

LUKS Full Disk Encryption Upside-Down

Where full disk encryption do not do what you expected



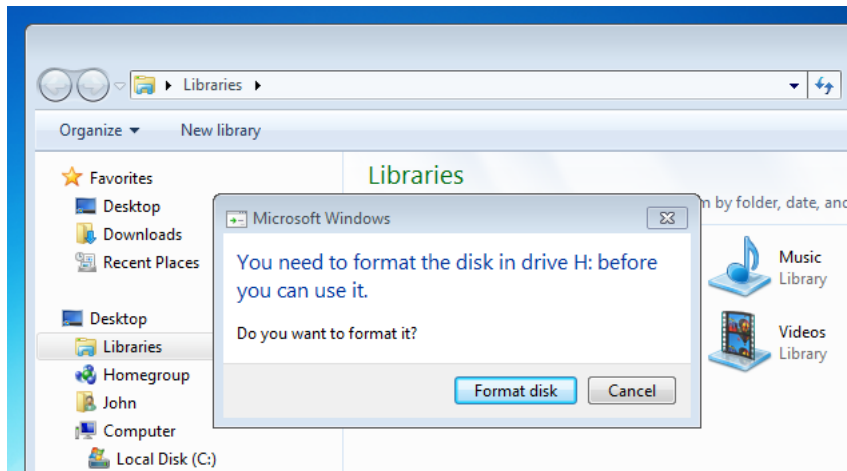
CIRCL

Computer Incident
Response Center
Luxembourg

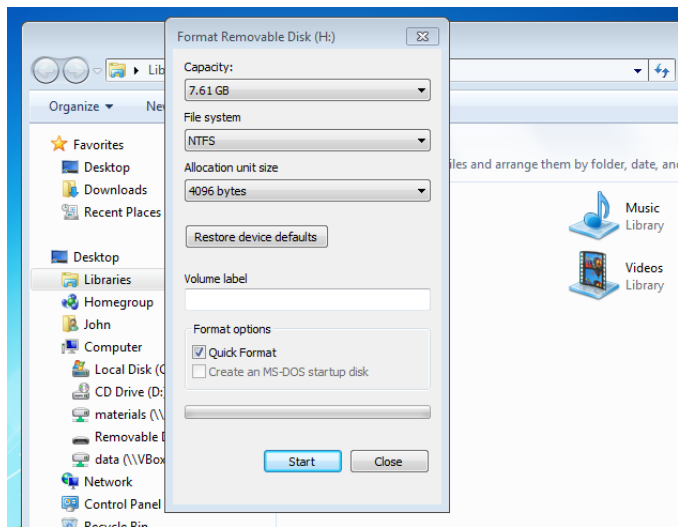
Michael Hamm

TLP:CLEAR

1. Create NTFS partition



1. Create NTFS partition



2 Create plaintext data



Name **8AC81CEBC81CD773**

Type **Folder**

Contents **202 items, totalling 5,0 GB**

Parent folder **/media/michael**

Volume **8,2 GB Volume**

Modified **Mi 16 Okt 2024 13:25:06**

Created **Mi 16 Okt 2024 13:25:06**



5,1 GB used

3,1 GB free

Total capacity **8,2 GB**

Filesystem type

Open in Disks

2 Create plaintext data

Create 100 very small files:

```
smallFiles
    00.txt      48 Byte
    01.txt      48 Byte
    ....
    ....
    99.txt      48 Byte
```

Create 100 large files:

```
bigFiles
    00.txt      48 MByte
    01.txt      48 MByte
    ....
    ....
    99.txt      48 MByte
```

Content:

```
My small secret message is 01234567890987654321
```

Hide content somewhere:

```
echo -n "My small secret message is 01234567890987654321" | dd of=/dev/sda seek=10
```

3 Re-format the disk

Create partition table (default parameter):

```
umount /dev/sda
fdisk /dev/sda

                d d d d
                n
                w

fdisk -l /dev/sda

    Device      Boot Start          End  Sectors  Size Id Type
    /dev/sda1                2048 15974399 15972352   7,6G 83 Linux
```

Activate full disk encryption:

```
cryptsetup luksFormat /dev/sda1

WARNING!
=====
This will overwrite data on /dev/sda1 irrevocably.

Are you sure? (Type 'yes' in capital letters): YES
```

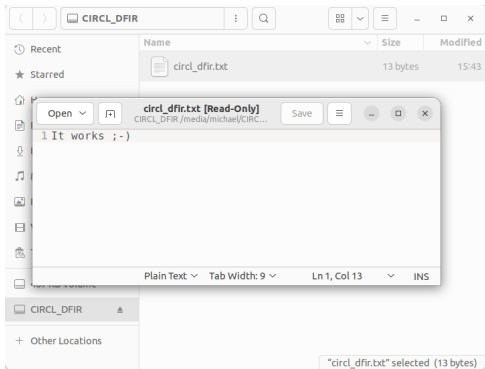
Create file system:

```
cryptsetup luksOpen /dev/sda1 myusb
mkfs.ext4 /dev/mapper/myusb -L CIRCL_DFIR
cryptsetup luksClose myusb
```

3 Re-format the disk

Mount encrypted partition and create a file:

```
echo "It works ;-)" > /media/michael/CIRCL_DFIR/circl_dfir.txt
```



Unmount encrypted partition:

```
umount /media/michael/CIRCL_DFIR/  
cryptsetup luksClose luks-01b0d6a2-.....
```

4 Launch investigation

Create a forensic copy:

```
dd if=/dev/sda of=2_ext4.raw bs=$((4*4096)) status=progress
```

Read partition table:

```
fdisk -l 2_ext4.raw
```

Device	Boot	Start	End	Sectors	Size	Id	Type
2_ext4.raw1		2048	15974399	15972352	7,6G	83	Linux

Analyze the MBR:

```
dd if=2_ext4.raw count=1 | xxd | less
```

```
00000000: eb52 904e 5446 5320 2020 2000 0208 0000  .R.NTFS      .....
00000010: 0000 0000 00f8 0000 3f00 ff00 0000 0000  .....?.....
.....
.....
000001b0: 4752 2069 7320 6d69 7373 696e 6700 0021  GR is missing..!
000001c0: 0300 836b e6fe 0008 0000 00b8 f300 0000  ...k.....
000001d0: 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001e0: 0000 0000 0000 0000 0000 0000 0000 0000  .....
000001f0: 0000 0000 0000 0000 0000 0000 0000 55aa  .....U.
```


4 Launch investigation

What about our hidden message at sector 11:

```
dd if=2_ext4.raw skip=10 count=1 | xxd | less
00000000: 4d79 2073 6d61 6c6c 2073 6563 7265 7420  My small secret
00000010: 6d65 7373 6167 6520 6973 2030 3132 3334  message is 01234
00000020: 3536 3738 3930 3938 3736 3534 3332 3100  567890987654321.
00000030: 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000040: 0000 0000 0000 0000 0000 0000 0000 0000  .....
.....
.....
```

Analyze the 1th sector of the partition:

```
dd if=2_ext4.raw skip=2048 | xxd | less
00000000: 4c55 4b53 babe 0002 0000 0000 0000 4000  LUKS.....@.
00000010: 0000 0000 0000 0003 0000 0000 0000 0000  .....
00000020: 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000030: 0000 0000 0000 0000 0000 0000 0000 0000  .....
00000040: 0000 0000 0000 0000 7368 6132 3536 0000  ..... sha256..
.....
.....
```

4 Launch investigation

Scroll down to the first data:

```
dd if=2_ext4.raw skip=2048 | xxd | less

00008000: 5144 6c6c 2d71 d434 47cf 31c3 ba28 92b1 QDl1-q.4G.1..(..
00008010: 3c44 bacb 4ea2 29b6 0a13 dda0 f846 f563 <D..N.).....F.c
00008020: 5e6c eda0 1dcc a58d 5202 edda 47d7 cf34 ^l.....R...G..4
00008030: ab02 7e8b 3c4b acb2 ecf3 3911 29e6 453f ..~.<K....9.)E?
00008040: d222 c53e 1d1f b959 6ac6 f789 495d 75fd ."..>...Yj...I]u.
.....
.....
```

Scroll down \approx 52 MByte:

```
dd if=2_ext4.raw skip=2048 | xxd | less

033e3fd0: d834 f1e0 844a 0dc8 d900 41f2 3142 a20f .4...J....A.1B..
033e3fe0: 0e7c b383 faca 8549 0ed9 a30b cd6e 7d13 .|....I.....n}.
033e3ff0: bc82 a73b 9078 67a5 db05 61b8 bd68 77fc ...;xg...a..hw.
033e4000: 4d79 2073 6d61 6c6c 2073 6563 7265 7420 My small secret
033e4010: 6d65 7373 6167 6520 6973 2030 3132 3334 message is 01234
033e4020: 3536 3738 3930 3938 3736 3534 3332 310a 567890987654321.
033e4030: 4d79 2073 6d61 6c6c 2073 6563 7265 7420 My small secret
.....
```

5 Conclusion

- MBR is not encrypted
- Space outside the partition is not encrypted
- Other partitions on the disk are not encrypted
- Misleading term: Full Disk Encryption

- Encryption is applied in 52 MByte chunks
- Encrypted area is growing with more data stored over time
- Without wiping old data stay on the disk until overwritten

- You need to wipe before using Full Disk Encryption