Maria Valencia

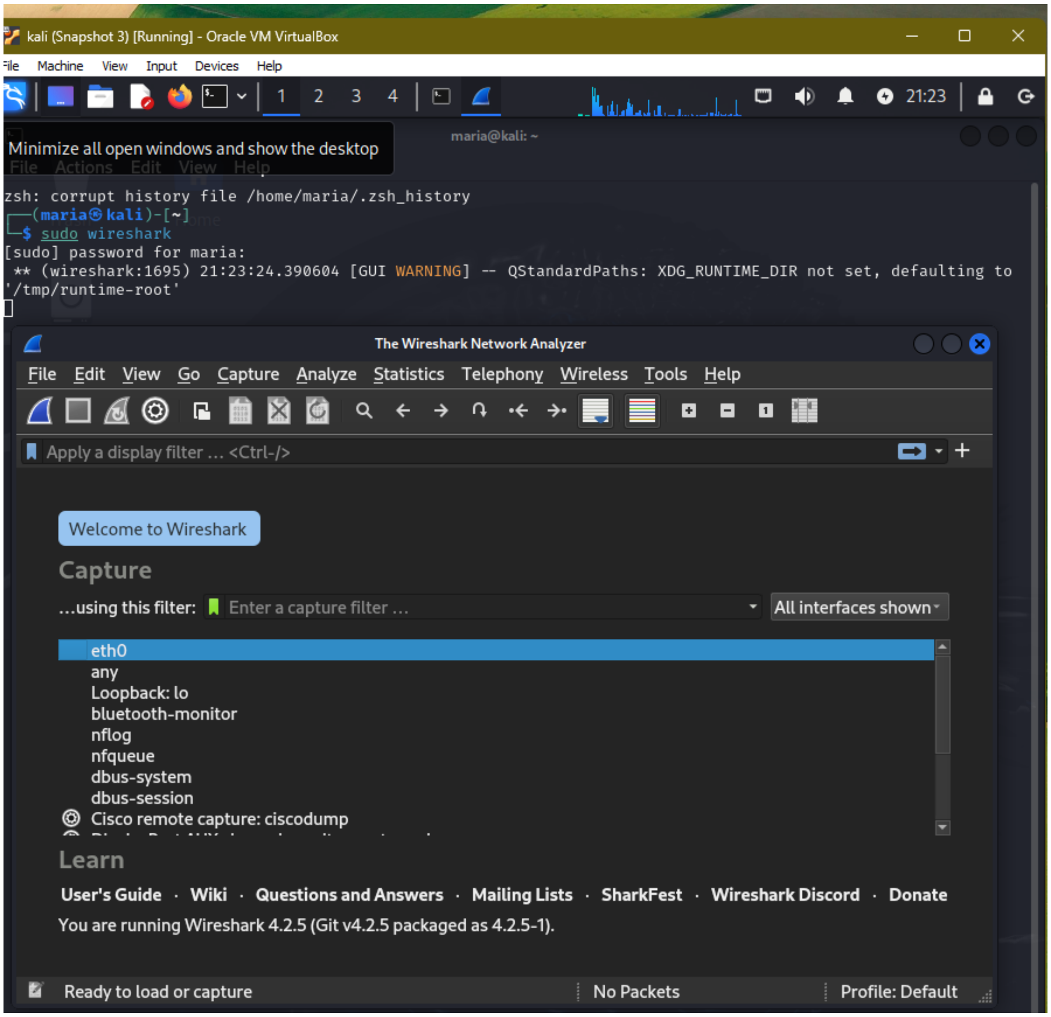
Lab 03

CSC 154

Network Security

3.1 Wireshark Packet Capture

In this exercise, I logged into my Kali VM and launched the terminal. I ran the command "Sudo Wireshark" to start up Wireshark adn started capturing packets.



kali (Snapshot 3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
21:23
Minimize all open windows and show the desktop
maria@kali: ~
File Actions Edit View Help

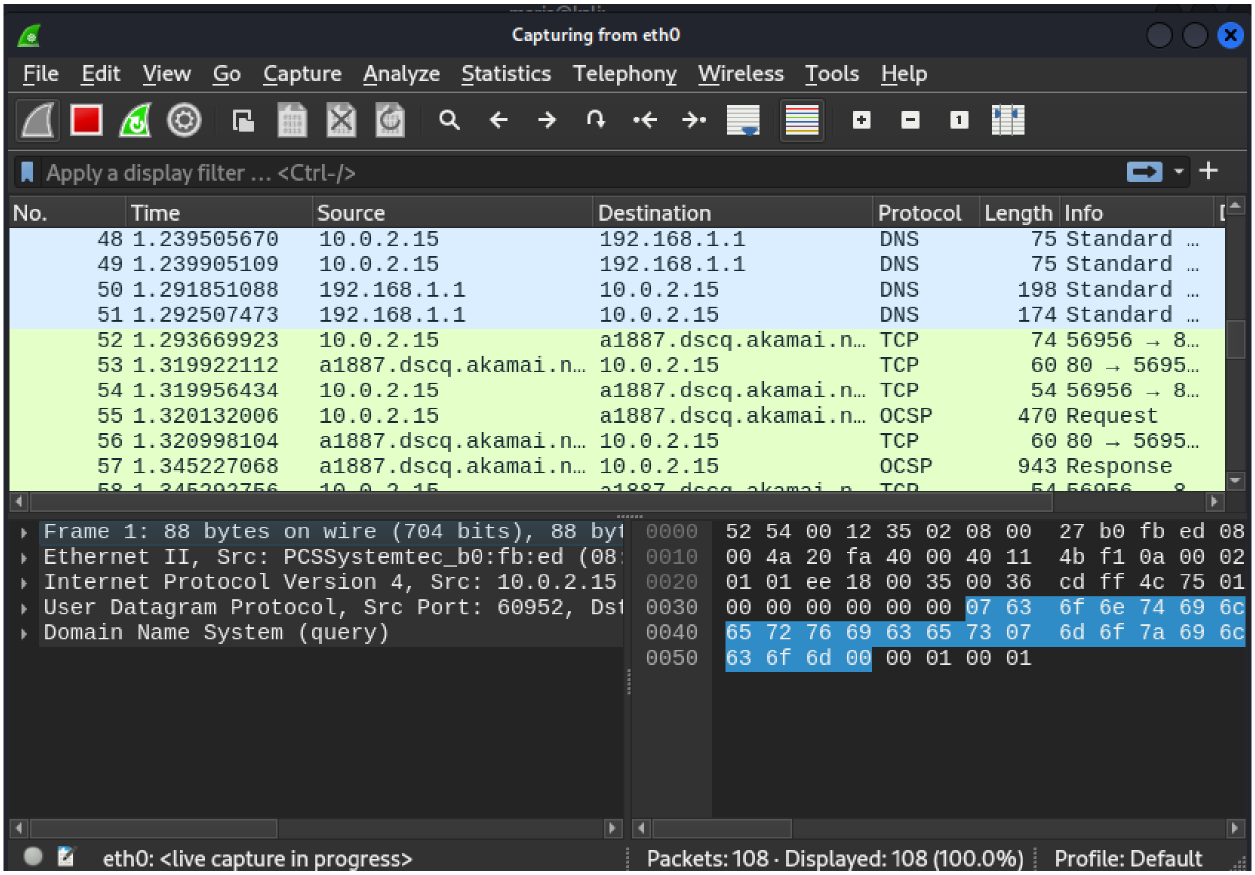
```
zsh: corrupt history file /home/maria/.zsh_history
┌──(maria㉿kali)-[~]
└─$ sudo wireshark
[sudo] password for maria:
 ** (wireshark:1695) 21:23:24.390604 [GUI WARNING] -- QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to
'/tmp/runtime-root'
```

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark
Capture
…using this filter:
Enter a capture filter …
All interfaces shown
eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate

You are running Wireshark 4.2.5 (Git v4.2.5 packaged as 4.2.5-1).

Ready to load or capture | No Packets | Profile: Default

After I started capturing packets, i stopped capturing packets by pressing the red stop button.


Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

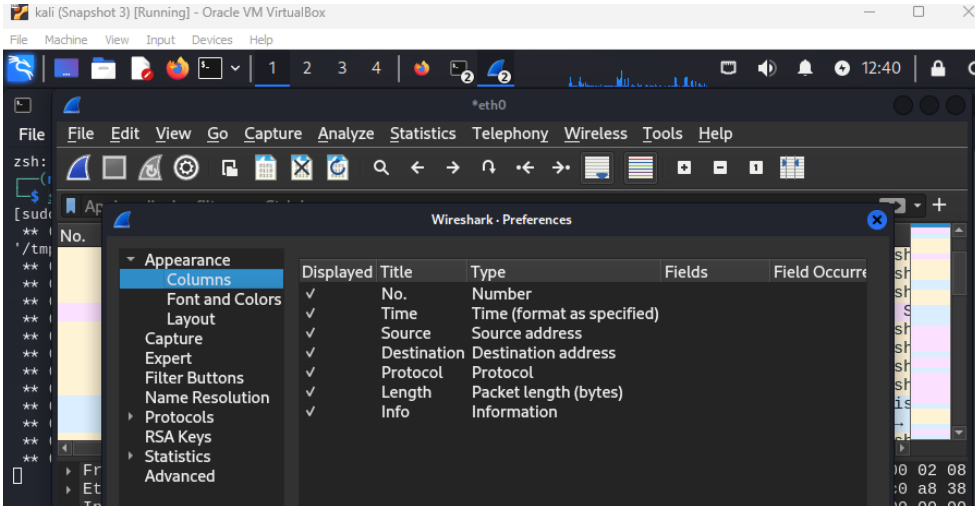
```
No.    Time          Source               Destination          Protocol  Length Info
    48 1.239505670   10.0.2.15            192.168.1.1          DNS          75 Standard …
    49 1.239905109   10.0.2.15            192.168.1.1          DNS          75 Standard …
    50 1.291851088   192.168.1.1          10.0.2.15            DNS         198 Standard …
    51 1.292507473   192.168.1.1          10.0.2.15            DNS         174 Standard …
    52 1.293669923   10.0.2.15            a1887.dscq.akamai.n… TCP          74 56956 → 8…
    53 1.319922112   a1887.dscq.akamai.n… 10.0.2.15            TCP          60 80 → 5695…
    54 1.319956434   10.0.2.15            a1887.dscq.akamai.n… TCP          54 56956 → 8…
    55 1.320132006   10.0.2.15            a1887.dscq.akamai.n… OCSP        470 Request
    56 1.320998104   a1887.dscq.akamai.n… 10.0.2.15            TCP          60 80 → 5695…
    57 1.345227068   a1887.dscq.akamai.n… 10.0.2.15            OCSP        943 Response
```

```
Frame 1: 88 bytes on wire (704 bits), 88 byt
Ethernet II, Src: PCSSystemtec_b0:fb:ed (08
Internet Protocol Version 4, Src: 10.0.2.15
User Datagram Protocol, Src Port: 60952, Dst
Domain Name System (query)
```

```
0000  52 54 00 12 35 02 08 00  27 b0 fb ed 08
0010  00 4a 20 fa 40 00 40 11  4b f1 0a 00 02
0020  01 01 ee 18 00 35 00 36  cd ff 4c 75 01
0030  00 00 00 00 00 00 07 63  6f 6e 74 69 6c
0040  65 72 76 69 63 65 73 07  6d 6f 7a 69 6c
0050  63 6f 6d 00 00 01 00 01
```

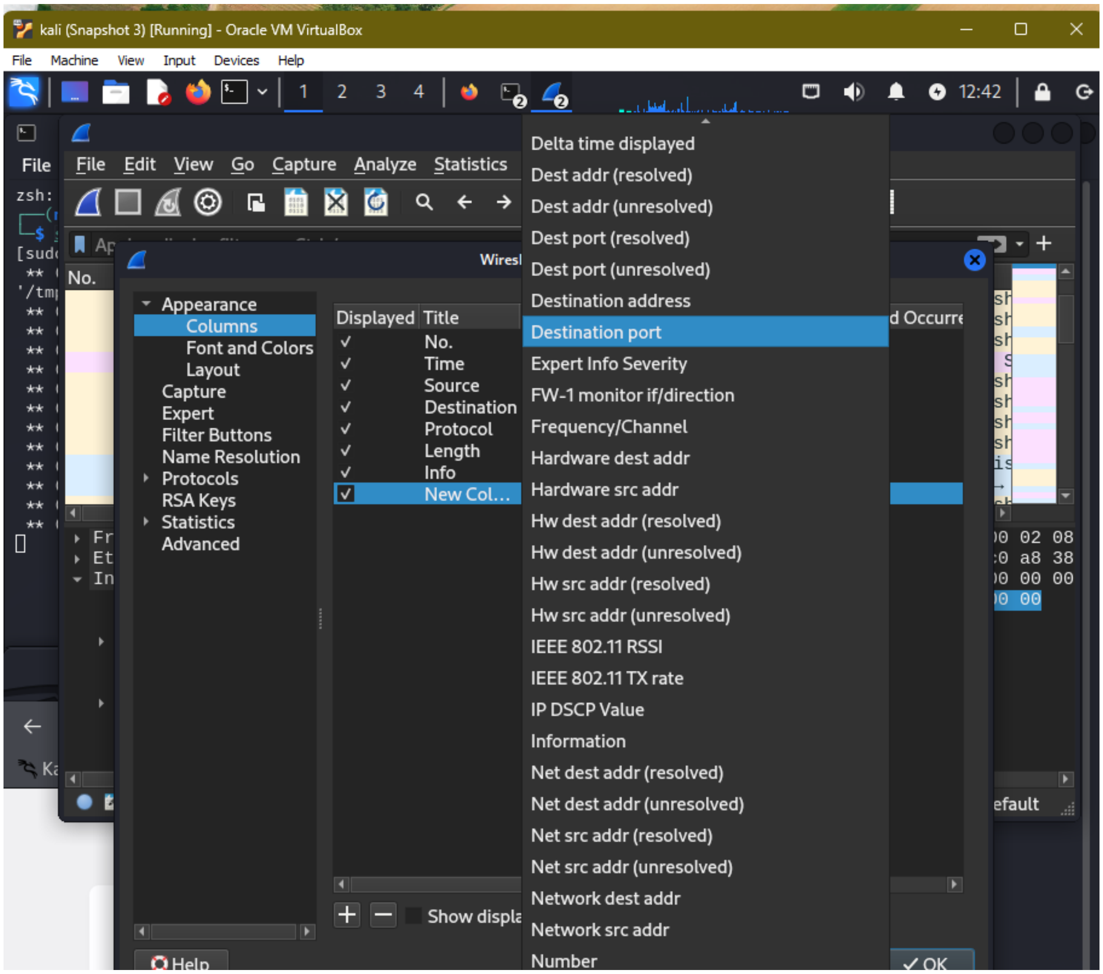
eth0: <live capture in progress>   Packets: 108 · Displayed: 108 (100.0%)   Profile: Default

I then went to add the source and destination ports as columns. To do this, I right clicked the column header and selected "Column Preferences" from the context menu.

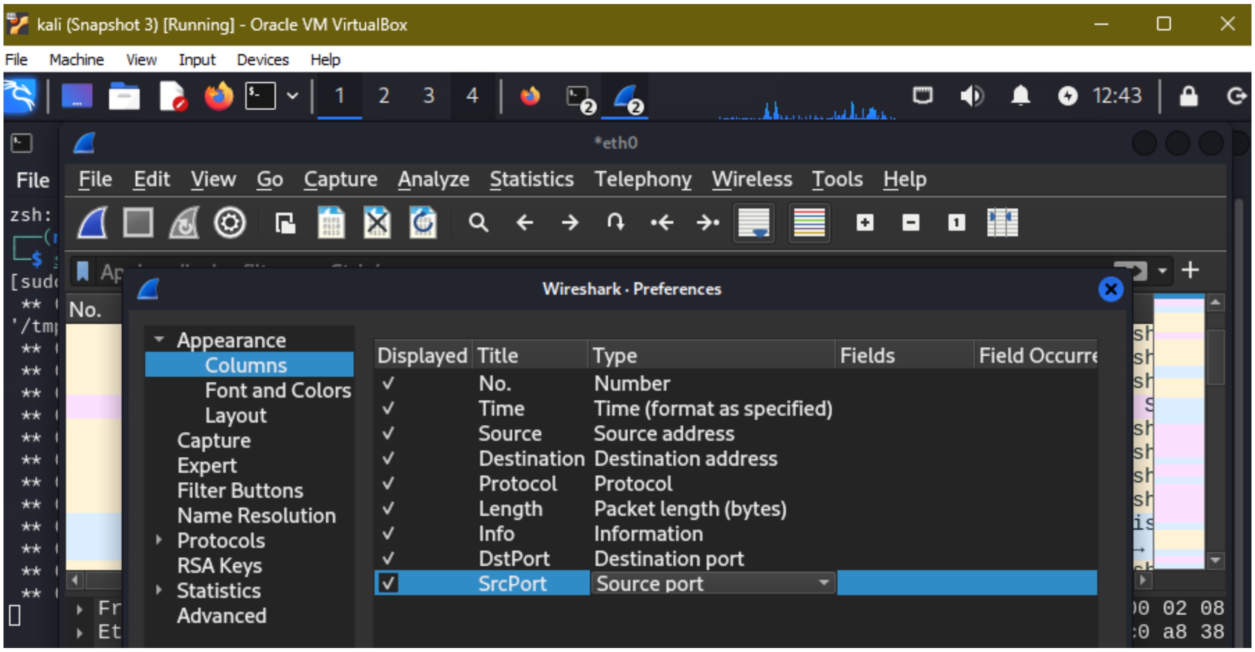

kali (Snapshot 3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
12:40
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Wireshark · Preferences
Appearance
Columns
Font and Colors
Layout
Capture
Expert
Filter Buttons
Name Resolution
Protocols
RSA Keys
Statistics
Advanced

Displayed	Title	Type	Fields	Field Occurre
✓	No.	Number		
✓	Time	Time (format as specified)		
✓	Source	Source address		
✓	Destination	Destination address		
✓	Protocol	Protocol		
✓	Length	Packet length (bytes)		
✓	Info	Information		

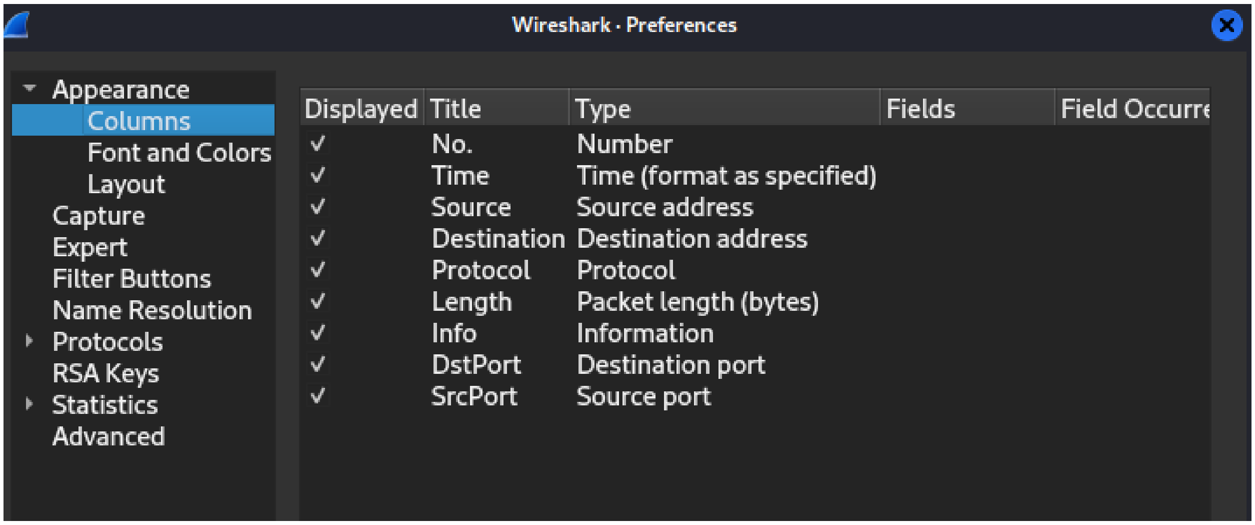
With the column preferences open, I pushed the “+” button at the bottom and double clicked “Number” Then selected “Destination Port” from the drop-down menu, double clicked the title and entered “DstPort”. I repeated these steps for the Source Port as well. Then pressed “OK” to complete the changes.



Displayed	Title
✓	No.
✓	Time
✓	Source
✓	Destination
✓	Protocol
✓	Length
✓	Info
☑	New Col…

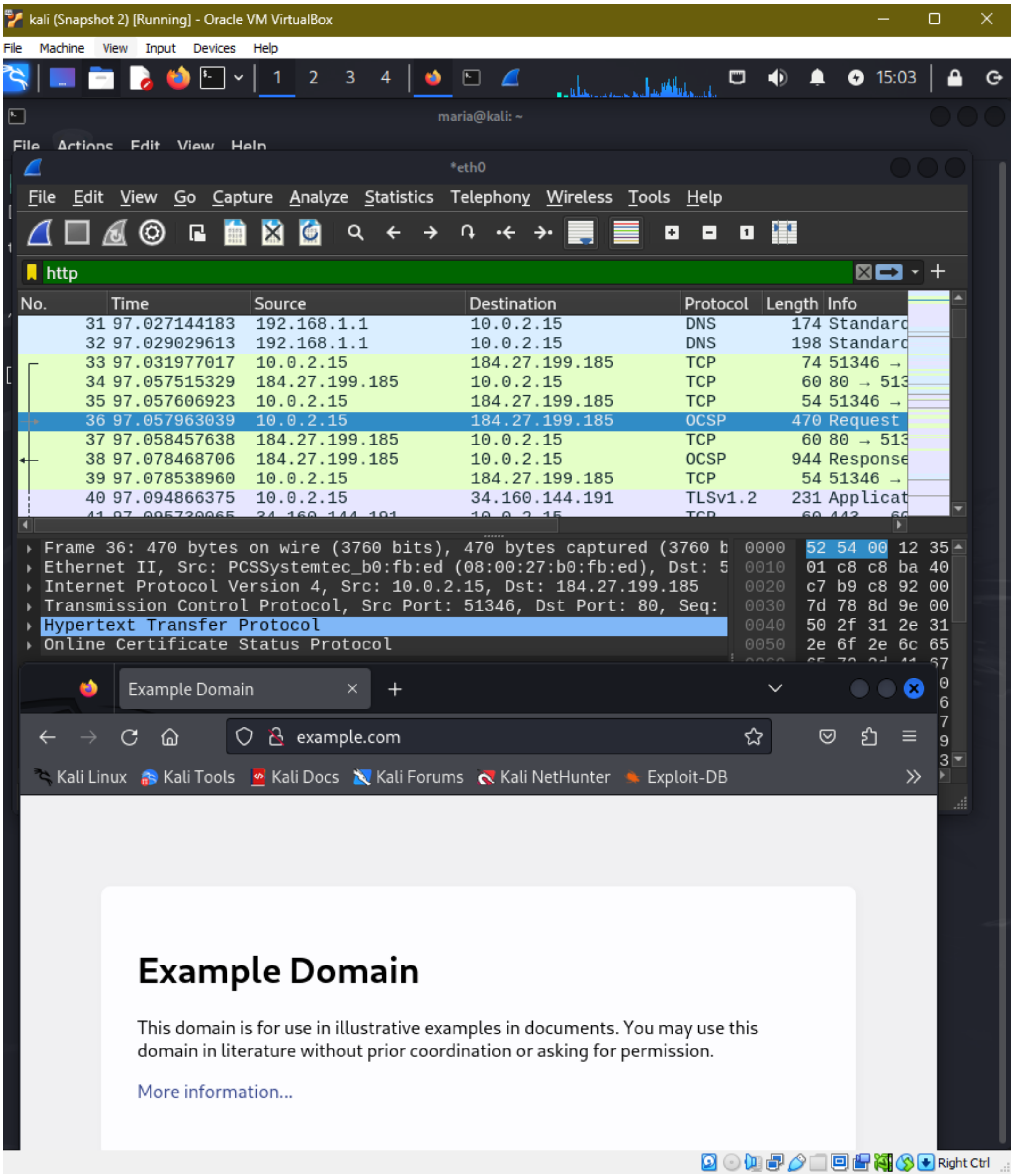

kali (Snapshot 3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
12:43
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Wireshark · Preferences
Appearance
Columns
Font and Colors
Layout
Capture
Expert
Filter Buttons
Name Resolution
Protocols
RSA Keys
Statistics
Advanced

Displayed	Title	Type	Fields	Field Occurre
✓	No.	Number		
✓	Time	Time (format as specified)		
✓	Source	Source address		
✓	Destination	Destination address		
✓	Protocol	Protocol		
✓	Length	Packet length (bytes)		
✓	Info	Information		
✓	DstPort	Destination port		
☑	SrcPort	Source port		



Displayed	Title	Type	Fields	Field Occurre
✓	No.	Number		
✓	Time	Time (format as specified)		
✓	Source	Source address		
✓	Destination	Destination address		
✓	Protocol	Protocol		
✓	Length	Packet length (bytes)		
✓	Info	Information		
✓	DstPort	Destination port		
✓	SrcPort	Source port		

I launched a browser from the Kali VM while the Wireshark packet capture was running. In the browser, I navigated to the given URLs. After each site loaded, I stopped the packet capture, found the related packets in Wireshark and viewed each stream using filters "http" and "tls".


kali (Snapshot 2) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
15:03
maria@kali: ~
File Actions Edit View Help
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Right Ctrl

```
http
```

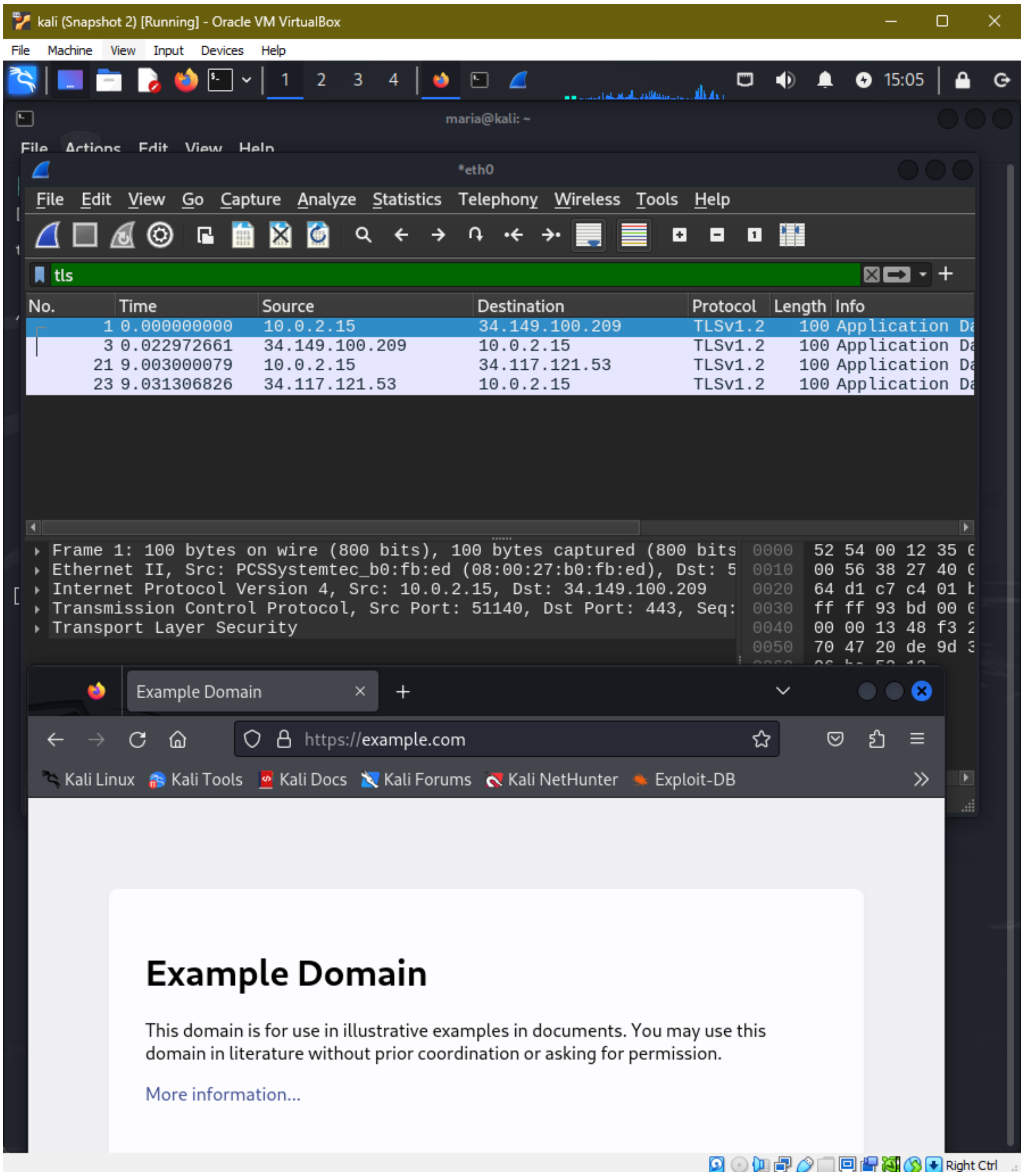
```
No.  Time           Source          Destination     Protocol  Length Info
  31 97.027144183   192.168.1.1     10.0.2.15       DNS          174 Standard
  32 97.029029613   192.168.1.1     10.0.2.15       DNS          198 Standard
  33 97.031977017   10.0.2.15       184.27.199.185  TCP           74 51346 →
  34 97.057515329   184.27.199.185  10.0.2.15       TCP           60 80 → 513
  35 97.057606923   10.0.2.15       184.27.199.185  TCP           54 51346 →
  36 97.057963039   10.0.2.15       184.27.199.185  OCSP         470 Request
  37 97.058457638   184.27.199.185  10.0.2.15       TCP           60 80 → 513
  38 97.078468706   184.27.199.185  10.0.2.15       OCSP         944 Response
  39 97.078538960   10.0.2.15       184.27.199.185  TCP           54 51346 →
  40 97.094866375   10.0.2.15       34.160.144.191  TLSv1.2      231 Applicat
```

```
› Frame 36: 470 bytes on wire (3760 bits), 470 bytes captured (3760 b   0000  52 54 00 12 35
› Ethernet II, Src: PCSSystemtec_b0:fb:ed (08:00:27:b0:fb:ed), Dst: 5   0010  01 c8 c8 ba 40
› Internet Protocol Version 4, Src: 10.0.2.15, Dst: 184.27.199.185      0020  c7 b9 c8 92 00
› Transmission Control Protocol, Src Port: 51346, Dst Port: 80, Seq:    0030  7d 78 8d 9e 00
› Hypertext Transfer Protocol                                           0040  50 2f 31 2e 31
› Online Certificate Status Protocol                                    0050  2e 6f 2e 6c 65
```

Example Domain

This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.

More information...



kali (Snapshot 2) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
15:05
maria@kali: ~
File Actions Edit View Help
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Example Domain
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Right Ctrl

tls

```
No.   Time          Source          Destination     Protocol Length Info
    1 0.000000000   10.0.2.15       34.149.100.209  TLSv1.2     100 Application Da
    3 0.022972661   34.149.100.209  10.0.2.15       TLSv1.2     100 Application Da
   21 9.003000079   10.0.2.15       34.117.121.53   TLSv1.2     100 Application Da
   23 9.031306826   34.117.121.53   10.0.2.15       TLSv1.2     100 Application Da
```

```
› Frame 1: 100 bytes on wire (800 bits), 100 bytes captured (800 bits
› Ethernet II, Src: PCSSystemtec_b0:fb:ed (08:00:27:b0:fb:ed), Dst: 5
› Internet Protocol Version 4, Src: 10.0.2.15, Dst: 34.149.100.209
› Transmission Control Protocol, Src Port: 51140, Dst Port: 443, Seq:
› Transport Layer Security
```

```
0000  52 54 00 12 35 0
0010  00 56 38 27 40 0
0020  64 d1 c7 c4 01 b
0030  ff ff 93 bd 00 0
0040  00 00 13 48 f3 2
0050  70 47 20 de 9d 3
```

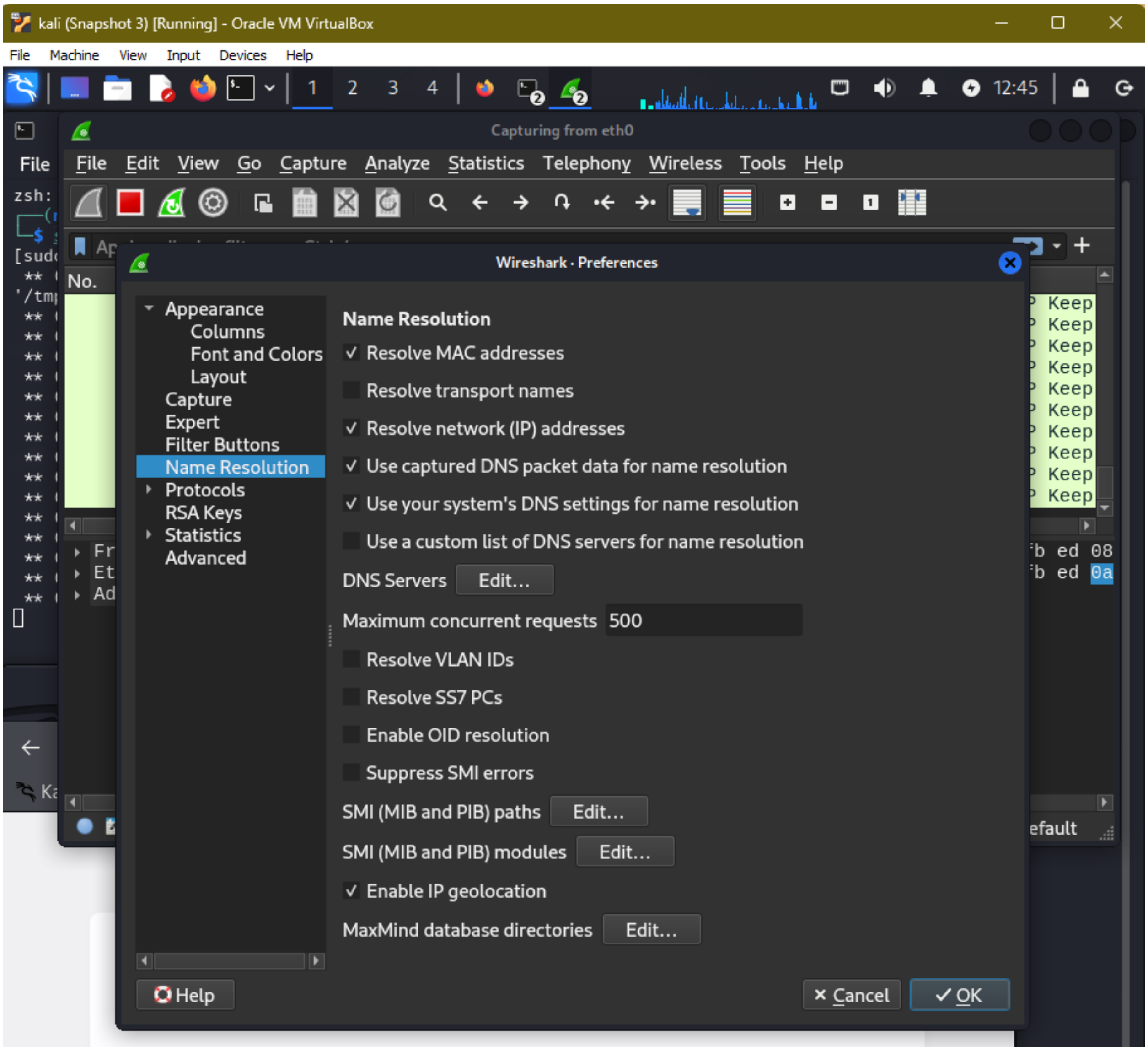
https://example.com

Example Domain

This domain is for use in illustrative examples in documents. You may use this domain in literature without prior coordination or asking for permission.

More information...

To enable domain names, I went to edit and preferences and selected “Name Resolution” on the window prompted. Then I checked the “Resolve network (IP) addresses option and I clicked “OK” to apply the setting.



Name Resolution

☑ Resolve MAC addresses

□ Resolve transport names

☑ Resolve network (IP) addresses

☑ Use captured DNS packet data for name resolution

☑ Use your system's DNS settings for name resolution

□ Use a custom list of DNS servers for name resolution

DNS Servers Edit…

Maximum concurrent requests 500

□ Resolve VLAN IDs

□ Resolve SS7 PCs

□ Enable OID resolution

□ Suppress SMI errors

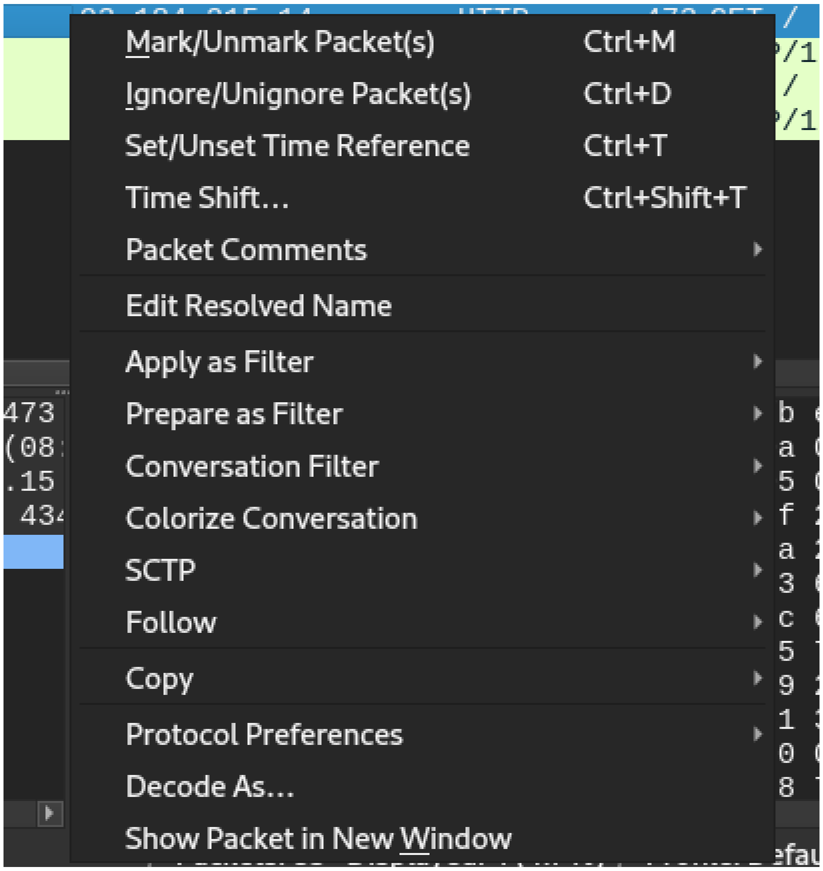
SMI (MIB and PIB) paths Edit…

SMI (MIB and PIB) modules Edit…

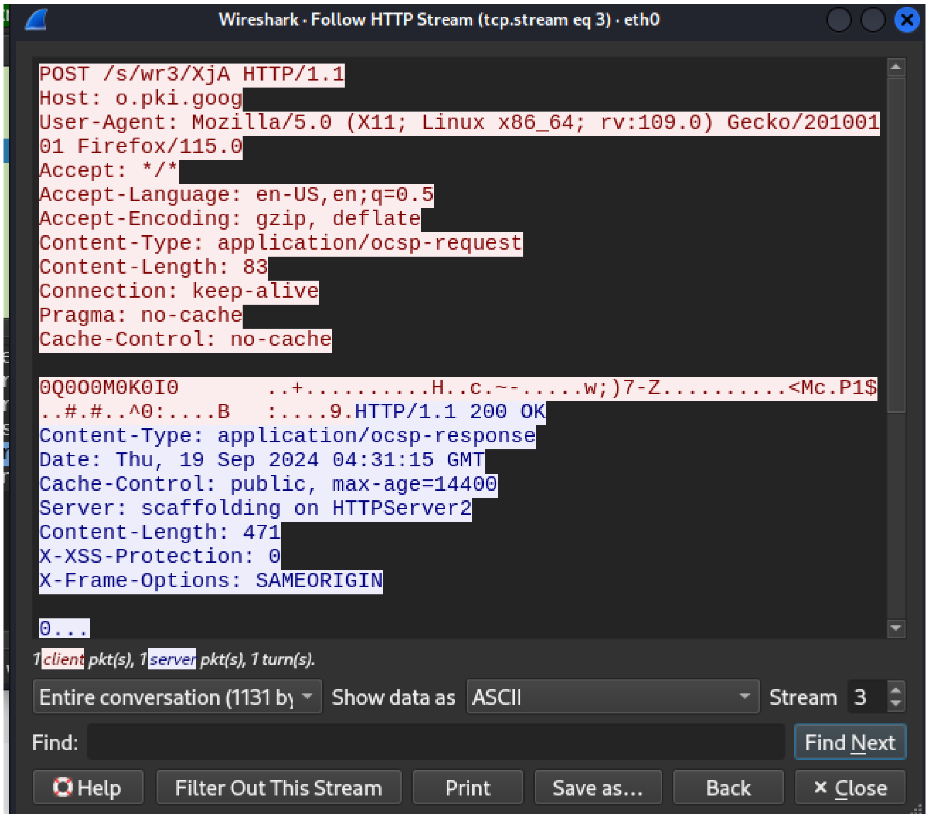
☑ Enable IP geolocation

MaxMind database directories Edit…

Then, I right clicked onm a packet and selected “follow” and then “HTTP STREAM” and I did the same for “TLS STREAM”.



These stream windows opened and displayed the request (red text) and the response (blue text) and any subsequent related packets.

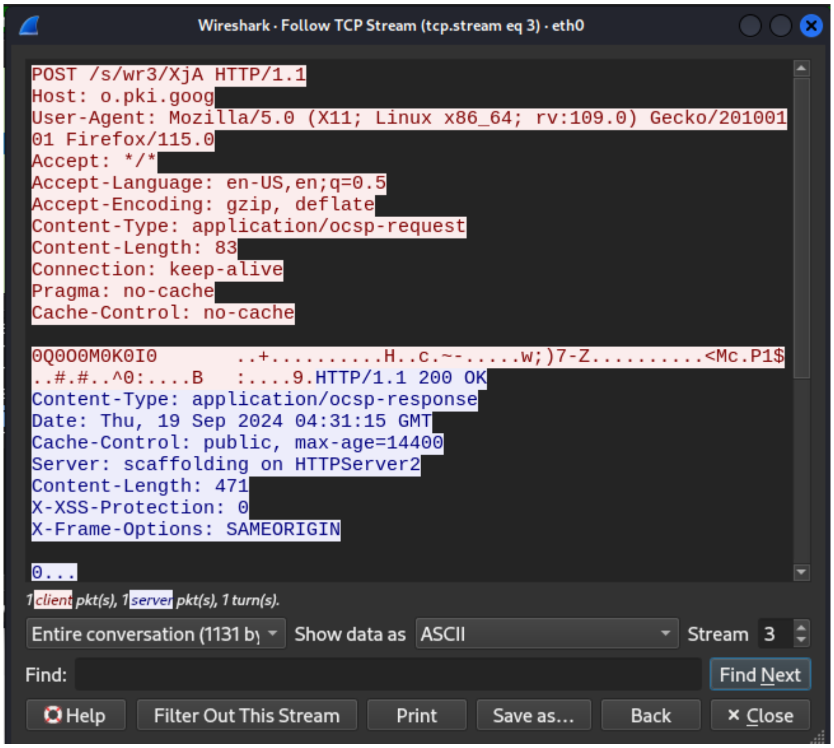


```
POST /s/wr3/XjA HTTP/1.1
Host: o.pki.goog
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/201001
01 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/ocsp-request
Content-Length: 83
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

0Q0O0M0K0I0       ..+..........H..c.~-.....w;)7-Z..........<Mc.P1$
..#.#..^0:....B   :....9.HTTP/1.1 200 OK
Content-Type: application/ocsp-response
Date: Thu, 19 Sep 2024 04:31:15 GMT
Cache-Control: public, max-age=14400
Server: scaffolding on HTTPServer2
Content-Length: 471
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN

0...
```

1 client pkt(s), 1 server pkt(s), 1 turn(s).

Entire conversation (1131 by ▾ Show data as ASCII ▾ Stream 3

Find: Find Next

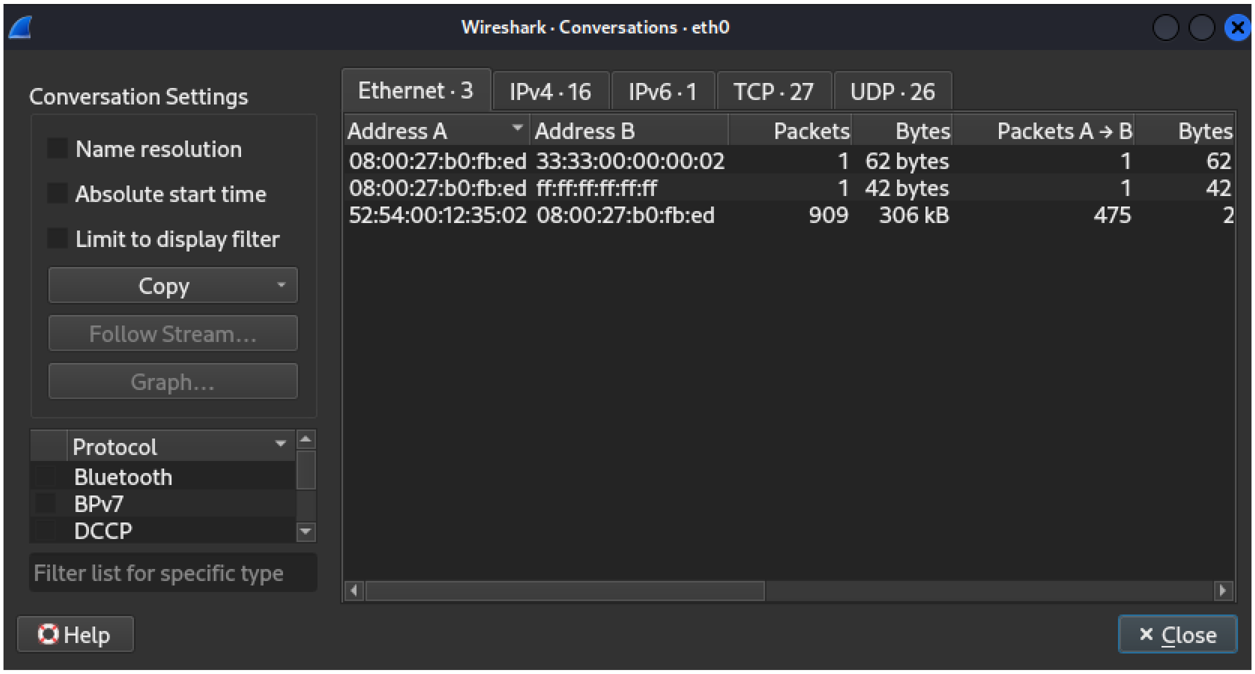


```
POST /s/wr3/XjA HTTP/1.1
Host: o.pki.goog
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/201001
01 Firefox/115.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/ocsp-request
Content-Length: 83
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache

0Q0O0M0K0I0       ..+..........H..c.~-.....w;)7-Z..........<Mc.P1$
..#.#..^0:....B   :....9.HTTP/1.1 200 OK
Content-Type: application/ocsp-response
Date: Thu, 19 Sep 2024 04:31:15 GMT
Cache-Control: public, max-age=14400
Server: scaffolding on HTTPServer2
Content-Length: 471
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN

0...
```

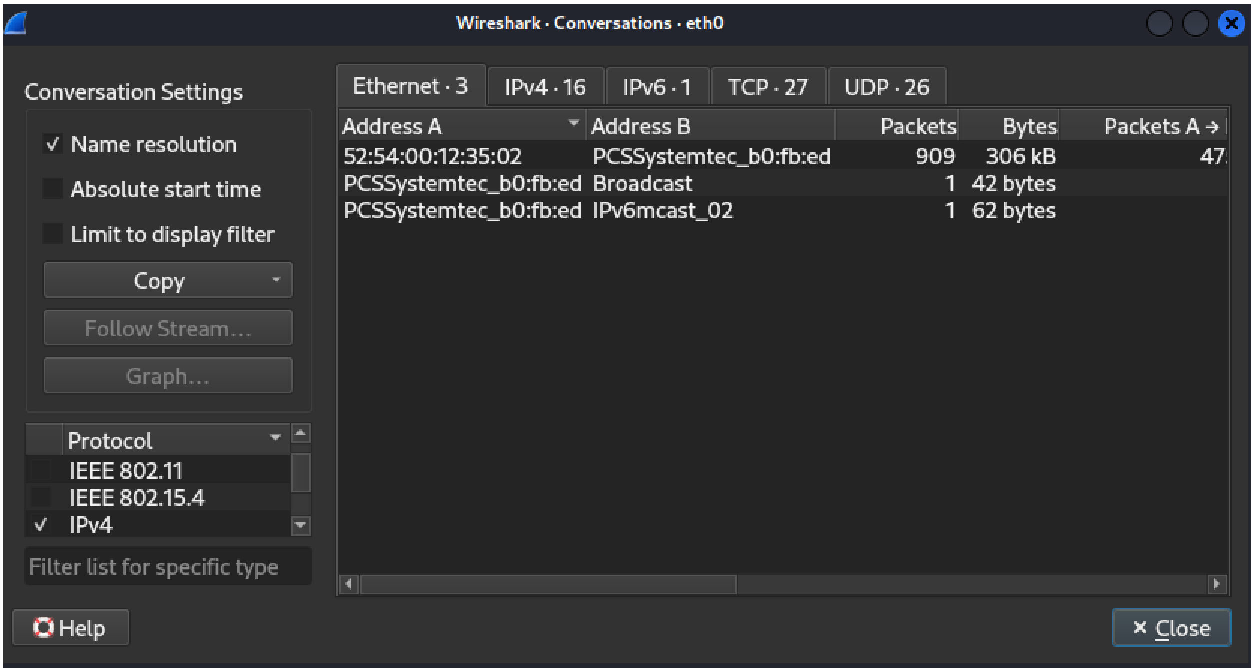
1 client pkt(s), 1 server pkt(s), 1 turn(s).

Entire conversation (1131 by Show data as ASCII Stream 3

I then closed the stream and deleted the http and tls filter then plressed eneter to display all the captured packets. Then I selected the "statistics" menu and chose "Conversations" I pressed the name "Name Resolution" option on the left menu and chose the IPV4 tab to observe the statistics from our connection to google.com.



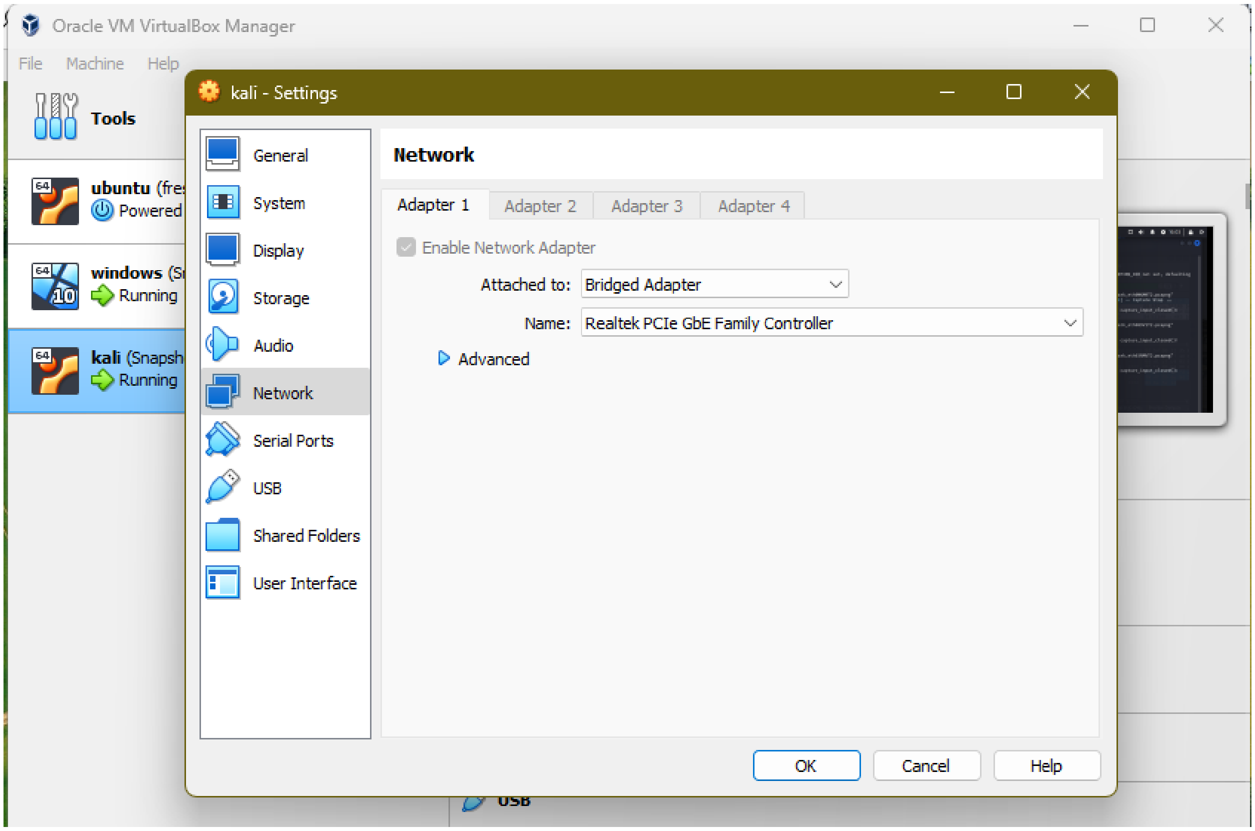
Address A	Address B	Packets	Bytes	Packets A → B	Bytes
08:00:27:b0:fb:ed	33:33:00:00:00:02	1	62 bytes	1	62
08:00:27:b0:fb:ed	ff:ff:ff:ff:ff:ff	1	42 bytes	1	42
52:54:00:12:35:02	08:00:27:b0:fb:ed	909	306 kB	475	2



Address A	Address B	Packets	Bytes	Packets A →
52:54:00:12:35:02	PCSSystemtec_b0:fb:ed	909	306 kB	47
PCSSystemtec_b0:fb:ed	Broadcast	1	42 bytes	
PCSSystemtec_b0:fb:ed	IPv6mcast_02	1	62 bytes	

3.2 Network Utilities

In this exercise, I used my Windows and Kali VMs in the network bridge adapter mode.

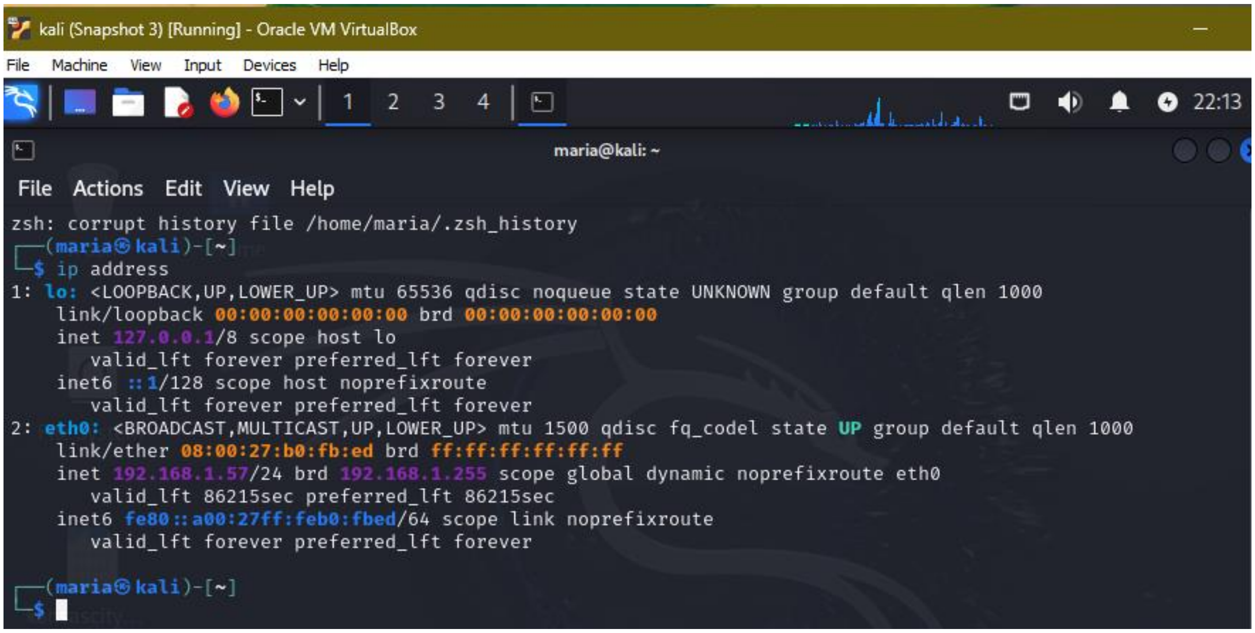


☑ Enable Network Adapter

Attached to: Bridged Adapter

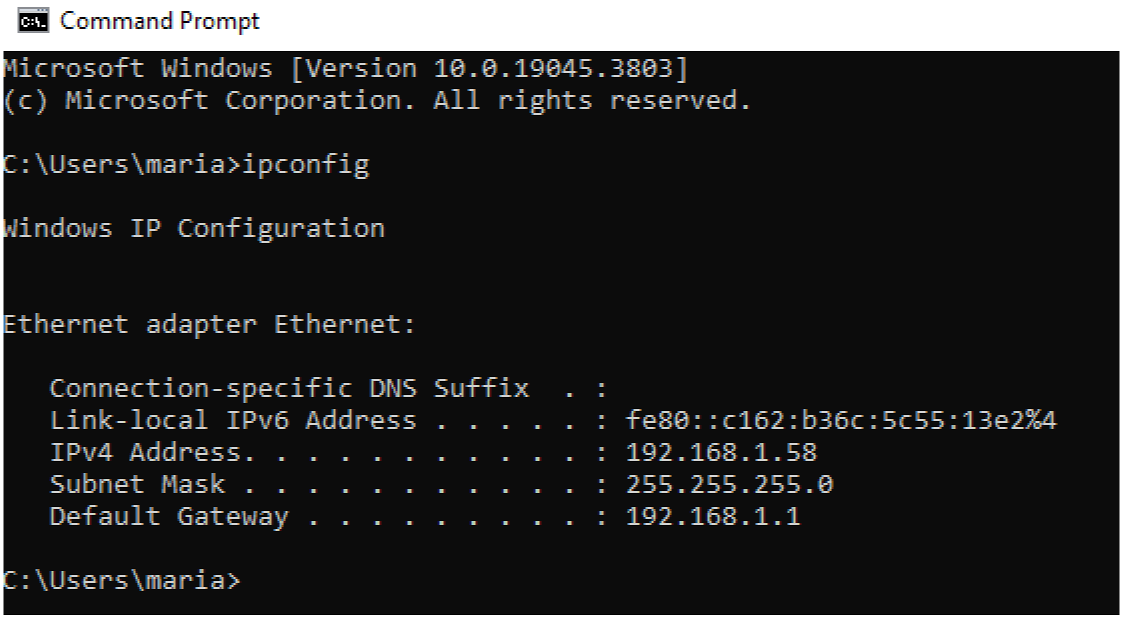
Name: Realtek PCIe GbE Family Controller

On my Kali VM, I launched a terminal and ran the “ip” command to identify the IP address


kali (Snapshot 3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
22:13
maria@kali: ~
File Actions Edit View Help

```
zsh: corrupt history file /home/maria/.zsh_history
┌──(maria㉿kali)-[~]
└─$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b0:fb:ed brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.57/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 86215sec preferred_lft 86215sec
    inet6 fe80::a00:27ff:feb0:fbed/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(maria㉿kali)-[~]
└─$ 
```

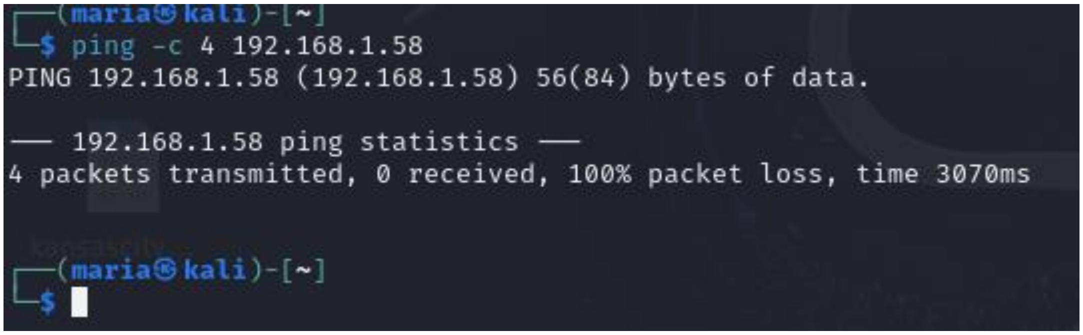
On my Windows VM, I launched the command “ipconfig” and identified the IP address.



Command Prompt

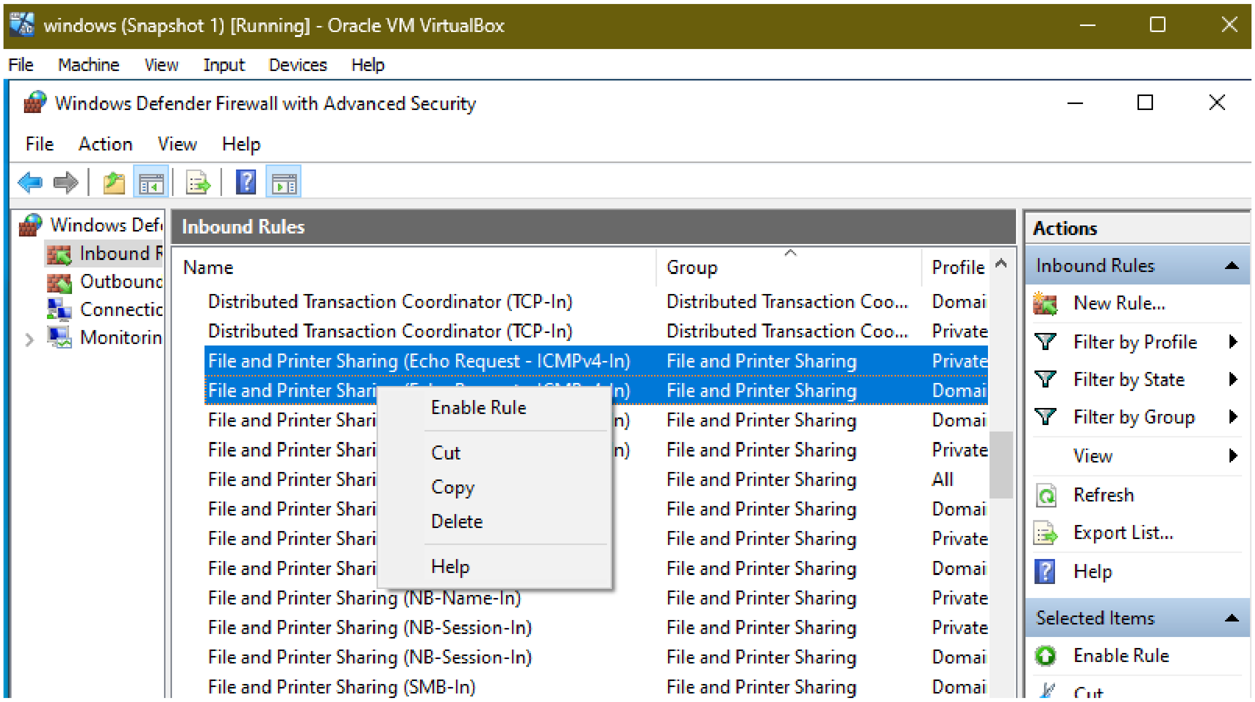
```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\maria>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::c162:b36c:5c55:13e2%4
   IPv4 Address. . . . . . . . . . . : 192.168.1.58
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1

C:\Users\maria>
```

I demonstrated that the Kali and Windows VM can connect with each other by using a connectivity test using the ping utility that sends a message over the ICMP protocol. From the Kali VM I ping with a count (-c) of four packets targeting the windows IP address. I see that there is 100% packet loss.



```
┌──(maria㉿kali)-[~]
└─$ ping -c 4 192.168.1.58
PING 192.168.1.58 (192.168.1.58) 56(84) bytes of data.

--- 192.168.1.58 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3070ms


┌──(maria㉿kali)-[~]
└─$ 
```

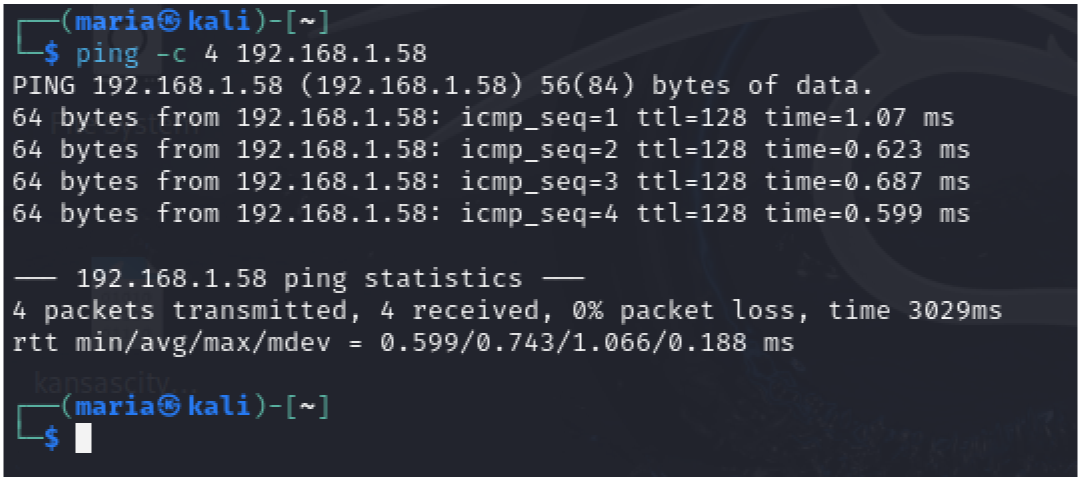
So, I launched the “Windows defender firewall with advanced security” application and selected “Inbound Rules”. And enabled the following in the screenshot.



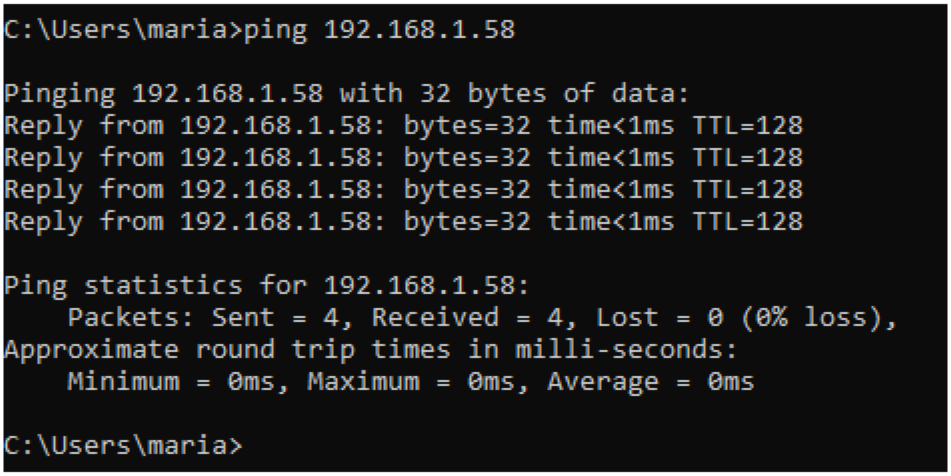
Name	Group	Profile
Distributed Transaction Coordinator (TCP-In)	Distributed Transaction Coo...	Domai
Distributed Transaction Coordinator (TCP-In)	Distributed Transaction Coo...	Private
File and Printer Sharing (Echo Request - ICMPv4-In)	File and Printer Sharing	Private
File and Printer Shari n)	File and Printer Sharing	Domai
File and Printer Shari n)	File and Printer Sharing	Domai
File and Printer Shari n)	File and Printer Sharing	Private
File and Printer Shari	File and Printer Sharing	All
File and Printer Shari	File and Printer Sharing	Domai
File and Printer Shari	File and Printer Sharing	Private
File and Printer Shari	File and Printer Sharing	Domai
File and Printer Sharing (NB-Name-In)	File and Printer Sharing	Private
File and Printer Sharing (NB-Session-In)	File and Printer Sharing	Private
File and Printer Sharing (NB-Session-In)	File and Printer Sharing	Domai
File and Printer Sharing (SMB-In)	File and Printer Sharing	Domai

I pinged the VMs to test the connectivity using the ICMP protocol to validate packets.

For windows I used the command “ping -c 4 <ip address> and for Kali VM i used the “ping ip address” command. This time there was no packet loss.

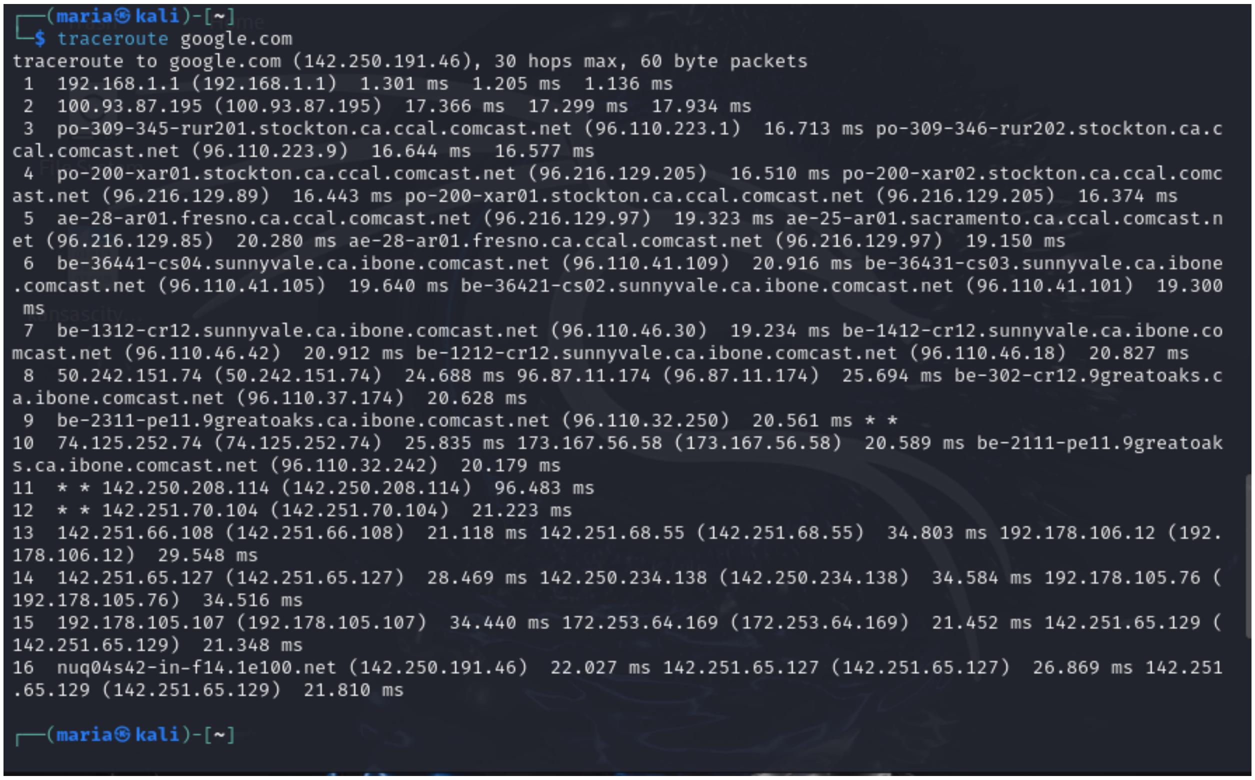


```
┌──(maria㉿kali)-[~]
└─$ ping -c 4 192.168.1.58
PING 192.168.1.58 (192.168.1.58) 56(84) bytes of data.
64 bytes from 192.168.1.58: icmp_seq=1 ttl=128 time=1.07 ms
64 bytes from 192.168.1.58: icmp_seq=2 ttl=128 time=0.623 ms
64 bytes from 192.168.1.58: icmp_seq=3 ttl=128 time=0.687 ms
64 bytes from 192.168.1.58: icmp_seq=4 ttl=128 time=0.599 ms

--- 192.168.1.58 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3029ms
rtt min/avg/max/mdev = 0.599/0.743/1.066/0.188 ms

┌──(maria㉿kali)-[~]
└─$ █
```



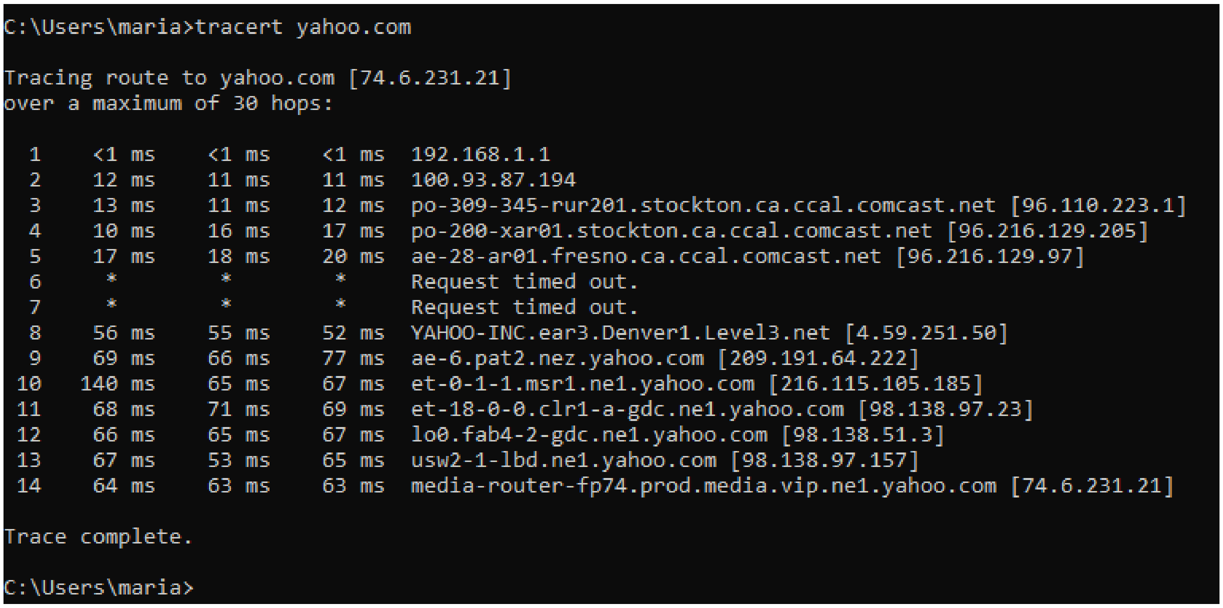
```
C:\Users\maria>ping 192.168.1.58

Pinging 192.168.1.58 with 32 bytes of data:
Reply from 192.168.1.58: bytes=32 time<1ms TTL=128
Reply from 192.168.1.58: bytes=32 time<1ms TTL=128
Reply from 192.168.1.58: bytes=32 time<1ms TTL=128
Reply from 192.168.1.58: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.58:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\maria>
```

From the Kali VM, I traced the route to Google’s webservers using the “traceroute google.com” command.



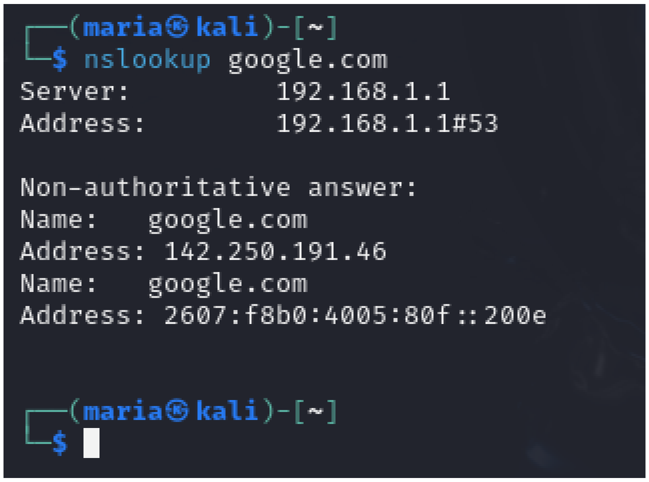
```
┌──(maria㉿kali)-[~]
└─$ traceroute google.com
traceroute to google.com (142.250.191.46), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  1.301 ms  1.205 ms  1.136 ms
 2  100.93.87.195 (100.93.87.195)  17.366 ms  17.299 ms  17.934 ms
 3  po-309-345-rur201.stockton.ca.ccal.comcast.net (96.110.223.1)  16.713 ms po-309-346-rur202.stockton.ca.c
cal.comcast.net (96.110.223.9)  16.644 ms  16.577 ms
 4  po-200-xar01.stockton.ca.ccal.comcast.net (96.216.129.205)  16.510 ms po-200-xar02.stockton.ca.ccal.comc
ast.net (96.216.129.89)  16.443 ms po-200-xar01.stockton.ca.ccal.comcast.net (96.216.129.205)  16.374 ms
 5  ae-28-ar01.fresno.ca.ccal.comcast.net (96.216.129.97)  19.323 ms ae-25-ar01.sacramento.ca.ccal.comcast.n
et (96.216.129.85)  20.280 ms ae-28-ar01.fresno.ca.ccal.comcast.net (96.216.129.97)  19.150 ms
 6  be-36441-cs04.sunnyvale.ca.ibone.comcast.net (96.110.41.109)  20.916 ms be-36431-cs03.sunnyvale.ca.ibone
.comcast.net (96.110.41.105)  19.640 ms be-36421-cs02.sunnyvale.ca.ibone.comcast.net (96.110.41.101)  19.300
 ms
 7  be-1312-cr12.sunnyvale.ca.ibone.comcast.net (96.110.46.30)  19.234 ms be-1412-cr12.sunnyvale.ca.ibone.co
mcast.net (96.110.46.42)  20.912 ms be-1212-cr12.sunnyvale.ca.ibone.comcast.net (96.110.46.18)  20.827 ms
 8  50.242.151.74 (50.242.151.74)  24.688 ms 96.87.11.174 (96.87.11.174)  25.694 ms be-302-cr12.9greatoaks.c
a.ibone.comcast.net (96.110.37.174)  20.628 ms
 9  be-2311-pe11.9greatoaks.ca.ibone.comcast.net (96.110.32.250)  20.561 ms * *
10  74.125.252.74 (74.125.252.74)  25.835 ms 173.167.56.58 (173.167.56.58)  20.589 ms be-2111-pe11.9greatoak
s.ca.ibone.comcast.net (96.110.32.242)  20.179 ms
11  * * 142.250.208.114 (142.250.208.114)  96.483 ms
12  * * 142.251.70.104 (142.251.70.104)  21.223 ms
13  142.251.66.108 (142.251.66.108)  21.118 ms 142.251.68.55 (142.251.68.55)  34.803 ms 192.178.106.12 (192.
178.106.12)  29.548 ms
14  142.251.65.127 (142.251.65.127)  28.469 ms 142.250.234.138 (142.250.234.138)  34.584 ms 192.178.105.76 (
192.178.105.76)  34.516 ms
15  192.178.105.107 (192.178.105.107)  34.440 ms 172.253.64.169 (172.253.64.169)  21.452 ms 142.251.65.129 (
142.251.65.129)  21.348 ms
16  nuq04s42-in-f14.1e100.net (142.250.191.46)  22.027 ms 142.251.65.127 (142.251.65.127)  26.869 ms 142.251
.65.129 (142.251.65.129)  21.810 ms

┌──(maria㉿kali)-[~]
```

From the Windows VM, I traced the route to Yahoo’s server using the “tracert yahoo.com” command.

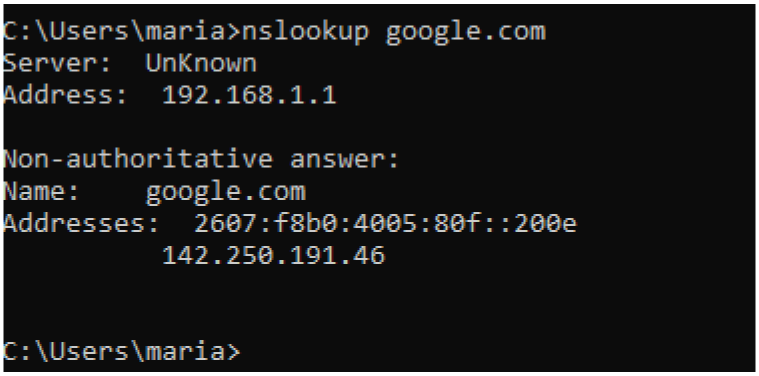


```
C:\Users\maria>tracert yahoo.com

Tracing route to yahoo.com [74.6.231.21]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  192.168.1.1
  2    12 ms    11 ms    11 ms  100.93.87.194
  3    13 ms    11 ms    12 ms  po-309-345-rur201.stockton.ca.ccal.comcast.net [96.110.223.1]
  4    10 ms    16 ms    17 ms  po-200-xar01.stockton.ca.ccal.comcast.net [96.216.129.205]
  5    17 ms    18 ms    20 ms  ae-28-ar01.fresno.ca.ccal.comcast.net [96.216.129.97]
  6     *        *        *     Request timed out.
  7     *        *        *     Request timed out.
  8    56 ms    55 ms    52 ms  YAHOO-INC.ear3.Denver1.Level3.net [4.59.251.50]
  9    69 ms    66 ms    77 ms  ae-6.pat2.nez.yahoo.com [209.191.64.222]
 10   140 ms    65 ms    67 ms  et-0-1-1.msr1.ne1.yahoo.com [216.115.105.185]
 11    68 ms    71 ms    69 ms  et-18-0-0.clr1-a-gdc.ne1.yahoo.com [98.138.97.23]
 12    66 ms    65 ms    67 ms  lo0.fab4-2-gdc.ne1.yahoo.com [98.138.51.3]
 13    67 ms    53 ms    65 ms  usw2-1-lbd.ne1.yahoo.com [98.138.97.157]
 14    64 ms    63 ms    63 ms  media-router-fp74.prod.media.vip.ne1.yahoo.com [74.6.231.21]

Trace complete.

C:\Users\maria>
```

From both the Kali and Windows VM, I looked up Google's IP address using the “nslokup google.com” command.

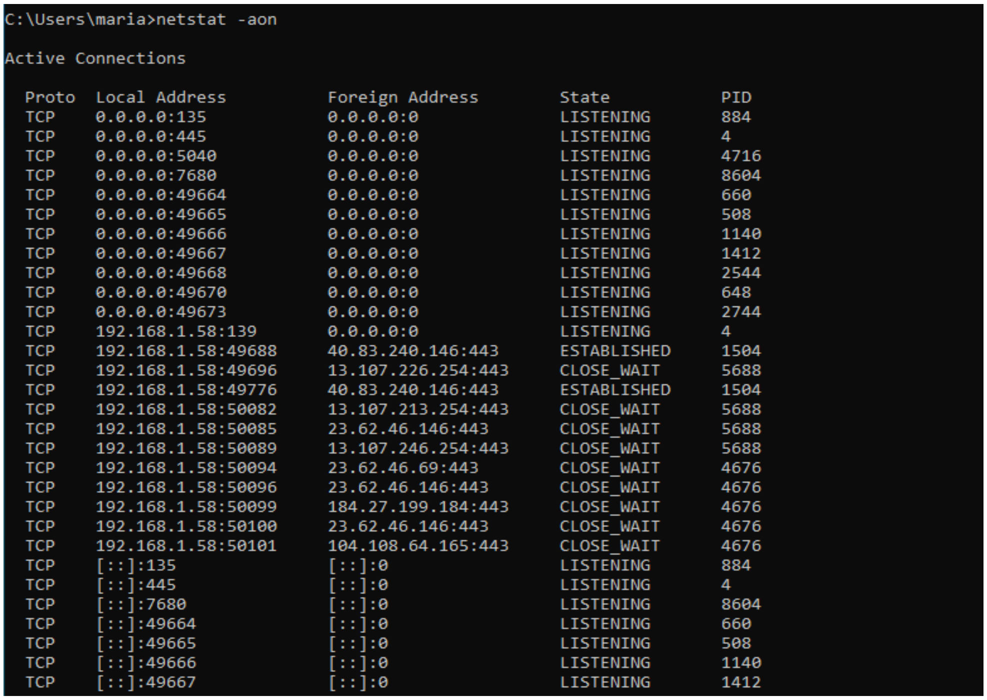


```
┌──(maria㉿kali)-[~]
└─$ nslookup google.com
Server:         192.168.1.1
Address:        192.168.1.1#53

Non-authoritative answer:
Name:   google.com
Address: 142.250.191.46
Name:   google.com
Address: 2607:f8b0:4005:80f::200e


┌──(maria㉿kali)-[~]
└─$ 
```

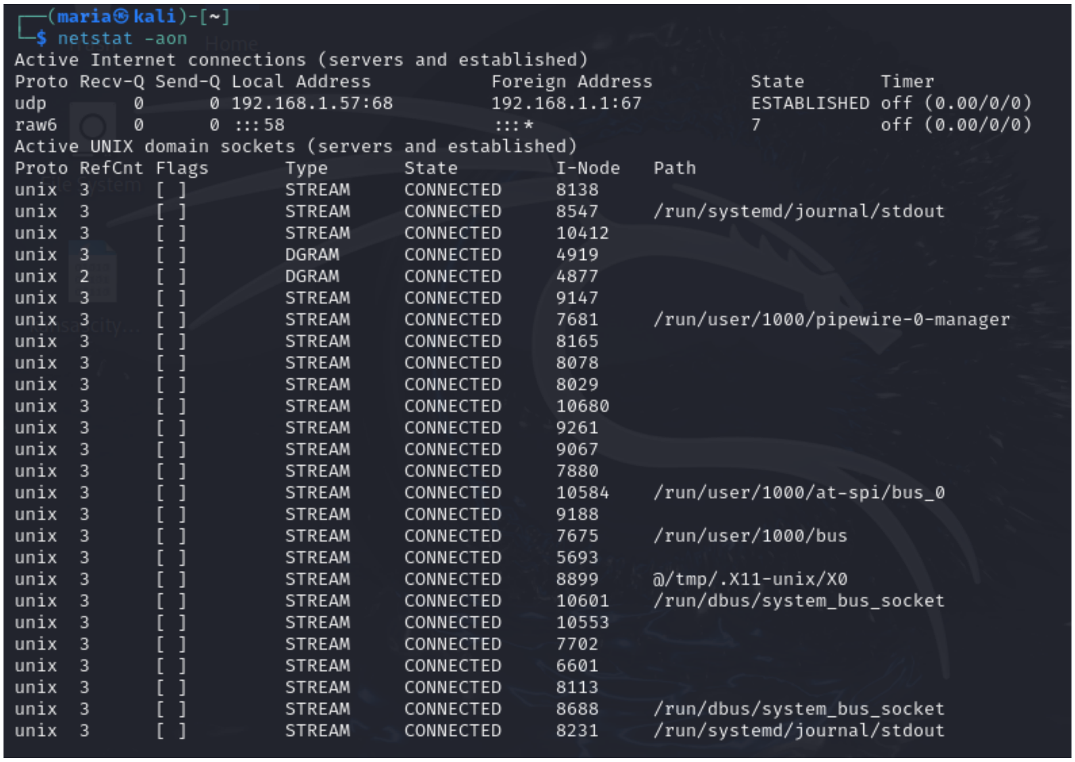


```
C:\Users\maria>nslookup google.com
Server:  UnKnown
Address:  192.168.1.1

Non-authoritative answer:
Name:    google.com
Addresses:  2607:f8b0:4005:80f::200e
          142.250.191.46


C:\Users\maria>
```

Here, I discovered which ports were open, services listening, and network connections made using the netstat command with the netstat –aon options on both VMs. I identitied windows port 4 which carries a video signal using the DisplayPort protocol, charge connected devices and allow for data transfers at speeds beyond what simple USB can manage.



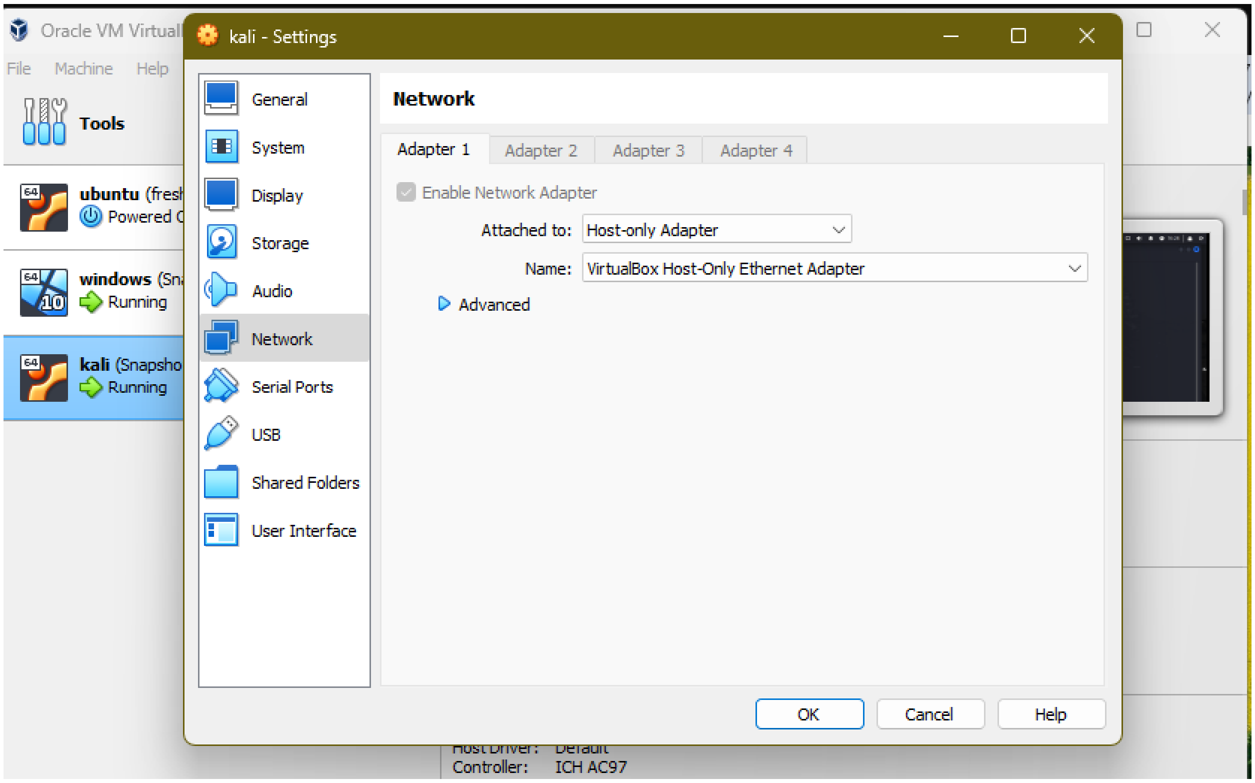
```
C:\Users\maria>netstat -aon

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING       884
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING       4
  TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING       4716
  TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING       8604
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING       660
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING       508
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING       1140
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING       1412
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING       2544
  TCP    0.0.0.0:49670          0.0.0.0:0              LISTENING       648
  TCP    0.0.0.0:49673          0.0.0.0:0              LISTENING       2744
  TCP    192.168.1.58:139       0.0.0.0:0              LISTENING       4
  TCP    192.168.1.58:49688     40.83.240.146:443      ESTABLISHED     1504
  TCP    192.168.1.58:49696     13.107.226.254:443     CLOSE_WAIT      5688
  TCP    192.168.1.58:49776     40.83.240.146:443      ESTABLISHED     1504
  TCP    192.168.1.58:50082     13.107.213.254:443     CLOSE_WAIT      5688
  TCP    192.168.1.58:50085     23.62.46.146:443       CLOSE_WAIT      5688
  TCP    192.168.1.58:50089     13.107.246.254:443     CLOSE_WAIT      5688
  TCP    192.168.1.58:50094     23.62.46.69:443        CLOSE_WAIT      4676
  TCP    192.168.1.58:50096     23.62.46.146:443       CLOSE_WAIT      4676
  TCP    192.168.1.58:50099     184.27.199.184:443     CLOSE_WAIT      4676
  TCP    192.168.1.58:50100     23.62.46.146:443       CLOSE_WAIT      4676
  TCP    192.168.1.58:50101     104.108.64.165:443     CLOSE_WAIT      4676
  TCP    [::]:135               [::]:0                 LISTENING       884
  TCP    [::]:445               [::]:0                 LISTENING       4
  TCP    [::]:7680              [::]:0                 LISTENING       8604
  TCP    [::]:49664             [::]:0                 LISTENING       660
  TCP    [::]:49665             [::]:0                 LISTENING       508
  TCP    [::]:49666             [::]:0                 LISTENING       1140
  TCP    [::]:49667             [::]:0                 LISTENING       1412
```



```
┌──(maria㉿kali)-[~]
└─$ netstat -aon
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       Timer
udp        0      0 192.168.1.57:68         192.168.1.1:67          ESTABLISHED off (0.00/0/0)
raw6       0      0 :::58                   :::*                    7           off (0.00/0/0)
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  3      [ ]         STREAM     CONNECTED     8138
unix  3      [ ]         STREAM     CONNECTED     8547     /run/systemd/journal/stdout
unix  3      [ ]         STREAM     CONNECTED     10412
unix  3      [ ]         DGRAM      CONNECTED     4919
unix  2      [ ]         DGRAM      CONNECTED     4877
unix  3      [ ]         STREAM     CONNECTED     9147
unix  3      [ ]         STREAM     CONNECTED     7681     /run/user/1000/pipewire-0-manager
unix  3      [ ]         STREAM     CONNECTED     8165
unix  3      [ ]         STREAM     CONNECTED     8078
unix  3      [ ]         STREAM     CONNECTED     8029
unix  3      [ ]         STREAM     CONNECTED     10680
unix  3      [ ]         STREAM     CONNECTED     9261
unix  3      [ ]         STREAM     CONNECTED     9067
unix  3      [ ]         STREAM     CONNECTED     7880
unix  3      [ ]         STREAM     CONNECTED     10584    /run/user/1000/at-spi/bus_0
unix  3      [ ]         STREAM     CONNECTED     9188
unix  3      [ ]         STREAM     CONNECTED     7675     /run/user/1000/bus
unix  3      [ ]         STREAM     CONNECTED     5693
unix  3      [ ]         STREAM     CONNECTED     8899     @/tmp/.X11-unix/X0
unix  3      [ ]         STREAM     CONNECTED     10601    /run/dbus/system_bus_socket
unix  3      [ ]         STREAM     CONNECTED     10553
unix  3      [ ]         STREAM     CONNECTED     7702
unix  3      [ ]         STREAM     CONNECTED     6601
unix  3      [ ]         STREAM     CONNECTED     8113
unix  3      [ ]         STREAM     CONNECTED     8688     /run/dbus/system_bus_socket
unix  3      [ ]         STREAM     CONNECTED     8231     /run/systemd/journal/stdout
```

3.3 Host and Service Discovery

In this exercise, I set both my VMs to "host-only adapter".



kali - Settings

Network

☑ Enable Network Adapter

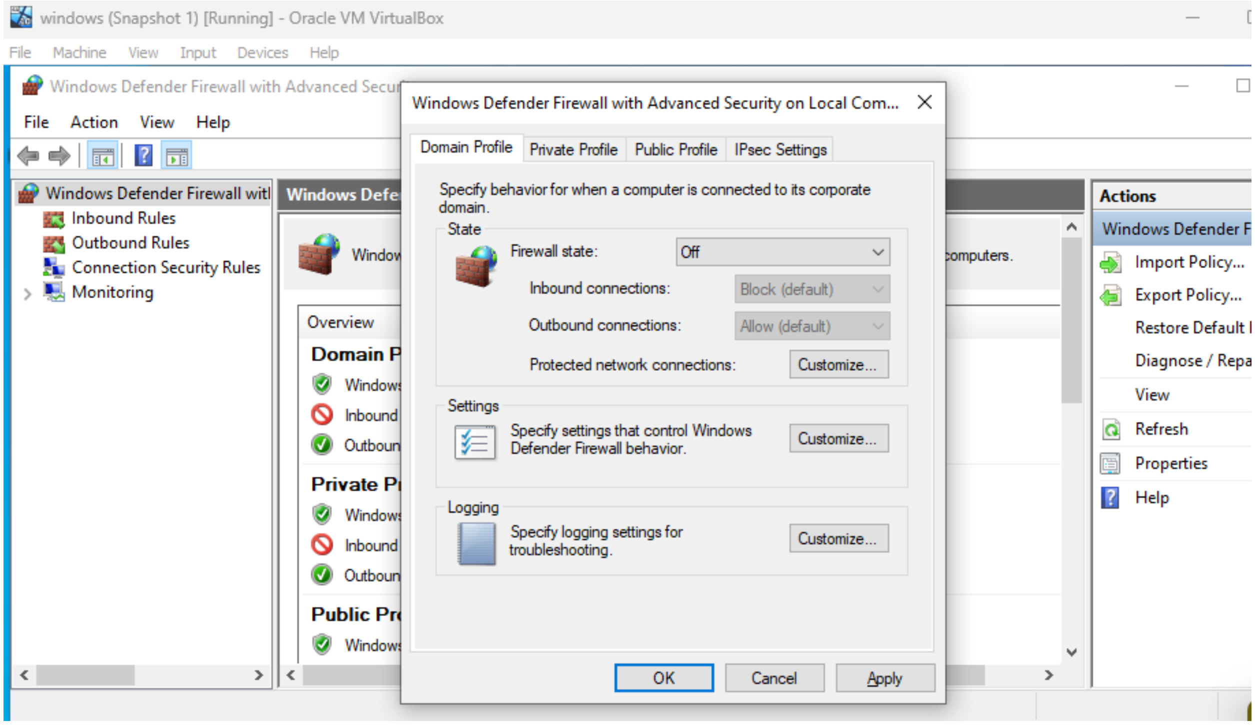
Attached to: Host-only Adapter

Name: VirtualBox Host-Only Ethernet Adapter

Host Driver: Default

Controller: ICH AC97

In the Windows VM, I opened the "Windows Defender Firewall with Advanced Security", selected the properties and set the firewall state to off for each profile tab.



Windows Defender Firewall with Advanced Security on Local Com...

Domain Profile | Private Profile | Public Profile | IPsec Settings

Specify behavior for when a computer is connected to its corporate domain.

State

Firewall state: Off

Inbound connections: Block (default)

Outbound connections: Allow (default)

Protected network connections: Customize...

Settings

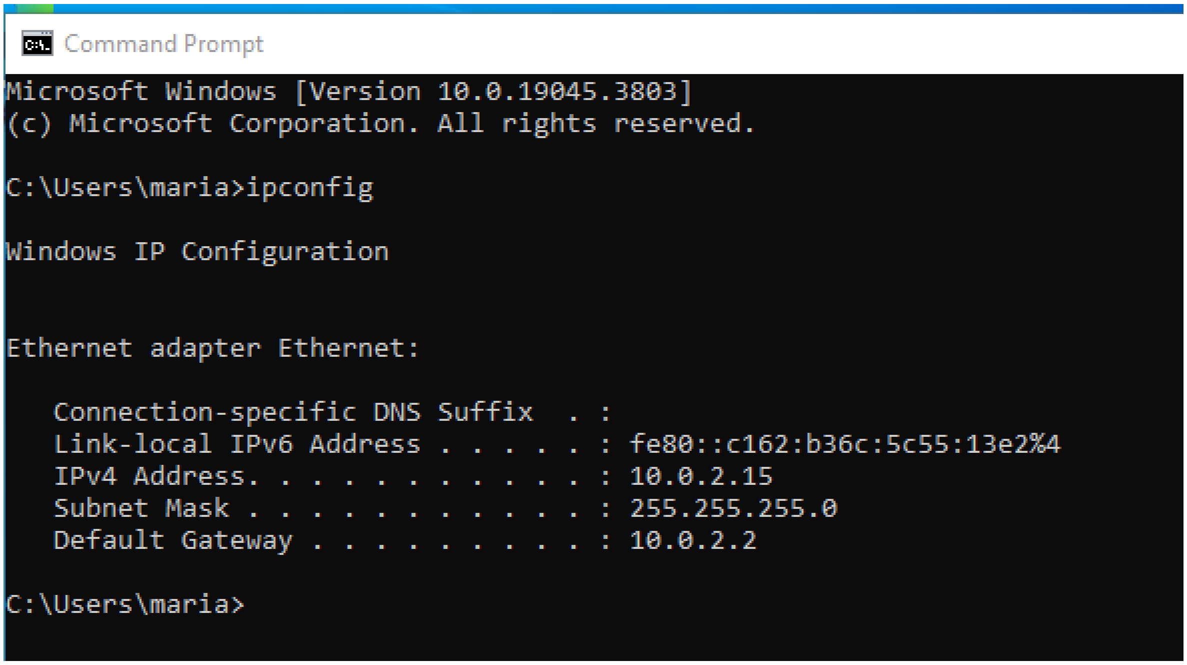
Specify settings that control Windows Defender Firewall behavior. Customize...

Logging

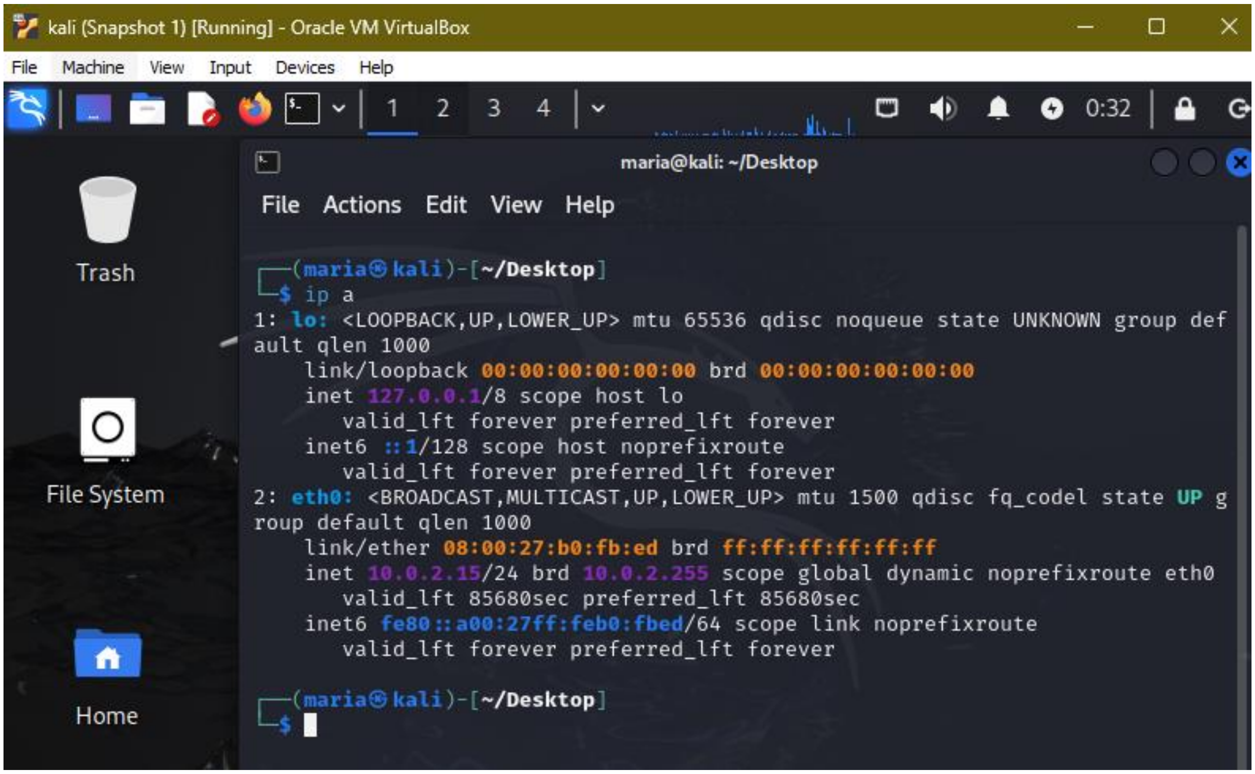
Specify logging settings for troubleshooting. Customize...

OK Cancel Apply

I checked the IP addresses of the Kali and Windows VM for reference and ensured each had a unique IP address in the subnet of each. On the Windows VM, I opened a command prompt and inserted “ipconfig” command and withing the Kali terminal I inserted the “ip a” command.

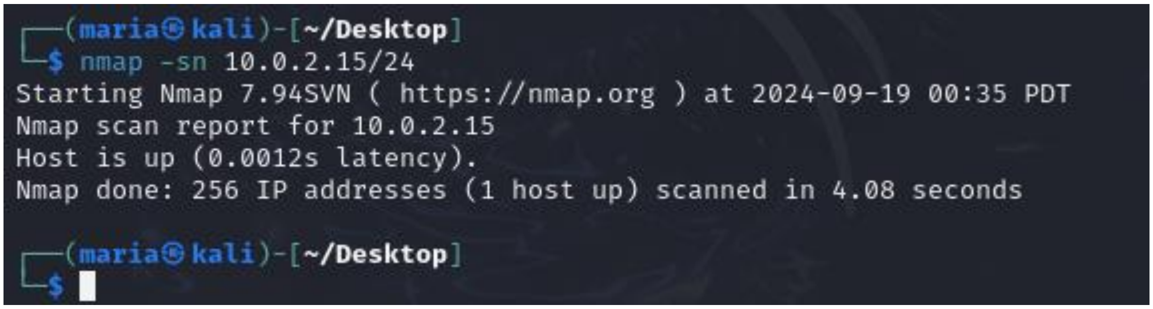


```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\maria>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::c162:b36c:5c55:13e2%4
   IPv4 Address. . . . . . . . . . . : 10.0.2.15
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.0.2.2

C:\Users\maria>
```


kali (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
0:32
Trash
File System
Home
maria@kali: ~/Desktop
File Actions Edit View Help

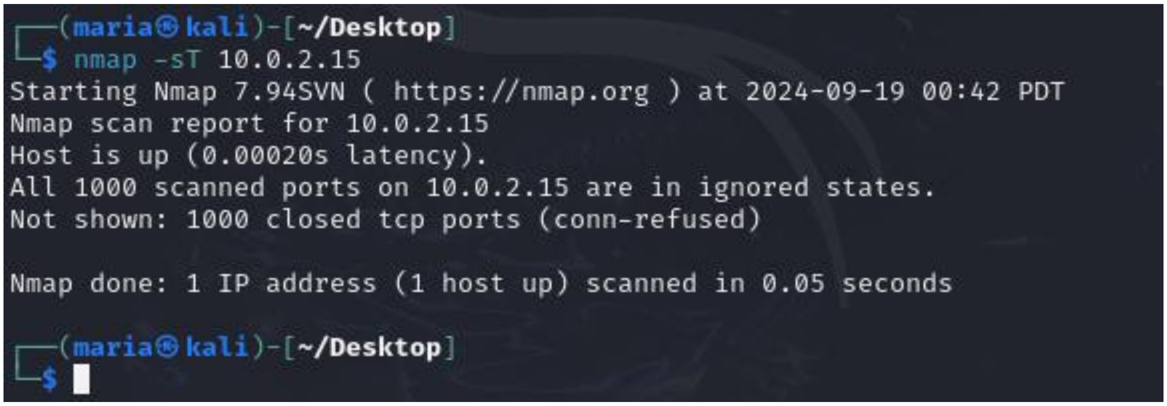
```
┌──(maria㉿kali)-[~/Desktop]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
roup default qlen 1000
    link/ether 08:00:27:b0:fb:ed brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 85680sec preferred_lft 85680sec
    inet6 fe80::a00:27ff:feb0:fbed/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(maria㉿kali)-[~/Desktop]
└─$ 
```

I discovered the Windows VM from the Kali VM using NMAP ping sweep. From the Kali terminal, I ran the command “nmap -sn”

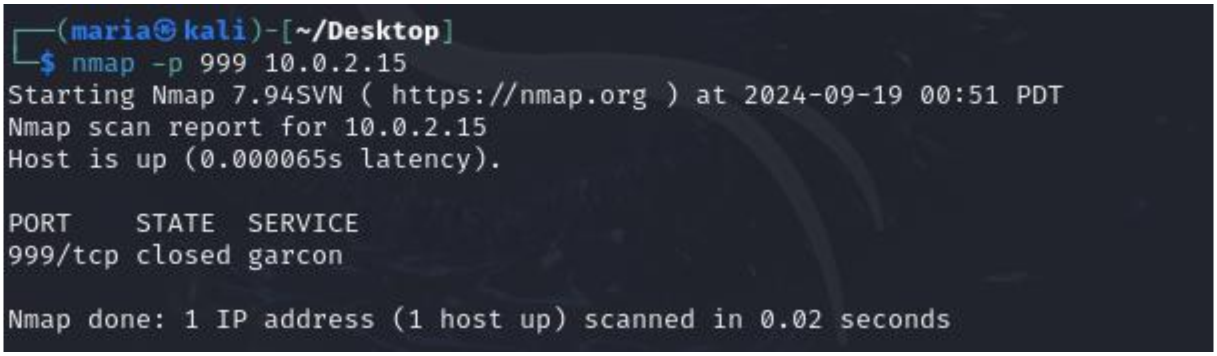


```
┌──(maria㉿kali)-[~/Desktop]
└─$ nmap -sn 10.0.2.15/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-19 00:35 PDT
Nmap scan report for 10.0.2.15
Host is up (0.0012s latency).
Nmap done: 256 IP addresses (1 host up) scanned in 4.08 seconds

┌──(maria㉿kali)-[~/Desktop]
└─$ 
```

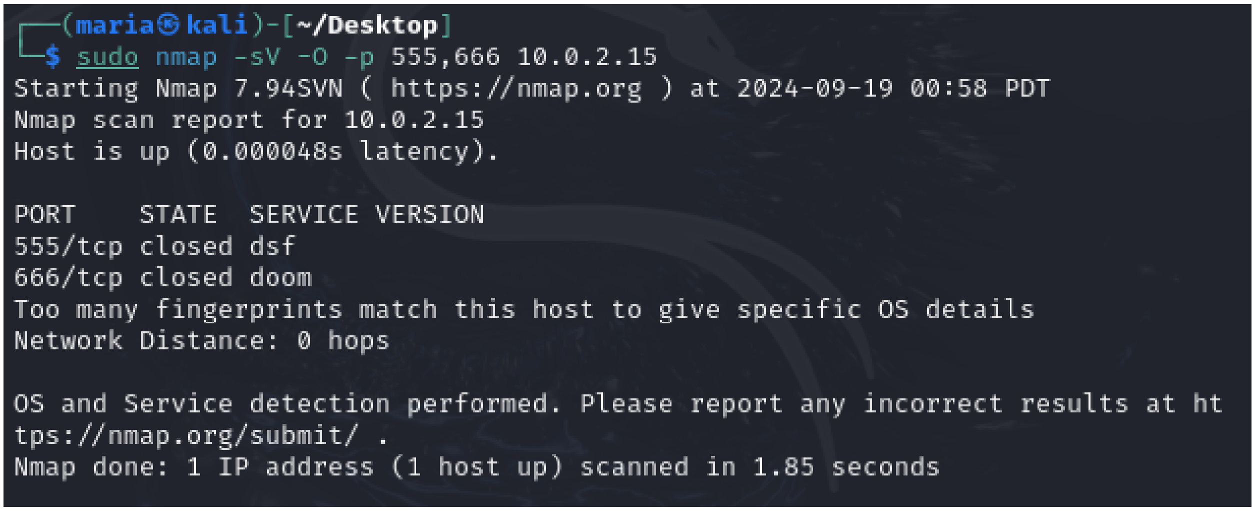
I scanned the open ports and services of the IP address (Windows) discovered during the Ping Sweep. I did this by running the “nmap -sT –sV –p-” command in the Kali terminal.



```
┌──(maria㉿kali)-[~/Desktop]
└─$ nmap -sT 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-19 00:42 PDT
Nmap scan report for 10.0.2.15
Host is up (0.00020s latency).
All 1000 scanned ports on 10.0.2.15 are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap done: 1 IP address (1 host up) scanned in 0.05 seconds

┌──(maria㉿kali)-[~/Desktop]
└─$ 
```



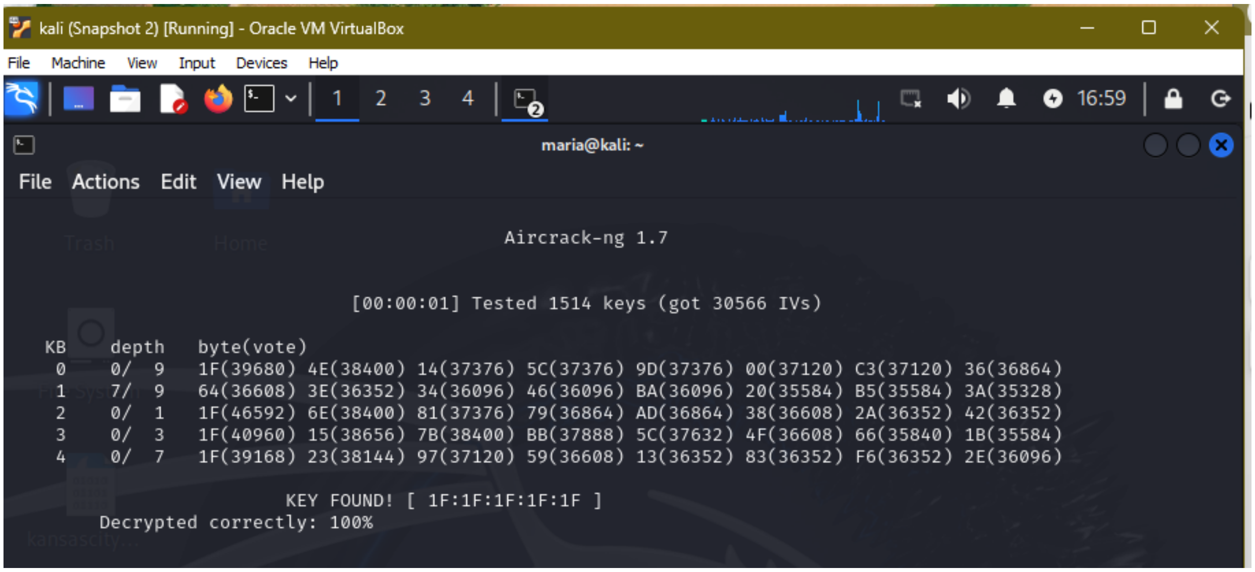
```
┌──(maria㉿kali)-[~/Desktop]
└─$ nmap -p 999 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-19 00:51 PDT
Nmap scan report for 10.0.2.15
Host is up (0.000065s latency).

PORT    STATE  SERVICE
999/tcp closed garcon

Nmap done: 1 IP address (1 host up) scanned in 0.02 seconds
```



```
┌──(maria㉿kali)-[~/Desktop]
└─$ sudo nmap -sV -O -p 555,666 10.0.2.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-19 00:58 PDT
Nmap scan report for 10.0.2.15
Host is up (0.000048s latency).

PORT    STATE  SERVICE VERSION
555/tcp closed dsf
666/tcp closed doom
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS and Service detection performed. Please report any incorrect results at ht
tps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.85 seconds
```

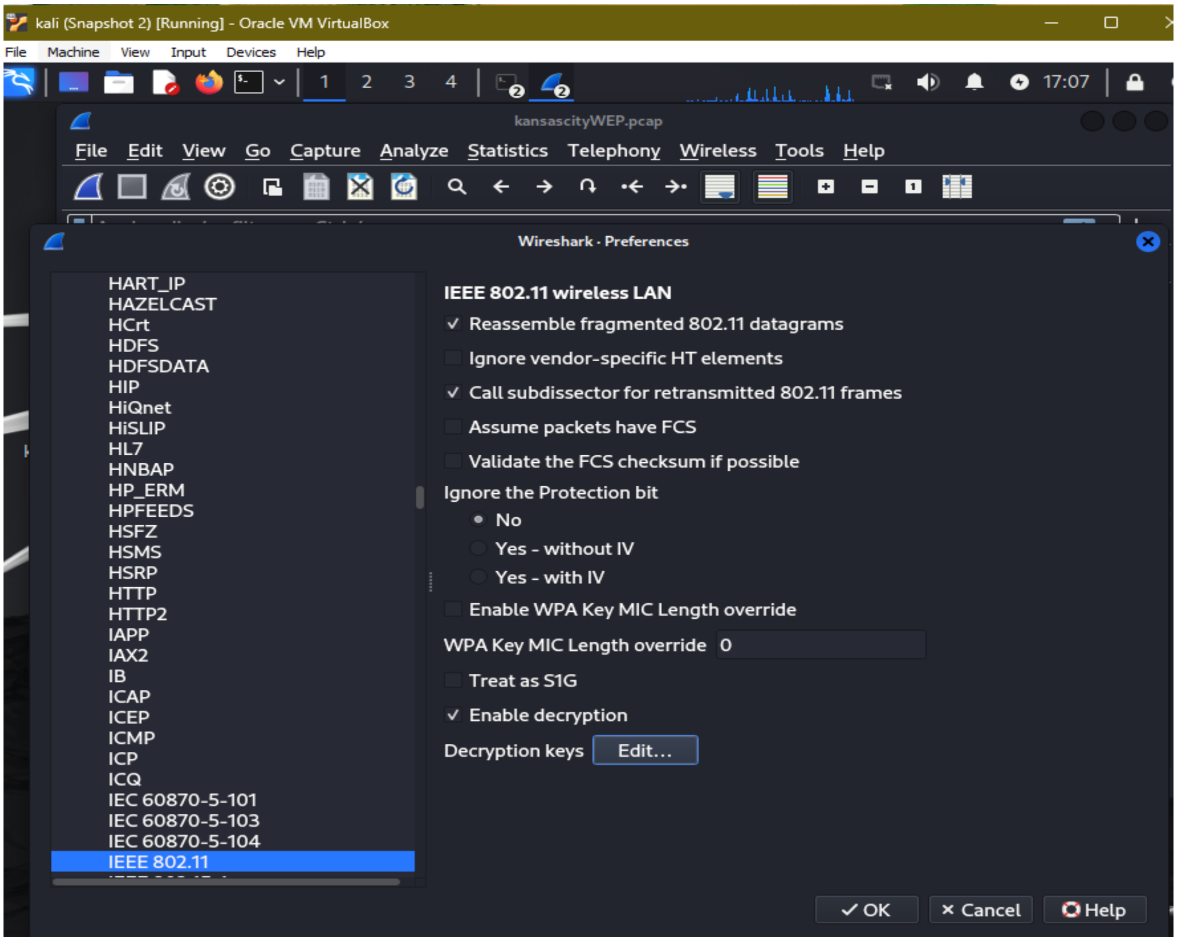
3.4 Wi-Fi WEP Cracking

In this exercise, I downloaded the kansascityWEP.pcap to the desktop of the Kali VM. I launched the terminal and cracked the WEP encryption using air crack-ng and observed the cracked encryption key.


kali (Snapshot 2) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
16:59
maria@kali: ~
File Actions Edit View Help

```
                                 Aircrack-ng 1.7

                     [00:00:01] Tested 1514 keys (got 30566 IVs)

   KB    depth   byte(vote)
    0    0/  9   1F(39680) 4E(38400) 14(37376) 5C(37376) 9D(37376) 00(37120) C3(37120) 36(36864)
    1    7/  9   64(36608) 3E(36352) 34(36096) 46(36096) BA(36096) 20(35584) B5(35584) 3A(35328)
    2    0/  1   1F(46592) 6E(38400) 81(37376) 79(36864) AD(36864) 38(36608) 2A(36352) 42(36352)
    3    0/  3   1F(40960) 15(38656) 7B(38400) BB(37888) 5C(37632) 4F(36608) 66(35840) 1B(35584)
    4    0/  7   1F(39168) 23(38144) 97(37120) 59(36608) 13(36352) 83(36352) F6(36352) 2E(36096)

                        KEY FOUND! [ 1F:1F:1F:1F:1F ]
        Decrypted correctly: 100%
```

After cracking the encryption key, I launched Wireshark. I opened the kansascityWEP.pcap file in Wireshark and enabled the Wireless toolbar. Then I selected enable decryption and added the decryption key in the key field.

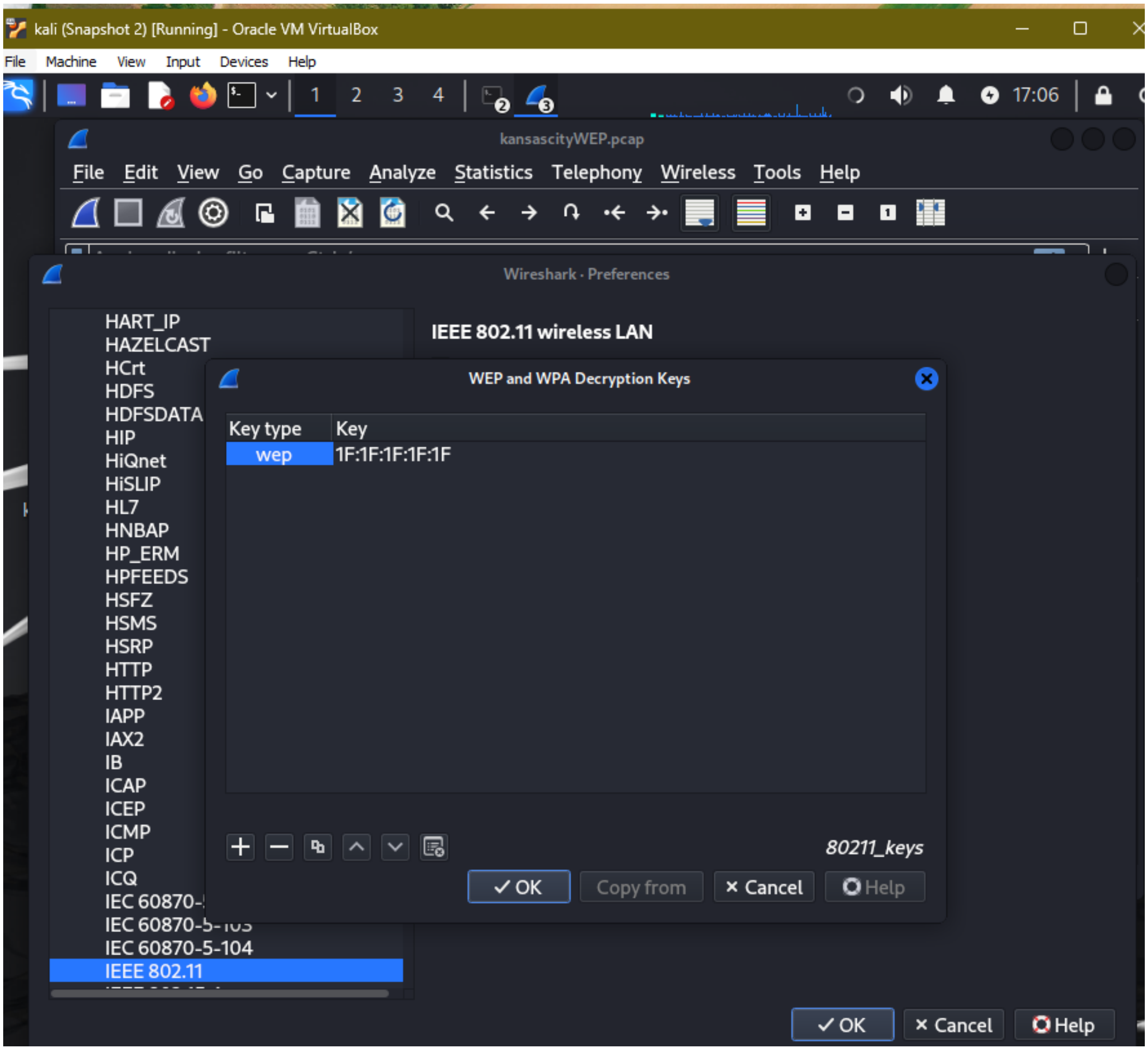


IEEE 802.11 wireless LAN

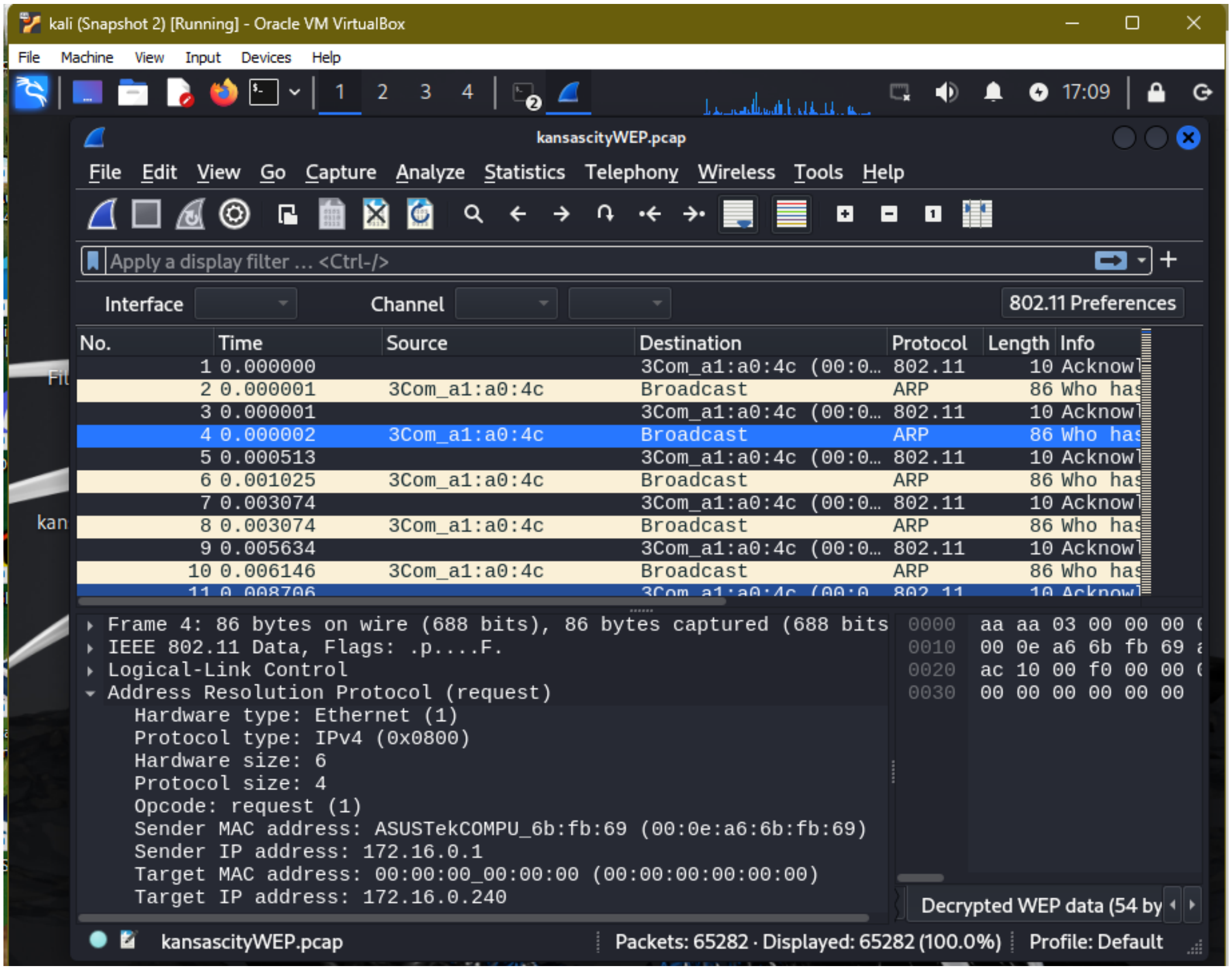
☑ Reassemble fragmented 802.11 datagrams
□ Ignore vendor-specific HT elements
☑ Call subdissector for retransmitted 802.11 frames
□ Assume packets have FCS
□ Validate the FCS checksum if possible

Ignore the Protection bit
● No
○ Yes - without IV
○ Yes - with IV

□ Enable WPA Key MIC Length override
WPA Key MIC Length override 0
□ Treat as S1G
☑ Enable decryption
Decryption keys Edit…



Key type	Key
wep	1F:1F:1F:1F:1F



kali (Snapshot 2) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
17:09
kansascityWEP.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Interface
Channel
802.11 Preferences

```
No.   Time      Source           Destination                Protocol Length Info
    1 0.000000                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
    2 0.000001  3Com_a1:a0:4c    Broadcast                  ARP        86 Who has
    3 0.000001                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
    4 0.000002  3Com_a1:a0:4c    Broadcast                  ARP        86 Who has
    5 0.000513                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
    6 0.001025  3Com_a1:a0:4c    Broadcast                  ARP        86 Who has
    7 0.003074                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
    8 0.003074  3Com_a1:a0:4c    Broadcast                  ARP        86 Who has
    9 0.005634                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
   10 0.006146  3Com_a1:a0:4c    Broadcast                  ARP        86 Who has
   11 0.008706                   3Com_a1:a0:4c (00:0…       802.11     10 Acknowl
```

```
› Frame 4: 86 bytes on wire (688 bits), 86 bytes captured (688 bits
› IEEE 802.11 Data, Flags: .p....F.
› Logical-Link Control
⌄ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: ASUSTekCOMPU_6b:fb:69 (00:0e:a6:6b:fb:69)
    Sender IP address: 172.16.0.1
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 172.16.0.240
```

```
0000  aa aa 03 00 00 0
0010  00 0e a6 6b fb 69 a
0020  ac 10 00 f0 00 00 0
0030  00 00 00 00 00 00
```

Decrypted WEP data (54 by
kansascityWEP.pcap
Packets: 65282 · Displayed: 65282 (100.0%)
Profile: Default