

Image-Erstellung mit dd

Inhaltsverzeichnis

- 1 Image-Erstellung mit dd
 - 1.1 ISO-Images und CD/DVD RAW-Images
 - 1.2 Diskettenimages
 - 1.3 Festplattenimages
 - 1.4 Festplattenimages mounten
 - 1.5 Festplattenreplikation
- 2 Bootsektorsicherung
- 3 Alternativen zu dd
- 4 Datenvernichtung(!) und Festplattenrecycling
- 5 Siehe auch
- 6 Weblinks

Image-Erstellung mit dd

Mit Hilfe von dd lassen sich Images diverser Medien erstellen. Ob es sich dabei um Abbilddateien einer Festplatte, Diskettenimages oder ISO-/RAW-Images einer CD oder DVD erstellen ist für den eigentlichen syntaktischen Aufbau des Befehls unerheblich.

Der Befehl dd ist nicht speziell für diesen Zweck konzipiert, vielmehr tut er nichts anderes, als Daten auf eine bestimmte Weise von einer Datei in die andere zu schaufeln. Der Umstand, dass jedes Gerät unter Unix und unixoiden Systemen wie Linux als herkömmliche Datei (z.B. /dev/dvd, /dev/sda) behandelt wird, macht den Einsatz von dd in der hier beschriebenen Weise möglich.

ISO-Images und CD/DVD RAW-Images

Für die Erstellung eines kompletten Abbilds einer CD/DVD dient beispielsweise folgender Befehl:

```
dd if=/dev/dvd of=/pfad/wo/das/Image/hin/soll/Image.img
```

Danach sollte man mit Hilfe des befehls cmp überprüfen, ob beim Auslesen des Ursprungsmediums auch alle Daten korrekt gelesen wurden.

```
cmp Image.img /dev/dvd
```

Im Falle einer Daten-CD/DVD ist das Ergebnis ein ISO-Image, es kann also mit

```
mv Image.img Image.iso
```

problemlos den passenden Namen bekommen. Wenn es sich jedoch um eine SVCD gehandelt haben sollte, sollte man bedenken, dass oben im dd-Befehl keine Blocksize angegeben wurde und dd in diesem Fall die Blocksize selbst herausgefunden hat, die bei SVCDs nun mal keine 2048 wie in einem ISO-Format beträgt, sondern 2352. Dann liegt natürlich kein wirkliches ISO-Image vor sondern ein RAW-Image, welches dann als solches durchaus auch per K3b oder cdrdao wieder auf eine CD gebrannt werden könnte. Mit dd selbst kann jedoch nicht auf CD oder DVD gebrannt werden.

Diskettenimages

Mit dem dd Befehl kann man jedoch auch Diskettenimages anlegen und auch vorhandene Diskettenimages auf Diskette zurückkopieren. Der Befehl zum Anlegen lautet auch hier wieder:

```
dd if=/dev/fd0 of=/pfad/wo/das/Image/hin/soll/Diskimage.img
```

Der Befehl zum zurück kopieren eines solchen Diskettenimages lautet dann passenderweise:

```
dd if=Diskimage.img of=/dev/fd0
```

Dabei ist es unerheblich, ob auf der Diskette bereits ein Dateisystem existiert oder nicht, da dies gnadenlos von dd überschrieben wird. Man sollte jedoch auch hier wieder mit cmp überprüfen, ob die Daten korrekt gelesen bzw. geschrieben wurden. Sollte ein auf Diskette zurückgeschriebenes Image Fehler aufweisen, so sollte die entsprechende Diskette umgehend entsorgt werden, da dann Medienfehler die Ursache sein dürften und diese wohl kaum reparierbar sind (es sei denn, man formatiert die Diskette mehrfach im sogenannten Lowlevel, also ohne Dateisystem, was aber recht zeitaufwändig ist und den Aufwand gegenüber dem Wert einer Diskette meist nicht rechtfertigt).

Festplattenimages

Des Weiteren dient dd auch zum Erstellen von Festplattenimages. Man kann damit ganze Festplatten in ein RAW-Image sichern:

```
dd if=/dev/hdb of=/pfad/wo/das/Image/hin/soll/hdb-Image.img
```

Oder beispielsweise einzelne Partitionen:

```
dd if=/dev/hda6 of=/pfad/wo/das/Image/hin/soll/hda6-Image
```

Achtung: Die zu kopierende Festplatte/Partition darf hierbei *nicht* gemountet sein, da sonst etwaige Schreibzugriffe während des Kopievorgangs zu defekten Image-Dateien führen können!

Festplattenimages mounten

Um erstellte Festplattenimages auf einem anderen Linuxsystem zu mounten, kann man wie folgt vorgehen: image = Pfad zum Image z.B.: /mnt/server.img

```
# kpartx -a -v image
add map loop0p1 (253:0): 0 433692 linear /dev/loop0 63
add map loop0p2 (253:1): 0 64260 linear /dev/loop0 433755
#
# ls -l /dev/mapper/
insgesamt 0
lrwxrwxrwx 1 root root    16 7. Dez 02:18 control -> ../device-mapper
brw----- 1 root root 253, 0 7. Dez 14:13 loop0p1
brw----- 1 root root 253, 1 7. Dez 14:13 loop0p2
```

Nun kann man sich seine Partition mounten

```
# mount /dev/mapper/loop0p1 /mnt/
```

Festplattenreplikation

Man kann jedoch diesen Befehl ebensogut zum 1:1 kopieren von ganzen Festplatten nutzen indem man identische(!) Festplatten in ein System integriert, beispielsweise mit externen USB-Festplattengehäusen, und dann die Rohdaten einfach 1:1 per dd vom Eingangsmedium auf das Ausgangsmedium schaufelt:

```
dd if=/dev/sda of=/dev/sdb
```

Je nach Festplattengröße dauert das natürlich einige Zeit.

Festplattenimages lassen sich aber auch direkt komprimiert anlegen indem man die Angabe über das Ausgangsmedium weglässt und die daraus resultierende Ausgabe an einen Packer piped:

```
dd if=/dev/hda | gzip --best >hda-Image.gz
```

Zurückkopieren ließe sich das dann mit:

```
gunzip -c hda-Image.gz | dd of=/dev/hda
```

Mit Hilfe der Pipefähigkeit von dd lassen sich durchaus komplexe Scripte erstellen mit deren Hilfe z.B. solche Images über Netzwerke hinweg angelegt und zurückgespielt werden können, siehe dazu auch in der [manpage](#) von netcat.

Bootsektorsicherung

Mit Angabe der Blockgröße und der Anzahl der zu lesenden Blöcke kann man per dd aber auch den Masterbootrecord einer Festplatte sichern um zum Beispiel [GRUB](#) zu sichern.

```
dd if=/dev/hda bs=512 count=1 of=grub.img
```

Zurückspielen eines so angelegten Images geht äquivalent zum Erstellen mit:

```
dd if=grub.img bs=512 count=1 of=/dev/hda
```

Alternativen zu dd

Die Alternative zu dd für den Fall, dass es sich um Medien handelt, die Fehler aufweisen, lautet ddrescue. Dessen Syntax ist identisch zu dd, jedoch muss man dabei beachten, dass beim Einlesen im Fehlerfall natürlich die an der defekten Stelle ehemals vorhandenen Daten nicht aus dem Nichts heraus restauriert werden können, sondern durch ddrescue schlichtweg durch Nullen ersetzt werden. Des Weiteren kann es dabei auch je nach Fehlerart zwingend notwendig werden die Blockgröße sowie die Anzahl der zu lesenden Datenblöcke anzugeben.

Datenvernichtung(!) und Festplattenrecycling

Der Befehl dd kann auch dazu genutzt werden alte Daten unwiderruflich zu löschen, ähnlich wie wipe für Dateien. Benötigt wird so was z.B. wenn Firmen Altgeräte weiterverkaufen wollen aber auch sichergehen wollen, dass die ehemals gespeicherten sensiblen Firmendaten nicht in die falschen Hände geraten können.

```
dd if=/dev/zero of=/dev/sda
```

Auch wenn hiermit ausgeschlossen ist, dass einzelne Dateien oder gar die gesamte Festplatte wieder herstellbar sind, so kann man dennoch – aus „psychologischen Gründen“, /dev/urandom verwenden.

```
dd if=/dev/urandom of=/dev/sda
```

Ein Wiederholen dieses Vorgangs hat in beiden Fällen allerdings nur einen Effekt: Man verschwendet Zeit. Aufgrund der Bauart heutiger Platten ist es praktisch ausgeschlossen, dass bereits nach einmaligem überschreiben mit Nullen noch Daten wiederhergestellt werden können. (vgl. verlinkter Artikel aus [heise security](#)).

Siehe auch

- [ISO-Image mounten](#)

Weblinks

- <http://www.linux-user.de/ausgabe/2004/08/048-dd-rescue> 
- [Sicheres Löschen: Einmal überschreiben genügt](http://www.heise.de/security/meldung/Sicheres-Loeschen-Einmal-ueberschreiben-genuegt-198816.html) (<http://www.heise.de/security/meldung/Sicheres-Loeschen-Einmal-ueberschreiben-genuegt-198816.html>) 
- [Images unter Linux erstellen](http://www.wiggleLeaks.de/35/images-unter-linux-erstellen.html) (<http://www.wiggleLeaks.de/35/images-unter-linux-erstellen.html>) 

Dieser Artikel (oder Teile davon) ist eine Kopie des Artikels [Imageerstellung mit dd und ddrescue](#) (http://wiki.linux-club.de/Imageerstellung_mit_dd_und_ddrescue) aus dem [Linux Club Wiki](#) (<http://wiki.linux-club.de/Hauptseite>) und steht unter der [GNU-Lizenz für freie Dokumentation](#). Im Linux Club Wiki ist eine [Liste der Autoren](http://wiki.linux-club.de/index.php?title=Imageerstellung_mit_dd_und_ddrescue&action=history) (http://wiki.linux-club.de/index.php?title=Imageerstellung_mit_dd_und_ddrescue&action=history) verfügbar.

Abgerufen von „https://wiki.archlinux.de/index.php?title=Image-Erstellung_mit_dd&oldid=19767“

- Diese Seite wurde zuletzt am 29. Juni 2017 um 12:14 Uhr bearbeitet.
- Der Inhalt ist verfügbar unter der Lizenz [GNU Free Documentation License 1.2](#), sofern nicht anders angegeben.