

# DRBD + Pacemaker & Corosync NFS Cluster Centos7

## On Both Nodes

### Host file

```
vim /etc/hosts
```

```
“ 10.1.2.114 nfs1 nfs1.localdomain.com
   10.1.2.115 nfs2 nfs2.localdomain.com
```

Corosync will not work if you add something like this: **127.0.0.1 nfs1 nfs2.localdomain.com** - however you do not need to delete 127.0.0.1 localhost

### Firewall

#### Option 1 **Firewalld**

```
systemctl start firewalld
systemctl enable firewalld
firewall-cmd --permanent --add-service=nfs
firewall-cmd --permanent --add-service=rpc-bind
firewall-cmd --permanent --add-service=mountd
firewall-cmd --permanent --add-service=high-availability
```

#### On **NFS1**

```
firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.1.2.115" port port="7789" protocol="tcp" accept'
```

```
firewall-cmd --reload  
firewall-cmd --reload
```

### On **NFS2**

```
firewall-cmd --permanent --add-rich-rule='rule family="ipv4" source address="10.1.2.114" port  
port="7789" protocol="tcp" accept'  
firewall-cmd --reload  
firewall-cmd --reload
```

## Disable SELINUX

```
vim /etc/sysconfig/selinux
```

```
SELINUX=disabled
```

## Pacemaker Install

### Install PaceMaker and Corosync

```
yum install -y pacemaker pcs
```

### Authenticate as the hacluster user

```
echo "H@xorP@assWD" | passwd hacluster --stdin
```

### Start and enable the service

```
systemctl start pcsd  
systemctl enable pcsd
```

## ON NFS1

### Test and generate the Corosync configuration

```
pcs cluster auth nfs1 nfs2 -u hacluster -p H@xorP@assWD
```

```
pcs cluster setup --start --name mycluster nfs1 nfs2
```

## ON BOTH NODES

## Start the cluster

```
systemctl start corosync
systemctl enable corosync
pcs cluster start --all
pcs cluster enable --all
```

## Verify Corosync installation

Master should have ID 1 and slave ID 2

```
corosync-cfgtool -s
```

### ON NFS1

## Create a new cluster configuration file

```
pcs cluster cib mycluster
```

## Disable the Quorum & STONITH policies in your cluster configuration file

```
pcs -f /root/mycluster property set no-quorum-policy=ignore
pcs -f /root/mycluster property set stonith-enabled=false
```

## Prevent the resource from failing back after recovery as it might increases downtime

```
pcs -f /root/mycluster resource defaults resource-stickiness=300
```

## LVM partition setup

### Both Nodes

## Create a empty partition

```
fdisk /dev/sdb
```

“ Welcome to fdisk (util-linux 2.23.2).

```
Command (m for help): n
Partition type:
p primary (0 primary, 0 extended, 4 free)
e extended
Select (default p):(ENTER)
Partition number (1-4, default 1): (ENTER)
First sector (2048-16777215, default 2048): (ENTER)
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-16777215, default 16777215):
(ENTER)
Using default value 16777215
Partition 1 of type Linux and of size 8 GiB is set

Command (m for help): w
The partition table has been altered!
```

### Create LVM partition

```
pvcreeate /dev/sdb1
vgcreate vg00 /dev/sdb1
lvcreate -l 95%FREE -n drbd-r0 vg00
```

### View LVM partition after creation

```
pvdisplay
```

Look in "/dev/mapper/" find the name of your LVM disk

```
ls /dev/mapper/
```

### OUTPUT:

```
control vg00-drbd--r0
```

**\*\*You will use "vg00-drbd--r0" in the "drbd.conf" file in the below steps**

## DRBD Installation

Install the DRBD package

```
rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
rpm -Uvh http://www.elrepo.org/elrepo-release-7.0-3.el7.elrepo.noarch.rpm
```

```
yum install -y kmod-drbd84 drbd84-utils
modprobe drbd
echo drbd > /etc/modules-load.d/drbd.conf
```

Edit the DRBD config and add the to hosts it will be connecting to (NFS1 and NFS2)

```
vim /etc/drbd.conf
```

Delete all and replace for the following

```
## include "drbd.d/global_common.conf";
include "drbd.d/*.res";

global {
usage-count no;
}

resource r0 {
protocol C;
startup {
degr-wfc-timeout 60;
outdated-wfc-timeout 30;
wfc-timeout 20;
}
disk {
on-io-error detach;
}
net {
cram-hmac-alg sha1;
shared-secret "Daveisc00l123313";
}
on nfs1.localdomain.com {
device /dev/drbd0;
disk /dev/mapper/vg00-drbd--r0;
address 10.1.2.114:7789;
meta-disk internal;
}
on nfs2.localdomain.com {
device /dev/drbd0;
disk /dev/mapper/vg00-drbd--r0;
address 10.1.2.115:7789;
meta-disk internal;
}
```

```
}
```

```
vim /etc/drbd.d/global_common.conf
```

Delete all and replace for the following

```
“ common {
    handlers {
    }
    startup {
    }
    options {
    }
    disk {
    }
    net {
        after-sb-0pri discard-zero-changes;
        after-sb-1pri discard-secondary;
        after-sb-2pri disconnect;
    }
}
```

## On NFS1

Create the DRBD partition and assign it primary on NFS1

```
drbdadm create-md r0
drbdadm up r0
drbdadm primary r0 --force
drbdadm -- --overwrite-data-of-peer primary all
drbdadm outdate r0
mkfs.ext4 /dev/drbd0
```

## On NFS2

Configure r0 and start DRBD on NFS2

```
drbdadm create-md r0
drbdadm up r0
```

```
drbdadm secondary all
```

## Pacemaker cluster resources

### On NFS1

Add resource r0 to the cluster resource

```
pcs -f /root/mycluster resource create r0 ocf:linbit:drbd drbd_resource=r0 op monitor  
interval=10s
```

Create an additional clone resource r0-clone to allow the resource to run on both nodes at the same time

```
pcs -f /root/mycluster resource master r0-clone r0 master-max=1 master-node-max=1 clone-max=2  
clone-node-max=1 notify=true
```

Add DRBD filesystem resource

```
pcs -f /root/mycluster resource create drbd-fs Filesystem device="/dev/drbd0"  
directory="/data" fstype="ext4"
```

Filesystem resource will need to run on the same node as the r0-clone resource, since the pacemaker cluster services that runs on the same node depend on each other we need to assign an infinity score to the constraint:

```
pcs -f /root/mycluster constraint colocation add drbd-fs with r0-clone INFINITY with-rsc-  
role=Master
```

Add the Virtual IP resource

```
pcs -f /root/mycluster resource create vip1 ocf:heartbeat:IPAddr2 ip=10.1.2.116  
cidr_netmask=24 op monitor interval=10s
```

The VIP needs an active filesystem to be running, so we need to make sure the DRBD resource starts before the VIP

```
pcs -f /root/mycluster constraint colocation add vip1 with drbd-fs INFINITY  
pcs -f /root/mycluster constraint order drbd-fs then vip1
```

Verify that the created resources are all there

```
pcs -f /root/mycluster resource show
pcs -f /root/mycluster constraint
```

And finally commit the changes

```
pcs cluster cib-push mycluster
```

## On Both Nodes

# Installing NFS

Install nfs-utils

```
yum install nfs-utils -y
```

Stop all services

```
systemctl stop nfs-lock && systemctl disable nfs-lock
```

Setup service

```
pcs -f /root/mycluster resource create nfsd nfsserver nfs_shared_infodir=/data/nfsinfo
pcs -f /root/mycluster resource create nfsroot exportfs clientspec="10.1.2.0/24"
options=rw,sync,no_root_squash directory=/data fsid=0
pcs -f /root/mycluster constraint colocation add nfsd with vip1 INFINITY
pcs -f /root/mycluster constraint colocation add vip1 with nfsroot INFINITY
pcs -f /root/mycluster constraint order vip1 then nfsd
pcs -f /root/mycluster constraint order nfsd then nfsroot
pcs -f /root/mycluster constraint order promote r0-clone then start drbd-fs
pcs resource cleanup
pcs cluster cib-push mycluster
```

Test failover

```
pcs resource move drbd-fs nfs2
```

# Other notes on DRBD

To update a resource after a commit

```
cibadmin --query > tmp.xml
```

Edit with vi tmp.xml or do a pcs -f tmp.xml %do your thing%

```
cibadmin --replace --xml-file tmp.xml
```

Delete a resource

```
pcs -f /root/mycluster resource delete db
```

Delete cluster

```
pcs cluster destroy
```

## Recover a split brain

### Secondary node

```
drbdadm secondary all
```

```
drbdadm disconnect all
```

```
drbdadm -- --discard-my-data connect all
```

### Primary node

```
drbdadm primary all
```

```
drbdadm disconnect all
```

```
drbdadm connect all
```

### On both

```
drbdadm status
```

```
cat /proc/drbd
```

---

Revision #9

Created 2019-05-17 19:38:28 UTC by Dave

Updated 2019-09-25 19:05:31 UTC by Dave