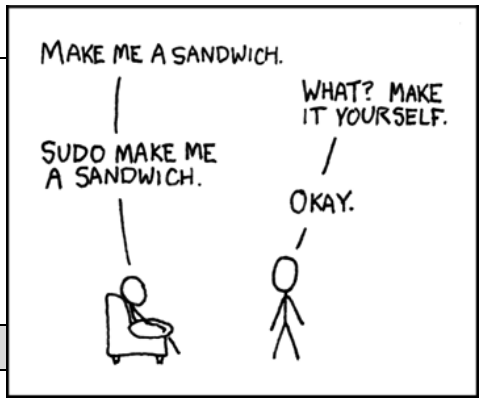
Linux-commands for investigators



Do not forget the «sudo» command if you need elevated rights.

Read the manual if you are lost: `man program`

Disk management:

```
fdisk -l
```

Shows all disks and partitions recognized by the OS.

```
df -h
```

Shows you mounted disks and their mount points with human readable sizes.

```
mkdir evidence
```

Creates the folder «evidence»

```
mount -o ro /dev/sdb1 evidence
```

Mounts the partition sdb1 read only (-o ro) in the folder “evidence”.

Note: For imaging only, you don’t need to mount the disk!

```
mount /dev/sdc1 storage
```

Mounts the partition sdc1 writeable in the folder “storage”

```
cd storage
dcfldd if=/dev/sdb conv=noerror,sync hash=md5 hashlog=123456.md5 of=123456.dd
```

Enter the folder «storage».

Image the disk **sdb** and create a md5-hash. Hash is stored in the file 123456.md5, while image is stored as 123456.dd

```
fdisk -lu badguy.dd
```

Gives an overview of partitions in a disk image. startposition = sector * sectorsize in bytes

```
mount -o ro,offset=$((48*512)) badguy.dd /mnt/xp/
mount -o ro,offset=24576 badguy.dd /mnt/xp/
```

Mounts a partition from a diskimage read-only

Sleuthkit:

```
mmls badguy.dd
```

Shows disk layout like fdisk – but also with gaps

```
fsstat -o 48 badguy.dd
```

Info about file system starting at sect. 48

```
fls -o 48 -r -d
  +++ r/r 5866-128-3:    Stonehenge.jpg
icat -r -o 48 badguy.dd 5866-128-3 > Stonehenge.jpg
```

Shows recursively deleted files

One of the results from last command

Extracts the deleted file

```
fls -o 48 -r -m "C:/" bad-guy.dd > fillayout.badguy.txt
```

Creates a basis for a time line, and sets the partition to C:/ . NB! Not backslash «/» because of compatibility…

```
mactime -b fillayout.badguy.txt > timeline.badguy.txt
```

Creates the timeline.

Text and strings:

```
strings –t d  largfile.bin
```

Finds all text from a large file and shows hit with the offset to the string in decimal

```
strings -n 8 –e S memdump.dd > pass.txt
```

Finds all strings longer than 8 characters from a memorydump and store this to the file pass.txt

```
cat tekst.txt | grep hob | grep -v hobbit  | awk '{print $2}' | sort | uniq > newtext.txt
```

lists the content of the file tekst.txt, pics only lines with the word «hob», omits lines with the word «hobbit», prints column 2 of the text, sorts this alfabetical, keeps only uniqe ones and stores all in the file newtext.txt

```
cat tekst.txt | egrep -o '[^[:space:]]+\>@[a-zA-Z_\.]+\.[a-zA-Z]{2,3}' | sort | uniq -c
```

finds all mailadresses from a file, sorts them and shows only unique ones. The switch «-c» enables counting, describing how many of each was found.

Media:

```
mplayer -ao null -frames 10 -sstep 60 -vo jpeg .  filmnavn.avi
```

makes 10 screendumps with 60 seconds between each. Notice the "." before the file name; describes where pictures should be stored.

```
find /mnt/xp -iname *.avi -exec mplayer -frames 200 -ss 20 '{}' \;
```

Finds recursively all files ending with avi in the folder /mnt/xp, and shows 200 frames from each film, 20 seconds into the film

```
exiftool –CreateDate bilde.jpg
```

Shows the date a picture was taken, from the Exif information

Other tools

```
shred –n 1 -v -u /media/disk/filename
```

Securely deletes 1 file

```
shred –n 1 -v –z /dev/sdb2
```

Deletes and wipes a partition, but could also be used on a disk. The switch «-z» overwrites everything with zeroes at the end..

```
foremost –t jpg –o pictures badguy.dd
```

Carves all pictures of the type jpg and puts them in the folder «pictures».

```
apt-cache search exiftool
```

Search for packets containing the word "exiftool"

```
apt-cache show libimage-exiftool-perl
```

Show info about the packet you got as a result from the previous command

```
apt-get install libimage-exiftool-perl
```

Install the program tha contains «exiftool»

```
./pf32 –v NOTEPAD++.EXE-76BDB3.pf
```

Runs the program «pf32» downloaded from tzworks.net against the prefetch-file NOTEPAD++.EXE-76BDB3.pf