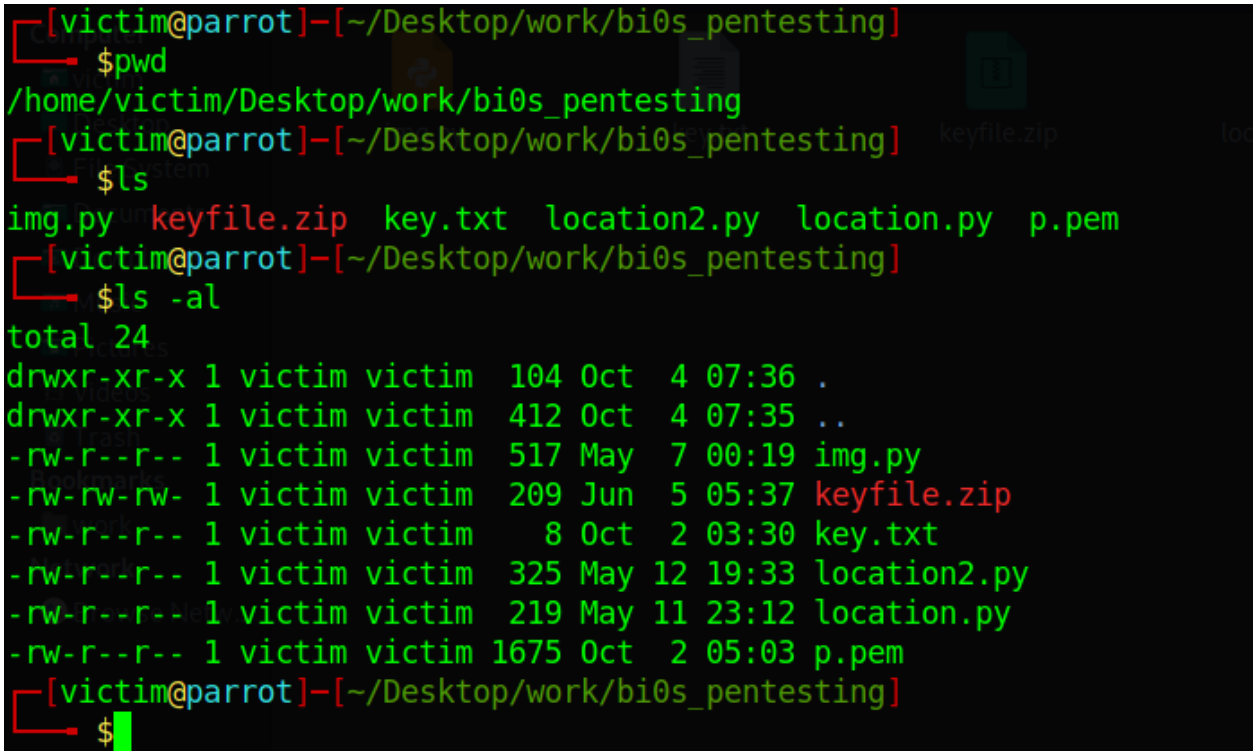
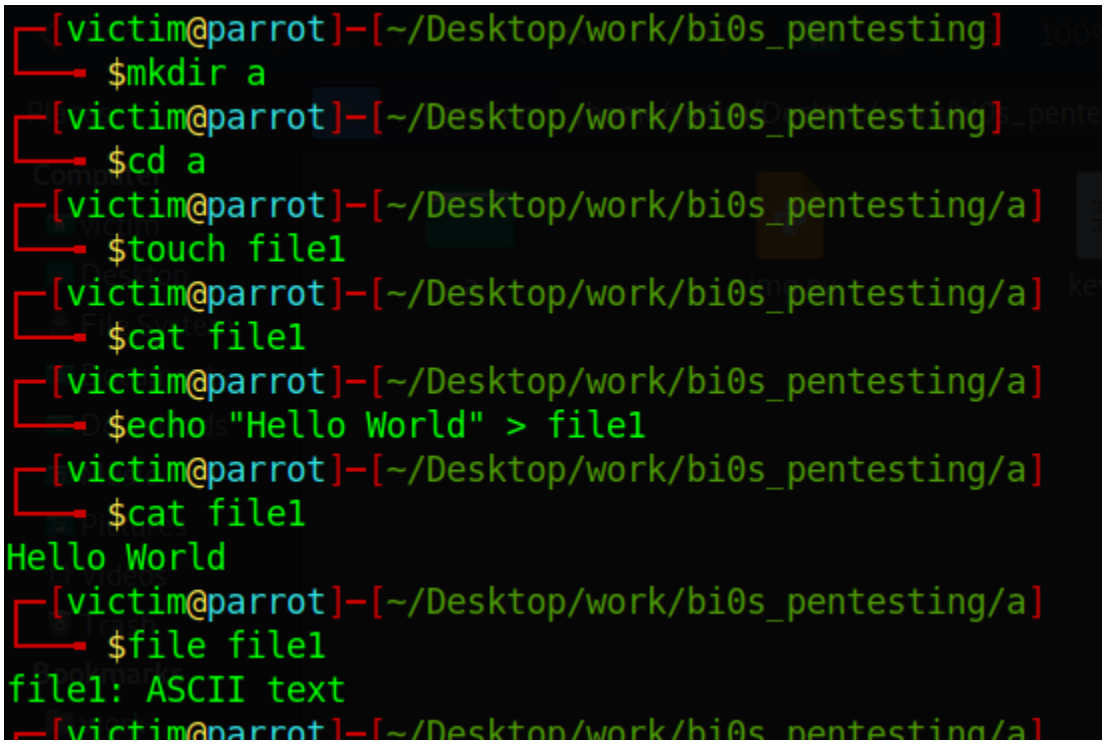
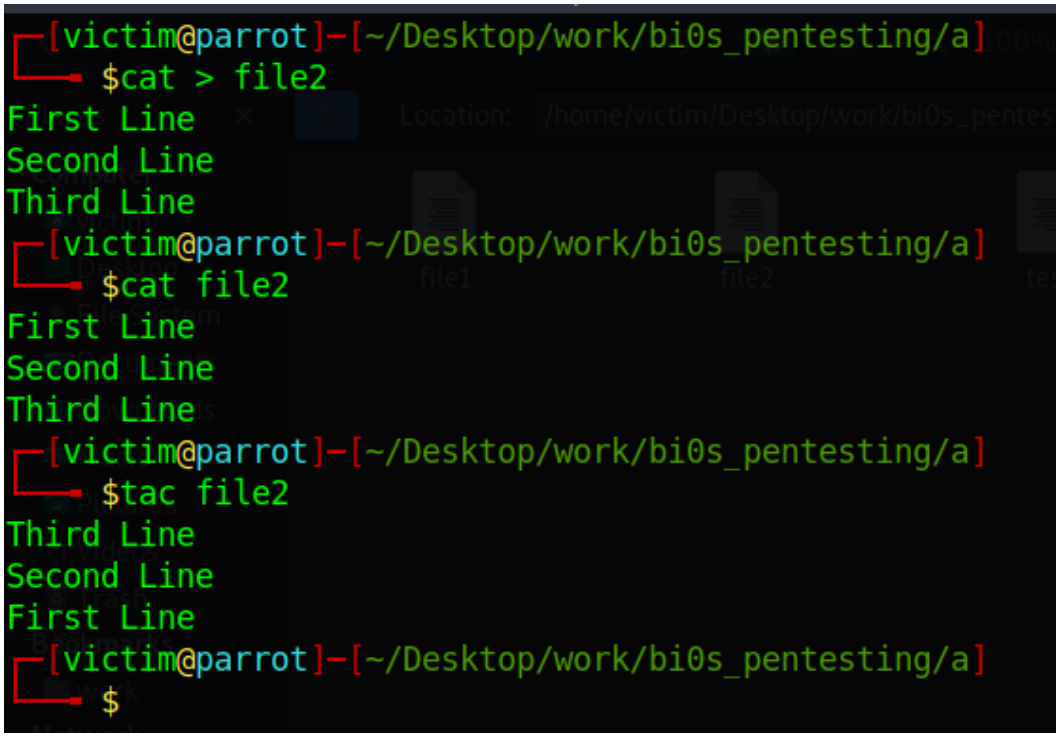
1.


[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$pwd
/home/victim/Desktop/work/bi0s_pentesting
[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$ls
img.py keyfile.zip key.txt location2.py location.py p.pem
[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$ls -al
total 24
drwxr-xr-x 1 victim victim 104 Oct 4 07:36 .
drwxr-xr-x 1 victim victim 412 Oct 4 07:35 ..
-rw-r--r-- 1 victim victim 517 May 7 00:19 img.py
-rw-rw-rw- 1 victim victim 209 Jun 5 05:37 keyfile.zip
-rw-r--r-- 1 victim victim 8 Oct 2 03:30 key.txt
-rw-r--r-- 1 victim victim 325 May 12 19:33 location2.py
-rw-r--r-- 1 victim victim 219 May 11 23:12 location.py
-rw-r--r-- 1 victim victim 1675 Oct 2 05:03 p.pem
[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$

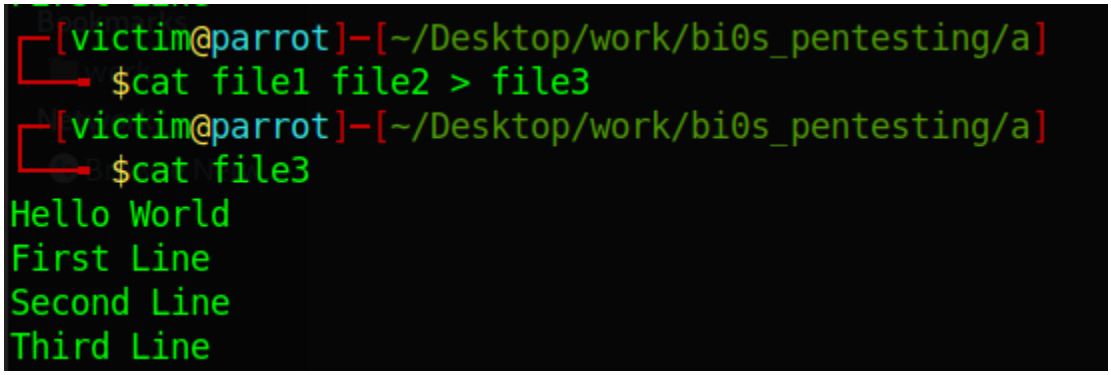
2.


[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$mkdir a
[victim@parrot]-[~/Desktop/work/bi0s_pentesting]
$cd a
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$touch file1
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat file1
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$echo "Hello World" > file1
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat file1
Hello World
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$file file1
file1: ASCII text

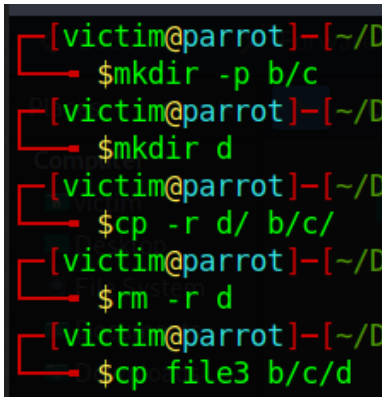
3.


[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat > file2
First Line
Second Line
Third Line
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat file2
First Line
Second Line
Third Line
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$tac file2
Third Line
Second Line
First Line
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$

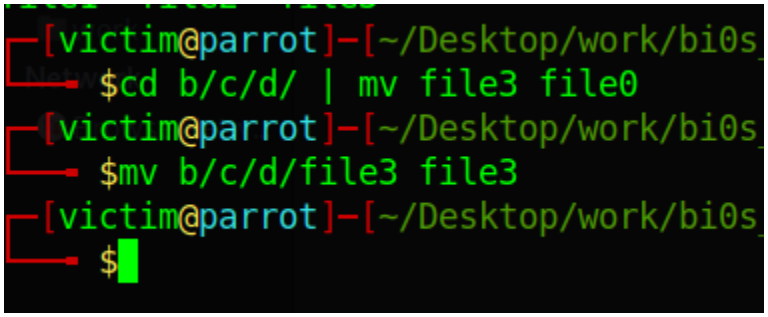
4.


[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat file1 file2 > file3
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cat file3
Hello World
First Line
Second Line
Third Line

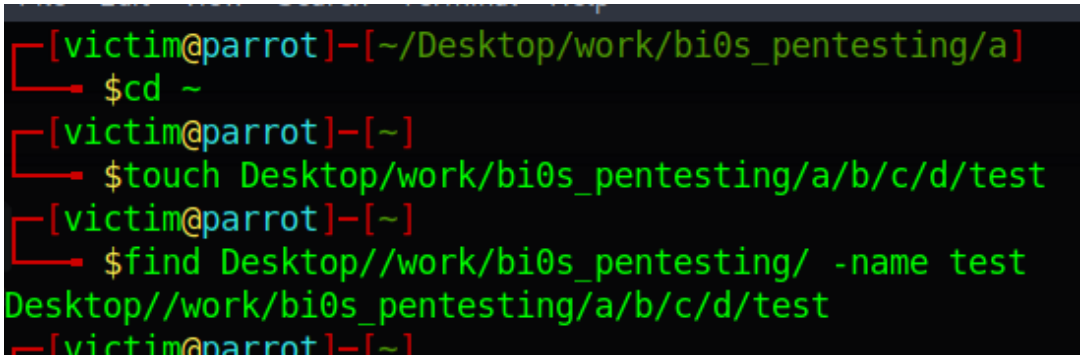
5.


[victim@parrot]-[~/D
$mkdir -p b/c
[victim@parrot]-[~/D
$mkdir d
[victim@parrot]-[~/D
$cp -r d/ b/c/
[victim@parrot]-[~/D
$rm -r d
[victim@parrot]-[~/D
$cp file3 b/c/d

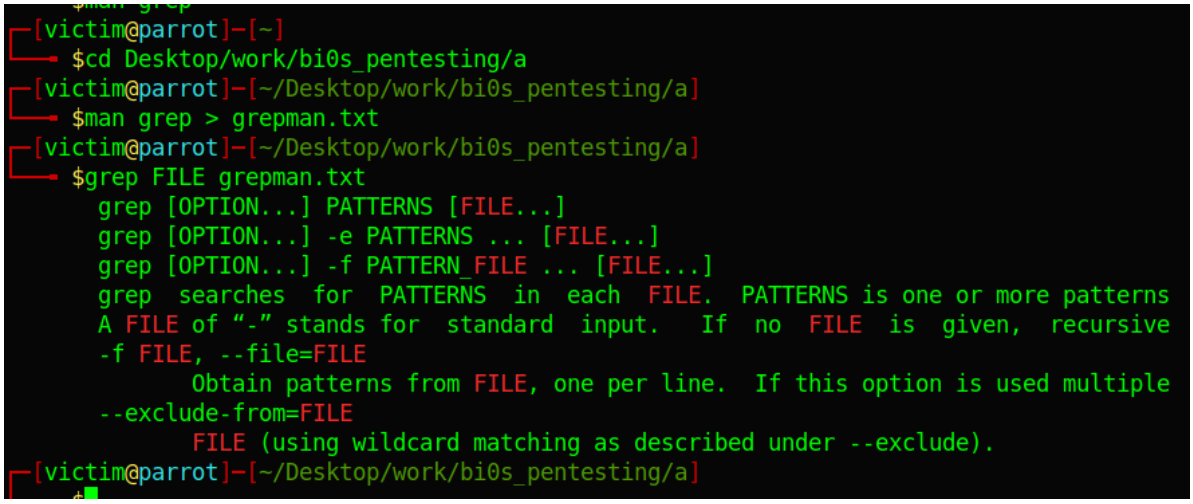
6.


[victim@parrot]-[~/Desktop/work/bi0s
$cd b/c/d/ | mv file3 file0
[victim@parrot]-[~/Desktop/work/bi0s
$mv b/c/d/file3 file3
[victim@parrot]-[~/Desktop/work/bi0s
$

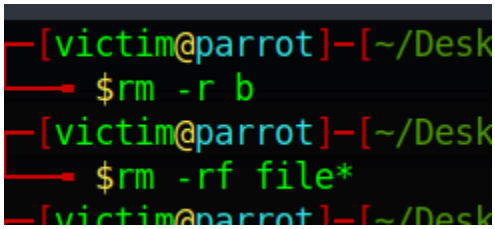
7.


[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$cd ~
[victim@parrot]-[~]
$touch Desktop/work/bi0s_pentesting/a/b/c/d/test
[victim@parrot]-[~]
$find Desktop//work/bi0s_pentesting/ -name test
Desktop//work/bi0s_pentesting/a/b/c/d/test

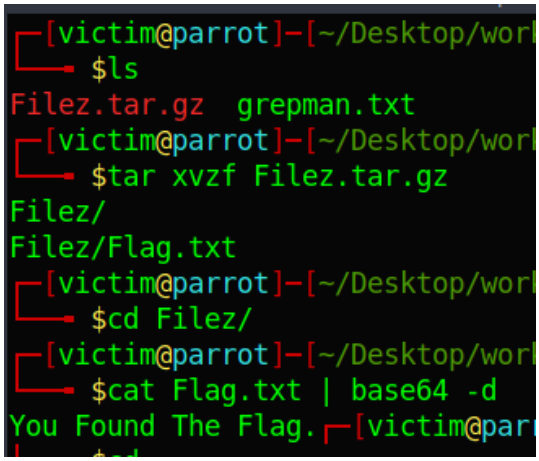
8.


[victim@parrot]-[~]
$cd Desktop/work/bi0s_pentesting/a
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$man grep > grepman.txt
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]
$grep FILE grepman.txt
grep [OPTION...] PATTERNS [FILE...]
grep [OPTION...] -e PATTERNS ... [FILE...]
grep [OPTION...] -f PATTERN_FILE ... [FILE...]
grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns
A FILE of "-" stands for standard input. If no FILE is given, recursive
-f FILE, --file=FILE
Obtain patterns from FILE, one per line. If this option is used multiple
--exclude-from=FILE
FILE (using wildcard matching as described under --exclude).
[victim@parrot]-[~/Desktop/work/bi0s_pentesting/a]

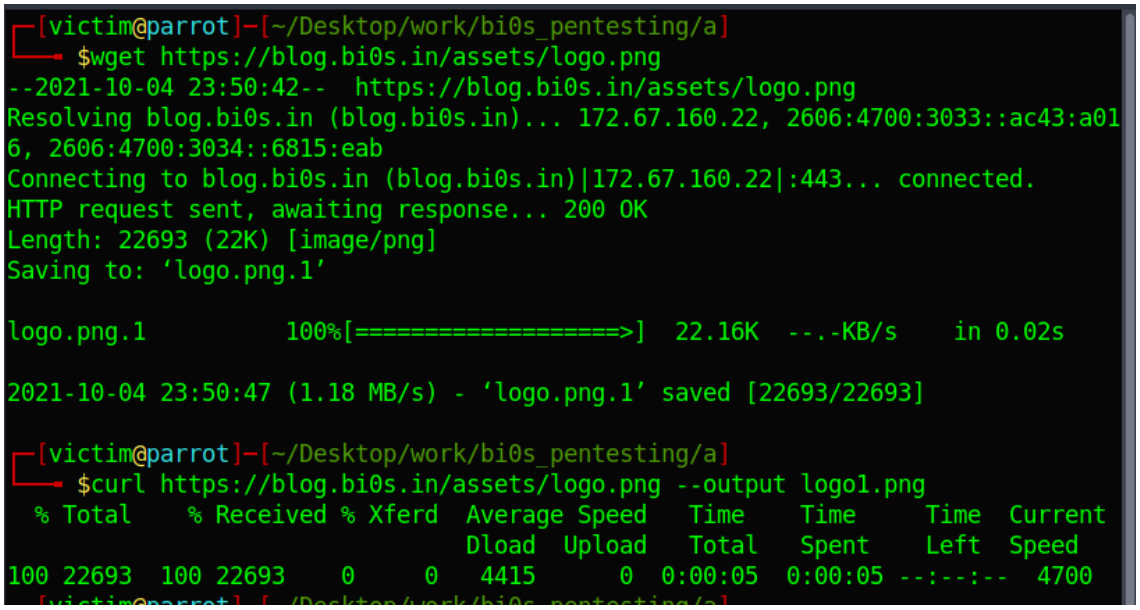
9.


[victim@parrot]-[~/Desk
$rm -r b
[victim@parrot]-[~/Desk
$rm -rf file*

10.

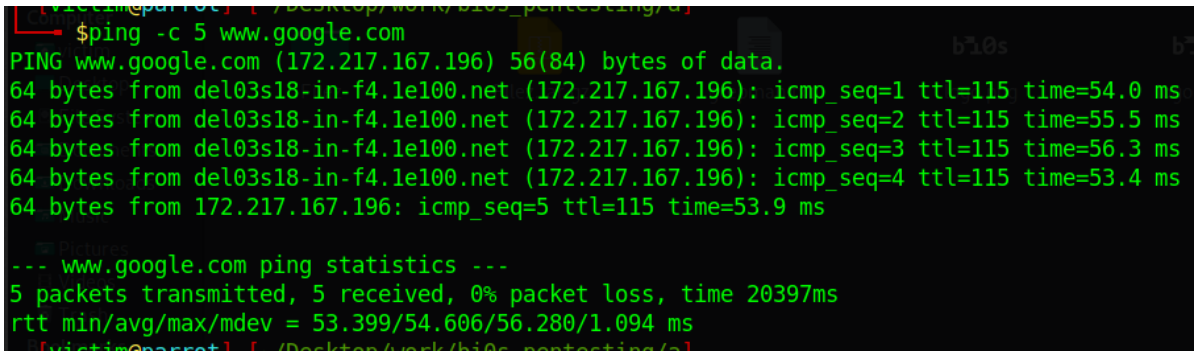


11.



12.

A.



Min = 53.399 ms

```
└─$ ping -c 6 www.google.com
PING www.google.com (172.217.167.196) 56(84) bytes of data.
64 bytes from del03s18-in-f4.1e100.net (172.217.167.196): icmp_seq=1 ttl=115 time=54.9 ms
64 bytes from del03s18-in-f4.1e100.net (172.217.167.196): icmp_seq=2 ttl=115 time=54.2 ms
64 bytes from del03s18-in-f4.1e100.net (172.217.167.196): icmp_seq=3 ttl=115 time=55.1 ms
64 bytes from del03s18-in-f4.1e100.net (172.217.167.196): icmp_seq=4 ttl=115 time=55.1 ms
64 bytes from del03s18-in-f4.1e100.net (172.217.167.196): icmp_seq=5 ttl=115 time=54.8 ms
64 bytes from 172.217.167.196: icmp_seq=6 ttl=115 time=54.2 ms

--- www.google.com ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 25528ms
rtt min/avg/max/mdev = 54.191/54.705/55.087/0.360 ms
```

Avg = 54.705 ms

13.

bandit0:

ssh bandit0@bandit.labs.overthewire.org -p 2220

connect to the above port and gave password bandit0

bandit0:

there is a readme file : `boJ9jbbUNNfktd78OOpsqOltutMc3MY1`

this is the password for the bandit2

bandit1:

there is a file but named as "-" so used command cat < - to read the file

got new password `CV1DtqXWVFXTvM2F0k09SHz0YwRINYA9`

bandit2:

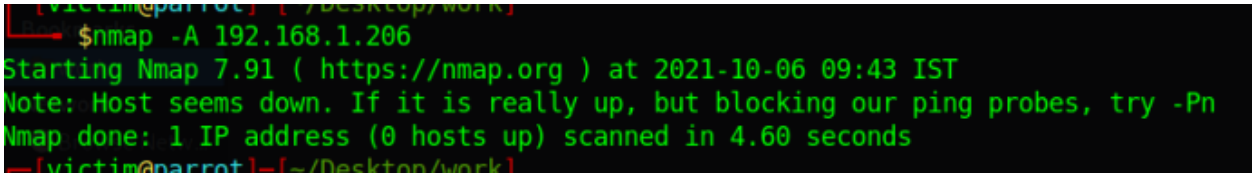
here there is a fie but with spaces in between so added \ insted of spaces cat spaces\ in\ this\ filename

the password is `UmHadQclWmgdLOKQ3YNgjWxGoRMb5luK`

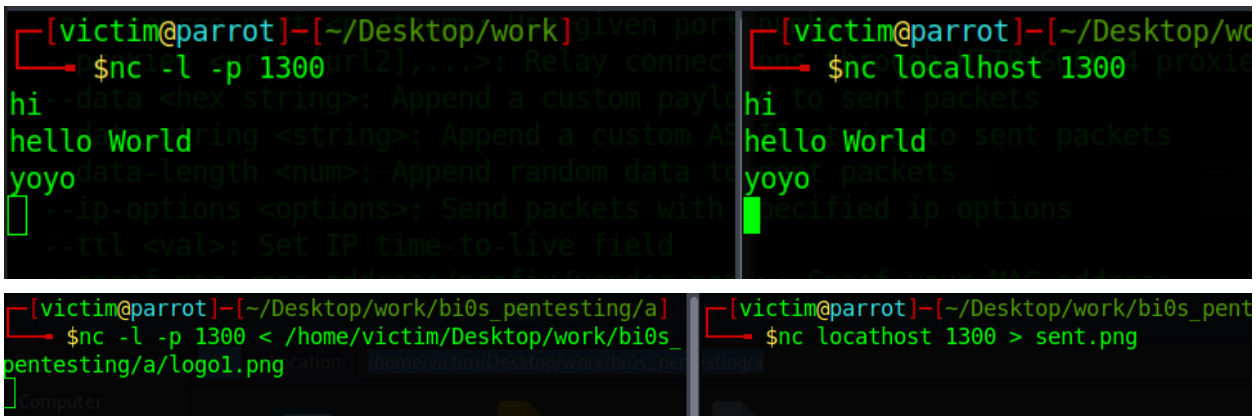
14.

```
┌──[✗]─[victim@parrot]─[~/Desktop/work]
└─$ telnet localhost 5432
Trying ::1...
Connected to localhost.
Escape character is '^]'.
```

15.



16.



Shell Scripting

1.

```
# !/bin/bash

echo -n "Enter a : "
read a
echo -n "Enter b : "
read b

echo "Choose which Opertaion :"
echo "1. Addition"
echo "2. Subtraction"
echo "3. Multiplication"
echo "4. Division"
echo "5. Average"
echo -n "Enter the option : "
read opt

case $opt in
  1)cal=`expr $a + $b `
  ;;
  2)cal=`expr $a - $b `
  ;;
  3)cal=`expr $a \* $b `
  ;;
  4)cal=`echo "scale=2; $a / $b" | bc `
  ;;
  5)cal=`echo "scale=2; ($a+$b)/2" | bc `
esac
echo "Result : $cal"
```

```
$./calculator.sh
Enter a : 6
Enter b : 9
Choose which Opertaion :
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Average
Enter the option : 4
Result : .66
```

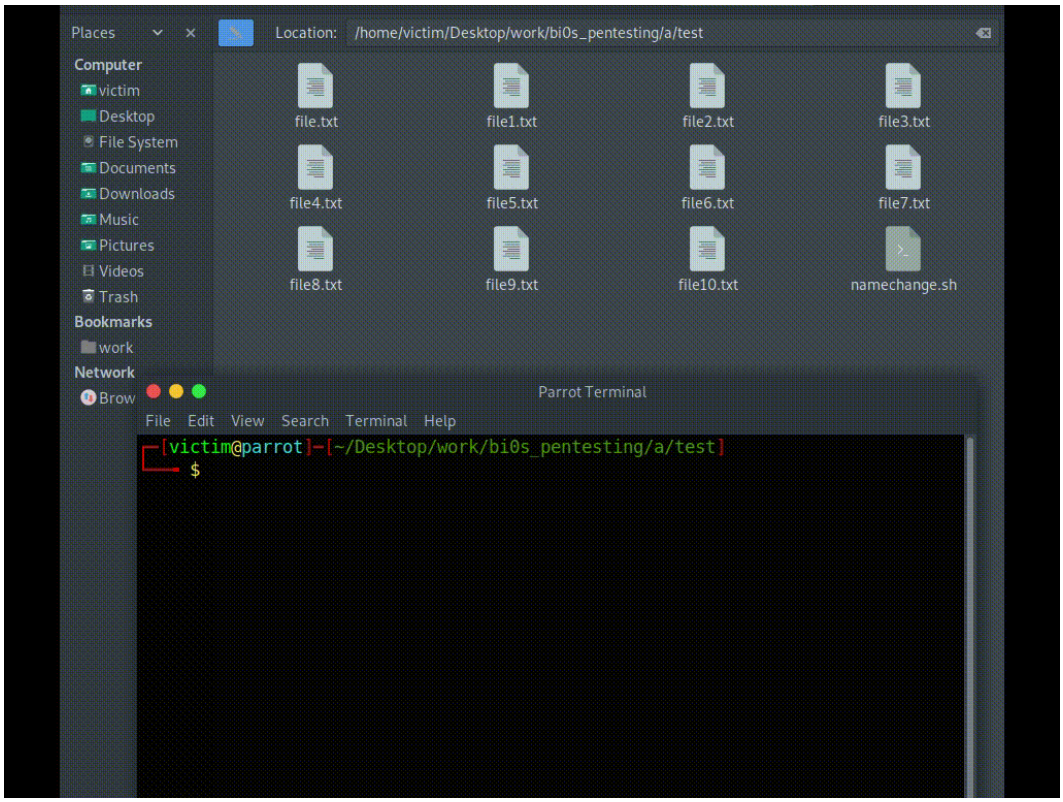
2.

```
# !/bin/bash

echo "1. ROT13 encoder"
echo "2. ROT13 decoder"

echo -n "Choose 1 or 2: "
read opt
if [[ $opt -eq 1 || $opt -eq 2 ]];
then
    echo -n "Enter the String: "
    read string

    case $opt in
        1)enc=`echo $string | tr 'A-Za-z' 'N-ZA-Mn-za-m'`
            ;;
        2)enc=`echo $string | tr 'n-za-mN-ZA-M' 'a-zA-Z'`
            ;;
    esac
    echo "Result : $enc"
else
    echo "exit"
fi
```

```
└─$ ./rot13.sh
1. ROT13 encoder
2. ROT13 decoder
Choose 1 or 2: 1
Enter the String: manohar
Result : znabune
┌─[victim@parrot]─[~/Desktop/wo
└─$ ./rot13.sh
1. ROT13 encoder
2. ROT13 decoder
Choose 1 or 2: 2
Enter the String: znabune
Result : manohar
```

3.

```
# !/bin/bash

for file in *.txt; do
    VAR=$file
    VAR1=`echo $VAR | cut -c1-4`
    VAR2=`echo "-" `
    now=`date +"%d%m"`
    if [ "$VAR1" != "$now" ]; then
        `mv $VAR $now$VAR2$VAR`
    fi
done
```



4.

```
# !/bin/ bash
echo -n "Number of elenemts in array: "
read n

echo "enter elements into array:"
for ((i= 0; i< $n; i++)); do
    read num[$i]
done

for ((i = 0; i < $n; i++)); do
    for ((j = $i; j < $n; j++)); do
        if [[ ${num[$i]} -gt ${num[$j]} ]];
        then
            t=${num[$i]}
            num[$i]=${num[$j]}
            num[$j]=$t
        fi
    done
done

echo "Sorted Numbers: "
for ((i = 0; i < $n; i++)); do
    echo ${num[$i]}
done
```

```
┌─[victim@parrot]─[~/Desktop/work
└──╼ $./bubblesort.sh
Number of elenemts in array: 3
enter elements into array:
3
2
1
Sorted Numbers:
1
2
3
```

5.

```
# !/bin/bash
echo -n "Enter the String: "
read str
rts=`echo "$str" | rev `

if [ "$str" = "$rts" ]; then
    echo "is a Palindrom."
else
    echo "Not a Palindrom."
fi
```

```
┌─[victim@parrot]-[~/Deskto
└──╼ $./palindrom.sh
Enter the String: 12321
is a Palindrom.
┌─[victim@parrot]-[~/Deskto
└──╼ $./palindrom.sh
Enter the String: 12345
Not a Palindrom.
┌─[victim@parrot]-[~/Deskto
```