Configuration Manual

MSc Research Project

Cybersecurity

Hari Haran Rajendran

Student ID: X21156077

School of Computing

National College of Ireland

Supervisor: Imran Khan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Hari Haran Rajendran
Student ID: X21156077
Programme: Cybersecurity **Year:** 2022

Module: Internship
Supervisor: Imran Khan
Submission Due Date: 15th December 2022
Project Title: Enhance MITM attack detection with response time in Secure web communication
Word count 890
Page count: 11

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……………………… R. Hari Haran ………………………………………

Date: ……………………………12/15/2022………………………………………….

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Hari Haran Rajendran
X21156077

1 Introduction

This document describes the proper implementation and execution of Enhance MITM attack detection with response time in Secure web communication. The experiment is implemented using Virtualization setup and manually followed the below procedure to collect the sample dataset and done comparative analysis using python code.

2 System requirements

The study is carried out on an Oracle VM Virtual Box virtualization environment with two Kali Linux and Windows 10 VMs and Internet router. VM machines are configured as bridged adapter mode to connect to the physical Network.

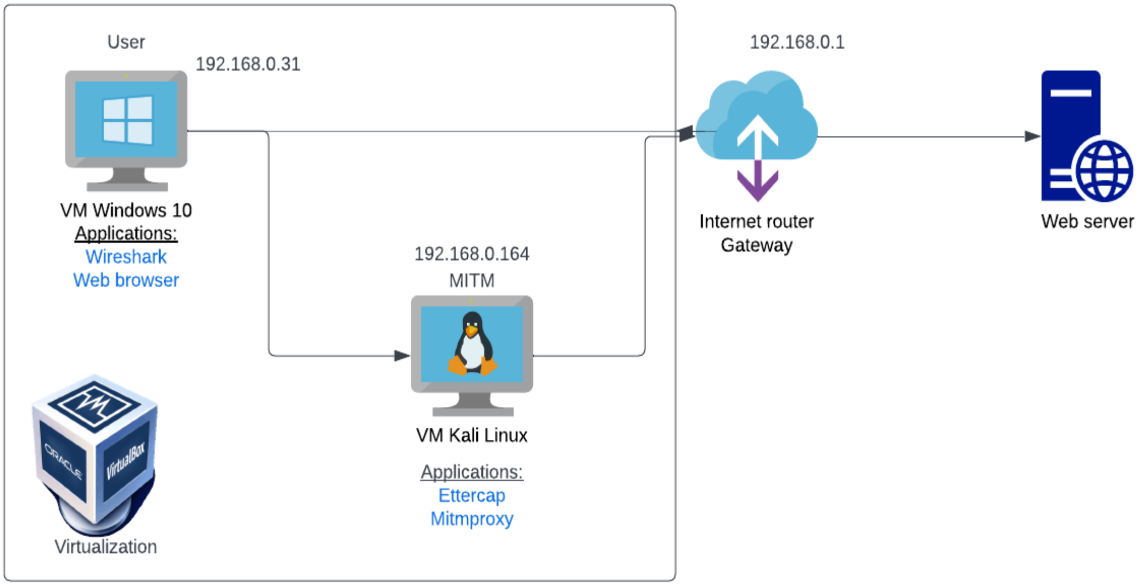
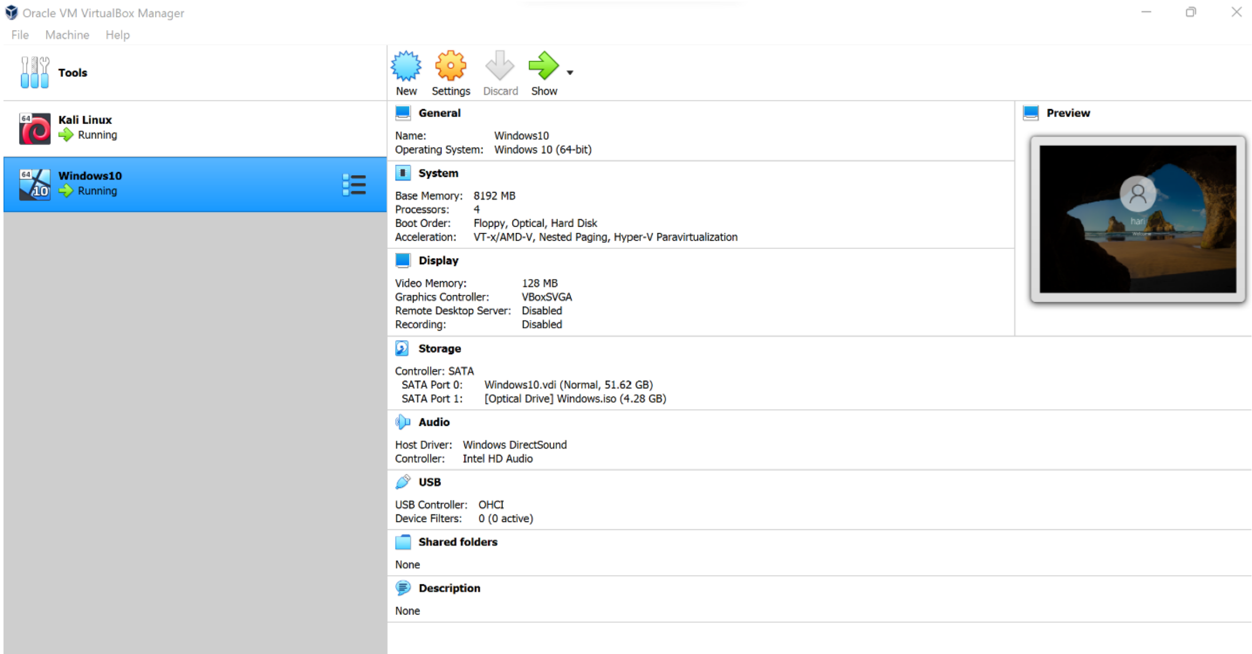


Figure 1: Implementation step



General

Name: Windows10
Operating System: Windows 10 (64-bit)

System

Base Memory: 8192 MB
Processors: 4
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, Hyper-V Paravirtualization

Display

Video Memory: 128 MB
Graphics Controller: VBoxSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: SATA
SATA Port 0: Windows10.vdi (Normal, 51.62 GB)
SATA Port 1: [Optical Drive] Windows.iso (4.28 GB)

Audio

Host Driver: Windows DirectSound
Controller: Intel HD Audio

USB

USB Controller: OHCI
Device Filters: 0 (0 active)

Shared folders

None

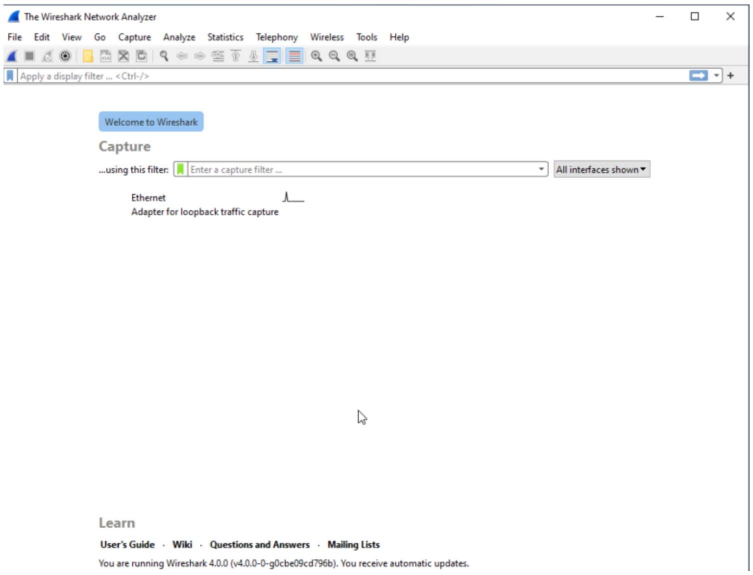
Description

None

Figure 2: Virtualization setup

2.1 Applications used:

Wireshark – Application is installed on Windows 10 to capture packets from network connections.

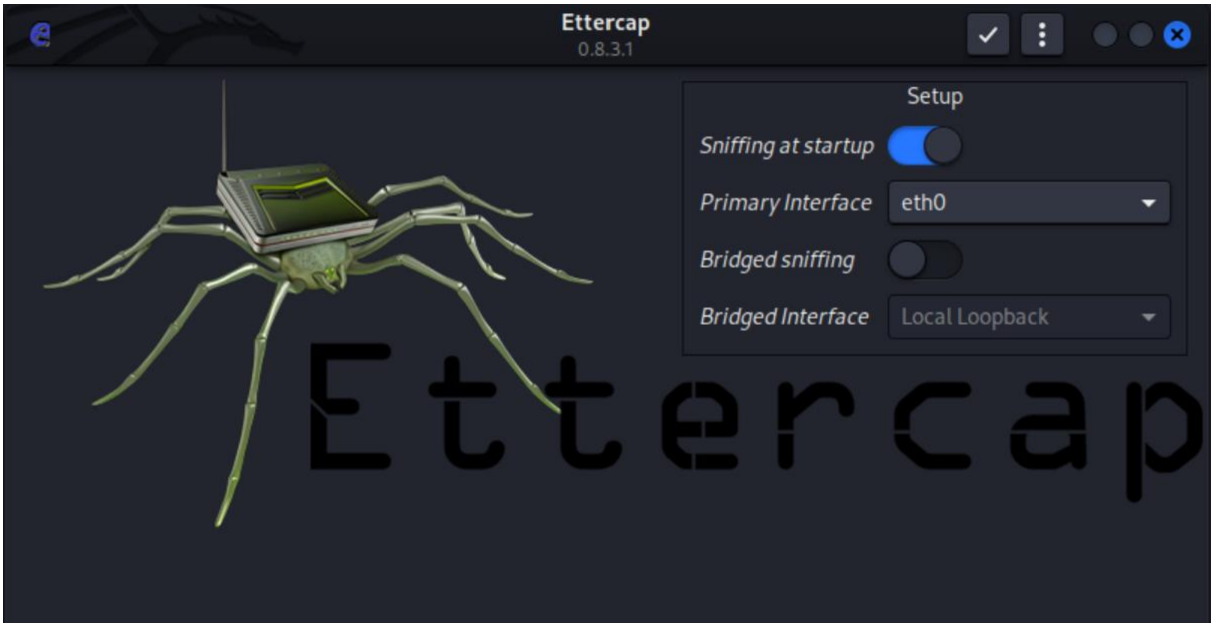

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Welcome to Wireshark
Capture
...using this filter:
Enter a capture filter ...
All interfaces shown
Ethernet
Adapter for loopback traffic capture
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists

You are running Wireshark 4.0.0 (v4.0.0-0-g0cbe09cd796b). You receive automatic updates.

Figure 3: Wireshark Network Analyzer

Web browser – This is to establish the web application connection.

Ettercap – This is enabled on Kali Linux to execute ARP Poisoning Man in the middle attack in the setup.



Setup

Sniffing at startup ●

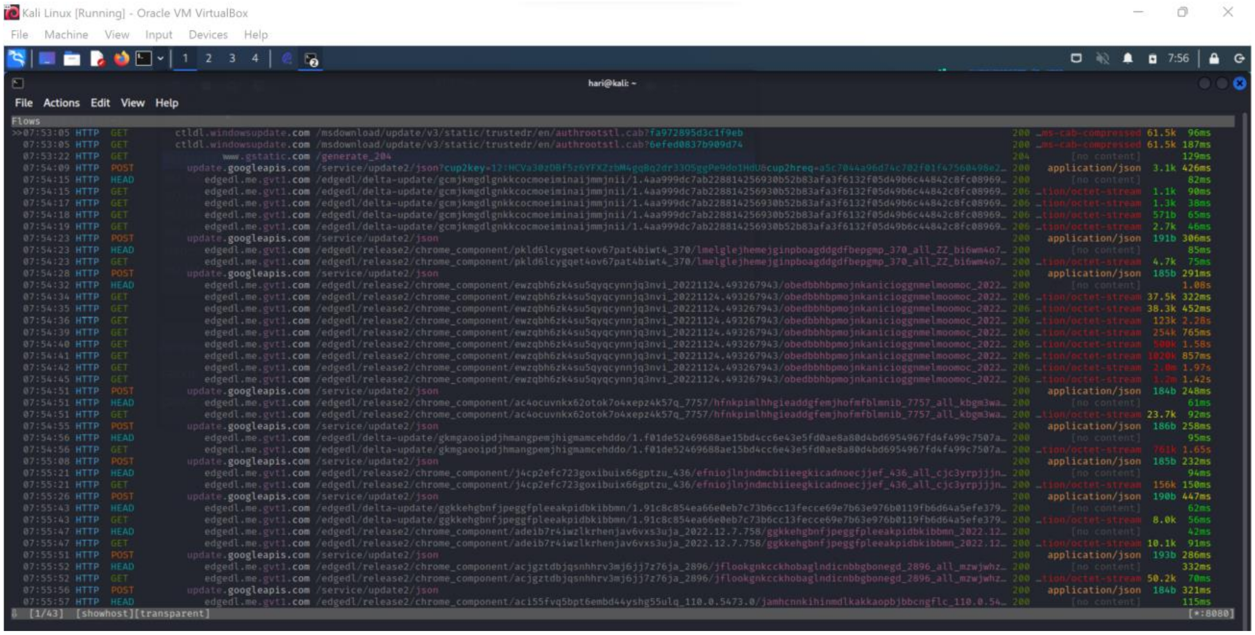
Primary Interface eth0

Bridged sniffing ○

Bridged Interface Local Loopback

Figure 4: Ettercap for ARP Poisoning MITM

Mitmproxy – This is enabled on Kali Linux to act as an interactive HTTPS proxy.


Kali Linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
7:56
hari@kali: ~
File Actions Edit View Help

```
Flows
>>07:53:05 HTTP GET        ctldl.windowsupdate.com /msdownload/update/v3/static/trustedr/en/authrootstl.cab?fa972895d3c1f9eb                                200 …ms-cab-compressed 61.5k  96ms
  07:53:05 HTTP GET        ctldl.windowsupdate.com /msdownload/update/v3/static/trustedr/en/authrootstl.cab?6efed0837b909d74                                200 …ms-cab-compressed 61.5k 187ms
  07:53:22 HTTP GET                 www.gstatic.com /generate_204                                                                                    204       [no content]       129ms
  07:54:09 HTTP POST          update.googleapis.com /service/update2/json?cup2key=12:HCVa30z0Bf5z6YFXZzbM4gqBq2dr3305ggPe9do1HdU6cup2hreq=a5c7044a96d74c702f01f47560498e2… 200  application/json  3.1k 426ms
  07:54:15 HTTP HEAD            edgedl.me.gvt1.com /edgedl/delta-update/gcmjkmgdlgnkkcocmoeiminaijmmjnii/1.4aa999dc7ab228814256930b52b83afa3f6132f05d49b6c44842c8fc08969… 200       [no content]        82ms
  07:54:15 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/gcmjkmgdlgnkkcocmoeiminaijmmjnii/1.4aa999dc7ab228814256930b52b83afa3f6132f05d49b6c44842c8fc08969… 206 …tion/octet-stream  1.1k  90ms
  07:54:17 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/gcmjkmgdlgnkkcocmoeiminaijmmjnii/1.4aa999dc7ab228814256930b52b83afa3f6132f05d49b6c44842c8fc08969… 206 …tion/octet-stream  1.3k  38ms
  07:54:18 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/gcmjkmgdlgnkkcocmoeiminaijmmjnii/1.4aa999dc7ab228814256930b52b83afa3f6132f05d49b6c44842c8fc08969… 206 …tion/octet-stream  571b  65ms
  07:54:19 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/gcmjkmgdlgnkkcocmoeiminaijmmjnii/1.4aa999dc7ab228814256930b52b83afa3f6132f05d49b6c44842c8fc08969… 206 …tion/octet-stream  2.7k  46ms
  07:54:23 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  191b 306ms
  07:54:23 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/pkld6lcyggqet4ov67pat4biwt4_370/lmelglejhemejginpboagddgdfbepgmp_370_all_ZZ_bi6wm4o7… 200       [no content]        85ms
  07:54:23 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/pkld6lcyggqet4ov67pat4biwt4_370/lmelglejhemejginpboagddgdfbepgmp_370_all_ZZ_bi6wm4o7… 200 …tion/octet-stream  4.7k  75ms
  07:54:28 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  185b 291ms
  07:54:32 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 200       [no content]       1.06s
  07:54:34 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream 37.5k 322ms
  07:54:35 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream 38.3k 452ms
  07:54:36 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream  123k 2.28s
  07:54:39 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream  254k 765ms
  07:54:40 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream  500k 1.58s
  07:54:41 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream 1020k 857ms
  07:54:42 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream  2.0m 1.97s
  07:54:45 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ewzqbh6zk4su5qyqcynnjq3nvi_20221124.493267943/obedbbhbpmojnkanicioggnmelmoomoc_2022… 206 …tion/octet-stream  1.2m 1.42s
  07:54:51 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  184b 248ms
  07:54:51 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/ac4ocuvnkx62otok7o4xepz4k57q_7757/hfnkpimlhhgieaddgfemjhofmfblmnib_7757_all_kbgm3wa… 200       [no content]        61ms
  07:54:51 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/ac4ocuvnkx62otok7o4xepz4k57q_7757/hfnkpimlhhgieaddgfemjhofmfblmnib_7757_all_kbgm3wa… 200 …tion/octet-stream 23.7k  92ms
  07:54:55 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  186b 258ms
  07:54:56 HTTP HEAD            edgedl.me.gvt1.com /edgedl/delta-update/gkmgaooipdjhmangpemjhigmamcehddo/1.f01de52469688ae15bd4cc6e43e5fd0ae8a80d4bd6954967fd4f499c7507a… 200       [no content]        95ms
  07:54:56 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/gkmgaooipdjhmangpemjhigmamcehddo/1.f01de52469688ae15bd4cc6e43e5fd0ae8a80d4bd6954967fd4f499c7507a… 200 …tion/octet-stream  761k 1.65s
  07:55:08 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  185b 232ms
  07:55:21 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/j4cp2efc723goxibuix66gptzu_436/efniojlnjndmcbiieegkicadnoecjjef_436_all_cjc3yrpjjjn… 200       [no content]        94ms
  07:55:21 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/j4cp2efc723goxibuix66gptzu_436/efniojlnjndmcbiieegkicadnoecjjef_436_all_cjc3yrpjjjn… 200 …tion/octet-stream  156k 150ms
  07:55:26 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  190b 447ms
  07:55:43 HTTP HEAD            edgedl.me.gvt1.com /edgedl/delta-update/ggkkehgbnfjpeggfpleeakpidbkibbmn/1.91c8c854ea66e0eb7c73b6cc13fecce69e7b63e976b0119fb6d64a5efe379… 200       [no content]        62ms
  07:55:43 HTTP GET             edgedl.me.gvt1.com /edgedl/delta-update/ggkkehgbnfjpeggfpleeakpidbkibbmn/1.91c8c854ea66e0eb7c73b6cc13fecce69e7b63e976b0119fb6d64a5efe379… 200 …tion/octet-stream  8.0k  56ms
  07:55:47 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/adeib7r4iwzlkrhenjav6vxs3uja_2022.12.7.758/ggkkehgbnfjpeggfpleeakpidbkibbmn_2022.12… 200       [no content]        42ms
  07:55:47 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/adeib7r4iwzlkrhenjav6vxs3uja_2022.12.7.758/ggkkehgbnfjpeggfpleeakpidbkibbmn_2022.12… 200 …tion/octet-stream 10.1k  91ms
  07:55:51 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  193b 286ms
  07:55:52 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/acjgztdbjqsnhhrv3mj6jj7z76ja_2896/jflookgnkcckhobaglndicnbbgbonegd_2896_all_mzwjwhz… 200       [no content]       332ms
  07:55:52 HTTP GET             edgedl.me.gvt1.com /edgedl/release2/chrome_component/acjgztdbjqsnhhrv3mj6jj7z76ja_2896/jflookgnkcckhobaglndicnbbgbonegd_2896_all_mzwjwhz… 200 …tion/octet-stream 50.2k  70ms
  07:55:56 HTTP POST          update.googleapis.com /service/update2/json                                                                            200  application/json  184b 321ms
  07:55:57 HTTP HEAD            edgedl.me.gvt1.com /edgedl/release2/chrome_component/aci55fvq5bpt6embd44yshg55ulq_110.0.5473.0/jamhcnnkihinmdlkakkaopbjbbcngflc_110.0.54… 200       [no content]       115ms
⇩  [1/43]  [showhost][transparent]                                                                                                                                                   [*:8080]
```

Figure 5: MITM proxy flow

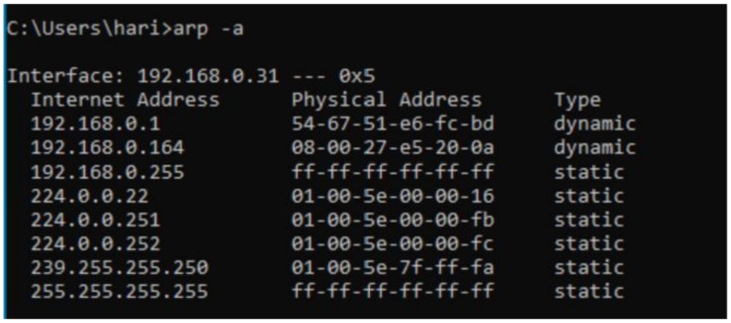
3 Procedure

3.1 Capturing the traffic without MITM attack

In this phase no configuration is required on Kali Linux. In Windows 10, Enable Wireshark to capture the traffic on ethernet then initiate web applications connection from the browser. It is important to clear the browser history and cache for each connection.

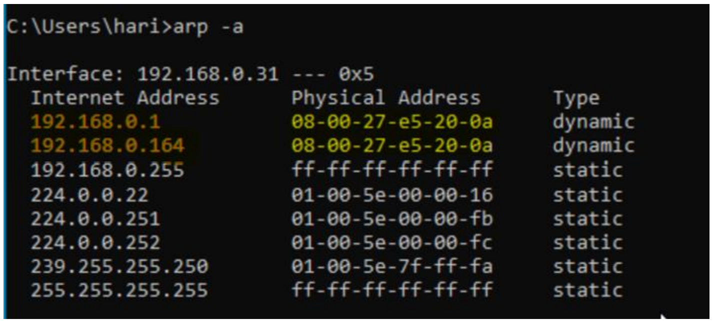
3.2 Capturing the traffic during MITM attack

Firstly, collect the arp table information from windows 10. Then enable Ettercap on Kali Linux and configure target 1 as Windows 10 and target 2 as network gateway. Use the options available in Ettercap to start an MITM attack using ARP poisoning. Verify the arp table on Window 10 you would notice the MAC address for the gateway is changed as Kali Linux MAC address Which denotes successful execution of MITM attack.



```
C:\Users\hari>arp -a

Interface: 192.168.0.31 --- 0x5
  Internet Address      Physical Address      Type
  192.168.0.1           54-67-51-e6-fc-bd     dynamic
  192.168.0.164         08-00-27-e5-20-0a     dynamic
  192.168.0.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

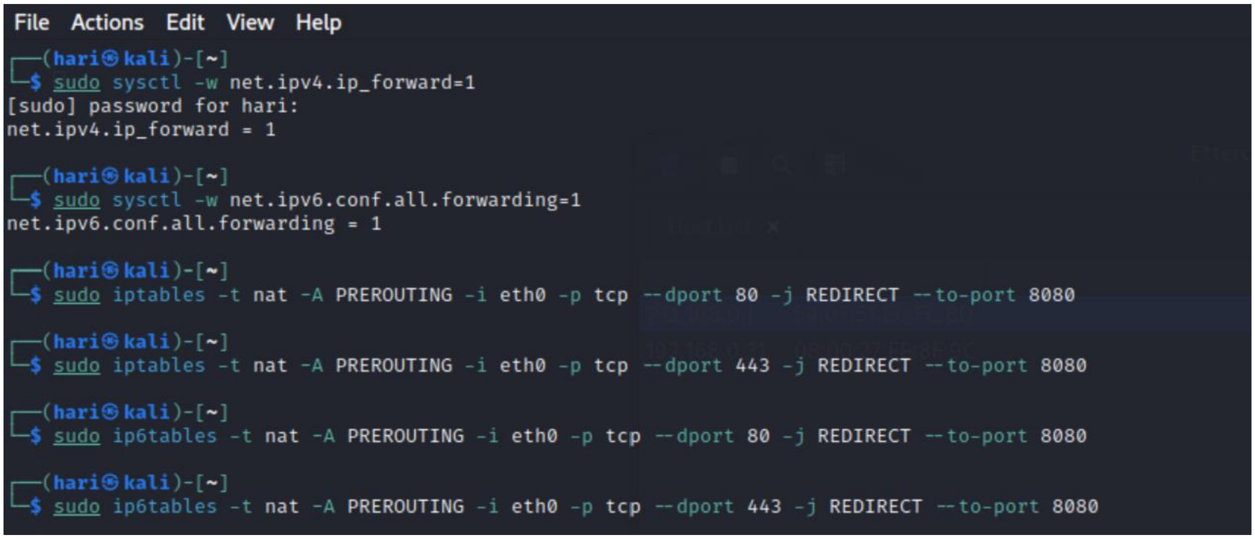
Figure 6: ARP table before MITM



```
C:\Users\hari>arp -a

Interface: 192.168.0.31 --- 0x5
  Internet Address      Physical Address      Type
  192.168.0.1           08-00-27-e5-20-0a     dynamic
  192.168.0.164         08-00-27-e5-20-0a     dynamic
  192.168.0.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

Figure 7: ARP table after MITM

Secondly, we will configure all the prerequisites to enable Mitmproxy on Kali Linux to intercept the https connection. Configure IPV4 and IPV6 forwarding on Kali Linux. Configure IP table rule set to redirect the http and https traffic to the port number tcp:8080 where the Mitm proxy listens. Enable Mitmproxy in transparent mode on Kali Linux. Traffic is routed into a proxy at the network layer when a transparent proxy is optioned to opt; clients do not need to be configured.



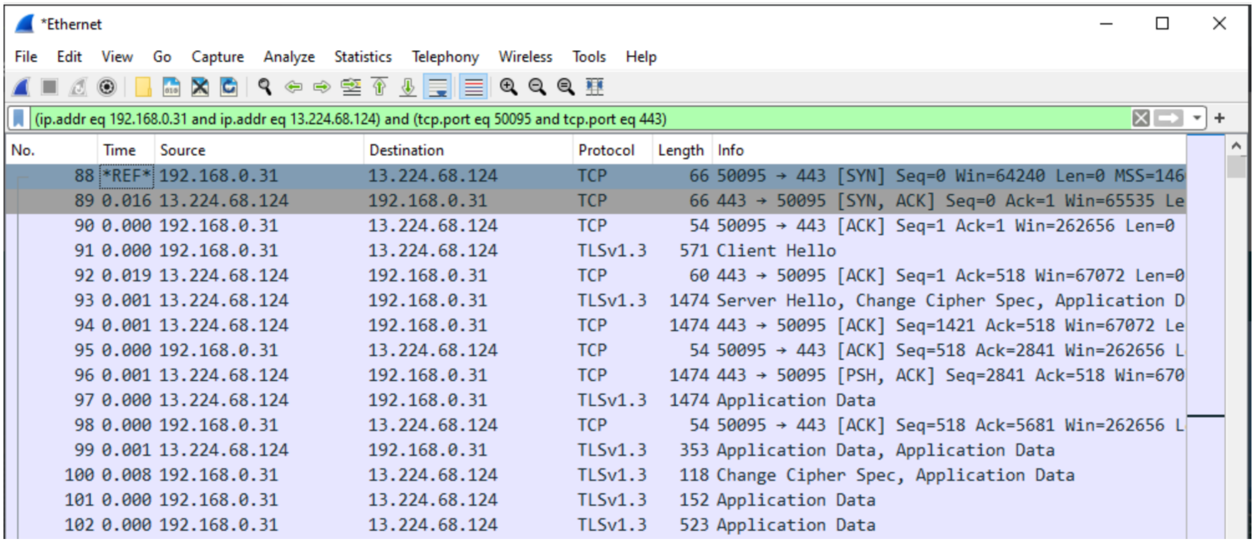
```
┌──(hari㉿kali)-[~]
└─$ sudo sysctl -w net.ipv4.ip_forward=1
[sudo] password for hari:
net.ipv4.ip_forward = 1

┌──(hari㉿kali)-[~]
└─$ sudo sysctl -w net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1

┌──(hari㉿kali)-[~]
└─$ sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080

┌──(hari㉿kali)-[~]
└─$ sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j REDIRECT --to-port 8080

┌──(hari㉿kali)-[~]
└─$ sudo ip6tables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080

┌──(hari㉿kali)-[~]
└─$ sudo ip6tables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j REDIRECT --to-port 8080
```

Figure 8: IP forwarding and IP table rule sets

Lastly capture the traffic on Wireshark from Windows 10 during the MITM attack and make sure to clear the browser history and cache for each connection.

3.3 Sample data collection

We have captured the network packets on Wireshark for 30 web application connections without and with MITM attack. We fetch the IP address of each url by nslookup or ping command on cmd prompt on Windows. Then we used the IP address of websites and clients in the Wireshark conversation filter and filtered the tcp and https steam to get time stamps for TCP and SSL shake. Then set up reference point to first SYN packet of TCP handshake to get packets in sequences and selection option to view time in milliseconds. Tabulated the collected sample data in the spreadsheet.

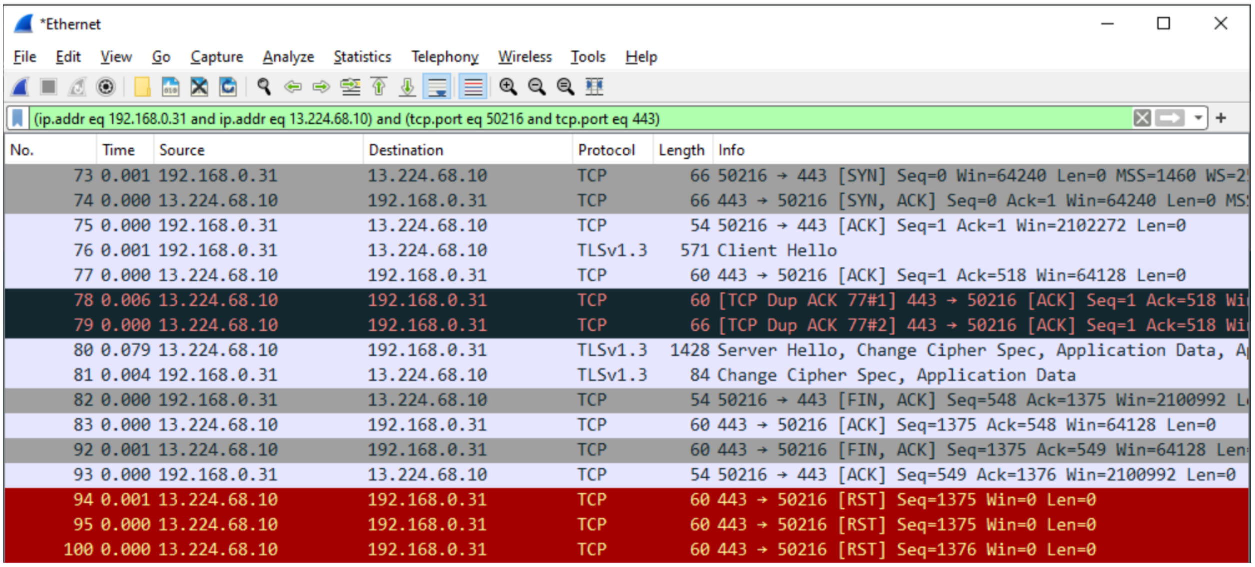


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

(ip.addr eq 192.168.0.31 and ip.addr eq 13.224.68.124) and (tcp.port eq 50095 and tcp.port eq 443)

```
No.     Time   Source           Destination      Protocol  Length Info
    88 *REF* 192.168.0.31     13.224.68.124    TCP          66 50095 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=146
    89 0.016 13.224.68.124    192.168.0.31     TCP          66 443 → 50095 [SYN, ACK] Seq=0 Ack=1 Win=65535 Le
    90 0.000 192.168.0.31     13.224.68.124    TCP          54 50095 → 443 [ACK] Seq=1 Ack=1 Win=262656 Len=0
    91 0.000 192.168.0.31     13.224.68.124    TLSv1.3     571 Client Hello
    92 0.019 13.224.68.124    192.168.0.31     TCP          60 443 → 50095 [ACK] Seq=1 Ack=518 Win=67072 Len=0
    93 0.001 13.224.68.124    192.168.0.31     TLSv1.3    1474 Server Hello, Change Cipher Spec, Application D
    94 0.001 13.224.68.124    192.168.0.31     TCP        1474 443 → 50095 [ACK] Seq=1421 Ack=518 Win=67072 Le
    95 0.000 192.168.0.31     13.224.68.124    TCP          54 50095 → 443 [ACK] Seq=518 Ack=2841 Win=262656 L
    96 0.001 13.224.68.124    192.168.0.31     TCP        1474 443 → 50095 [PSH, ACK] Seq=2841 Ack=518 Win=670
    97 0.000 13.224.68.124    192.168.0.31     TLSv1.3    1474 Application Data
    98 0.000 192.168.0.31     13.224.68.124    TCP          54 50095 → 443 [ACK] Seq=518 Ack=5681 Win=262656 L
    99 0.001 13.224.68.124    192.168.0.31     TLSv1.3     353 Application Data, Application Data
   100 0.008 192.168.0.31     13.224.68.124    TLSv1.3     118 Change Cipher Spec, Application Data
   101 0.000 192.168.0.31     13.224.68.124    TLSv1.3     152 Application Data
   102 0.000 192.168.0.31     13.224.68.124    TLSv1.3     523 Application Data
```

Figure 9: Packet captured before MITM attack


*Ethernet
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

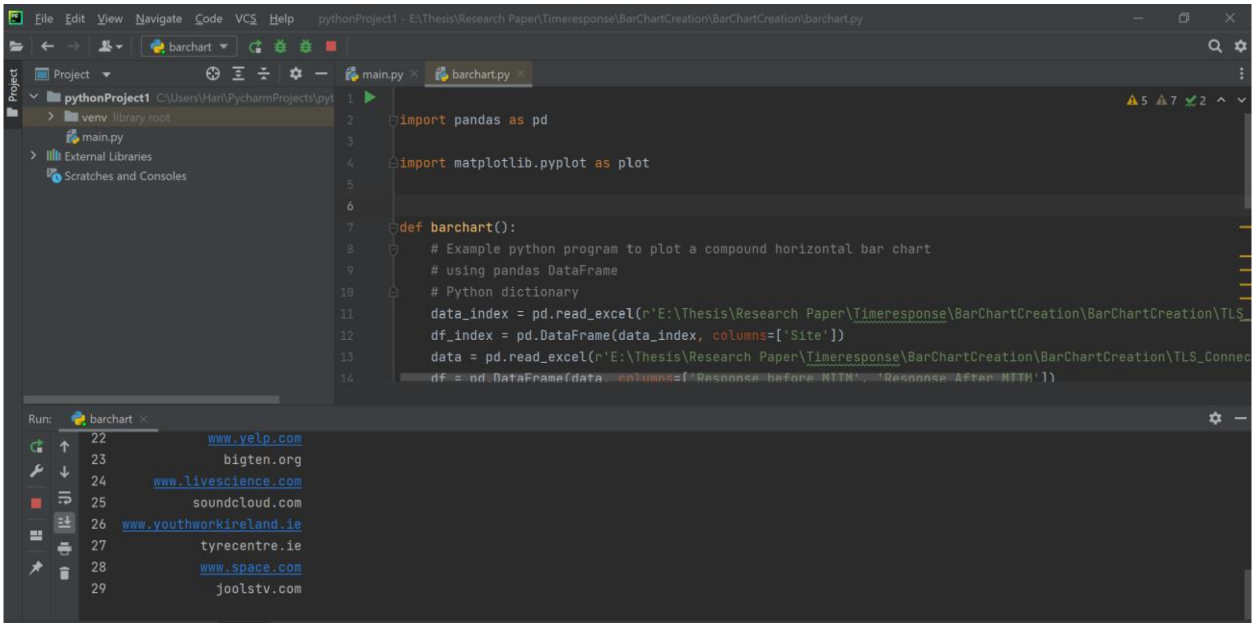
```
(ip.addr eq 192.168.0.31 and ip.addr eq 13.224.68.10) and (tcp.port eq 50216 and tcp.port eq 443)
```

```
No.     Time  Source        Destination   Protocol  Length Info
     73 0.001 192.168.0.31  13.224.68.10  TCP          66 50216 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=2
     74 0.000 13.224.68.10  192.168.0.31  TCP          66 443 → 50216 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MS
     75 0.000 192.168.0.31  13.224.68.10  TCP          54 50216 → 443 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
     76 0.001 192.168.0.31  13.224.68.10  TLSv1.3     571 Client Hello
     77 0.000 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [ACK] Seq=1 Ack=518 Win=64128 Len=0
     78 0.006 13.224.68.10  192.168.0.31  TCP          60 [TCP Dup ACK 77#1] 443 → 50216 [ACK] Seq=1 Ack=518 Wi
     79 0.000 13.224.68.10  192.168.0.31  TCP          66 [TCP Dup ACK 77#2] 443 → 50216 [ACK] Seq=1 Ack=518 Wi
     80 0.079 13.224.68.10  192.168.0.31  TLSv1.3    1428 Server Hello, Change Cipher Spec, Application Data, A
     81 0.004 192.168.0.31  13.224.68.10  TLSv1.3      84 Change Cipher Spec, Application Data
     82 0.000 192.168.0.31  13.224.68.10  TCP          54 50216 → 443 [FIN, ACK] Seq=548 Ack=1375 Win=2100992 L
     83 0.000 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [ACK] Seq=1375 Ack=548 Win=64128 Len=0
     92 0.001 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [FIN, ACK] Seq=1375 Ack=549 Win=64128 Len
     93 0.000 192.168.0.31  13.224.68.10  TCP          54 50216 → 443 [ACK] Seq=549 Ack=1376 Win=2100992 Len=0
     94 0.001 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [RST] Seq=1375 Win=0 Len=0
     95 0.000 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [RST] Seq=1375 Win=0 Len=0
    100 0.000 13.224.68.10  192.168.0.31  TCP          60 443 → 50216 [RST] Seq=1376 Win=0 Len=0
```

Figure 10: Packet capture after MITM

4 Evaluation

Populate the sample data spreadsheet into a python program for evaluation. The figure shows the results of using our written program to visually represent the round-trip timing variance for TCP and SSL/TLS connections with and without MITM attack for the 30 selected domains.


File Edit View Navigate Code VCS Help
pythonProject1 - E:\Thesis\Research Paper\Timeresponse\BarChartCreation\BarChartCreation\barchart.py
barchart
Project
pythonProject1 C:\Users\Hari\PycharmProjects\pyt
venv library root
main.py
External Libraries
Scratches and Consoles
main.py
barchart.py
5 7 2
Run: barchart

```python

import pandas as pd

import matplotlib.pyplot as plot


def barchart():
    # Example python program to plot a compound horizontal bar chart
    # using pandas DataFrame
    # Python dictionary
    data_index = pd.read_excel(r'E:\Thesis\Research Paper\Timeresponse\BarChartCreation\BarChartCreation\TLS_
    df_index = pd.DataFrame(data_index, columns=['Site'])
    data = pd.read_excel(r'E:\Thesis\Research Paper\Timeresponse\BarChartCreation\BarChartCreation\TLS_Connec
    df = pd.DataFrame(data, columns=['Response before MITM', 'Response After MITM'])
```

```
22           www.yelp.com
23             bigten.org
24    www.livescience.com
25         soundcloud.com
26 www.youthworkireland.ie
27           tyrecentre.ie
28           www.space.com
29           joolstv.com
```

Figure 11: Python code for comparative analysis

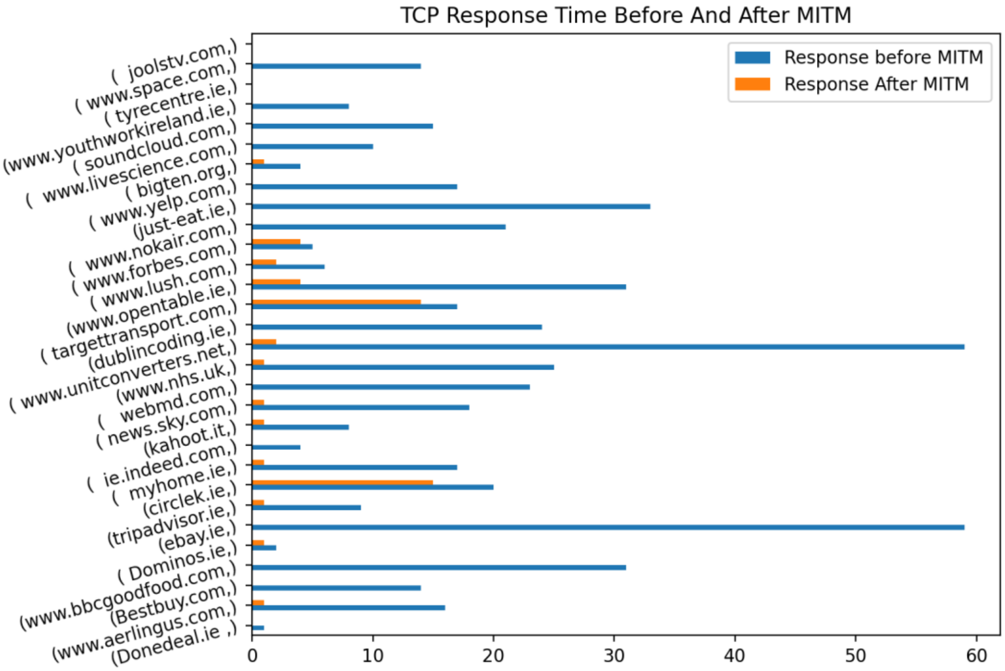


Figure 12: TCP response time with and without MTIM

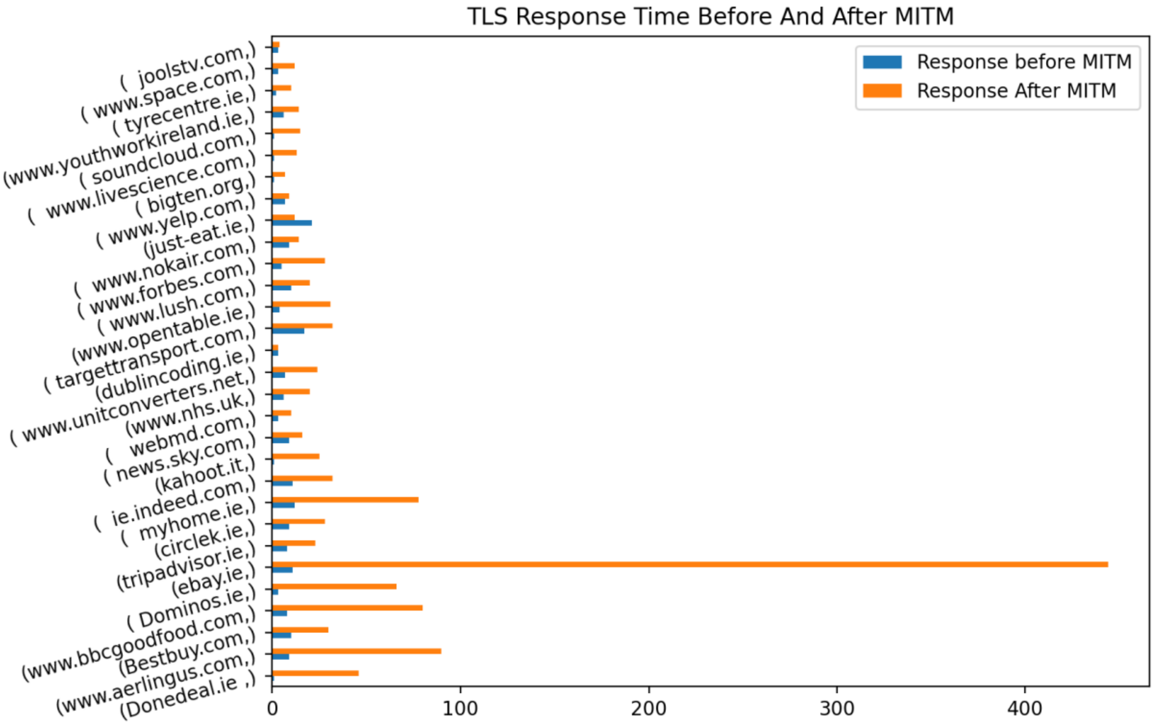


Figure 13: TLS response time with and without MITM