

SSDs anzeigen

Anzeigen, welche Blockdevices auf rotierenden Geräten (z.B. magnetische Festplatten) liegen und welche nicht (z.B. SSDs):

```
lsblk -o NAME,ROTA
```

Geräte mit 1 in der Spalte „ROTA“ liegen auf rotierenden Geräten, Geräte mit 0 nicht.

Daten auf einer Partition oder SSD löschen & TRIM auslösen

1)

Daten auf Partition /dev/sda2 löschen und **TRIM** auslösen:

```
blkdiscard /dev/sda2
```

2)

manuell freie Bereiche eines Dateisystems "trimmen"

3)

freie Bereiche des Dateisystems unter /mnt/ext4 „trimmen“:

```
fstrim -v /mnt/ext4
```

ausgabe

```
/mnt/ext4: 9,3 GiB (9950195712 bytes) trimmed
```

4)

regelmäßig "trimmen"

Laut [Debian Wiki](#) ab Version 10 (Buster) nicht mehr nötig, unter Ubuntu 18.04 default:

```
systemctl edit --force fstrim.timer
```

[/etc/systemd/system/fstrim.timer](#)

```
[Unit]
Description=Discard unused blocks once a week
Documentation=man:fstrim

[Timer]
OnCalendar=weekly
AccuracySec=1h
Persistent=true

[Install]
WantedBy=timers.target
```

```
systemctl cat fstrim.timer
```

```
systemctl edit --force fstrim.service
```

[/etc/systemd/system/fstrim.service](#)

```
[Unit]
Description=Discard unused blocks

[Service]
Type=oneshot
ExecStart=/sbin/fstrim -av
```

```
systemctl cat fstrim.service
```

kontinuierlich freie Bereiche eines Dateisystems "trimmen"

TODO: verlangsamt laut diversen Webseiten die Geschwindigkeit in der Dateien gelöscht werden, da nach dem Löschen jeder Datei die SSD informiert wird → prüfen

```
mount -o discard /dev/sda2 /mnt/ext4
```

5)

kontinuierlich freie Bereiche im LVM "trimmen"

Beim löschen und verkleinern von Logical Volumes frei werdende Bereiche „trimmen“: TODO: warum ist das nicht default?

</etc/lvm/lvm.conf>

```
...
devices {
...
issue_discards = 1
...
}
```

1)

Optional (damit man es besser sieht) Einsen auf die Partition schreiben:

```
tr '\0' '\377' < /dev/zero | dd of=/dev/sda2 bs=1MiB status=progress
fdisk -l /dev/sda
```

ausgabe

Device	Boot	Start	End	Sectors	Size	Id	Type
...							
/dev/sda2		444416	21415935	20971520	10G	83	Linux

```
hdparm --read-sector 444416 /dev/sda
hdparm --read-sector 21415935 /dev/sda
```

→ sollte lauter Einsen anzeigen TODO: hier prüfe ich nur den ersten und den letzten Sektor, besser wäre ein Test ob alle Sektoren Einsen enthalten

2)

Prüfen:

```
hdparm --read-sector 444416 /dev/sda
hdparm --read-sector 21415935 /dev/sda
```

→ sollte lauter Nullen anzeigen TODO: hier prüfe ich nur den ersten und den letzten Sektor, besser wäre ein Test ob alle Sektoren Nullen enthalten

3)

Optional: Datei mit lauter Einsen anlegen und wieder löschen:

```
tr '\0' '\377' < /dev/zero | dd of=file1 count=100 bs=512k iflag=fullblock
oflag=direct
```

```
od -x file1
```

ausgabe

```
00000000 ffff ffff ffff ffff ffff ffff ffff ffff
*
24000000
```

```
hdparm --fibmap file1
```

ausgabe

```
...
byte_offset  begin_LBA  end_LBA  sectors
           0  14874624  14884863  10240
```

```
rm file1
hdparm --read-sector 14874624 /dev/sda
hdparm --read-sector 14884863 /dev/sda
```

→ sollte lauter Einsen ausgeben

4)

Prüfen:

```
hdparm --read-sector 14874624 /dev/sda
hdparm --read-sector 14884863 /dev/sda
```

→ sollte lauter Nullen ausgeben

5)

Prüfen:

```
Datei mit lauter Einsen anlegen und wieder löschen:
tr '\0' '\377' < /dev/zero | dd of=file1 count=100 bs=512k iflag=fullblock
oflag=direct
od -x file1
```

ausgabe

```
00000000 ffff ffff ffff ffff ffff ffff ffff ffff
*
24000000
```

```
hdparm --fibmap file1
```

ausgabe

```
...
```

byte_offset	begin_LBA	end_LBA	sectors
0	14874624	14884863	10240

```
hdparm --read-sector 14874624 /dev/sda
hdparm --read-sector 14884863 /dev/sda
```

→ sollte lauter Einsen ausgeben

```
rm file1
sync
```

Prüfen:

```
watch hdparm --read-sector 14874624 /dev/sda
```

→ sollte (spätestens nach ein paar Minuten) lauter Nullen ausgeben

From:
<https://wiki.lab.linuxhotel.de/> - **Linuxhotel Wiki**

Permanent link:
<https://wiki.lab.linuxhotel.de/doku.php/lpi2:ssd?rev=1546439613>

Last update: **2019/01/02 16:28**

