

Dateisysteme:

Erstellen von Dateisystemen: Beispiel eine SATA-Platte

Ext2:

mkfs.ext2 /dev/sda1 → Erstellt ein ext2 Dateisystem auf sda1

mke2fs /dev/sda1 → Erstellt ein ext2 Dateisystem auf sda1

Optionen zu "mke2fs"

- m 15** → Reserviert 15 % des Platzes für Root
- c** → Die Platte wird vor dem erstellen des Dateisystems, einem Read-Test unterzogen
- c -c** → Die Platte wird vor dem erstellen des Dateisystems, einem Read-Write-Test unterzogen
- F** → Erstellt das Dateisystem auch, wenn es sich nicht um ein Block-Device handelt
- L TESTDISK** → Gibt dem Dateisystem das Label "TESTDISK"
- v** → Verbose!

Ext3:

mkfs.ext3 /dev/sda1 → Erstellt ein ext3 Dateisystem auf sda1

mkfs -t ext3 /dev/sda1 → Erstellt ein ext3 Dateisystem auf sda1

mke2fs -j /dev/sda1 → Erstellt ein ext3 Dateisystem auf sda1

Reiserfs:

mkfs.reiserfs /dev/sda1

mkfs -t reiserfs /dev/sda1

mkreiserfs /dev/sda1

Optionen zu "mkreiserfs"

- b 512** → Stellt die Blockgröße auf 512 byte
- l TESTDISK** → Gibt dem Dateisystem das Label "TESTDISK"
- j /dev/sda2** → Das Journal für Das Dateisystem wird auf /dev/sda2 geschrieben
- f** → Erstellt das Dateisystem auch, wenn es sich nicht um ein Block-Device handelt

Ändern von ext2/3 Dateisystemoptionen:

tune2fs -C 2 → Endert den Mountcount auf 2 (Das dateisystem denkt es wäre erst 2 mal gemountet worden, seit dem letzten "fsck")

tune2fs -c 30 → Das Dateisystem wird alle 30 mounts mit "fsck" geprüft

tune2fs -i 20 → Das Dateisystem wird alle 20 Tage mit "fsck" geprüft

tune2fs -L → Ändert das Label

tune2fs -m → Ändert die Reservierung für Super-User

tune2fs -M → Ändert den "Last mountet on" eintrag im Superblock

tune2fs -o → Ändert die Default-Mount Optionen

`tune2fs -j` → macht aus einem ext2 ein ext3 Dateisystem

Ändern von Reiser Dateisystemoptionen:

`reiserfstune -l` → Ändert das Label
`reiserfstune -j` → Ändert den Speicherort des Journals

Dateisysteme in Imagedateien: z.B.:

`dd if=/dev/zero of=/tmp/part1.img bs=1M count=500`

zum erzeugen einer 500 MB Großen Datei. Diese Datei nun mit einer Device-Datei verbinden:

`losetup /dev/loop0 /tmp/part1.img`

Nun kann der Befehl "`mkfs.<dateisystem>`" auf diese Device-Datei angewendet werden. z.B.: für ext3

`"mkfs.ext3 /dev/loop0"` oder `"mke2fs -j /dev/loop0"`

Das ganze wäre auch möglich ohne eine Verbindung zur Device-Datei z.B.:

`"mkfs.ext3 -F /tmp/part1.img"` oder `"mke2fs -j -F /tmp/part1.img"`

Nun kann entweder das Device `"/dev/loop0"`

`mount /dev/loop0 /mnt`

oder Die zuvor erstellte Image-Datei gemountet werden

`mount -o loop /tmp/part1.img /mnt`

So können auch ganze Datenträger als Image auf der Festplatte abgelegt werden und als loop-Device verwendet werden. z.B.: eine CD-ROM

`dd if=/dev/cdrom of=/daten/cdrom.iso`
`losetup /dev/loop1 /daten/cdrom.iso`
`mount -t iso9660 /dev/loop1 /media/cdrom`
`mount -t iso9660 -o loop /daten/cdrom.iso /media/cdrom`

Nun kann ohne das die CD-ROM eingelegt ist, über den Ordner `/media/cdrom` auf das Abbild der CD-ROM zugegriffen werden.

Dateisysteme Überprüfen:

ext2:

`fsck /dev/sda1`
`fsck.ext2 /dev/sda1`
`fsck -t ext2 /dev/sda1`
`e2fsck /dev/sda1`

ext3:

```
fsck /dev/sda1  
fsck.ext3 /dev/sda1  
fsck -t ext3 /dev/sda1  
e2fsck /dev/sda1
```

reiserfs:

```
fsck /dev/sda1  
fsck.reiserfs /dev/sda1  
fsck -t reiserfs /dev/sda1  
reiserfsck /dev/sda1
```

Der befehl "fsck" testet das Dateisystem und entscheidet dann welches Programm zur Überprüfung gestartet werden soll.
(fsck.<dateisystem>)

Fsck ohne Option Schaut nur in den Superblock ob der Filesystem state auf clean steht. Ist dies der Fall wird das Dateisystem nicht überprüft.

Optionen für fsck:

```
fsck -f /dev/sda1 → Überprüft das dateisystem auch wenn der  
Filesystem State auf clean steht.  
fsck -a → Überprüft alle in der /etc/fstab einge-  
tragenen Dateisysteme  
fsck -y → Antwortet auf alle Fragen mit yes (!)
```

Defekte Blöcke auf der Festplatte suchen:

```
badblocks /dev/sda1 → Einfacher lese-Test  
badblocks -n /dev/sda1 → Zerstörungsfreier Schreib/Lese Test  
badblocks -w /dev/sda1 → Genauerer Schreib/Lese Test, der alle  
daten auf /dev/sda1 werden Zerstört  
badblocks -w /dev/sda1 -o datei → Defekte Blöcke werden in die  
Datei geschrieben
```

SuperBlock Anzeigen:

```
dumpe2fs -h /dev/sda1  
tune2fs -l /dev/sda1
```

Debugfs:

Debugfs ist ein Tool zur Manipulation des Dateisystems und sollte mit Vorsicht behandelt werden. Debugfs sollte nur auf nicht gemountete Dateisysteme angewendet werden.

```
debugfs /dev/sda1
```

Öffnet den Interaktiven Modus von debugfs, das Dateisystem wird Read-Only verwendet, Änderungen sind somit nicht möglich.

```
debugfs -w /dev/sda1
```

Hier wird das Dateisystem mit Schreibrechten geöffnet.

Im interaktiven Modus von debugfs:

- help** → Eine Liste aller zur Verfügung stehenden Befehle
- lsdel** → Zeigt die I-Node Nummern von gelöschten Dateien
- dump** → Speichert den Inhalt einer gelöschten Datei ab
z.B. "dump <14> /tmp/test.txt"
- undel** → Stellt eine gelöschte Datei unter dem angegebenen
Namen wieder her. z.B.:
"undel <14> backup.txt"