Redundancy - Installing PCS Cluster (Debian/Win2019)

PCS Cluster is required for:

- The role LB (Load Balancer), if two load balancers are used for redundancy
- The role STORE (file storage), if two file stores are setup using DRBD for redundancy

If the system does not contain redundancy, do not install this. Only install on the above roles.

Install PCS Services (Both nodes)

Install the PCS packages and stop the services.

```
apt-get update
apt-get -y install pacemaker corosync pcs haveged

systemctl stop pcsd
systemctl stop pacemaker
systemctl stop corosync
```

Disable Managed Services (Both nodes)

Disable smb since this will be managed by pacemaker:

```
systemctl disable smbd
```

Configuration Settings (Both nodes)

Next configure the names of the machines and the virtual IP address which will be shared in the cluster:

CAUTION PASSWORD

```
JT_HOST1=acd-lb1
JT_HOST2=acd-lb2
PASSWORD=<password>
```

Configure the Firewall (Both nodes)

Next configure the firewall for ha services:

```
ufw allow 2224/tcp
ufw allow 3121/tcp
ufw allow 5403/tcp
ufw allow 5404/udp
ufw allow 5405/udp
```

Change user password (Both nodes)

Change the password of the hacluster user (replace <password> with the chosen password):

```
echo hacluster:${PASSWORD} | chpasswd
```

Cluster Configuration

Node 1 - Create Cluster Key

Create a key for the cluster and copy to server 2:

```
# On Server 1
corosync-keygen
scp /etc/corosync/authkey jtel@acd-lb2:/home/jtel/
```

Node 2 - Move Cluster Key

Move the cluster key to the configuration directory and setup rights:

```
# On Server 2
mv /home/jtel/authkey /etc/corosync/
chown root:root /etc/corosync/authkey
chmod 400 /etc/corosync/authkey
```

Both Nodes - Create Corosync Configuration

Note: the hosts file must be configured for this to work.

```
mv /etc/corosync/corosync.conf /etc/corosync/corosync.conf.orig
cat << EOFF > /etc/corosync/corosync.conf
totem {
  version: 2
  cluster_name: jtel_cluster
  transport: knet
  crypto_cipher: aes256
  crypto_hash: sha256
  token: 4000
}

nodelist {
  node {
    ring0_addr: acd-lb1
    name: acd-lb1
    nodeid: 1
  }

  node {
    ring0_addr: acd-lb2
    name: acd-lb2
    nodeid: 2
  }
}

quorum {
  provider: corosync_votequorum
  two_node: 1
}

logging {
  to_logfile: yes
  logfile: /var/log/corosync/corosync.log
  to_syslog: yes
  timestamp: on
}
EOFF
```

Start Cluster - Both Nodes

```
systemctl enable corosync
systemctl enable pacemaker
systemctl enable pcsd

systemctl start corosync
systemctl start pacemaker
systemctl start pcsd
```

Resource Cleanup - One Node

```
pcs resource cleanup
pcs status
```

Check if the output is OK.

Configure Cluster - One Node

```
pcs property set stonith-enabled=false
pcs property set no-quorum-policy=ignore
pcs resource defaults migration-threshold=1
```

Test

Check the results on both machines:

```
root@test-lb1:/home/jtel# pcs status
Cluster name: jtel_cluster
Stack: corosync
Current DC: acd-lb1 (version 2.0.1-9e909a5bdd) - partition with quorum
Last updated: Tue Feb 23 07:49:26 2021
Last change: Tue Feb 23 07:40:58 2021 by root via cibadmin on acd-lb1

2 nodes configured
0 resources configured

Online: [ acd-lb1 acd-lb2 ]

No resources


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

```
root@acd-store1-test:/home/jtel# pcs config
Cluster Name: jtel_cluster
Corosync Nodes:
 acd-store1-test acd-store2-test
Pacemaker Nodes:
 acd-store1-test acd-store2-test

Resources:

Stonith Devices:
Fencing Levels:

Location Constraints:
Ordering Constraints:
Colocation Constraints:
Ticket Constraints:

Alerts:
 No alerts defined

Resources Defaults:
  Meta Attrs: rsc_defaults-meta_attributes
    migration-threshold=1
Operations Defaults:
  No defaults set

Cluster Properties:
 cluster-infrastructure: corosync
 cluster-name: debian
 dc-version: 2.0.5-ba59be7122
 have-watchdog: false
 no-quorum-policy: ignore
 stonith-enabled: false

Tags:
 No tags defined

Quorum:
  Options:
```