CCF Lab Assignment:
Hiding and Filesystems

1 Steganography

1. Look for 3 different tools that can hide files in images.

(a) How do the tools hide the files?

Answer:

Steghide is a steganography program that is able to hide data in various kinds of image- and audio-files. The color- respectivly sample-frequencies are not changed thus making the embedding resistant against first-order statistical tests.

Features: *) compression of embedded data *) encryption of embedded data *) embedding of a checksum to verify the integrity of the extraced data *) support for JPEG, BMP, WAV and AU files

Crypture is another command line tool that performs Steganography. You can use this tool to hide your sensitive data inside a BMP image file. But there is one requirement. BMP file should be eight times larger than the data file which you want to hide inside the BMP file. If you have a small amount of data to hide, you can use this tool. This tool is very small and is only 6KB in size. It does not need any kind of installation.

rSteg is a Java based tool that lets you hide textual data inside an image. It has two buttons: one to encrypt and second to decrypt the text. Just select the image file, enter the PIN, and then enter the text which you want to hide in the image. It will generate a target image file with hidden text inside. If you want to read that text again, use this tool and select decrypt option.

(b) Using very exotic tools usually leads to better results. Explain why this is.

Answer:

exotic tools are most likely to be proprietary software (not open source) reason for that is these tools are made for military, hight profile security experts and so on. So the chances that a forensics tools or methods to detect is really low. They are less likely to be used by all people and as exotic tools means they used really robust algorithm to implement the steganography.

© Are there any detection tools that can detect the tools you name?

Answer:

a one possible answer for this is **Stegdetect**, it can detect steghide, rSteg, and Crypture stenographies (since they all used somehow the same technique). However, it is not fully successful according to http://theevilbit.blogspot.nl/2013/01/backtrack-forensics-steganoghraphy.html it is most likely to detect images created by these tools.

Sources:

1- https://github.com/StefanoDeVuono/steghide

2- https://sourceforge.net/projects/crypture/

2. Pick your favorite tool and use this to send over a file to another student. Be creative in the way of transportation.

Answer:

I will use steghide for this:

```
kotaiba@bristol:~$ steghide embed -ef hiddenInfo.txt -cf q2.jpg
Enter passphrase:
Re-Enter passphrase:
embedding "hiddenInfo.txt" in "q2.jpg"... done
```

The image that I will sent to another student:

3. Install a detection package and use it to detect whether your secret is exposed. Write down the steps you take.

Answer:

I will use stegdetect for this (https://blog.robseder.com/2015/08/27/steganography-with-linux/)

```
root@bristol:/home/kotaiba# stegdetect q2.jpg
q2.jpg : negative
```

stegdetect couldn't detect it

4. Decode a file from one of your colleagues. Explain the steps that you used.

Answer:

I will decode Sjorst image, 😃.

What I did is that I tried many tools and different script that (crack steghide (Sjorst stated that for me) passphrase. However, I wasted a lot of time on it without any result. I remembered that Sjorst told me that the passphrase is exist, So I though ok let me try to check the metadata of the image:

```
root@bristol:/home/kotaiba# exiftool q3.jpg
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
	LANGUAGE = (unset),
	LC_ALL = (unset),
	LC_CTYPE = "UTF-8",
	LANG = "en_US.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to a fallback locale ("en_US.UTF-8").
ExifTool Version Number         : 10.10
File Name                       : q3.jpg
Directory                       : .
File Size                       : 1763 kB
File Modification Date/Time     : 2018:02:18 20:50:50+01:00
File Access Date/Time           : 2018:02:18 20:51:05+01:00
File Inode Change Date/Time     : 2018:02:18 20:50:50+01:00
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : None
X Resolution                    : 1
Y Resolution                    : 1
Comment                         : The passphrase is hidden somewhere
Image Width                     : 4032
Image Height                    : 3024
Encoding Process                : Baseline DCT, Huffman coding
Bits Per Sample                 : 8
Color Components                : 3
Y Cb Cr Sub Sampling            : YCbCr4:2:0 (2 2)
Image Size                      : 4032x3024
Megapixels                      : 12.2

root@bristol:/home/kotaiba# steghide extract -sf q3.jpg
```

```
Enter passphrase:
wrote extracted data to "mamma-mia.txt".
root@bristol:/home/kotaiba# cat mamma-mia.txt

kotaiba@bristol:~$ cat mamma-mia.txt
I been cheated by you since you know when
So I made up my mind, it must come to an end
Look at me now, will I ever learn?
I don't know how but I suddenly lose control
There's a fire within my soul
Just one look and I can hear a bell ring
One more look and I forget everything
Mamma mia, here I go again
My my, how can I resist you?
Mamma mia, does it show again
My my, just how much I've missed you?
Yes, I've been brokenhearted
Blue since the day we parted
Why, why did I ever let you go?
Mamma mia, now I really know
My my, I could never let you go
I've been angry and sad about things that you do
I can't count all the times that I've told you "we're through"
And when you go, when you slam the door
I think you know that you won't be away too long
You know that I'm not that strong
Just one look and I can hear a bell ring
One more look and I forget everything
Mamma mia, here I go again
My my, how can I resist you?
Mamma mia, does it show again
My my, just how much I've missed you?
Yes, I've been brokenhearted
Blue since the day we parted
Why, why did I ever let you go?
Mamma mia, even if I say
"Bye bye, leave me now or never"
Mamma mia, it's a game we play
"Bye bye" doesn't mean forever
Mamma mia, here I go again
My my, how can I resist you?
Mamma mia, does it show again
My my, just how much I've missed you?
Yes, I've been brokenhearted
Blue since the day we parted
Why, why did I ever let you go?
Mamma mia, now I really know
My my, I could never let you go
```

😆 😆 after that I tried few combination and I ended up with the password **hidden somewhere** and It works 🤐 .

5. Download Dave’s usbkey.hdd from https://software.os3.nl/CCF/. Make sure you check the files.

Answer:

```
caine@caine:~/Desktop$ wget https://software.os3.nl/CCF/usbkey.hdd
caine@caine:~/Desktop$ wget https://software.os3.nl/CCF/usbkey.hdd.sha2
caine@caine:~/Desktop$ cat usbkey.hdd.sha2
1481d1633dfff916b54bf55647f1085a2b981d01de5df8c250bd07cafbde396e  usbkey.hdd
caine@caine:~/Desktop$ shasum -a 256 usbkey.hdd
1481d1633dfff916b54bf55647f1085a2b981d01de5df8c250bd07cafbde396e  usbkey.hdd
```

6. How would you approach the detection of steganography in a large set of files? Use this method on Dave’s USB key

Answer:

First of all I mounted the image:

```
mount usbkey.hdd /mnt/tmp
```

then I created a bash script to search through specifies extension within the files:

```
 for file in `find /mnt/tmp -iname "*jpg"`; do
        ./stegdetect $file | grep -v "skipped" | grep -v "negative";
    done
```

```
root@caine:~# ./daveDetector.sh
/mnt/tmp/y3mxyzca2l76_wd640.jpg : jphide(***)
/mnt/tmp/uboxtd0a654j_wd640.jpg : jphide(***)
/mnt/tmp/65txnxfapqtg_wd640.jpg : jphide(***)
/mnt/tmp/yq5x204azxo2_wd1280.jpg : jphide(*)
/mnt/tmp/3nrx8jaai9qa_std320.jpg : jphide(**)
/mnt/tmp/j7txrlza2pzl_wd640.jpg : jphide(*)
/mnt/tmp/qwdxt7taixbe_std320.jpg : jphide(*)
/mnt/tmp/z3fxwala42q4_wd640.jpg : jphide(*)
/mnt/tmp/ha6xqpvao7ui_wd640.jpg : jphide(**)
/mnt/tmp/geix6amasoe6_std320.jpg : jphide(*)
/mnt/tmp/k3yxwodasvvf_wd640.jpg : jphide(*)
/mnt/tmp/3xuxkziawwgm_wd640.jpg : jphide(*)
/mnt/tmp/cr3xhirayrbn_wd1280.jpg : jphide(*)
/mnt/tmp/5smxen2ajuib_wd640.jpg : jphide(*)
/mnt/tmp/a6dxcqia90b6_wd640.jpg : jphide(*)
/mnt/tmp/pbhx1bhanypr_wd640.jpg : jphide(*)
/mnt/tmp/cr3xhirayrbn_wd640.jpg : jphide(*)
```

```
/mnt/tmp/vlmx4cyajwb9_std320.jpg : jphide(*)
/mnt/tmp/3jgxkj8abx8y_wd1280.jpg : jphide(*)
/mnt/tmp/vw0xy6pa9axu_wd640.jpg : jphide(***)
/mnt/tmp/pk1xawtar9gc_std320.jpg : jphide(***)
/mnt/tmp/pp4xqw3ankzq_wd640.jpg : jphide(*)
/mnt/tmp/no3x08kaf5pa_wd640.jpg : jphide(*)
/mnt/tmp/o22xi52afpo7_wd640.jpg : jphide(*)
/mnt/tmp/024x57kaxkmx_std320.jpg : jphide(***)
/mnt/tmp/xryxh44aqad8_wd640.jpg : jphide(*)
/mnt/tmp/ldexakeak48i_wd1280.jpg : jphide(**)
/mnt/tmp/y3mxyzca2l76_wd1280.jpg : jphide(*)
/mnt/tmp/ymhxt5raczew_std320.jpg : jphide(*)
/mnt/tmp/9z5xgoqa1iuv_std320.jpg : jphide(*)
/mnt/tmp/m1mxot4a7mys_std320.jpg : jphide(***)
/mnt/tmp/ckcx0ihat90i_wd640.jpg : jphide(*)
/mnt/tmp/6mkxznpa2rwh_std320.jpg : jphide(*)
/mnt/tmp/f0bx16ia3fj0_wd640.jpg : jphide(*)
/mnt/tmp/y1rx3abavx8c_std320.jpg : jphide(***)
/mnt/tmp/qcmxuszazd6e_wd640.jpg : jphide(*)
/mnt/tmp/njuxgrja2eiv_wd640.jpg : jphide(***)
/mnt/tmp/w7jxeuiakwo4_wd640.jpg : jphide(*)
/mnt/tmp/4wqxemia57kb_std320.jpg : jphide(*)
/mnt/tmp/a0bxd30apq2s_std320.jpg : jphide(*)
/mnt/tmp/8pyxcijathw4_std320.jpg : jphide(*)
/mnt/tmp/ynyxwo2ahw5m_wd640.jpg : jphide(*)
/mnt/tmp/pyrx6vka637z_wd640.jpg : jphide(*)
/mnt/tmp/yq4xlokadfhm_wd640.jpg : jphide(***)
```

7. Dave confessed that he used the steghide tool. Try to find out what file Dave hid.

Answer:

From assignment one, Dave password: XKCDWindmillh@ck, I extracted Dave RSA private key from image ikxx557aww99_wd1280.jpg to secret.key file:

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEArkedYkhHKHFmebENbVbaBgWyLwgGLGZImnBH7Qi5alAM2Qt+
qZj6IjsXoTiGFdGgUzxgqpZQLDOLN9RUblo4HD1qsntiP4f7vKf/vY+qYTRa42ZZ
dfBK+FxgDRmixYQ4Qmns2QwKNzv/AgfyDB7XgNZ999Lo8uXuCrI//wfqA8Xd3rqz
9PytCYjZVJHiZ3wq9TS2+ybN3IzGTuV2mWcyB+7z0Bar4kQBgGODzXY9QdT8ZLPO
TSpMMpGP3BNfetiGzknahnIP4woVS4yhNGq27qLOWy0ZZcpictUj5H1Ryn4r5vAl
6IhlH71icybCerBgchIwvz31coY5ASvYCwUB/QIDAQABAoIBAGNsn7DOIypDZ37N
LyfNgJwm67xBC6SQxaK1o5LqgzzPZkT8dcozZ5/XrmdfY79W0+woac1n610MGsRC
8dfyyEf0Eca49RfhcA/0l8WBDGZgA+nvHeJWr654pKNUmLOt9pqM6333jGxuYdC6
z69yeOcbdsnHHPZVjjOz9SZ9UVfMvDtExXQ/WfZ9B7IPaUiV1Z5uDvZNePJz1iHx
Nq7QMqETA/YvgJzVWsPuqvmoUlHiMt57q0dc4FU8VQehPYulcXSUmTptKLPRhV09
6kDLdpNBuKMYar8yMtVwNl6yq7kBVlEn6HMdTfKU/2Cx1vbp6z+RhdrVDgRvkrcI
```

```
1KgIIFECgYEA5NisL8h4XCaz2JC6GuFVMoT81W8Bh8nQ7vLBrx2vmzV9jrU6rDUG
lykjW0sSmqW3TahOOXDHUNezmVXcIyo/myBG72c+yKyuhb3JvqetyXB/+9cmx1jS
Prprm9xzM49H4FWNtNCs7YZ52BnLvGh1Fo+1GKaZFi1+lhdh/hgMQysCgYEAwvVz
0kBeNBaTZIUhIVSx16AKbx2vWBhl1mOfDOmwg1cS89FoPGUHyqE+k8KluUW6qwio
A4tX1jqnVboS8BYvRph4Ugbq84GpJOlabcYv76pIwY6B23YUavzVzJ6dO3wBhbJR
UB/JtlGtdzFjMBR4rcKBbcrXQtgKTIMcuyGU23cCgYEAqEM0H5IcBU2jsNmBLSB/
XzzvFhOfoXLff8HYbWS7aLik0BgqwtHePajOyWJilHjCVYQpuAxXUPa4pEbALM70
o5/Q6FgWjsCBNe789oUdv95LDCX+6lZBiEPTuW8W+VMhey4MmmVQsPjOf/k/lxGK
/gK+GhjsuKTMzZj1wTl3Uq8CgYBJ5lSS3AdZYz1Xmwcl5T7MZ0PNPslacVUY4QZH
FMXt4zGx7iy+x+UeL+TSibPb+Mx7THqzbTxMXktTuYa4LxCYh+8D2M9yojGFZlb6
yWceR8Pwap5am/W9YD2CpJUhGS5SiXc9Ee+aBnfkeHoKnZfo9ZOuFHdoRRASVJit
bltInQKBgHcLMTj53brjYqPgjBi2/Wd3rfWybkYqn7ASU0m7tRaT8Myq2XW3FPeJ
LMgyztO9xWskcyY+U6qxjEXD7ZCd9CFUK7V85oKU2kWbX4advH//VGRyIfAN8okM
STg9YOK60y+LzWW6HcNV9b32tQ0v/L06h36bFozJP1x6dZSVhWVE
-----END RSA PRIVATE KEY-----
```

8. Write a small paragraph of maximum 400 words about your approach and findings.

Answer:

As a findings, I found 40 .jpg images that contains possible hidden information using stegdetect tool. However, if check the output again we notice that 10 of these images has the highest indication (number of asterisks is 3) which indicate the confidence level. I tried Dave hard disk password from assignment one on the images, it turned up a file called ikxx557aww99_wd1280.jpg, that contains DAVE RSA private key in a file called “secret_key” (the head and trailer inside the file).

2 Filesystem

9. Read up on EXT4. Write a small paragraph of maximum 400 words answering the following questions:

(a) What is ext4’s on-disk layout?

Answer:

According to source 1: an ext4 file system is split into a series of block groups. To reduce performance difficulties due to fragmentation, the block allocator tries very hard to keep each file's blocks within the same group, thereby reducing seek times. The size of a block group is specified in sb.s_blocks_per_group blocks, though it can also calculated as 8 * block_size_in_bytes. With the default block size of 4KiB, each group will contain 32,768 blocks, for a length of 128MiB. The number of block groups is the size of the device divided by the size of a block group. All fields in ext4 are written to disk in little-endian order ext4 allocates storage space in units of “blocks”. A block is a group of sectors between 1KiB and 64KiB, and the number of sectors must be an integral power of 2. Blocks are in turn grouped into larger units called block groups. Block size is specified at mkfs time and typically is 4KiB. You may experience mounting problems if block size is greater than page size (i.e. 64KiB blocks on a i386 which only has 4KiB memory pages). By default a filesystem can contain 2^32 blocks; if the '64bit' feature is enabled, then a filesystem can have 2^64 blocks.

Disk layout:

1. Group 0 Padding (1024 bytes)
2. ext4 Super Block (1 block)
3. Group Descriptors (many blocks)
4. Reserved GDT Blocks (many blocks)
5. Data Block Bitmap (1 block)
6. inode Bitmap (1 block)
7. inode Table (many blocks)
8. Data Blocks (many more blocks)

(b) How does ext4 use of a log affect your work as a forensic investigator?

Answer:

The most interesting part in forensics investigation is the Super blocks ext4, it contains wealth amount of information about the file system (Cryptography in use, Operating system and so on) and also it contains location of the journal file which has interesting information in forensics (e.g. deleted files and so on).

Sources:

1- https://ext4.wiki.kernel.org/index.php/Ext4_Disk_Layout

2- https://www.dfrws.org/sites/default/files/session-files/paper-an_analysis_of_ext4_for_digital_forensics.pdf

10. Detect whether there is an encrypted container on the USB key. This can be done by calculating the entropy. Hint: binwalk

Answer:

First I will install binwalk (always in OS3 assignment follow the HINTS) and check:

```
root@caine:~/Desktop# apt install binwalk

root@caine:~/Desktop# binwalk -E usbkey.hdd --save

DECIMAL       HEXADECIMAL     ENTROPY
----------------------------------------------------------------------------
----
0             0x0             Falling entropy edge (0.025572)
8670208       0x844C00        Rising entropy edge (0.994581)
9549824       0x91B800        Rising entropy edge (0.996502)
12808192      0xC37000        Rising entropy edge (0.955464)
13033472      0xC6E000        Rising entropy edge (0.960487)
14025728      0xD60400        Rising entropy edge (0.993650)
16291840      0xF89800        Rising entropy edge (0.985669)
16686080      0xFE9C00        Rising entropy edge (0.997077)
25131008      0x17F7800       Falling entropy edge (0.818926)
25447424      0x1844C00       Rising entropy edge (0.995588)
31513600      0x1E0DC00       Rising entropy edge (0.995864)
33554432      0x2000000       Rising entropy edge (0.996959)
37748736      0x2400000       Rising entropy edge (0.995209)
41345024      0x276E000       Rising entropy edge (0.967035)
41682944      0x27C0800       Rising entropy edge (0.990215)
41795584      0x27DC000       Falling entropy edge (0.000000)
42224640      0x2844C00       Rising entropy edge (0.995003)
46137344      0x2C00000       Rising entropy edge (0.990564)
49135616      0x2EDC000       Falling entropy edge (0.477071)
```

As we can see above the entropy is really high (close to 1) at parts on the filesystem. Which possibly indicates encrypted files

11. Using a hex editor, can you detect that something is off?

Answer:

First, I checked all the filesystem for signatures:

```
root@caine:~/Desktop# binwalk usbkey.hdd | grep -v JPEG

DECIMAL       HEXADECIMAL     DESCRIPTION
----------------------------------------------------------------------------
----
0             0x0             Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
1160          0x488           Unix path: /home/mick/bin/mount
8388608       0x800000        Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
8389768       0x800488        Unix path: /home/mick/bin/mount
12890121      0xC4B009        Unix path: /home/mick/bin/lol
```

```
14671230      0xDFDD7E        Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
16186538      0xF6FCAA        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16223262      0xF78C1E        TIFF image data, big-endian, offset of first
image directory: 8
16223478      0xF78CF6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16228382      0xF7A01E        TIFF image data, big-endian, offset of first
image directory: 8
16228598      0xF7A0F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16233502      0xF7B41E        TIFF image data, big-endian, offset of first
image directory: 8
16233718      0xF7B4F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16237598      0xF7C41E        TIFF image data, big-endian, offset of first
image directory: 8
16237814      0xF7C4F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16262174      0xF8241E        TIFF image data, big-endian, offset of first
image directory: 8
16262390      0xF824F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16266270      0xF8341E        TIFF image data, big-endian, offset of first
image directory: 8
16266486      0xF834F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16313374      0xF8EC1E        TIFF image data, big-endian, offset of first
image directory: 8
16313590      0xF8ECF6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16347166      0xF9701E        TIFF image data, big-endian, offset of first
image directory: 8
16347382      0xF970F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16352286      0xF9841E        TIFF image data, big-endian, offset of first
image directory: 8
16352502      0xF984F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16359454      0xF9A01E        TIFF image data, big-endian, offset of first
```

```
image directory: 8
16359670      0xF9A0F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16393246      0xFA241E        TIFF image data, big-endian, offset of first
image directory: 8
16393462      0xFA24F6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16415774      0xFA7C1E        TIFF image data, big-endian, offset of first
image directory: 8
16415990      0xFA7CF6        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
16647180      0xFE040C        TIFF image data, little-endian offset of first
image directory: 8
16647438      0xFE050E        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:xmpMM="http
16648388      0xFE08C4        Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
16671756      0xFE640C        TIFF image data, little-endian offset of first
image directory: 8
16672014      0xFE650E        Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:xmpMM="http
18121758      0x114841E       TIFF image data, big-endian, offset of first
image directory: 8
18121974      0x11484F6       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
18158622      0x115141E       TIFF image data, big-endian, offset of first
image directory: 8
18158838      0x11514F6       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
18218014      0x115FC1E       TIFF image data, big-endian, offset of first
image directory: 8
18218230      0x115FCF6       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:about="uuid:faf5bdd5-ba3d-11da-ad31-d33d75182f1b" xmlns:dc="http://p
19194892      0x124E40C       TIFF image data, little-endian offset of first
image directory: 8
19195150      0x124E50E       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:xmpMM="http
25165824      0x1800000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
25166984      0x1800488       Unix path: /home/mick/bin/mount
28935550      0x1B9857E       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
```

```
31194142      0x1DBFC1E       TIFF image data, big-endian, offset of first
image directory: 8
31198876      0x1DC0E9C       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://
31204096      0x1DC2300       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
33285150      0x1FBE41E       TIFF image data, big-endian, offset of first
image directory: 8
33300483      0x1FC2003       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:xap="htt
33304451      0x1FC2F83       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
35638302      0x21FCC1E       TIFF image data, little-endian offset of first
image directory: 8
35638764      0x21FCDEC       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://
37035038      0x2351C1E       TIFF image data, big-endian, offset of first
image directory: 8
37053192      0x2356308       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://
37058015      0x23575DF       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
39064606      0x254141E       TIFF image data, little-endian offset of first
image directory: 8
39065080      0x25415F8       Copyright string: "copyright."
39066004      0x2541994       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
39079986      0x2545032       Copyright string: "Copyright (c) 1998 Hewlett-
Packard Company"
39090627      0x25479C3       Unix path:
/www.w3.org/1999/02/22-rdf-syntax-ns#"> <rdf:Description rdf:about=""
xmlns:xmp_1_="http://ns.abobe.com/xap/1.0/" xmlns:aux="htt
39092507      0x254811B       Copyright string: "CopyrightFlag="true"
photoshop:ColorMode="3" photoshop:ICCProfile="Adobe RGB (1998)"
dc:format="image/jpeg" xmpMM:InstanceID="xm"
39096343      0x2549017       Copyright string: "copyright.</rdf:li>
</rdf:Alt> </dc:rights> <dc:description> <rdf:Alt> <rdf:li xml:lang="x-
default">Colombia, Bolivar, Cartagena"
39100666      0x254A0FA       Copyright string: "Copyright 1999 Adobe
Systems Incorporated"
41943040      0x2800000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
41944200      0x2800488       Unix path: /home/mick/bin/mount
43941918      0x29E801E       TIFF image data, little-endian offset of first
image directory: 8
50335744      0x3001000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
```

```
50336904      0x3001488       Unix path: /home/mick/bin/mount
50341888      0x3002800       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
50343048      0x3002C88       Unix path: /home/mick/bin/mount
58720256      0x3800000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
58721416      0x3800488       Unix path: /home/mick/bin/mount
75497472      0x4800000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=df8d5c63-b78c-4237-b637-6a4f99579957
75498632      0x4800488       Unix path: /home/mick/bin/mount
104857600     0x6400000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=47b4aed5-1915-4762-9a43-45e9f692f692
104858760     0x6400488       Unix path: /home/mick/lab3/mount
105252864     0x6460800       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=47b4aed5-1915-4762-9a43-45e9f692f692
105254024     0x6460C88       Unix path: /home/mick/lab3/mount
113246208     0x6C00000       Linux EXT filesystem, rev 1.0, ext4 filesystem
data, UUID=47b4aed5-1915-4762-9a43-45e9f692f692
```

Found this **/home/mick/bin/mount** which a signature lead to a hidden mounted container.

Also saw that on hex output:

```
00000480: 0000 0000 0000 0000 2f68 6f6d 652f 6d69  ......../home/mi
00000490: 636b 2f62 696e 2f6d 6f75 6e74 0000 0000  ck/bin/mount....
```

Which contains same values.

12. Try to find out what happened to the filesystem.

Answer:

I extracted everything from usbkey.hdd:

```
caine@caine:~/Desktop$ binwalk -Me usbkey.hdd

caine@caine:~/Desktop/_usbkey.hdd.extracted$ ls
0.ext        3001000.ext  4800000.ext  6C00000.ext  ext-root-0
1800000.ext  3002800.ext  6400000.ext  800000.ext
2800000.ext  3800000.ext  6460800.ext  ext-root

caine@caine:~/Desktop/_usbkey.hdd.extracted/ext-root-0$ ls
usbkey.hdd.md5  usbkey.hdd.sha1  usbkey.hdd.sha2
```

I noticed that the original hashes value are different.

here:

```
caine@caine:~/Desktop/_usbkey.hdd.extracted/ext-root-0$ cat usbkey.hdd.sha2
ff100e4e632738b590578d22a0df6591bdff72ff0a0c77405e3cd917ca122e56  usbkey.hdd
```

Original:

```
caine@caine:~/Desktop$ cat usbkey.hdd.sha2
1481d1633dfff916b54bf55647f1085a2b981d01de5df8c250bd07cafbde396e  usbkey.hdd
```

Which is something questionable and need further investigations.

I got help from colleague on this

13. Write a small paragraph of maximum 200 words about your findings.

Answer:

There was a EXT4 filesystem configured on usbkey.hdd, we found images inside, on of these images contains Dave RSA private key inside file “secret.key”. However, we couldn't guaranty that the secret.key contains a real private key, but we based our assumption on the signature in the header and trailer. Extracting data from usbkey.hdd produced hashes of the file. Finally, the entropy calculation, shows that the filesystem contains data with a high entropy which indicates the possibility to find encrypted information. In addition to that, we can see an existence of another file system if we look at signature from output from Binwalk.