTC
357 | 192.168.229.135 | 38762 | 31.13.80.12 [scontent.xx.fbcdn.net] [static.xx.fbcdn.n | 443 | SSL | 2024-07-16 23:38:14 UTC
556 | 192.168.229.135 | 51278 | 31.13.80.12 [scontent.xx.fbcdn.net] [static.xx.fbcdn.n | 443 | SSL | 2024-07-16 23:38:14 UTC
553 | 192.168.229.135 | 38778 | 31.13.80.12 [scontent.xx.fbcdn.net] [static.xx.fbcdn.n | 443 | SSL | 2024-07-16 23:38:14 UTC
1108 | 192.168.229.135 | 38780 | 31.13.80.12 [scontent.xx.fbcdn.net] [static.xx.fbcdn.n | 443 | SSL | 2024-07-16 23:38:16 UTC
1329 | 192.168.229.135 | 58374 | 142.251.33.170 [safebrowsing.googleapis.com] | 443 | SSL | 2024-07-16 23:38:16 UTC
1517 | 192.168.229.135 | 53054 | 142.251.41.46 [www3.l.google.com] [ogs.google.com | 443 | SSL | 2024-07-16 23:38:18 UTC
1655 | 192.168.229.135 | 55648 | 142.251.41.67 [ssl.gstatic.com] | 443 | SSL | 2024-07-16 23:38:18 UTC
1661 | 192.168.229.135 | 55660 | 142.251.41.67 [ssl.gstatic.com] | 443 | SSL | 2024-07-16 23:38:19 UTC
21 | 192.168.229.135 | 41452 | 192.229.211.108 | 80 | | 2024-07-16 23:38:07 UTC
Case Panel
Filename | MD5
capture.pc 2fcc291b
Reload Case Files
Buffered Frames to Parse:
File Edit View
Name: Ishan Aakash Patel
StudentID: 146151238
Ln 2, Col 21 45 characters 100% Windov UTF-8
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
77°F
Mostly sunny
Search
ENG
IN
20:21
7/16/2024

Learning Experience

This lab was an enriching experience that allowed me to dive deep into the world of network traffic analysis using tcpdump. Initially, I was somewhat apprehensive about using a command-line tool for such a technical task. However, the step-by-step instructions provided a clear path to follow, which made the process manageable and even enjoyable. From installing tcpdump to capturing and analyzing packets, each step built on the previous one, reinforcing my understanding of network interfaces and packet structures. By the end of the lab, I felt much more confident in my ability to monitor and troubleshoot network issues using tcpdump.

One of the most valuable aspects of this lab was the hands-on practice with different filtering options. Learning how to capture only specific types of traffic, such as HTTP GET requests or DNS queries, provided practical skills that I can apply in real-world scenarios. Additionally, saving captured packets to a file and analyzing them later with Wireshark opened my eyes to the power of combining different tools for comprehensive network analysis. Overall, this lab not only enhanced my technical skills but also boosted my confidence in using command-line tools to perform detailed network diagnostics.