Role STORE - Redundant (CentOS8/Win2019)

Notes

We no longer support DRBD on the base file system. We always install with LVM for maintenance purposes.

We use the whole disk, so increasing size is done using LVM by adding new disks.

It is recommended to create the STORE machine without the disk for the storage being mounted by the installation routines.

Disk and DRBD Setup

Create LVM Physical Volume, Volume Group and Logical Volume **(Both Nodes)**

The commands below assume that **/dev/sdb** will be used for the DRBD on top of LVM configuration, and that the disks are **EXACTLY** the same size.

```
# Create the phsyical volume - this is based on sdb assuming it is the second drive on the system
lvm pvcreate /dev/sdb

# Create the volume group
lvm vgcreate "vg_drbd_jtelshared" /dev/sdb

# Create the logical volume
lvm lvcreate -l +100%FREE vg_drbd_jtelshared -n lv_drbd_jtelshared
```

Configure Firewall for DRBD **(Both Nodes)**

```
# Prepare the firewall
firewall-cmd --zone=public --add-port=7788-7799/tcp --permanent
firewall-cmd --reload
```

Prepare Mount Point **(Both Nodes)**

The data should be mounted to the directory **/srv/jtel/shared**.

The following commands prepare for this:

```
mkdir /srv/jtel
mkdir /srv/jtel/shared
chown -R jtel:jtel /srv/jtel
```

Install ELrepo and DRBD **(Both Nodes)**

We now install DRBD 9. This requires using the ELrepo. Also, SELinux must be set to permissive for DRBD.

```
dnf -y install https://www.elrepo.org/elrepo-release-8.el8.elrepo.noarch.rpm
rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
dnf -y install drbd90-utils kmod-drbd90
systemctl enable drbd
semanage permissive -a drbd_t
systemctl start drbd
```

Configure DRBD **(Both Nodes)**

DRBD must be configured with static ip addresses and correct hostnames.

The IP addresses below must be modified:

CAUTION IP ADDRESSES CAUTION HOSTNAMES

```
# Configure DRBD
cat <<EOFF > /etc/drbd.d/jtelshared.res
resource jtelshared {
    protocol C;
    meta-disk internal;
    device /dev/drbd0;
    syncer {
        verify-alg sha1;
    }
    net {
        allow-two-primaries;
    }
    on acd-store1.jtel.local {
        disk   /dev/vg_drbd_jtelshared/lv_drbd_jtelshared;
        address 10.1.1.1:7789;
    }
    on acd-store2.jtel.local {
        disk   /dev/vg_drbd_jtelshared/lv_drbd_jtelshared;
        address 10.1.1.2:7789;
    }
}
EOFF
```

Create Metadata and Start **(Both Nodes)**

```
# Create metadata and start DRBD
drbdadm create-md jtelshared
drbdadm up jtelshared
```

Make ONE Node Primary

```
# Make ONE node primary
drbdadm primary jtelshared --force
```

Wait for Sync

DRBD will now sync. This might take some time, however, you do not have to wait, as long as you make sure the primary is active in the PCS cluster below.

Note: with DRBD9 we currently have no options to tune the transfer.

You can watch the initial sync with the following command:

```
drbdadm status jtelshared
```

You will see outbout like this:

```
jtelshared role:Primary
  disk:UpToDate
  acd-store2 role:Secondary
    replication:SyncSource peer-disk:Inconsistent done:7.19
```

This means the following:

- The local machine is primary.
- The disk in the local machine is up to date.
- acd-store2 is secondary
- It is inconsistent and syncing, 7.19% done

You can continue, but usually it is best to wait for it to complete.

Make Second Node Primary **(SECOND NODE ONLY)**

```
drbdadm primary jtelshared
```

Create the Filesystem **(FIRST NODE ONLY)**

```
mkfs.xfs -L data /dev/drbd/by-res/jtelshared/0
```

Create fstab entry for file system **(Both Nodes)**

This command adds a line to **/etc/fstab**

```
cat << EOFF >> /etc/fstab
/dev/drbd/by-res/jtelshared/0  /srv/jtel/shared         xfs noauto,noatime,nodiratime  0   0
EOFF
```

Test DRBD

Now, we can test the DRBD setup.

Mount the file system (FIRST Node)

```
mount /srv/jtel/shared
```

Create a test file and Unmount (FIRST Node)

```
cat <<EOFF > /srv/jtel/shared/test.txt
test 123
EOFF
umount /srv/jtel/shared
```

Mount the file system and check test file (SECOND Node)

```
mount /srv/jtel/shared
cat /srv/jtel/shared/test.txt

# Check contents of file before proceeding

rm /srv/jtel/shared/test.txt
umount /srv/jtel/shared
```

Do not proceed unless you can see the contents of the test file.

Comment Mount out in fstab (BOTH nodes) and disable DRBD

```
sed -i '/jtelshared/s/^/#/' fstab
systemctl disable drbd
umount /srv/jtel/shared
```

Install PCS Cluster **(BOTH NODES)**

If you have not installed Pacemaker / Corosync on both LB machines, do this now - see here: Redundancy - Installing PCS Cluster

Install and Configure Samba

Installation **(BOTH NODES)**

These commands install the samba server and client and lsof.

```
dnf -y install samba samba-client lsof
```

Configure Samba **(BOTH NODES)**

The following creates a samba configuration file with a minimum configuration.

```
# SMB Conf
cat <<EOFF > /etc/samba/smb.conf
[global]
    workgroup = JTEL
    security = user
    passdb backend = tdbsam
    min protocol = SMB3
    reset on zero vc = yes
[shared]
    comment = jtel ACD Shared Directory
    public = no
    read only = no
    writable = yes
    locking = yes
    path = /srv/jtel/shared
    create mask = 0644
    directory mask = 0755
    force user = jtel
    force group = jtel
    acl allow execute always = True
EOFF
```

Setup SeLinux and the Firewall **(BOTH NODES)**

The following command sets up selinux for the smb service and opens the necessary ports in the firewall:

```
setsebool -P samba_enable_home_dirs=on samba_export_all_rw=on use_samba_home_dirs=on use_nfs_home_dirs=on
firewall-cmd --zone=public --add-port=445/tcp --add-port=139/tcp --add-port=138/udp --add-port=137/udp --permanent
firewall-cmd --reload
```

Link /home/jtel/shared **(BOTH NODES)**

Link the /home/jtel/shared folder.

```
ln -s /srv/jtel/shared /home/jtel/shared
```

Setup Access to the Samba Server

jtel User Access **(BOTH NODES)**

The following command creates the smb credentials for the jtel user.

CAUTION PASSWORD

```
printf '<password>\n<password>\n' | smbpasswd -a -s jtel
```

Further User Access **(BOTH NODES)**

If necessary, add further users to samba - replacing password with the actual password for the user. Here, for example, the windows administrator user:

CAUTION PASSWORD

```
useradd -m Administrator
printf '<password>\n<password>\n' | smbpasswd -a -s Administrator
```

Configure Cluster Resources

Now all resources will be configured in the pacemaker cluster.

Setup virtual IP (One Node Only!)

Change the following to set the virtual IP which should be shared between the nodes.

CAUTION IP ADDRESS

```
JT_VIP=10.1.1.100
```

Configure PCS Resources for DRBD Mount, DRBD Primary / Secondary, Samba and Virtual IP Address **(One Node Only!)**

Configure the PCS resources with the following commands:

```
# Configure using a file jtel_cluster_config
cd
pcs cluster cib jtel_cluster_config
# DRBD Primary Secondary
pcs -f jtel_cluster_config resource create DRBDClusterMount ocf:linbit:drbd drbd_resource=jtelshared op monitor interval=60s
pcs -f jtel_cluster_config resource promotable DRBDClusterMount promoted-max=1 promoted-node-max=1 clone-max=2 clone-node-max=1 notify=true
# DRBD File System Mount
pcs -f jtel_cluster_config resource create DRBDClusterFilesystem ocf:heartbeat:Filesystem device="/dev/drbd/by-res/jtelshared/0" directory="/srv/jtel/shared"
fstype="xfs"
# Colocation of File System Mount with Primary DRBD instance
pcs -f jtel_cluster_config constraint colocation add DRBDClusterFilesystem with DRBDClusterMount-clone INFINITY with-rsc-role=Master
# Promote first, then start filesystem
pcs -f jtel_cluster_config constraint order promote DRBDClusterMount-clone then start DRBDClusterFilesystem
# Resource for Samba
pcs -f jtel_cluster_config resource create Samba systemd:smb op monitor interval=30s
# Resource for virtual IP
pcs -f jtel_cluster_config resource create ClusterIP ocf:heartbeat:IPaddr2 ip=${JT_VIP} cidr_netmask=32 op monitor interval=30s
# Samba must be with active DRBD filesystem
pcs -f jtel_cluster_config constraint colocation add Samba with DRBDClusterFilesystem INFINITY
# Cluster IP must be with Samba
pcs -f jtel_cluster_config constraint colocation add ClusterIP with Samba INFINITY
# Start DRBD File system then start Samba
pcs -f jtel_cluster_config constraint order DRBDClusterFilesystem then Samba
# Start Samba then start Cluster IP
pcs -f jtel_cluster_config constraint order Samba then ClusterIP
```

Check the configuration:

```
# Check the config file
pcs -f jtel_cluster_config config
```

Push the configuration to the cluster:

```
# Push the config to the cluster
pcs cluster cib-push jtel_cluster_config --config
```

Tests

Test pcs status

First of all, we test the cluster status:

```
pcs status
```

You should see output similar to this:

```
Cluster name: jtel_cluster
Cluster Summary:
  * Stack: corosync
  * Current DC: acd-lb1 (version 2.0.3-5.el8_2.1-4b1f869f0f) - partition with quorum
  * Last updated: Sat Oct  3 12:59:34 2020
  * Last change:  Sat Oct  3 12:31:22 2020 by root via cibadmin on acd-lb2
  * 2 nodes configured
  * 5 resource instances configured

Node List:
  * Online: [ acd-lb1 acd-lb2 ]

Full List of Resources:
  * Clone Set: DRBDClusterMount-clone [DRBDClusterMount] (promotable):
    * Masters: [ acd-lb1 ]
    * Stopped: [ acd-lb2 ]
  * DRBDClusterFilesystem       (ocf::heartbeat:Filesystem):    Started acd-lb1
  * Samba       (systemd:smb):  Started acd-lb1
  * ClusterIP   (ocf::heartbeat:IPaddr2):       Started acd-lb1

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

Make sure all of the resources are started and both nodes are online.

Test File Mount

You should now be able to access **\\acd-store\shared** from the windows machines for example.

If you want to test from linux, you will need to mount STORE as described here: Mounting STORE - All Linux except for STORE (CentOS8/Win2019)

Test Failover and Failback

You can test failover and failback with any of the following commands:

Standby and Unstandby

```
pcs node standby acd-lb1

# TEST

pcs node unstandby acd-lb1

# TEST

pcs node standby acd-lb1

# TEST

pcs node unstandby acd-lb1

# TEST
```

Stop Cluster Node

```
pcs cluster stop acd-lb1

# TEST

pcs cluster start acd-lb1

# TEST

pcs cluster stop acd-lb2

# TEST

pcs cluster start acd-lb2

# TEST
```

Reboot

Rebooting is also a good way to test.

Power Off

This is the best way to test, but be aware, you may cause split brain on DRBD and need to repair it.