Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Setting up Lustre MDS/MDT Server with DRBD and Heartbeat.
This is meant as a general setup procedure.

Your mileage may vary.

Content

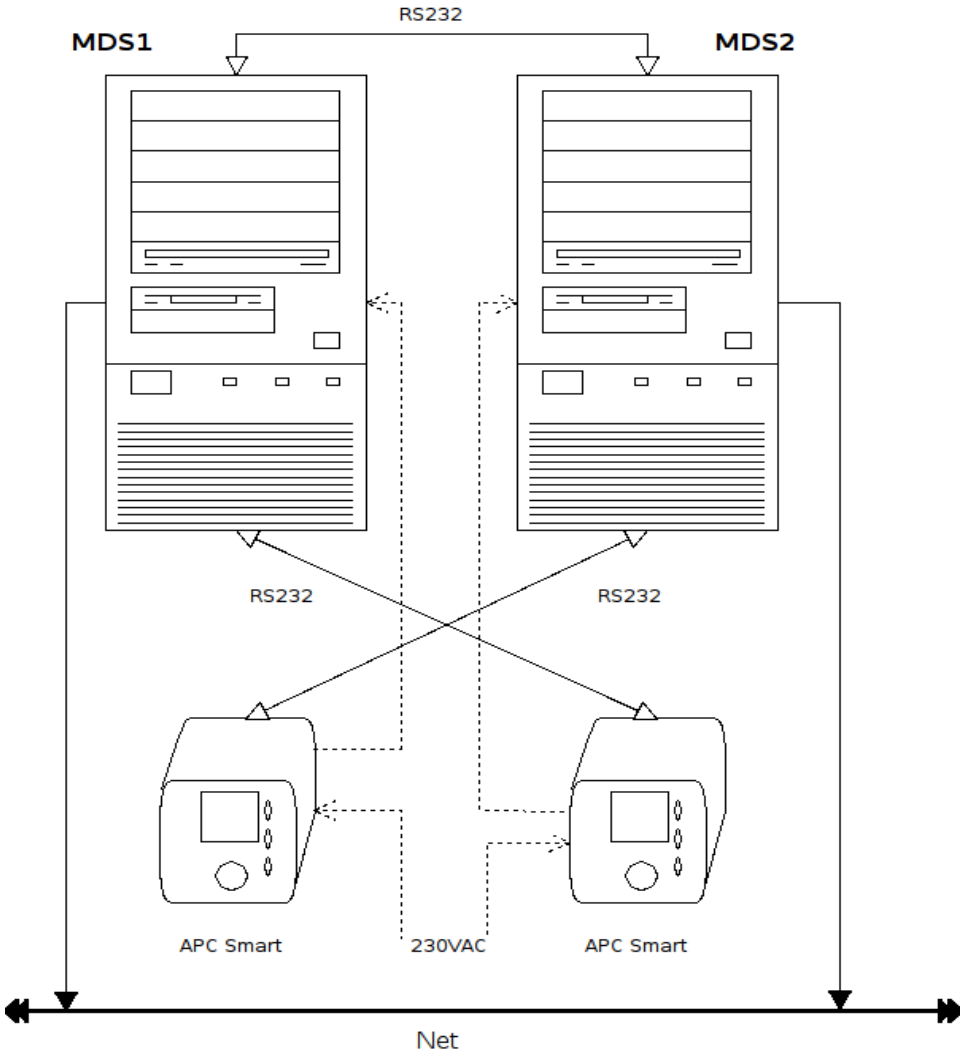
Scenario....2
Partitions....3
MDADM....4
DRBD....5
Lustre....7
HEARTBEAT....8
STONITH....10
Starting....11
Stop....12
Troubleshooting....13

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Scenario:

Hardware: 2 Servers with 2 SATA Disks for MDS/MDT, 2 Serial Ports, 1 GB Network Port

System used:
Gentoo 2008.0
Kernel vanilla with lustre patches 2.6.22.19
lustre source 1.6.5.1, drbd 8.0.12, md tools 2.6.4, heartbeat 2.0.8



Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Partitions:

We place the MGS and MDT on one machine (Yes, we know that this is not the lustre way. But we use the lustre system as a huge data archive with very little write access during the day).

We couldn't use /dev/sdb or /dev/sdc natively and had to create partitions. Check out if you can get along with /dev/sdb etc. It is recommended by the lustre dev team.

Create partitions on MDS1 and MDS2 /dev/sdb resp. /dev/sdc.

The general device generation of this scenario is:
/dev/sdb1 + /dev/sdc1 --> /dev/md0 --> /dev/drbd0 (the lustre device in the end)

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

MDADM setup with Raid1:

On MDS1 and MDS2 enter:
mdadm --create --verbose /dev/md0 –level=1 --raid-devices=2 /dev/sdb1 /dev/sdc1
Save the config:
mdadm --detail --scan > /etc/mdadm.conf

evtl. start the Raid manually when no startscript is given:
mdadm -As /dev/md0
Evtl. add this to the Startscript to get mdadm starting the raid correctly:
mdadm --assemble /dev/md0

Check if md is up:
cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 sdb1[0] sdc1[1]
      185558656 blocks [2/2] [UU]

unused devices: <none>

Content of /etc/mdadm.conf
ARRAY /dev/md0 level=raid1 num-devices=2 UUID=d048e8ee:7ec085a8:167087a3:27dba536
(Your UUID is different of course)

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

DRBD (needs network up and running):
Long intro:
We couldn't get an additonal net aka 10.0.0.x for DRBD (as stated in the examples) up and running because even when an IP(number) connection in the config had been configured drbd wanted the hosts resolved by name. So we decided that drbd should share the net with lustre. So we might be bashed here by the drbd gurus.
Create the config file given below in /etc.

We do use Primary/Secondary setup. Not Primary/Primary. This is meant for something different.
On both mds do (You may choose a different resource name than r0, name it in /etc/drbd.conf):
drbdadm create-md r0

The DRBD Raid is now inconsistent on mds1 and mds2, so who is our clean primary ? :
mds1 ~# drbdadm primary all (This may not be enough.Check the status for the drbd device.)
(drbdadm state r0)

You may need to force now the primary state on mds1 and just overwrite the data on mds2:
mds1 ~# drbdadm -- --overwrite-data-of-peer primary drbd0
and
mds1 ~# drbdadm primary all

check status on mds1:
drbdadm state r0
output --> Primary/Secondary (or Primary/Inconsistent if no sync yet)
(The DRBD notification is: localnode/remotenode. So the output on mds2 is switched.)

cat /proc/drbd gives full info what is going on
Let the sync finish before continuing !
Othwerwise Gremlins are known to attack your system.

NOTE: If you don't have a clean primary/secondary state here there is no use in carrying on. It won't just work.

Our content of /etc/drbd.conf on mds1 and mds2:

```
#
global { usage-count no; }

common {  syncer {  rate 100M; } }

resource r0 {
       protocol C;
       startup {
         degr-wfc-timeout 60;
         wfc-timeout 120;
#          wait-after-sb;
```

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

```
        }
        disk {
                on-io-error   detach;
        }
        net {
        #       allow-two-primaries; ← DON'T
                after-sb-0pri discard-younger-primary;# Auto sync from the node that was primary before
the split brain situation happened.
                after-sb-1pri discard-secondary;      # Always honour the outcome of the after-sb-0pri
algorithm
                after-sb-2pri call-pri-lost-after-sb; # reboot after sb pri lost
        }
        on mds1 {
                device    /dev/drbd0;
                disk      /dev/md0;
                address   192.168.16.122:7789;
                meta-disk internal;
        }
        on mds2 {
                device    /dev/drbd0;
                disk      /dev/md0;
                address   192.168.16.124:7789;
                meta-disk internal;
        }
}
```

Please check the docs for additonal infos on the net{} section. The setup above is for what to do in case of the various split-brain and recovery situations.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Lustre:

Now create lustre MDS/MDT.

Make sure you have the lustre e2fsprogs installed !
Otherwise heartbeat will not mount /dev/drbd0 and you get endless failover between mds1 and mds2. Heartbeat does a file system check on starting up and needs a success there.

i.e. On MDS1 do:
mkfs.lustre --fsname=foo --mdt --mgs --failnode=mds2 /dev/drbd0

There is no need to format the /dev/drbd0 on MDS2.
DRBD takes care of the mirroring.

Check the status of drbd „cat /proc/drbd“ .
If it is still „syncing“ than let it finish before continuing.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

HEARTBEAT (after drbd has come up and is running !):
We manually manage the IP's of the Network Interfaces and not let HA do it.
Also we use the Heartbeat Version1 Config file style.
RS232 and net used here for heartbeating. Yes, we know that there are various pros and cons of how to do it differently especially avoiding RS232.

/etc/ha.d/ha.cf(_Must_ be the same on MDS1 and MDS2):

```
#logfacility local7
# logfile /var/log/ha-log
# debugfile /var/log/ha-debug
use_logd yes
udpport 694
keepalive 1 # 1 second
deadtime 30
initdead 80
bcast eth1
serial /dev/ttyS0 #if you use serial
baud 19200 # if you use serial
node mds1
node mds2
# crm yes
crm no
# crm respawn
auto_failback yes
# watchdog /dev/watchdog
stonith_host mds1 apcsmart /dev/ttyS1 mds2
stonith_host mds2 apcsmart /dev/ttyS1 mds1
```

/etc/ha.d/haresources (_Must_ be the same on MDS1 and MDS2):

```
mds1  drbddisk::r0 Filesystem::/dev/drbd0::/mnt/mdsmdt::lustre
```

Here you specify the mountpoint. In this example /mnt/mdsmdt

Check the serial link:
On mds2:

```
cat < /dev/ttyS0
```

On mds1:

```
echo test > /dev/ttyS0
```

The actual mount of the lustre file system to /mnt/mdsmdt (you name it) is done by the heartbeat script.
We specify what to do in the /etc/ha.d/haresources.

The needed scripts "drbddisk" and "Filesystem" should be in /etc/ha.d/resource.d/.
It should be installed automatically when installing heartbeat.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Start heartbeat:
/etc/init.d/heartbeat start

check the messages:
tail -f /var/log/messages

After a while you should get a filesystem check info and (hopefully a success).
After this the /dev/drbd0 gets mounted to /mnt/mdsmdt in this example.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

STONITH (configured in /etc/ha.d/ha.cf)
Power cycle the other node if something gets out of order.

(APC smart here with serial cable)
It's all in the config file:
"stonith_host mds1 apcsmart /dev/ttyS1 mds2
stonith_host mds2 apcsmart /dev/ttyS1 mds1"

meaning „stonith_host mds1 handles the UPS for mds2“ and vice versa. That's it.

Make sure that stonith can find the progs needed. In our case the install tool placed them here:
/usr/lib64/stonith/plugins/stonith2/apcmaster.so
/usr/lib64/stonith/plugins/stonith2/apcsmart.a
/usr/lib64/stonith/plugins/stonith2/apcsmart.so
/usr/lib64/stonith/plugins/stonith2/apcmaster.a
/usr/lib64/stonith/plugins/stonith2/apcsmart.la
/usr/lib64/stonith/plugins/stonith2/apcmaster.la

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Starting:

1) Fire up md raid
2) drbd (may take a few seconds)
and after drbd has finished !
3) heartbeat (may take a few minutes. Be patient. Have a look at the messages.)

In Gentoo you can place the order and dependencies of the startscripts in the startscripts itself:

example from /etc/init.d/heartbeat
<snip>
```
depend() {
        use logger
        need net drbd
        # After DRBD edited by HD
        after drbd
}
```
<snap>
Check out your Distro of how to do it.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Stop:
1) stop heartbeat on MDS2
2) stop drbd on MDS2

Now you can change things on MDS1.
Do not forget to sync the Conf files if needed !

Evtl. do
3) stop heartbeat on MDS1
4) stop drbd on MDS1
if needed for major operation

Have a shell open and watch the messages and/or dmesg on MDS1 and MDS2 for the syncing.

Universität Bremen

Universität Bremen, IUP, Otto-Hahn-Allee 1, 28359 Bremen
Heiko Schröter, schroete@iup.physik.uni-bremen.de

Troubleshooting:

1) Are the config files in sync ?
2) Status of md Raid ?
3) Status of DRBD ?
4) Can Heartbeat do the e2fsck during startup ?
5) Is the serial link ok ?

Test the failover !

1) Kill Heartbeat on MDS1. MDS2 should take over. Restart Heartbeat. MDS1 should be primary again.
2) Reboot MDS1. MDS2 should take over. MDS1 should be primary again after reboot.
3) Unplug network and serial heartbeat link. Is the system stable ?
4) Unplug power.