

Een volledige backup maken van mijn computer laptopca, host=ulefr01-server

Laatst verwerkt op 02-01-15 11:03:08 CET door gebruiker ulefr01

Index

1. [Mijn partitietabel](#)
2. [Al mijn procedures](#)
 1. [procedure t00](#)
 2. [procedure t01](#)
 3. [procedure t02](#)
 4. [procedure t03](#)
 5. [procedure t03-end](#)
 6. [procedure mkout_to-html](#)
 7. [procedure bckupmbr](#)
 8. [procedure tune part](#)
 9. [procedure tuneLV](#)
 10. [procedure tune_ext4](#)
 11. [procedure tune_ext3](#)
 12. [procedure tune_ext2](#)
 13. [procedure bckupdick](#)
 14. [procedure bckupntfs](#)
 15. [procedure archive-start](#)
 16. [procedure archive-mint](#)
 17. [procedure archive-end](#)
3. [Al mijn gebruikte commandos](#)
 1. [command dislocker](#)
 2. [command losetup](#)
 3. [command ntfscione](#)
 4. [command vgchange](#)
 5. [command vgscan](#)
 6. [command vgextend](#)
 7. [command pvs](#)
 8. [command lvs](#)
 9. [command dd](#)
 10. [command sfdisk](#)
 11. [command fsarchiver](#)
 12. [command tune2fs](#)
 13. [command e2fsck](#)
 14. [command ntfsinfo](#)

15. [command ntfsfix](#)

4. [Het uiteindelijke resultaat](#)

5. [De uitvoering in details](#)

Partitietabel /dev/sda

```
# partition table of /dev/sda
unit: sectors
```

```
/dev/sda1 : start=      2048, size=650117120, Id=83
/dev/sda2 : start=650119168, size= 53163440, Id=83
/dev/sda3 : start=        0, size=        0, Id= 0
/dev/sda4 : start=        0, size=        0, Id= 0
```

Partitietabel /dev/sdb

```
# partition table of /dev/sdb
unit: sectors
```

```
/dev/sdb1 : start=        1, size= 3953663, Id= c, bootable
/dev/sdb2 : start=        0, size=        0, Id= 0
/dev/sdb3 : start=        0, size=        0, Id= 0
/dev/sdb4 : start=        0, size=        0, Id= 0
```

Gebruikte bash/shell procedures

bash shell script t00

```
#!/bin/bash
#
# only run as root
```

```

RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
echo "Terminal 00" | tee term_00.out
echo "-----" | tee -a term_00.out
(date) | tee -a term_00.out
#
pwd
#
echo "vgchange -a y " | tee -a term_00.out
(vgchange -a y) | tee -a term_00.out
#
# echo "./laptopca-t00 linux linux_boot linux_root linux_home "
# (./laptopca-t00 linux linux_boot linux_root linux_home 2>&1) | tee term_00.out
#
echo "bckupmbr devices" | tee -a term_00.out
echo "-----" | tee -a term_00.out
(./bckupmbr sda 2>&1) | tee -a term_00.out
(./bckupmbr sdb 2>&1) | tee -a term_00.out
(./bckupmbr sdc 2>&1) | tee -a term_00.out
(./bckupmbr sdd 2>&1) | tee -a term_00.out
(date) | tee -a term_00.out
echo "tune partitions" | tee -a term_00.out
(./tunepart /dev/sda1 dlck 2>&1) | tee -a term_00.out
(./tunepart /dev/sda3 ntfs 2>&1) | tee -a term_00.out
(./tunepart /dev/sda4 ntfs 2>&1) | tee -a term_00.out
(./tuneLV linux linux_boot ext4 2>&1) | tee -a term_00.out
(./tuneLV linux linux_root ext4 2>&1) | tee -a term_00.out
(./tuneLV linux linux_home ext4 2>&1) | tee -a term_00.out
(./tuneLV linux linux_data ext4 2>&1) | tee -a term_00.out
# (./tuneLV linux linux_boot ext4 2>&1) | tee -a term_00.out
RES=0
echo "-----" | tee -a term_00.out
exit ${RES}
#

```

bash shell script t01

```

#!/bin/bash
#
# -----

```

```
PATH="${PATH}:/usr/sbin:/sbin:/usr/bin:/bin"
TIMESTAMP="$(date +%Y%m%d-%Hh%M)"

# only run as root
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
echo "Terminal 01" | tee term_01.out
echo "-----" | tee -a term_01.out
(date) | tee -a term_01.out
# | tee term_01.out
# | tee -a term_01.out
echo "(./bckupdlck backup-C /dev/sda1 " | tee -a term_01.out
(./bckupdlck /dev/sda1 backup-C 2>&1) | tee -a term_01.out
(date) | tee -a term_01.out
RES=0
exit ${RES}
```

bash shell script t02

```
#!/bin/bash
#
# -----

PATH="${PATH}:/usr/sbin:/sbin:/usr/bin:/bin"
TIMESTAMP="$(date +%Y%m%d-%Hh%M)"

# only run as root
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
echo "Terminal 02" | tee term_02.out
echo "-----" | tee -a term_02.out
(date) | tee -a term_02.out
# | tee term_02.out
# | tee -a term_02.out
echo "(./bckupntfs /dev/sda3 backup-D " | tee -a term_02.out
```

```
(./bckupntfs /dev/sda3 backup-D 2>&1) | tee -a term_02.out
echo "(./bckupntfs /dev/sda4 backup-E " | tee -a term_02.out
(./bckupntfs /dev/sda4 backup-E 2>&1) | tee -a term_02.out
(date) | tee -a term_02.out
RES=0
exit ${RES}
```

bash shell script t03

```
#!/bin/bash
#
# only run as root
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
#
echo "Terminal 3" | tee term_03.out
echo "-----" | tee -a term_03.out
(date) | tee -a term_03.out
#
pwd
#
(./archive-start linux /dev/sdc4 2>&1) | tee -a term_03.out
(date) | tee -a term_03.out
(./archive-mint linux linux_boot boot 2>&1) | tee -a term_03.out
(date) | tee -a term_03.out
(./archive-mint linux linux_root root 2>&1) | tee -a term_03.out
(date) | tee -a term_03.out
(./archive-mint linux linux_home home 2>&1) | tee -a term_03.out
(date) | tee -a term_03.out
# (./archive-mint linux linux_data data 2>&1) | tee -a term_03.out
# (date) | tee -a term_03.out
(./archive-end linux /dev/sdc4 2>&1) | tee -a term_03.out
#
(date | tee -a term_03.out)
RES=0
echo "-----"
exit ${RES}
#
```

bash shell script t03-end

```
#!/bin/bash
#
# only run as root
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
#
#
echo "Terminal 03_end" | tee term_03-end.out
echo "-----" | tee -a term_03-end.out
DATE="$(date +%Y%m%d)"
DAT2="$(date --date="yesterday" +%Y%m%d)"
(date) | tee -a term_03-end.out
#
pwd
#
(ls *${DATE}* -la --sort=n) | tee term_-a 03-end.out
# (ls *${DAT2}* -la --sort=n) | tee -a term_03-end.out
#
echo "term_00" | tee -a term_03-end.out
cat term_00.out | grep -o '[0-9][0-9]:[0-9][0-9]:[0-9][0-9]' | tee -a term_03-end.out
echo "term_01" | tee -a term_03-end.out
cat term_01.out | grep -o '[0-9][0-9]:[0-9][0-9]:[0-9][0-9]' | tee -a term_03-end.out
echo "term_02" | tee -a term_03-end.out
cat term_02.out | grep -o '[0-9][0-9]:[0-9][0-9]:[0-9][0-9]' | tee -a term_03-end.out
echo "term_03" | tee -a term_03-end.out
cat term_03.out | grep -o '[0-9][0-9]:[0-9][0-9]:[0-9][0-9]' | tee -a term_03-end.out
echo "-----"

cat term_00.out term_01.out term_02.out term_03.out term_03-end.out >laptopca-${DATE}.out
RES=0
#libreoffice --convert-to pdf --writer laptopca-${DATE}.out
#
echo "-----"
exit ${RES}
#
```

bash shell script mkout_to-html

```
#!/bin/bash
#
# make PDF
ARGS=1
E_BADARGS=65
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` outfile"
    exit $E_BADARGS
fi
outfile=$1
echo 'make PDF'
echo '-----'
echo "outfile= $outfile"
libreoffice --convert-to html --writer $outfile
echo 'gedaan'
exit 0
```

bash shell script bckupmbr

```
#!/bin/bash
#
# -----
#
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
ARGS=1
E_BADARGS=65
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` device "
    exit $E_BADARGS
fi

device=$1
# -----
echo "bckupmbr device =" $device
```

```

echo "-----"

DATE="$(date +%Y%m%d)"
date

# backup mbr ${device}
echo "doe mv ${device}.mbr ${device}_${DATE}.mbr"
mv ${device}.mbr ${device}_${DATE}.mbr
echo "doe mv ${device}65.mbr ${device}65_${DATE}.mbr"
mv ${device}65.mbr ${device}65_${DATE}.mbr
echo "save eerste sector"
echo "doe dd if=/dev/${device} of=${device}.mbr count=1 bs=512"
dd if=/dev/${device} of=${device}.mbr count=1 bs=512
echo "save 65 eerste sectoren"
echo "doe dd if=/dev/${device} of=${device}65.mbr count=65 bs=512"
dd if=/dev/${device} of=${device}65.mbr count=65 bs=512
#
echo "save printout partitietabel"
echo "doe mv parttbl${device}.sf parttbl${device}_${DATE}.sf"
mv parttbl${device}.sf parttbl${device}_${DATE}.sf
echo "doe sfdisk -d /dev/${device} >parttbl${device}.sf"
sfdisk -d /dev/${device} >parttbl${device}.sf
#
date
#
RES=0
echo "-----"
exit ${RES}
#

```

bash shell script tunepart

```

#!/bin/bash
#
# -----
#
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
ARGS=2
E_BADARGS=65

```



```
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` device type "
    exit $E_BADARGS
fi

device=$1
type=$2
# -----
echo "tunepart"
echo "-----"
echo "tunepart " $device " type " $type
DATE="$(date +%Y%m%d)"
date

case "$2" in
'ntfs')
    echo "tune ntfs"
    echo "doe ntfsinfo -m " $device
    ntfsinfo -m $device
    echo "doe ntfsfix " $device
    ntfsfix $device
;;
'dlck')
    echo "dislocker"
    mkdir /mnt/tmp
    echo "dislocker -v -V " $device " -pxyz /mnt/tmp"
    dislocker -v -V $device -pxyxyxy-xyxyxy-xyxyxy-xyxyxy-xyxyxy-xyxyxy-xyxyxy-xyxyxy /mnt/tmp
    echo "losetup /dev/loop1 /mnt/tmp/dislocker-file"
    losetup /dev/loop1 /mnt/tmp/dislocker-file
    echo "doe ntfsinfo encrypted partitie " $device
    ntfsinfo -m /mnt/tmp/dislocker-file
;;
esac
#
date
#
RES=0
echo "-----"
exit ${RES}
#
```

bash shell script tuneLV

```
#!/bin/bash
#
# -----
#
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
ARGS=2
E_BADARGS=65
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` volumegroup lv type "
    exit $E_BADARGS
fi

volumegroup=$1
lv=$2
type=$3
# -----
echo "tuneLV"
echo "-----"
echo "tuneLV " $volumegroup " lv " $lv " type " $type
DATE="$(date +%Y%m%d)"
date

case $type in
'ext4')
echo "tuneLV ext4"
(./tune_ext4 /dev/mapper/${volumegroup}-${lv})
;;
'ext3')
echo "tuneLV ext3"
(./tune_ext3 /dev/mapper/${volumegroup}-${lv})
;;
'ext2')
echo "tuneLV ext2"
(./tune_ext2 /dev/mapper/${volumegroup}-${lv})
;;
esac
#
date
#
```

```
RES=0
echo "-----"
exit ${RES}
```

bash shell script tune_ext4

```
#!/bin/bash
# tune_ext4
ARGS=1
E_BADARGS=65
RES=1
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` device"
    exit $E_BADARGS
fi
device=$1
echo 'tune_ext4'
echo '-----'
echo "device= $device"
echo "doe : tune2fs -O has_journal $device"
# verwijder de journal en maak er een ext2 bestandensysteem
tune2fs -O ^has_journal $device
echo "doe : e2fsck -f $device"
# check ext2 bestandensysteem op fouten en herstel ze
e2fsck -f $device
echo "doe : tune2fs -J size=4 $device"
# hermaak journal en maak ext3
tune2fs -J size=4 $device
echo "doe : tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 $device"
# maak een ext3 -6m om de 6 maanden hercheck bestandensysteem of -c 200 200 keren mounten vooraleer hercheck
tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 $device
echo "doe : e2fsck -pFD $device"
# maak een ext4 bestandensysteem
e2fsck -pFD $device
RES=0
echo 'gedaan !'
echo "device= $device"
exit ${RES}
```

bash shell script tune_ext3

```
#!/bin/bash
# tune_ext3
ARGS=1
E_BADARGS=65
RES=1
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` device"
    exit $E_BADARGS
fi
device=$1
echo 'tune_ext3'
echo '-----'
echo "device= $device"
echo "doe : tune2fs -O has_journal $device"
# verwijder de journal en maak er een ext2 bestandensysteem
tune2fs -O ^has_journal $device
echo "doe : e2fsck -f $device"
# check ext2 bestandensysteem op fouten en herstel ze
e2fsck -f $device
echo "doe : tune2fs -J size=4 -i 6m -c 200 $device"
# hermaak journal en maak ext3
tune2fs -J size=4 -i 6m -c 200 $device
RES=0
echo 'gedaan !'
echo "device= $device"
exit ${RES}
```

bash shell script tune_ext2

```
#!/bin/bash
# tune_ext2
ARGS=1
E_BADARGS=65
RES=1
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` device"
    exit $E_BADARGS
fi
device=$1
```

```
echo 'tune_ext2'
echo '-----'
echo "device= $device"
echo "doe : e2fsck -f $device"
# check ext2 bestandensysteem op fouten en herstel ze
e2fsck -f $device
RES=0
echo 'gedaan !'
echo "device= $device"
exit ${RES}
```

bash shell script bckupdlck

```
#!/bin/bash
#
# -----
#
RES=1
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
ARGS=2
E_BADARGS=65
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` device backnam "
    exit $E_BADARGS
fi
TIMESTAMP="$(date +%Y%m%d-%H%M)"
device=$1
backnam=$2
# -----
echo "bckupdlck"
echo "-----"

date
# dislocker
echo "dislocker -v -V ${device} -pxxxxxxx /mnt/tmp"
dislocker -v -V ${device} -p640717-239745-376475-360107-404932-570207-527956-490567 /mnt/tmp
# mount -ovloop,rw /mnt/tmp/dislocker-file /mnt/winc# sudo mount -ovloop,rw /mnt/tmp/dislocker-file /mnt/winc
echo "losetup /dev/loop1 /mnt/tmp/dislocker-file"
losetup /dev/loop1 /mnt/tmp/dislocker-file
```

```
# use ntfscclone
echo "ntfscclone " $device
ntfscclone --save-image --output - /dev/loop1 | split -b 2000M - ${backnam}-${TIMESTAMP}.img
# | split -b 2000m
#
echo "losetup -d /dev/loop1"
losetup -d /dev/loop1
echo "umount /mnt/tmp"
umount /mnt/tmp
#
date
#
RES=0
echo "-----"
exit ${RES}
#
```

bash shell script bckupntfs

```
#!/bin/bash
#
# -----
#
RES=1
FSAOPTS='-z7 -j3 -s 2000'          # options to pass to fsarchiver
#
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit ${RES}
fi
ARGS=2
E_BADARGS=65
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` device backnam "
    exit $E_BADARGS
fi
TIMESTAMP="$(date +%Y%m%d-%H%M)"
device=$1
backnam=$2
# -----
echo "fsarchiver savefs -L${backnam} ${FSAOPTS} ${backnam}-${TIMESTAMP}.fsa ${device}"
# main command of the script that do the real stuff
```

```
if fsarchiver savefs -L${backnam} ${FSAOPTS} ${backnam}-${TIMESTAMP}.fsa ${device}
then
    echo "fsarchiver done"
    RES=0
else
    echo "fsarchiver failed"
    RES=1
fi
date
#
RES=0
echo "-----"
exit ${RES}
#
```

bash shell script archive-start

```
#!/bin/bash
#
ARGS=2
E_BADARGS=65
RES=1
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` volgroup device"
    exit $E_BADARGS
fi
VOLGROUP=$1                # name of volume group
device=$2                  # vrije LV
# -----

PATH="${PATH}:/usr/sbin:/sbin:/usr/bin:/bin"
TIMESTAMP="$(date +%Y%m%d-%Hh%M)"

# only run as root
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit 1
fi

# check the volumegroups
vgscan
#
```

```
#vgreduce $VOLGROUP $device
#
vgextend $VOLGROUP $device
RES=0
exit ${RES}
```

bash shell script archive-mint

```
#!/bin/bash
#
SNAPVOL='mysnap'           # name of the snapshot to create
SNAPSIZ='6G'               # space to allocate for the snapshot in the volume group
FSAOPTS='-z7 -j3 -s 2000 -A' # options to pass to fsarchiver
# -----

PATH="${PATH}:/usr/sbin:/sbin:/usr/bin:/bin"
TIMESTAMP="$(date +%Y%m%d-%H%M)"

ARGS=3
E_BADARGS=65
if [ $# -lt "$ARGS" ]
then
    echo "Usage: `basename $0` volumegroup logicalvolume backnam "
    exit $E_BADARGS
fi

RES=1
volumegroup=$1
logvol=$2
backnam=$3

# -----
echo 'archive-mint'
echo '-----'
echo "volumegroup =" $volumegroup
echo "logical volume =" $logvol
echo "backnam =" $backnam
echo '-----'
# -----

# only run as root
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
```



```
        exit 1
fi
#
# check that the snapshot does not already exist
if [ -e "/dev/${volumegroup}/${SNAPVOL}" ]
then
    echo "the lvm snapshot already exists, please destroy it by hand first"
    exit 1
fi

# create the lvm snapshot
if ! lvcreate -L${SNAPSIZ} -s -n ${SNAPVOL} /dev/${volumegroup}/${logvol} >/dev/null 2>&1
then
    echo "creation of the lvm snapshot failed"
    exit 1
fi

echo "fsarchiver savefs -L${logvol} ${FSAOPTS} ${backnam}-${TIMESTAMP}.fsa /dev/mapper/${volumegroup}-${logvol}"
# main command of the script that do the real stuff
if fsarchiver savefs -L${logvol} ${FSAOPTS} ${backnam}-${TIMESTAMP}.fsa /dev/mapper/${volumegroup}-${logvol}
then
    echo "fsarchiver done"
# md5sum overbodig
#    echo "start md5sum"
#    md5sum ${BACKDIR}/${BACKNAM}-${TIMESTAMP}.fsa > ${BACKDIR}/${BACKNAM}-${TIMESTAMP}.md5
    RES=0
else
    echo "fsarchiver failed"
    RES=1
fi

#
sync
#
#
echo "fsarchiver archinfo ${backnam}-${TIMESTAMP}.fsa 2>${backnam}-${TIMESTAMP}.info "
fsarchiver archinfo ${backnam}-${TIMESTAMP}.fsa 2>${backnam}-${TIMESTAMP}.info
#
lvremove -f /dev/${volumegroup}/${SNAPVOL}
RES=0
exit ${RES}
```

bash shell script archive-end

```
#!/bin/bash
#
ARGS=2
E_BADARGS=65
RES=1
if [ $# -ne "$ARGS" ]
then
    echo "Usage: `basename $0` volgroup device"
    exit $E_BADARGS
fi
VOLGROUP=$1                # name of the volume group
device=$2
# -----

PATH="{PATH}:/usr/sbin:/sbin:/usr/bin:/bin"
TIMESTAMP="{date +%Y%m%d-%H%M}"

# only run as root
if [ "$(id -u)" != '0' ]
then
    echo "this script has to be run as root"
    exit 1
fi

# check the volumegroups
vgscan
pvs
#
vgreduce $VOLGROUP $device
#
vgscan
RES=0
exit ${RES}
```

Gebruikte utilities

help dislocker command

```
dislocker --help
Usage: dislocker [-hqrV] [-l LOG_FILE] [-o OFFSET] [-V VOLUME DECRYPTMETHOD -F[N]] [-- ARGS...]
    with DECRYPTMETHOD = -p[RECOVERY_PASSWORD]|-f BEK_FILE|-u[USER_PASSWORD]|-k FVEK_FILE|-c
(v0.3)
Options:
  -c, --clearkey           decrypt volume using a clear key (default)
  -f, --bekfile BEKFILE   decrypt volume using the bek file (on USB key)
  -F, --force-block N     force use of metadata block number N (1, 2 or 3)
  -h, --help               print this help and exit
  -k, --fvek FVEK_FILE    decrypt volume using the FVEK directly
  -l, --logfile LOG_FILE  put messages into this file (stdout by default)
  -o, --offset OFFSET     BitLocker partition offset (default is 0)
  -p, --recovery-password[RECOVERY_PASSWORD]
                          decrypt volume using the recovery password method
  -q, --quiet              do NOT display anything
  -r, --readonly           do not allow to write on the BitLocker volume
  -u, --user-password     decrypt volume using the user password method
  -v, --verbosity         increase verbosity (CRITICAL errors are displayed by default)
  -V, --volume VOLUME     volume to get metadata and keys from

  --                       end of program options, beginning of FUSE's ones

  ARGS are any arguments you want to pass to FUSE. You need to pass at least
the mount-point.
```

help losetup command

```
losetup --help
```

Usage:

```
losetup [options] []
losetup [options] -f |
```

Options:

```
-a, --all                list all used devices
-d, --detach [...]     detach one or more devices
-D, --detach-all      detach all used devices
-f, --find              find first unused device
-c, --set-capacity     resize device
-j, --associated        list all devices associated with

-o, --offset            start at offset into file
  --sizelimit           device limited to bytes of the file
-P, --partscan         create partitioned loop device
-r, --read-only        setup read-only loop device
  --show               print device name after setup (with -f)
-v, --verbose          verbose mode

-l, --list              list info about all or specified
-O, --output            specify columns to output for --list
-n, --noheadings       don't print headings for --list output
  --raw                use raw --list output format

-h, --help             display this help and exit
-V, --version          output version information and exit
```

Available --list columns:

```
NAME    loop device name
AUTOCLEAR  autoclear flag set
BACK-FILE  device backing file
BACK-INO   backing file inode number
BACK-MAJ:MIN  backing file major:minor device number
MAJ:MIN   loop device major:minor number
OFFSET    offset from the beginning
PARTSCAN  partscan flag set
RO        read-only device
SIZELIMIT size limit of the file in bytes
```

For more details see `losetup(8)`.

help ntfsclone command

```
ntfsclone --help
ntfsclone v2014.2.15 (libntfs-3g)
```

Usage: `ntfsclone [OPTIONS] SOURCE`
Efficiently clone NTFS to a sparse file, image, device or standard output.

```
-o, --output FILE      Clone NTFS to the non-existent FILE
-O, --overwrite FILE  Clone NTFS to FILE, overwriting if exists
-s, --save-image      Save to the special image format
-r, --restore-image    Restore from the special image format
  --rescue            Continue after disk read errors
-m, --metadata        Clone *only* metadata (for NTFS experts)
-n, --no-action        Test restoring, without outputting anything
  --ignore-fs-check   Ignore the filesystem check result
  --new-serial        Set a new serial number
  --new-half-serial   Set a partial new serial number
-t, --preserve-timestamps Do not clear the timestamps
-q, --quiet           Do not display any progress bars
-f, --force           Force to progress (DANGEROUS)
-h, --help            Display this help
```

If FILE is '-' then send the image to the standard output. If SOURCE is '-' and --restore-image is used then read the image from the standard input.

Developers' email address: ntfs-3g-devel@lists.sf.net
News, support and information: <http://tuxera.com>

help vgchange command

vgchange --help

WARNING: lvmtools is running but disabled. Restart lvmtools before enabling it!

vgchange: Change volume group attributes

vgchange

```
[-A|--autobackup {y|n}]
[--alloc AllocationPolicy]
[-P|--partial]
[--commandprofile ProfileName]
[-d|--debug]
[--detachprofile]
[-h|--help]
[--ignorelockingfailure]
[--ignoremonitoring]
[--ignoreskippedcluster]
[-K|--ignoreactivationskip]
[--metadataprofile ProfileName]
[--monitor {y|n}]
[--[vg]metadaticopies #copies]
[--poll {y|n}]
[--noudevsync]
```

```
[--refresh]
[--sysinit]
[-t|--test]
[-u|--uuid]
[-v|--verbose]
[--version]
{-a|--activate [a|e|l]{y|n} |
[--activationmode {complete|degraded|partial}]
-c|--clustered {y|n} |
-x|--resizeable {y|n} |
-l|--logicalvolume MaxLogicalVolumes |
-p|--maxphysicalvolumes MaxPhysicalVolumes |
-s|--physicalextentsize PhysicalExtentSize[bBsSkKmMgGtTpPeE] |
--addtag Tag |
--deltag Tag}
[VolumeGroupName...]
```

help vgscan command

```
vgscan --help
```

```
WARNING: lvmtools is running but disabled. Restart lvmtools before enabling it!
```

```
vgscan: Search for all volume groups
```

```
vgscan [--cache]
        [--commandprofile ProfileName]
        [-d|--debug]
        [-h|--help]
        [--ignorelockingfailure]
        [--mknodes]
        [-P|--partial]
        [-v|--verbose]
        [--version]
```

help vgextend command

```
vgextend --help
```

```
WARNING: lvmtools is running but disabled. Restart lvmtools before enabling it!
```

```
vgextend: Add physical volumes to a volume group
```

vgextend

```
[-A|--autobackup y|n]
[--restoremissing]
[--commandprofile ProfileName]
[-d|--debug]
[-f|--force]
[-h|--help]
[-t|--test]
[-v|--verbose]
[--version]
[-y|--yes]
[ PHYSICAL DEVICE OPTIONS ]
VolumeGroupName PhysicalDevicePath [PhysicalDevicePath...]
```

help pvs command

pvs --help

```
WARNING: lvm2 is running but disabled. Restart lvm2 before enabling it!
pvs: Display information about physical volumes
```

pvs

```
[-a|--all]
[--aligned]
[--binary]
[--commandprofile ProfileName]
[-d|--debug]
[-h|-?|--help]
[--ignorelockingfailure]
[--ignorestalecluster]
[--nameprefixes]
[--noheadings]
[--nosuffix]
[-o|--options [+]Field[,Field]]
[-O|--sort [+|-]key1[,[+|-]key2[, ...]]]
[-P|--partial]
[--readonly]
[--rows]
[--segments]
[-S|--select Selection]
[--separator Separator]
[--trustcache]
[--unbuffered]
```

```
[--units hHbBsSkKmMgGtTpPeE]
[--unquoted]
[-v|--verbose]
[--version]
[PhysicalVolume [PhysicalVolume...]]
```

help lvs command

```
lvs --help
```

```
WARNING: lvm2 is running but disabled. Restart lvm2 before enabling it!
```

```
lvs: Display information about logical volumes
```

```
lvs
```

```
[-a|--all]
[--aligned]
[--binary]
[--commandprofile ProfileName]
[-d|--debug]
[-h|--help]
[--ignorelockingfailure]
[--ignorestalecluster]
[--nameprefixes]
[--noheadings]
[--nosuffix]
[-o|--options [+]Field[,Field]]
[-O|--sort [+|-]key1[,[+|-]key2[, ...]]]
[-P|--partial]
[--readonly]
[--rows]
[--segments]
[-S|--select Selection]
[--separator Separator]
[--trustcache]
[--unbuffered]
[--units hHbBsSkKmMgGtTpPeE]
[--unquoted]
[-v|--verbose]
[--version]
[LogicalVolume[Path] [LogicalVolume[Path]...]]
```


help dd command

dd --help

Usage: dd [OPERAND]...

or: dd OPTION

Copy a file, converting and formatting according to the operands.

bs=BYTES	read and write up to BYTES bytes at a time
cbs=BYTES	convert BYTES bytes at a time
conv=CONVS	convert the file as per the comma separated symbol list
count=N	copy only N input blocks
ibs=BYTES	read up to BYTES bytes at a time (default: 512)
if=FILE	read from FILE instead of stdin
iflag=FLAGS	read as per the comma separated symbol list
obs=BYTES	write BYTES bytes at a time (default: 512)
of=FILE	write to FILE instead of stdout
oflag=FLAGS	write as per the comma separated symbol list
seek=N	skip N obs-sized blocks at start of output
skip=N	skip N ibs-sized blocks at start of input
status=WHICH	WHICH info to suppress outputting to stderr; 'noxfers' suppresses transfer stats, 'none' suppresses all

N and BYTES may be followed by the following multiplicative suffixes:

c =1, w =2, b =512, kB =1000, K =1024, MB =1000*1000, M =1024*1024, xM =M
GB =1000*1000*1000, G =1024*1024*1024, and so on for T, P, E, Z, Y.

Each CONV symbol may be:

ascii	from EBCDIC to ASCII
ebcdic	from ASCII to EBCDIC
ibm	from ASCII to alternate EBCDIC
block	pad newline-terminated records with spaces to cbs-size
unblock	replace trailing spaces in cbs-size records with newline
lcase	change upper case to lower case
ucase	change lower case to upper case
sparse	try to seek rather than write the output for NUL input blocks
swab	swap every pair of input bytes
sync	pad every input block with NULs to ibs-size; when used with block or unblock, pad with spaces rather than NULs
excl	fail if the output file already exists
nocreat	do not create the output file
notrunc	do not truncate the output file
noerror	continue after read errors
fdatasync	physically write output file data before finishing
fsync	likewise, but also write metadata

Each FLAG symbol may be:

```
append      append mode (makes sense only for output; conv=notrunc suggested)
direct      use direct I/O for data
directory   fail unless a directory
dsync       use synchronized I/O for data
sync        likewise, but also for metadata
fullblock   accumulate full blocks of input (iflag only)
nonblock    use non-blocking I/O
noatime     do not update access time
nocache     discard cached data
noctty      do not assign controlling terminal from file
nofollow    do not follow symlinks
count_bytes treat 'count=N' as a byte count (iflag only)
skip_bytes  treat 'skip=N' as a byte count (iflag only)
seek_bytes  treat 'seek=N' as a byte count (oflag only)
```

Sending a USR1 signal to a running 'dd' process makes it print I/O statistics to standard error and then resume copying.

```
$ dd if=/dev/zero of=/dev/null& pid=$!
$ kill -USR1 $pid; sleep 1; kill $pid
18335302+0 records in
18335302+0 records out
9387674624 bytes (9.4 GB) copied, 34.6279 seconds, 271 MB/s
```

Options are:

```
--help      display this help and exit
--version   output version information and exit
```

Report dd bugs to bug-coreutils@gnu.org

GNU coreutils home page:

General help using GNU software:

For complete documentation, run: `info coreutils 'dd invocation'`

help sfdisk command

```
sfdisk --help
```

Usage:

```
sfdisk [options] [...]
```

Options:

```
-s, --show-size      list size of a partition
```

```

-c, --id                change or print partition Id
  --change-id           change Id
  --print-id           print Id
-l, --list              list partitions of each device
-d, --dump              idem, but in a format suitable for later input
-i, --increment         number cylinders etc. from 1 instead of from 0
-u, --unit              units to be used; can be one of
                        S (sectors), C (cylinders), B (blocks), or M (MB)
-1, --one-only         reserved option that does nothing currently
-T, --list-types       list the known partition types
-D, --DOS              for DOS-compatibility: waste a little space
-E, --DOS-extended     DOS extended partition compatibility
-R, --re-read          make the kernel reread the partition table
-N                     change only the partition with this
-n                     do not actually write to disk
-O                     save the sectors that will be overwritten to
-I                     restore sectors from
-V, --verify           check that the listed partitions are reasonable
-v, --version          display version information and exit
-h, --help             display this help text and exit

```

Dangerous options:

```

-f, --force            disable all consistency checking
  --no-reread         do not check whether the partition is in use
-q, --quiet           suppress warning messages
-L, --Linux           do not complain about things irrelevant for Linux
-g, --show-geometry   print the kernel's idea of the geometry
-G, --show-pt-geometry print geometry guessed from the partition table
-A, --activate[=]    activate bootable flag
-U, --unhide[=]      set partition unhidden
-x, --show-extended  also list extended partitions in the output,
                    or expect descriptors for them in the input
  --leave-last       do not allocate the last cylinder
  --IBM              same as --leave-last
  --in-order         partitions are in order
  --not-in-order     partitions are not in order
  --inside-outer     all logicals inside outermost extended
  --not-inside-outer not all logicals inside outermost extended
  --nested           every partition is disjoint from all others
  --chained          like nested, but extended partitions may lie outside
  --onesector        partitions are mutually disjoint

```

Override the detected geometry using:

```

-C, --cylinders       set the number of cylinders to use
-H, --heads           set the number of heads to use
-S, --sectors         set the number of sectors to use

```

For more details see sfdisk(8).

help fsarchiver command

```
fsarchiver --help
====> fsarchiver version 0.6.19 (2014-03-01) - http://www.fsarchiver.org <====
Distributed under the GPL v2 license (GNU General Public License v2).
* usage: fsarchiver [] [ [ [...]]]

* savefs: save filesystems to an archive file (backup a partition to a file)
* restfs: restore filesystems from an archive (overwrites the existing data)
* savedir: save directories to the archive (similar to a compressed tarball)
* restdir: restore data from an archive which is not based on a filesystem
* archinfo: show information about an existing archive file and its contents
* probe [detailed]: show list of filesystems detected on the disks

-o: overwrite the archive if it already exists instead of failing
-v: verbose mode (can be used several times to increase the level of details)
-d: debug mode (can be used several times to increase the level of details)
-A: allow to save a filesystem which is mounted in read-write (live backup)
-a: allow running savefs when partition mounted without the acl/xattr options
-e : exclude files and directories that match that pattern
-L : set the label of the archive (comment about the contents)
-z : compression level from 1 (very fast) to 9 (very good) default=3
-s : split the archive into several files of megabytes each
-j : create more than one compression thread. useful on multi-core cpu
-c : encrypt/decrypt data in archive, "-c -" for interactive password
-h: show help and information about how to use fsarchiver with examples
-V: show program version and exit

* Support included for: lzo=yes, lzma=yes
* support for ntfs filesystems is unstable: don't use it for production.

* [1msave only one filesystem (/dev/sda1) to an archive:[0m
fsarchiver savefs /data/myarchive1.fsa /dev/sda1
* [1msave two filesystems (/dev/sda1 and /dev/sdb1) to an archive:[0m
fsarchiver savefs /data/myarchive2.fsa /dev/sda1 /dev/sdb1
* [1mrestore the first filesystem from an archive (first = number 0):[0m
fsarchiver restfs /data/myarchive2.fsa id=0,dest=/dev/sda1
* [1mrestore the second filesystem from an archive (second = number 1):[0m
fsarchiver restfs /data/myarchive2.fsa id=1,dest=/dev/sdb1
* [1mrestore two filesystems from an archive (number 0 and 1):[0m
fsarchiver restfs /data/arch2.fsa id=0,dest=/dev/sda1 id=1,dest=/dev/sdb1
* [1mrestore a filesystem from an archive and convert it to reiserfs:[0m
```

```
fsarchiver restfs /data/myarchive1.fsa id=0,dest=/dev/sda1,mkfs=reiserfs
* [1msave the contents of /usr/src/linux to an archive (similar to tar):[0m
fsarchiver savedir /data/linux-sources.fsa /usr/src/linux
* [1msave a filesystem (/dev/sda1) to an archive split into volumes of 680MB:[0m
fsarchiver savefs -s 680 /data/myarchive1.fsa /dev/sda1
* [1msave a filesystem and exclude all files/dirs called 'pagefile.*':[0m
fsarchiver savefs /data/myarchive.fsa /dev/sda1 --exclude='pagefile.*'
* [1mgeneric exclude for 'share' such as '/usr/share' and '/usr/local/share':[0m
fsarchiver savefs /data/myarchive.fsa --exclude=share
* [1mabsolute exclude valid for '/usr/share' but not for '/usr/local/share':[0m
fsarchiver savefs /data/myarchive.fsa --exclude=/usr/share
* [1msave a filesystem (/dev/sda1) to an encrypted archive:[0m
fsarchiver savefs -c mypassword /data/myarchive1.fsa /dev/sda1
* [1mSame as before but prompt for password in the terminal:[0m
fsarchiver savefs -c - /data/myarchive1.fsa /dev/sda1
* [1mextract an archive made of simple files to /tmp/extract:[0m
fsarchiver restdir /data/linux-sources.fsa /tmp/extract
* [1mshow information about an archive and its file systems:[0m
fsarchiver archinfo /data/myarchive2.fsa
```

help tune2fs command

```
tune2fs --help
tune2fs: invalid option -- '-'
Usage: tune2fs [-c max_mounts_count] [-e errors_behavior] [-g group]
      [-i interval[d|m|w]] [-j] [-J journal_options] [-l]
      [-m reserved_blocks_percent] [-o [^]mount_options[,...]] [-p mmp_update_interval]
      [-r reserved_blocks_count] [-u user] [-C mount_count] [-L volume_label]
      [-M last_mounted_dir] [-O [^]feature[,...]]
      [-E extended-option[,...]] [-T last_check_time] [-U UUID]
      [ -I new_inode_size ] device
tune2fs 1.42.12 (29-Aug-2014)
```

help e2fsck command

```
e2fsck --help
e2fsck: invalid option -- '-'
Usage: e2fsck [-panyrcdfvtDFV] [-b superblock] [-B blocksize]
      [-I inode_buffer_blocks] [-P process_inode_size]
      [-l|-L bad_blocks_file] [-C fd] [-j external_journal]
```

[-E extended-options] device

Emergency help:

```
-p          Automatic repair (no questions)
-n          Make no changes to the filesystem
-y          Assume "yes" to all questions
-c          Check for bad blocks and add them to the badblock list
-f          Force checking even if filesystem is marked clean
-v          Be verbose
-b superblock  Use alternative superblock
-B blocksize  Force blocksize when looking for superblock
-j external_journal  Set location of the external journal
-l bad_blocks_file  Add to badblocks list
-L bad_blocks_file  Set badblocks list
```

help ntfsinfo command

ntfsinfo --help

Usage: ntfsinfo [options] device

```
-i, --inode NUM  Display information about this inode
-F, --file FILE  Display information about this file (absolute path)
-m, --mft        Dump information about the volume
-t, --notime     Don't report timestamps

-f, --force      Use less caution
-q, --quiet      Less output
-v, --verbose    More output
-V, --version    Display version information
-h, --help      Display this help
```

Developers' email address: ntfs-3g-devel@lists.sf.net

News, support and information: <http://tuxera.com>

Failed to parse command line options

help ntfsfix command

ntfsfix --help

ntfsfix v2014.2.15 (libntfs-3g)

Usage: ntfsfix [options] device
Attempt to fix an NTFS partition.

-b, --clear-bad-sectors Clear the bad sector list
-d, --clear-dirty Clear the volume dirty flag
-h, --help Display this help
-n, --no-action Do not write anything
-V, --version Display version information

For example: ntfsfix /dev/hda6

Developers' email address: ntfs-3g-devel@lists.sf.net
News, support and information: <http://tuxera.com>

Het resultaat

Het resultaat met cijfers

ulefr01server

externe harddisk
lacie2

Datum = 2014/12/30

**Startup
Rescuecd**

l) ./t00

backup MBR &
partitietabel

bckupmbr
sda

bckupmbr
device

dd if=/dev/sda
of=sda.mbr count=1
bs=512

dd
if=/dev/sda
of=sda65.m
br count=65
bs=512
sfdisk -d /dev/sda
>parttblsda.sf

bckupmbr
sdb

bckupmbr
device

dd if=/dev/sdb
of=sdb.mbr count=1
bs=512

dd
if=/dev/sdb
of=sdb65.m
br count=65
sfdisk -d /dev/sdb
>parttblsdb.sf

bs=512

tune LV1	tuneLV linux3 linux_boot ext4	TuneLV VG LV type	tune_ext4 /dev/mapper/linux3- linux_boot
tune LV2	tuneLV linux3 linux_root ext4	TuneLV VG LV type	tune_ext4 /dev/mapper/linux3- linux_root
tune LV3	tuneLV linux3linux_h ome ext4	TuneLV VG LV type	tune_ext4 /dev/mapper/linux3- linux_home
tune LV4	tuneLV linux3 linux_data ext4	TuneLV VG LV type	tune_ext4 /dev/mapper/linux3- linux_data

II III) no

IV) ./t03

	archive-start	archive-start VG device	vgextend linux /dev/sdc4
backup LV1	archive-mint linux linux_boot boot	archive-mint VG LV backnam	fsarchiver saveefs -L\$ {logvol} \${FSAOPTS} \${backnam}- {TIMESTAMP}.fsa /dev/mapper/\$ {volumegroup}- {logvol}
backup LV2	archive-mint linux linux_root	archive-mint VG LV backnam	fsarchiver saveefs -L\$ {logvol} \${FSAOPTS} \${backnam}-

	root		{TIMESTAMP}.fsa /dev/mapper/\$ {volumegroup}-\$ {logvol}
backup LV3	archive-mint linux linux_home home	archive-mint VG LV backnam	fsarchiver saveefs -L\$ {logvol} \${FSAOPTS} \${backnam}-\$ {TIMESTAMP}.fsa /dev/mapper/\$ {volumegroup}-\$ {logvol}
backup LV4	archive-mint linux linux_data data	archive-mint VG LV backnam	fsarchiver saveefs -L\$ {logvol} \${FSAOPTS} \${backnam}-\$ {TIMESTAMP}.fsa /dev/mapper/\$ {volumegroup}-\$ {logvol}
	archive-end	archive-end VG device	vgreduce linux /dev/sdc4

V) ./t03-end

(ls *\${DATE}* -la --sort=n)	cat term_00.out term_01.out term_02.out term_03.out term_03- end.out
---------------------------------	--

**VI) ./mkout_to-
PDF**

mkout_to-PDF
laptopca-

yyyymmdd,out

Partitie	Type	files	Directories	Symlinks	hardlnks	specials	starttime	endtime	duur	term
linux_boot	ext4	295	7	0	0	0	22:29:00	22:31:00	00:02:00	t3
linux_root	ext4	283701	36151	75271	19	100	22:31:00	22:59:22	00:28:22	t3
linux_home	ext4	28714	2872	91	0	4	22:59:22	23:17:38	00:18:16	t3
linux_data	ext4	60830	9266	50	1	0	23:17:38	23:47:31	00:29:53	t3

De uitvoering

De uiteindelijke uitvoering

```
bckupmbrs
```

```
-----
```

```
sdx (x=a,b,...) device = sda
```

```
-----
```

```
Tue Dec 30 22:28:43 UTC 2014
```

```
doe mv sdas.mbr sdas_20141230.mbr
```

```
doe mv sdas65.mbr sdas65_20141230.mbr
```

```
doe dd if=/dev/sda of=sdas.mbr count=1 bs=512
```

```
1+0 records in
```

```
1+0 records out
```

```
512 bytes (512 B) copied, 0.000411156 s, 1.2 MB/s
```

```
doe dd if=/dev/sda of=sdas65.mbr count=65 bs=512
```

```
65+0 records in
```

```
65+0 records out
```

```
33280 bytes (33 kB) copied, 0.00105111 s, 31.7 MB/s
```

```
doe mv parttblsdas.sf parttblsdas_20141230.sf
```

```
doe sfdisk -d /dev/sda >parttblsdas.sf
Tue Dec 30 22:28:43 UTC 2014
-----
bckupmbrs
-----
sdx (x=a,b,...) device = sdb
-----
Tue Dec 30 22:28:43 UTC 2014
doe mv sdb.sdb mbr sdb_sdb_20141230.mbr
mv: cannot stat : No such file or directory
doe mv sdb65.mbr sdb65_20141230.mbr
mv: cannot stat : No such file or directory
doe dd if=/dev/sdb of=sdb.sdb count=1 bs=512
1+0 records in
1+0 records out
512 bytes (512 B) copied, 0.000352489 s, 1.5 MB/s
doe dd if=/dev/sdb of=sdb65 count=65 bs=512
65+0 records in
65+0 records out
33280 bytes (33 kB) copied, 0.00109169 s, 30.5 MB/s
doe mv parttblsdb.sf parttblsdb_20141230.sf
doe sfdisk -d /dev/sdb >parttblsdb.sf
Tue Dec 30 22:28:43 UTC 2014
-----
bckupmbrs
-----
sdx (x=a,b,...) device = sdc
-----
Tue Dec 30 22:28:43 UTC 2014
doe mv sdc.sdb mbr sdc_sdb_20141230.mbr
mv: cannot stat : No such file or directory
doe mv sdc65.mbr sdc65_20141230.mbr
mv: cannot stat : No such file or directory
doe dd if=/dev/sdc of=sdc.sdb count=1 bs=512
dd: failed to open : No medium found
doe dd if=/dev/sdc of=sdc65 count=65 bs=512
dd: failed to open : No medium found
```

```
doe mv parttblsdcs.sf parttblsdcs_20141230.sf
doe sfdisk -d /dev/sdc >parttblsdcs.sf
sfdisk: cannot open /dev/sdc for reading: No medium found
Tue Dec 30 22:28:43 UTC 2014
```

```
bckupmbrs
```

```
sdx (x=a,b,...) device = sdd
```

```
Tue Dec 30 22:28:43 UTC 2014
doe mv sdds.mbr sdds_20141230.mbr
mv: cannot stat : No such file or directory
doe mv sdds65.mbr sdds65_20141230.mbr
mv: cannot stat : No such file or directory
doe dd if=/dev/sdd of=sdds.mbr count=1 bs=512
dd: failed to open : No medium found
doe dd if=/dev/sdd of=sdds65.mbr count=65 bs=512
dd: failed to open : No medium found
doe mv parttblsdds.sf parttblsdds_20141230.sf
doe sfdisk -d /dev/sdd >parttblsdds.sf
sfdisk: cannot open /dev/sdd for reading: No medium found
Tue Dec 30 22:28:43 UTC 2014
```

```
tuneLV
```

```
tuneLV linux3 lv linux_boot type ext4
Tue Dec 30 22:28:43 UTC 2014
tuneLV ext4
tune_ext4
-----
device= /dev/mapper/linux3-linux_boot
doe : tune2fs -O has_journal /dev/mapper/linux3-linux_boot
tune2fs 1.42.12 (29-Aug-2014)
doe : e2fsck -f /dev/mapper/linux3-linux_boot
e2fsck 1.42.12 (29-Aug-2014)
e2fsck: need terminal for interactive repairs
doe : tune2fs -J size=4 /dev/mapper/linux3-linux_boot
```

```
tune2fs 1.42.12 (29-Aug-2014)
Creating journal inode: done
This filesystem will be automatically checked every 200 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
doe : tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 /dev/mapper/linux3-linux_boot
tune2fs 1.42.12 (29-Aug-2014)
Setting maximal mount count to 200
Setting interval between checks to 15552000 seconds
Setting reserved blocks percentage to 1% (2621 blocks)
doe : e2fsck -pFD /dev/mapper/linux3-linux_boot
/dev/mapper/linux3-linux_boot: 311/65536 files (1.3% non-contiguous), 103610/262144 blocks
gedaan !
device= /dev/mapper/linux3-linux_boot
Tue Dec 30 22:28:44 UTC 2014
-----
tuneLV
-----
tuneLV linux3 lv linux_root type ext4
Tue Dec 30 22:28:44 UTC 2014
tuneLV ext4
tune_ext4
-----
device= /dev/mapper/linux3-linux_root
doe : tune2fs -O has_journal /dev/mapper/linux3-linux_root
tune2fs 1.42.12 (29-Aug-2014)
doe : e2fsck -f /dev/mapper/linux3-linux_root
e2fsck 1.42.12 (29-Aug-2014)
e2fsck: need terminal for interactive repairs
doe : tune2fs -J size=4 /dev/mapper/linux3-linux_root
tune2fs 1.42.12 (29-Aug-2014)
Creating journal inode: done
This filesystem will be automatically checked every 200 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
doe : tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 /dev/mapper/linux3-linux_root
tune2fs 1.42.12 (29-Aug-2014)
Setting maximal mount count to 200
Setting interval between checks to 15552000 seconds
```

```
Setting reserved blocks percentage to 1% (262144 blocks)
doe : e2fsck -pFD /dev/mapper/linux3-linux_root
/dev/mapper/linux3-linux_root: 395232/6553600 files (0.1% non-contiguous), 3632029/26214400 blocks
gedaan !
device= /dev/mapper/linux3-linux_root
Tue Dec 30 22:29:13 UTC 2014
-----
tuneLV
-----
tuneLV linux3 lv linux_home type ext4
Tue Dec 30 22:29:13 UTC 2014
tuneLV ext4
tune_ext4
-----
device= /dev/mapper/linux3-linux_home
doe : tune2fs -O has_journal /dev/mapper/linux3-linux_home
tune2fs 1.42.12 (29-Aug-2014)
doe : e2fsck -f /dev/mapper/linux3-linux_home
e2fsck 1.42.12 (29-Aug-2014)
e2fsck: need terminal for interactive repairs
doe : tune2fs -J size=4 /dev/mapper/linux3-linux_home
tune2fs 1.42.12 (29-Aug-2014)
Creating journal inode: done
This filesystem will be automatically checked every 200 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
doe : tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 /dev/mapper/linux3-linux_home
tune2fs 1.42.12 (29-Aug-2014)
Setting maximal mount count to 200
Setting interval between checks to 15552000 seconds
Setting reserved blocks percentage to 1% (262144 blocks)
doe : e2fsck -pFD /dev/mapper/linux3-linux_home
/dev/mapper/linux3-linux_home: 31690/6553600 files (0.3% non-contiguous), 3453948/26214400 blocks
gedaan !
device= /dev/mapper/linux3-linux_home
Tue Dec 30 22:29:19 UTC 2014
-----
tuneLV
```

```
-----
tuneLV linux3 lv linux_data type ext4
Tue Dec 30 22:29:19 UTC 2014
tuneLV ext4
tune_ext4
-----
device= /dev/mapper/linux3-linux_data
doe : tune2fs -O has_journal /dev/mapper/linux3-linux_data
tune2fs 1.42.12 (29-Aug-2014)
doe : e2fsck -f /dev/mapper/linux3-linux_data
e2fsck 1.42.12 (29-Aug-2014)
e2fsck: need terminal for interactive repairs
doe : tune2fs -J size=4 /dev/mapper/linux3-linux_data
tune2fs 1.42.12 (29-Aug-2014)
Creating journal inode: done
This filesystem will be automatically checked every 200 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
doe : tune2fs -O extents,uninit_bg,dir_index -i 6m -c 200 -m 1 /dev/mapper/linux3-linux_data
tune2fs 1.42.12 (29-Aug-2014)
Setting maximal mount count to 200
Setting interval between checks to 15552000 seconds
Setting reserved blocks percentage to 1% (262144 blocks)
doe : e2fsck -pFD /dev/mapper/linux3-linux_data
/dev/mapper/linux3-linux_data: 70155/6553600 files (0.2% non-contiguous), 3129003/26214400 blocks
gedaan !
device= /dev/mapper/linux3-linux_data
Tue Dec 30 22:29:29 UTC 2014
-----
WARNING: lvm2 is running but disabled. Restart lvm2 before enabling it!
Reading all physical volumes. This may take a while...
Found volume group "linux3" using metadata type lvm2
WARNING: lvm2 is running but disabled. Restart lvm2 before enabling it!
Volume group "linux3" successfully extended
archive-mint
-----
volumegroup = linux3
logical volume = linux_boot
```

```
backnam = boot
-----
fsarchiver savefs -Llinux_boot -z7 -j3 -s 2000 -A boot-20141230-22h29.fsa /dev/mapper/linux3-linux_boot
Archive will be split into volumes of 2097152000 bytes (1.95 GB)
Statistics for filesystem 0
* files successfully processed:....regfiles=295, directories=7, symlinks=0, hardlinks=0, specials=0
* files with errors:.....regfiles=0, directories=0, symlinks=0, hardlinks=0, specials=0
fsarchiver done
fsarchiver archinfo boot-20141230-22h29.fsa 2>boot-20141230-22h29.info
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Logical volume "mysnap" successfully removed
archive-mint
-----
volumegroup = linux3
logical volume = linux_root
backnam = root
-----
fsarchiver savefs -Llinux_root -z7 -j3 -s 2000 -A root-20141230-22h31.fsa /dev/mapper/linux3-linux_root
Archive will be split into volumes of 2097152000 bytes (1.95 GB)
Statistics for filesystem 0
* files successfully processed:....regfiles=283701, directories=36151, symlinks=75271, hardlinks=19,
specials=100
* files with errors:.....regfiles=0, directories=0, symlinks=0, hardlinks=0, specials=0
fsarchiver done
fsarchiver archinfo root-20141230-22h31.fsa 2>root-20141230-22h31.info
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Logical volume "mysnap" successfully removed
archive-mint
-----
volumegroup = linux3
logical volume = linux_home
backnam = home
-----
fsarchiver savefs -Llinux_home -z7 -j3 -s 2000 -A home-20141230-22h59.fsa /dev/mapper/linux3-linux_home
Archive will be split into volumes of 2097152000 bytes (1.95 GB)
Statistics for filesystem 0
* files successfully processed:....regfiles=28714, directories=2872, symlinks=91, hardlinks=0, specials=4
```



```
* files with errors:.....regfiles=0, directories=0, symlinks=0, hardlinks=0, specials=0
fsarchiver done
fsarchiver archinfo home-20141230-22h59.fsa 2>home-20141230-22h59.info
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Logical volume "mysnap" successfully removed
archive-mint
-----
volumegroup = linux3
logical volume = linux_data
backnam = data
-----
fsarchiver savefs -Llinux_data -z7 -j3 -s 2000 -A data-20141230-23h17.fsa /dev/mapper/linux3-linux_data
Archive will be split into volumes of 2097152000 bytes (1.95 GB)
Statistics for filesystem 0
* files successfully processed:....regfiles=60830, directories=9266, symlinks=50, hardlinks=1, specials=0
* files with errors:.....regfiles=0, directories=0, symlinks=0, hardlinks=0, specials=0
fsarchiver done
fsarchiver archinfo data-20141230-23h17.fsa 2>data-20141230-23h17.info
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Logical volume "mysnap" successfully removed
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Reading all physical volumes. This may take a while...
Found volume group "linux3" using metadata type lvm2
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
PV VG Fmt Attr PSize PFree
/dev/sda1 linux3 lvm2 a-- 310.00g 8.00g
/dev/sdf4 linux3 lvm2 a-- 9.76g 9.76g
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Removed "/dev/sdf4" from volume group "linux3"
WARNING: lvmetad is running but disabled. Restart lvmetad before enabling it!
Reading all physical volumes. This may take a while...
Found volume group "linux3" using metadata type lvm2
-rw-r--r--+ 1 root root 392828373 Dec 30 22:31 boot-20141230-22h29.fsa
-rw-rw-rw--+ 1 root root 745 Dec 30 22:31 boot-20141230-22h29.info
-rw-r--r--+ 1 root root 2097028446 Dec 30 23:31 data-20141230-23h17.f01
-rw-r--r--+ 1 root root 2097054924 Dec 30 23:38 data-20141230-23h17.f02
-rw-r--r--+ 1 root root 2096939429 Dec 30 23:44 data-20141230-23h17.f03
```

```
-rw-r--r--+ 1 root root 799548969 Dec 30 23:47 data-20141230-23h17.f04
-rw-r--r--+ 1 root root 2097076686 Dec 30 23:24 data-20141230-23h17.fsa
-rw-rw-rw--+ 1 root root 746 Dec 30 23:47 data-20141230-23h17.info
-rw-r--r--+ 1 root root 2097149915 Dec 30 23:14 home-20141230-22h59.f01
-rw-r--r--+ 1 root root 262001778 Dec 30 23:17 home-20141230-22h59.f02
-rw-r--r--+ 1 root root 2097143439 Dec 30 23:06 home-20141230-22h59.fsa
-rw-rw-rw--+ 1 root root 746 Dec 30 23:17 home-20141230-22h59.info
-rw-rw-rw--+ 1 root root 249 Dec 30 21:19 parttblsdas_20141230.sf
-rw-rw-rw--+ 1 root root 0 Dec 30 21:19 parttblsdbas_20141230.sf
-rw-rw-rw--+ 1 root root 0 Dec 30 21:19 parttblsdcs_20141230.sf
-rw-rw-rw--+ 1 root root 0 Dec 30 21:19 parttblsdds_20141230.sf
-rw-r--r--+ 1 root root 2097138807 Dec 30 22:50 root-20141230-22h31.f01
-rw-r--r--+ 1 root root 1652608515 Dec 30 22:59 root-20141230-22h31.f02
-rw-r--r--+ 1 root root 2097096701 Dec 30 22:42 root-20141230-22h31.fsa
-rw-rw-rw--+ 1 root root 746 Dec 30 22:59 root-20141230-22h31.info
-rw-rw-rw--+ 1 root root 512 Dec 30 21:19 sdas_20141230.mbr
-rw-rw-rw--+ 1 root root 33280 Dec 30 21:19 sdas65_20141230.mbr
```

term_00s

22:28:43

22:28:43

22:28:43

22:28:43

22:28:43

22:28:43

22:28:43

22:28:43

22:28:43

22:28:44

22:28:44

22:29:13

22:29:13

22:29:19

22:29:19

22:29:29

term_03s