

LTFS en Tape

Tapedrive

Om informatie over de drive zelf uit te lezen kun je het volgende doen:

```
sudo sg_logs -a /dev/sg3
```

Dan krijg je mogelijk de volgende output:

```
Device statistics log page (ssc-3 and adc)
Lifetime media loads: 2200
Lifetime cleaning operations: 6
Lifetime power on hours: 61183
Lifetime media motion (head) hours: 11204
Lifetime metres of tape processed: 132126786
```

Wanneer het lang geleden is dat een tape gereinigd is, kun je een cleaning-tape laden. Dan wordt de drive automatisch gereinigd.

Er is een script waarmee je een test kunt doen om data te schrijven naar en lezen van tape. Hiervoor moet je een tape in de drive doen en het volgende commando geven (voor een test met 50Gb).:

```
sudo TAPE_DEV=/dev/nst0 SCSI_DEV=/dev/sg3 TEST_SIZE_MIB=51200 ./testdrive.sh
```

De map /mnt/data1/tmp/tapedrive moet bestaan en er moet minstens 100Gb geschreven kunnen worden.

WORM

Op een WORM-tape kun je geen LTFS schrijven. Daarom schrijven met bijv. tar. Hiervoor moet je eerst naar het einde van de tape omdat je vanaf daar met het tar-commando kunt schrijven:

```
sudo mt -f /dev/nst0 eod
```

Vrije ruimte

Vervolgens kun je uit de sg_logs informatie halen welke grootte nog vrij is. Daarvan kun je het beste wat extra ruimte vrijhouden.

Er is een sh-bestandje (onderaan de bron) waarmee de grootte getoond wordt. Dit gaat met het volgende commando:

```
./list_size.sh
```

Data schrijven

Daadwerkelijk schrijven van de data doe je met het tar commando. Dat gaat als volgt:

```
sudo tar -cvf /dev/nst0 "/mnt/data/sessions.zip"
```

of meerdere bestanden als één set:

```
sudo tar -cvf /dev/nst0 "/mnt/data/sessions.zip"  
"/mnt/data/sessions_bestandslijst.txt"
```

Anders worden het losse sets.

Data teruglezen

De data teruglezen vanaf tape gaat als volgt:

```
sudo mt -f /dev/st0 rewind  
sudo tar -xvf /dev/st0 -C /mnt/restore "mnt/data/music_bestandslijst.txt"
```

LTFS

Broncodes

list_size.sh

```
sudo sg_logs -p 0x31 /dev/sg3 \  
| awk -F': ' \  
/Main partition remaining capacity/ {rem=$2}  
END{  
    safe=rem*0.90  
    printf "Vrij:   %.1f GiB (%.1f GB)\n", rem/1024, rem/1024*1.073741824  
    printf "Veilig:  %.1f GiB (%.1f GB) [10%% marge]\n", safe/1024,  
    safe/1024*1.073741824  
}'
```

list_all_sets.sh

```
#!/usr/bin/env bash
set -euo pipefail

DEV=/dev/nst0

sudo mt -f "$DEV" rewind

i=1
while true; do
    echo
    echo "==== SET $i ===="
    if ! sudo tar -tvf "$DEV"; then
        echo "Einde tape of geen tar-set meer."
        break
    fi
    # spring naar volgende set (filemark overslaan)
    if ! sudo mt -f "$DEV" fsf 1; then
        echo "Kan niet verder spoelen (waarschijnlijk EOT)."
```

testdrive.sh

```
#!/usr/bin/env bash
set -euo pipefail

#####
# LT0-6 drive/tape test
# Destructief: schrijft naar tape en verifieert readback.
#
# Vereisten:
#   dnf install mt-st sg3_utils coreutils
#
# Devices:
#   TAPE_DEV   : /dev/nst0 (non-rewind tape)
#   SCSI_DEV   : /dev/sg3  (sg generic)
#####

TAPE_DEV="${TAPE_DEV:-/dev/nst0}"
SCSI_DEV="${SCSI_DEV:-/dev/sg3}"

# Test parameters
TEST_DIR="${TEST_DIR:-/mnt/data/tmp/tapetest}"
TEST_FILE="${TEST_FILE:-$TEST_DIR/test.bin}"
READBACK_FILE="${READBACK_FILE:-$TEST_DIR/test_readback.bin}"
HASH_FILE="${HASH_FILE:-$TEST_DIR/test.bin.sha256}"
```

```
# Size in MiB (10240 MiB = 10 GiB)
TEST_SIZE_MIB="${TEST_SIZE_MIB:-10240}"
BS_MIB="${BS_MIB:-1}"

log() { echo -e "\n== $* =="; }

require_root() {
    if [[ $EUID -ne 0 ]]; then
        echo "Run dit script als root (sudo)."        exit 1
    fi
}

check_cmds() {
    local missing=0
    for c in mt sg_logs dd sha256sum egrep awk; do
        if ! command -v "$c" >/dev/null 2>&1; then
            echo "Mist commando: $c"
            missing=1
        fi
    done
    [[ $missing -eq 0 ]] || exit 1
}

confirm_destructive() {
    cat <<EOF
WAARSCHUWING: Deze test schrijft ${TEST_SIZE_MIB} MiB naar ${TAPE_DEV}
en OVERSCHRIJFT ALLE DATA op de geladen tape.

Typ exact: YES (zonder quotes) om door te gaan.
EOF
    read -r answer
    if [[ "$answer" != "YES" ]]; then
        echo "Afgebroken."
        exit 0
    fi
}

mt_status() {
    sudo mt -f "$TAPE_DEV" status || true
}

pre_checks() {
    log "Drive/tape status (mt)"
    mt_status

    log "TapeAlert (actieve flags)"
    if sudo sg_logs --page=0x2e "$SCSI_DEV" | egrep -i ": 1$" ; then
        echo "Let op: er staan TapeAlert flags aan."
    else
        echo "Geen actieve TapeAlerts."
    fi
}
```

```
fi

log "Read counters (voor test)"
sudo sg_logs --page=0x03 "$SCSI_DEV" | egrep -i "Errors
corrected|rewrites|uncorrected|Hard read"

log "Write counters (voor test)"
sudo sg_logs --page=0x02 "$SCSI_DEV" | egrep -i "Errors
corrected|rewrites|uncorrected|Hard write"

log "Device statistics (kerncounters)"
sudo sg_logs --page=0x14 "$SCSI_DEV" | egrep -i "Lifetime cleaning|Hard
read errors|Hard write errors|since last successful cleaning|media
loads|power on hours" || true
}

generate_test_data() {
log "Maak testdata (${TEST_SIZE_MIB} MiB) + checksum"
mkdir -p "$TEST_DIR"
rm -f "$TEST_FILE" "$READBACK_FILE" "$HASH_FILE"

# gebruik /dev/urandom voor echte random testdata
dd if=/dev/urandom of="$TEST_FILE" bs="${BS_MIB}M" count="$TEST_SIZE_MIB"
status=progress
log "Genereren van hash"
sha256sum "$TEST_FILE" | tee "$HASH_FILE"
}

write_to_tape() {
log "Rewind + blocksize = variable"
mt -f "$TAPE_DEV" rewind
mt -f "$TAPE_DEV" setblk 0

log "Schrijf testdata naar tape"
dd if="$TEST_FILE" of="$TAPE_DEV" bs="${BS_MIB}M" status=progress

log "Schrijf filemark en rewind"
mt -f "$TAPE_DEV" weof 1
mt -f "$TAPE_DEV" rewind
}

readback_and_verify() {
log "Lees terug van tape"
dd if="$TAPE_DEV" of="$READBACK_FILE" bs="${BS_MIB}M" status=progress

log "Verifieer checksum"
sha256sum -c "$HASH_FILE"
}

post_checks() {
log "Read counters (na test)"
```

```
sudo sg_logs --page=0x03 "$SCSI_DEV" | egrep -i "Errors
corrected|rewrites|uncorrected|Hard read"

log "Write counters (na test)"
sudo sg_logs --page=0x02 "$SCSI_DEV" | egrep -i "Errors
corrected|rewrites|uncorrected|Hard write"

log "TapeAlert (actieve flags na test)"
if sudo sg_logs --page=0x2e "$SCSI_DEV" | egrep -i ": 1$" ; then
    echo "Let op: er staan TapeAlert flags aan."
else
    echo "Geen actieve TapeAlerts."
fi

log "Device statistics (na test)"
sudo sg_logs --page=0x14 "$SCSI_DEV" | egrep -i "Hard read errors|Hard
write errors|since last successful cleaning|Lifetime metres|media
motion|activity duty cycle" || true
}

summary() {
    log "SAMENVATTING / INTERPRETATIE"
    echo "1) Checksum OK? -> hierboven moet 'OK' staan."
    echo "2) Uncorrected read/write errors moeten 0 zijn."
    echo "3) TapeAlert na test liefst leeg."
    echo
    echo "Als je wilt dat ik oordeel geef: plak pre/post counters + TapeAlert
hier."
}

main() {
    require_root
    check_cmds
    confirm_destructive
    pre_checks
    generate_test_data
    write_to_tape
    readback_and_verify
    post_checks
    summary
}

main "$@"
```

From:
<https://info.kosternet.nl/> - **KosterNET Info**

Permanent link:
https://info.kosternet.nl/doku.php?id=publiek:ltfs_en_tape&rev=1764008351

Last update: **2025/11/24 19:19**



