

wendende Netzwerkschnittstelle ist die zusätzliche Gbit-cross-over-Verbindung.
Eine funktionierende Konfiguration von DRBD auf beiden Rechnern sieht wie folgt aus:

```
CONVG (etc/ha.d/ha.cf)
resource r1
  protocol C
  onco-dgr-cmd 'echo |DRBD| pri on con-dgr | wall : sleep 60 : halt -f'
  sampl 1
  wfc-timeout 60;
  dgr-wfc-timeout 120; # 2 minutes.
}
disk1
on-to-error detach;
}
net1
# timeout 60; # 6 seconds. (unit = 0.1 seconds)
on-fs-connect reconnect;
}
sync1
rate 30M;
group 1;
}
on links1
device /dev/drbd0;
disk /dev/sda3;
address 192.168.0.1:7788;
meta-disk internal;
}
on rechts1
device /dev/drbd0;
disk /dev/sda3;
address 192.168.0.2:7788;
meta-disk internal;
}
}
```

Listing 2

DRBD-Konfigurationsdatei

Die wichtigsten Parameter in /etc/drbd.conf

resource	Der Name des entsprechenden DRBD-Dienstes.
protocol	Standardprotokoll ist „C“ „A“ und „F“ sind nicht-richtig für synchrone Synchronisation und werden hier nicht weiter besprochen.
onco-dgr-cmd	Aktion im Fall der Feststellung einer „soft-heart“-Situation, also wenn beide Rechner primär sind. Standard ist; den entsprechenden Rechner in Moduli zu stoppen.
on-to-error	Was passieren soll, wenn eine Fehlerliste einen File/Ausgabefehler verursacht. Standard ist, das DRBD-Gerät zu dekonfigurieren.
on-disconnect	Definiert die Aktion, die unternommen wird, falls der sekundäre Rechner über das Netzwerk nicht erreichbar ist. Standard ist hier der Versuch, die Verbindung wiederherzustellen.
rate	Die Geschwindigkeit bei der Synchronisierung. Achtung: Die Angabe erfolgt in Byte pro Sekunde.
group	Falls mehrere Ressourcen definiert werden, kann mit diesem Parameter die Priorität festgelegt werden, welche Ressource nach einem Ausfall als erstes wieder synchronisiert werden soll.
meta-disk	Definiert, wo die Metadaten gespeichert werden. Auf kleineren Systemen ist dies üblicherweise „internal“, also auf derselben Partition. Alternativ kann eine eigene Partition pro Ressource angegeben werden. Vor allem bei größeren Systemen mit vielen Ressourcen und Partitionen kann dies geschwindigkeitssteigernd wirken.

Nun ist DRBD fertig konfiguriert und kann gestartet werden. Dazu muss auf beiden Servern der Befehl „`./etc/init.d/drbd start`“ ausgeführt werden. Falls keine Fehlermeldung erscheint, sind die beiden Server miteinander verbunden. Um die Daten das erste Mal zu synchronisieren, muss auf dem primären Server „`links`“ das DRBD-Gerät in den Status „`primary`“ gebracht werden. Da DRBD

zu diesem Zeitpunkt nicht feststellen kann, welcher Server der primäre ist, muss dies manuell erzwingen (forciert) werden. Dazu muss folgender Befehl eingegeben werden:

```
SHR1
links--# drbdsetup /dev/drbd0 primary --do-walr 1-sy

Listing 3
Unmittelbar nach Eingabe dieses Befehls beginnt die Synchronisierung. Der Fortschritt kann mit folgendem Befehl beobachtet werden:
SHR1
links--# cat /proc/drbd
version: 0.7.17 (api:77/proto:74)
SVN Revision: 2125 build by mlh@chp, 2006-03-27 17:40:22
0: csSyncTarget=Primary/Secondary/AllConsistent
net1 nr3028 drw3628 drd0 al0 km48 loc0 ped0 us0 ap0
  <-----> 1 sync ed. 1.1% (989428/393950K)
  finish: 0:01:37 speed: 3.628 B.628) K/sec
links--#
```

Listing 4

Der Erfolg der Synchronisation kann mit demselben Befehl kontrolliert werden:

```
SHR1
links--# cat /proc/drbd
version: 0.7.17 (api:77/proto:74)
SVN Revision: 2125 build by mlh@chp, 2006-03-27 17:40:22
0: csConnected=Primary/Secondary/AllConsistent
net0 nr393056 drw393056 drd0 al0 km72 loc0 ped0 us0 ap0
links--#
```

Listing 5

Das spezielle Augenmerk liegt bei „`Consistent`“. Das bedeutet, dass beide Rechner über denselben Datenbestand verfügen.

Nachdem die Synchronisation beendet wurde, kann das Dateisystem der Wahl erstellt werden. Dies geschieht auf die übliche Art und Weise, außer dass ein DRBD-Gerät anstelle einer Festplattepartition angegeben wird. Das Erstellen des Dateisystems muss nur auf einem Rechner, in diesem Beispiel auf dem primären Rechner „`links`“, erfolgen:

```
SHR1
links--# mkfs.ext3 /dev/drbd0
links--# tune2fs -o /dev/drbd0
```

Listing 6

Der zweite Befehl verhindert, dass das Dateisystem beim Einhängen unnötig neu geformatiert wird. Nach dem Anlegen des Dateisystems und des entsprechenden „`Mountpoints`“ kann das Gerät eingehängt werden:

```
SHR1
links--# mkdir /data
```

Listing 7

```
SHR1
links--# mount /dev/drbd0 /data
```

Listing 8

Falls dies ohne Fehlermeldung quittiert wurde, ist die wichtigste Voraussetzung für einen HA-Cluster bereits erreicht: ein funktionierendes, verteiltes Blockgerät.

Jetzt, da das DRBD-Gerät bereitsteht, können die Datenverzeichnisse und die entsprechenden Symlinks angelegt werden.

Erstellen der Datenverzeichnisse und Symlinks

In diesem Tutorial wird davon ausgegangen, dass das DRBD-Gerät in `/data` eingehängt wird. Dienste wie Apache oder MySQL speichern ihre Daten standardmäßig, aber an anderen Orten. Es gibt zwei Möglichkeiten, den Diensten den Standort ihrer Daten mitzuteilen: Erstens das Erstellen von Symlinks und zweitens die Anpassung in den Konfigurationsdateien. Ersteres ist zu bevorzugen, da dadurch die Originalkonfigurationen der Dienste nicht angepasst werden müssen.

Bei vielen neueren Linux-Distributionen wie OpenSuse sind die Daten von Apache in `/srv/www` abgelegt. In anderen, wie Debian Sarge oder CentOS unter `/var/www`. Bei MySQL ist meistens `/var/lib/mysql` als Ort der Datenspeicherung vorgesehen.

Erstellen der Verzeichnisse und Symlinks auf dem Rechner „`links`“:

```
SHR1
links--# mkdir /data/srv
links--# mkdir -p /data/var/lib
links--# mv /srv/www /data/srv/
links--# ln -s /data/srv/www /srv/www
links--# mv /var/lib/mysql /data/var/lib
links--# ln -s /data/var/lib/mysql /var/lib/mysql
```

Listing 9

Auf dem Rechner „`rechts`“ sind nur die Symlinks anzulegen. Diese Symlinks zeigen allerdings ins Leere, solange das DRBD-Gerät dort nicht eingehängt ist. Fehlermeldungen können ignoriert werden:

```
SHR1
rechts--# ln -s /srv/www
rechts--# ln -s /data/srv/www /srv/www
rechts--# ln -s /var/lib/mysql
rechts--# ln -s /data/var/lib/mysql /var/lib/mysql
```

Listing 10

Bevor nun Heartbeat konfiguriert und ein erstes Mal gestartet werden kann, muss das Dateisystem wieder getrennt und das DRBD-Gerät in den Status „`secondary`“ gebracht werden:

```
SHR1
links--# umount /data
links--# drbdadm secondary 0
```

Listing 11

Installation und Konfiguration von Heartbeat

Nachdem DRBD erfolgreich installiert und initialisiert sowie die Datenverzeichnisse erstellt wurden, kann der nächste Schritt in Angriff genommen werden: die Installation und Konfiguration von Heartbeat.

Heartbeat überwacht die Server und verwaltet die Cluster-Ressourcen und Dienste. Die meisten Linux-Distributionen enthalten vorkompilierte Pakete von Heartbeat, darum wird hier auf die Anleitung zum Kompilieren verzichtet. Wie Heartbeat installiert wird, hängt von der Distribution ab. Die entsprechende Anleitung kann im Handbuch der verwendeten Distribution gefunden werden.

Die Konfigurationsdateien sind in der Regel alle im eigenen Verzeichnis `/etc/ha.d/` zu finden. Heartbeat benötigt drei Konfigurationsdateien für den Betrieb:

- `/etc/ha.d/authkeys`
- `/etc/ha.d/ha.cf`
- `/etc/ha.d/haresources`

Diese Konfigurationsdateien müssen bis auf die Ausnahme der Datei „`ha.cf`“ auf beiden Servern identisch sein. Der Unterschied in der „`ha.cf`“ beschränkt sich auf die IP-Adresse des Partnerrechners. Die Datei „`authkeys`“ enthält die Hashmethode und einen Schlüssel, mit dem die Kommunikation zwischen den beiden Servern rudimentär verschlüsselt wird.

CONVG (etc/ha.d/authkeys)

```
auth 1
1 sha1 supersecret
```

Listing 12

In der zweiten Datei, `/etc/ha.d/ha.cf` (an dieser Stelle auf das Wesentliche gekürzt), werden die Netzwerkverbindungen und das Erkennen und Verhalten bei Ausfällen definiert.

CONVG (etc/ha.d/ha.cf)

```
logfile /var/log/ha-log
keepalive 2
deadtime 30
initdead 120
warnname 5
serial /dev/ttyS0
auto_failback off
ucast eth0 10.0.0.1
ucast eth1 192.168.0.2
node links
node rechts
```

Listing 13

Die wichtigsten Konfigurationsparameter in `/etc/ha.d/ha.cf` sind:

keepalive	Der zeitliche Abstand zwischen zwei Pings.
deadtime	Definiert die Zeit, nach welcher der zweite Server als tot deklariert wird.
initdead	Beim Start von Heartbeat wird etwas länger gewartet, bevor der zweite Server als tot deklariert wird.
warnname	Nach der Zeit, nach der kein Ping zurückkam, wird eine Warnung in die Logdatei geschrieben (siehe Heartbeat).
auto_failback	Definiert, ob nach der Wiederverfügbarkeit des primären Servers die Dienste automatisch auf den primären Server zurückschalten. Dies ist nützlich, wenn der sekundäre Server schlechter ausgebaut ist als der primäre.
ucast	Definiert die Netzwerkschnittstelle und die IP-Adresse des jeweils anderen Servers. Standardmäßig wird <code>ucast</code> (Multicast) verwendet, was allerdings nicht empfohlen wird. Die Anzahl der Linien mit diesem Parameter korrespondiert mit der Anzahl verwendeter Ethernet-Schnittstellen.
node	Bei den beiden Zeilen beginnend mit „ <code>node</code> “ werden die Namen der beiden Server angegeben. Achtung: Diese Namen müssen mit der Angabe des Befehls „ <code>uname -n</code> “ übereinstimmen.

Die dritte und wichtigste Datei `/etc/ha.d/haresources` enthält die Definitionen der Ressourcen, wie Service-IP-Adressen und die zu startenden Dienste.

CONVG (etc/ha.d/haresources)

```
links 10.0.0.3 192.168.0.31
drbdadm{c,r0}
Filesystem:/dev/drbd0:/data:ext3 \
```