

Main reference: <http://www.asplund.nu/xencluster/xen-cluster-howto.html>

Only the differences between the above reference and what we have really done will be explained here.

Install and configure DRBD

To install DRBD run both commands below:

```
root@serv1:~# apt-get install drbd8-utils root@serv1:~# apt-get install
drbd8-modules-2.6.26-2-xen-686
```

The configuration file for DRBD is located in `/etc/drbd.conf`. We will now configure it to use the LVs we created and later we will change the Xen configuration of our DomU to use the DRBD device instead of directly the LVs. Edit `/etc/drbd.conf`:

```
root@serv1:~# nano /etc/drbd.conf global { usage-count yes; } common { syncer {
rate 90M; } } resource lennyvm-root { protocol C; startup { } disk { on-io-error
detach; } net { after-sb-0pri discard-zero-changes; after-sb-1pri discard-secondary;
timeout 60; connect-int 10; ping-int 10; max-buffers 2048;
max-epoch-size 2048; } syncer { } on serv1.cae.locale { device /dev/drbd1;
disk /dev/serv1/lennyvm-root; address 192.168.1.100:7789; meta-disk
/dev/serv1/meta[0]; } on serv2.cae.locale { device /dev/drbd1; disk
/dev/serv2/lennyvm-root; address 192.168.1.101:7789; meta-disk /dev/serv2/meta[0];
} } resource lennyvm-swap { protocol C; startup { wfc-timeout 120;
degr-wfc-timeout 120; } disk { on-io-error detach; } net { after-sb-0pri
discard-zero-changes; after-sb-1pri discard-secondary; timeout 60; connect-int 10;
ping-int 10; max-buffers 2048; max-epoch-size 2048; } syncer { } on
serv1.cae.locale { address 192.168.1.100:7790; disk /dev/serv1/lennyvm-swap;
device /dev/drbd2; meta-disk /dev/serv1/meta[1]; } on serv2.cae.locale { device
/dev/drbd2; disk /dev/serv2/lennyvm-swap; meta-disk /dev/serv2/meta[1]; address
192.168.1.101:7790; } } resource lennyvm-home { protocol C; startup { wfc-timeout
120; degr-wfc-timeout 120; } disk { on-io-error detach; } net { after-sb-0pri
discard-zero-changes; after-sb-1pri discard-secondary; timeout 60; connect-int 10;
ping-int 10; max-buffers 2048; max-epoch-size 2048; } syncer { } on
serv1.cae.locale { device /dev/drbd3; disk /dev/serv1/lennyvm-home; address
192.168.1.100:7791; meta-disk /dev/serv1/meta[2]; } on serv2.cae.locale { device
/dev/drbd3; disk /dev/serv2/lennyvm-home; address 192.168.1.101:7791; meta-disk
/dev/serv2/meta[2]; } }
```

We have given the DRBD resource the same name as its corresponding LV. So, for example,

DRBD resource lennyvm-root is using LV lennyvm-root.

Next we will create a separate volume where we store DRBD's meta data. Meta data is used by DRBD to store information about the device. This can either be internal or external. Internal mode is easier to setup for a new devices but requires resizing operations when using an already formatted device. Further information at: <http://www.drbd.org/users-guide/ch-internals.html>

As we already have data on our LVs, created by XEN tools, we will use external meta data. So we create another LV with 1GB of space:

```
lvcreate -L 1G -n meta serv1
```

The next configuration steps are equal to those in the how-to, except from the differences in the names of the resources.

Then we have to configure DomU to use the DRBD resources

Configure DomU to use your DRBD device

The configuration files for your [DomUs?](#) are stored in /etc/xen/. So we first change to that directory:

```
cd /etc/xen
```

In here we have, for example, our debianvdd.cfg file. Edit /etc/xen/lennyvm.cfg

```
root@serv1:/# nano /etc/xen/lennyvm.cfg
```

In the section disk, we substitute all the entries "phy" with entries "drbd" followed by the names of the resources, as in following:

```
disk      = [          'drbd:lennyvm-root,sda1,w',          'drbd:lennyvm-swap,sda2,w',
                  'drbd:lennyvm-home,sda3,w',              #
'phy:/dev/serv1/lennyvm-root,sda1,w', #                    'phy:/dev/serv1/lennyvm-swap,sda2,w', #
```

```
'phy:/dev/serv1/lennyvm-home,sda3,w',          ]
```

Now we can follow again the mentioned how-to in configuring DomU on your other Dom0 node (copy conf files to the 2nd node) and configuring Live Migration.

By default XEN does not allow live migration, we have to enable this is `/etc/xen/xend-config.sxp`. Make sure the following line is commented, it should look like this:

```
 #(xend-relocation-hosts-allow '^localhost$ ^localhost\.localdomain$')
```

and that the following line is not commented, it should look like this:

```
 (xend-relocation-port 8002)
```

Put also this:

```
 (xend-relocation-server yes)
```

Make sure to do this two changes above on both nodes, both ha1 & ha2.

Install and Configure Heartbeat

We use Heartbeat, part of the Linux High Availability project (<http://www.linux-ha.org/>), to monitor our XEN resources and provide failover between our two nodes.

```
root@serv1:/# apt-get install heartbeat
```

Again only specific configurations for our job are outlined. For the rest just follow the above mentioned how-to and modify names, IPs and other server-specific options.

Edit `/etc/ha.d/authkeys`

```
root@serv1:~# nano /etc/ha.d/authkeys      auth 1      1 sha1 SecretKey123!!!
```

Note: Make sure to copy the exact same copy to your 2nd node.

Edit /etc/ha.d/ha.cf

```
root@serv1:~# nano /etc/ha.d/ha.cf    logfacility local0    udpport 694    keepalive 1    deadtime
10    warntime 3    initdead 20    ucast eth0 192.168.1.100    ucast eth0 192.168.1.101
auto_failback on    watchdog /dev/watchdog    debugfile /var/log/ha-debug    node
serv1.cae.locale    node serv2.cae.locale
```

Edit /etc/ha.d/haresources

```
root@serv1:~# nano /etc/ha.d/haresources    serv1.cae.locale xendomainsServ1
serv2.cae.locale.xendomainsServ2
```

Then we followed the how-to, with the requested modifications (names, IP, etc) according to our specific environment.

Bug in default xendomains script

We applied the patch reported here <http://xenbits.xensource.com/xen-unstable.hg?diff/01c8ccb551b0/tools/examples/init.d/xendomains>

Edit /etc/init.d/xendomains

```
root@serv1:~# nano /etc/init.d/xendomains    [...] stop() {    # Collect list of domains to shut
down    if test "$XENDOMAINS_AUTO_ONLY" = "true"; then        rdnames    fi    echo -n
"Shutting down Xen domains:"    while read LN; do        parseln "$LN"        if test $id = 0;
then continue; fi        echo -n " $name"        eval "        case "$name" in
($NAMES)        # nothing        ;;        (*)        echo -n '(skip)'
continue        ;;        esac        "    fi [...]
```

We still have some problems with that ... read more at <http://vdd-project.org/index.php/blog/comments/10>

{backbutton}