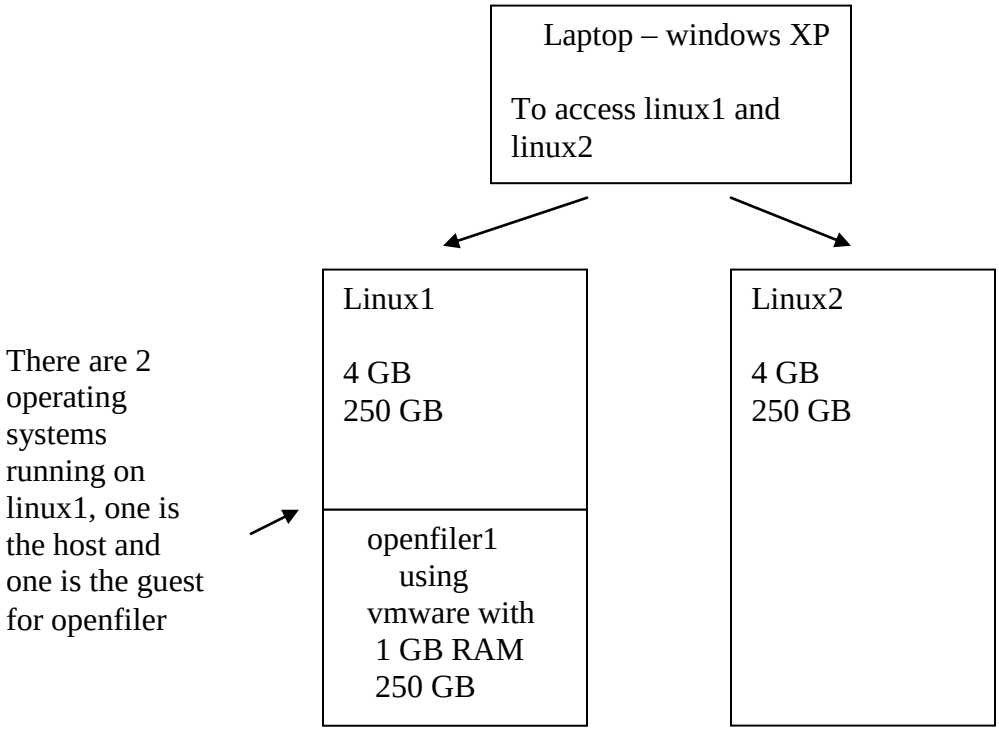
Author – *A.Kishore*
http:/www.appsdba.info

Build Your Own Oracle RAC 11*g* Release 2 Cluster on Red Hat 4 and iSCSI

Here is my setup which I have used for Installing Oracle RAC 11gR2



High Level Steps

> Install openfiler on linux1 using VMWARE
> Install Grid Infrastructure on linux1 and linux2
> Install Oracle 11gR2 software on linux1 and linux2
> Install Cluster database

Author – *A.Kishore*
http:/www.appsdba.info

```
-- Follow the below steps on linux1 and linux2

1> linux1 and linux2 /etc/hosts

# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1       localhost.localdomain   localhost
# Public Network - (eth0)
192.168.1.7     linux2.oracle.com       linux2
192.168.1.3     linux1.oracle.com       linux1

# Private Interconnect - (eth1)
192.168.0.101   linux1-priv.oracle.com  linux1-priv
192.168.0.102   linux2-priv.oracle.com  linux2-priv

# Public Virtual IP (VIP) addresses for - (eth0)
192.168.1.200   linux1-vip.oracle.com   linux1-vip
192.168.1.201   linux2-vip.oracle.com   linux2-vip

# Scan IP

192.168.1.187   racnode-cluster-scan

192.168.1.6     openfiler1.oracle.com   openfiler1



2> linux1 and linux2
vi /etc/sysctl.conf

kernel.core_uses_pid = 1
kernel.shmmax=4294967295
kernel.sem=256 32000 100 142
fs.file-max = 6815744
kernel.msgmnb = 65535
kernel.msgmni = 2878
net.core.rmem_default = 262144
net.core.rmem_max =  4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
net.ipv4.tcp_wmem = 262144 262144 262144
net.ipv4.tcp_rmem = 262144 262144 262144

fs.aio-max-nr = 1048576
net.ipv4.ip_local_port_range = 9000 65500

vi /etc/security/limits.conf

hard nofile 65535
soft nofile 4096
hard nproc 16384
soft nproc 2047
oracle hard nofile 65536
```

3> Install a Proper Linux Kernel - linux1 and linux2

cp /etc/grub.conf /etc/grub.conf.original
remove smp

```
[root@linux1 ~]# cat /etc/grub.conf
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You do not have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /, eg.
#          root (hd0,0)
#          kernel /boot/vmlinuz-version ro root=/dev/sda1
#          initrd /boot/initrd-version.img
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/boot/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux ES-up (2.6.9-67.EL)
        root (hd0,0)
        kernel /boot/vmlinuz-2.6.9-67.EL ro root=LABEL=/ rhgb quiet
        initrd /boot/initrd-2.6.9-67.EL.img
```

connect as below kernel
2.6.9-67.0.0.0.2.EL

4> Create "oracle" User and Directories (both nodes)

groupadd dba
groupadd oinstall
useradd -u 502 -g oinstall -G dba oracle
passwd oracle

5. Install missing packages

rpm -Uvh elfutils-libelf-devel-0.97.1-5.i386.rpm
rpm -Uvh iscsi-initiator-utils-4.0.3.0-6.i386.rpm
rpm -Uvh libaio-devel-0.3.105-2.i386.rpm
rpm -Uvh rsh-server-0.17-25.4.i386.rpm
rpm -Uvh sysstat-5.0.5-16.rhel4.i386.rpm
rpm -Uvh unixODBC-2.2.11-1.RHEL4.1.i386.rpm
rpm -Uvh unixODBC-devel-2.2.11-1.RHEL4.1.i386.rpm

6. Install remote shell – alternate to ssh

To enable the "rsh" service, the "disable" attribute in the /etc/xinetd.d/rsh file must be set to "no" and xinetd must be reloaded. Do that by running the following commands on all nodes in the cluster:

vi /etc/xinetd.d/rsh
Change disable from yes to no
change yes - no

```
[root@linux1 ~]# cat /etc/xinetd.d/rsh


# default: on
# description: The rshd server is the server for the rcmd(3) routine and, \
#       consequently, for the rsh(1) program.  The server provides \
#       remote execution facilities with authentication based on \
#       privileged port numbers from trusted hosts.
service shell
{
        disable = no
        socket_type             = stream
        wait                    = no
        user                    = root
        log_on_success          += USERID
        log_on_failure          += USERID
        server                  = /usr/sbin/in.rshd
}
```

```
/etc/init.d/xinetd restart

# su -
chkconfig rsh on
chkconfig rlogin on
service xinetd reload
Reloading configuration: [  OK  ]

To allow the "oracle" UNIX user account to be trusted among the RAC nodes,
create the /etc/hosts.equiv file on all nodes in the cluster:

# su -
touch /etc/hosts.equiv
chmod 600 /etc/hosts.equiv
chown root.root /etc/hosts.equiv

Now add all RAC nodes to the /etc/hosts.equiv file similar to the following
example for all nodes in the cluster:

vi /etc/hosts.equiv
+linux1 oracle
+linux2 oracle
+linux1-priv oracle
+linux2-priv oracle

# su -

# which rsh
/usr/kerberos/bin/rsh

cd /usr/kerberos/bin
mv rsh rsh.original

which rsh
/usr/bin/rsh

You should now test your connections and run the rsh command from the node that
will be performing the Oracle CRS and 10g RAC installation. We will use the
```

node linux1 to perform the install, so run the following commands from that node:

```
# su - oracle

$ rsh linux1 ls -l /etc/hosts.equiv
-rw-------    1 root     root           68 Jan 31 00:39 /etc/hosts.equiv

$ rsh int-linux1 ls -l /etc/hosts.equiv
-rw-------    1 root     root           68 Jan 31 00:39 /etc/hosts.equiv

$ rsh linux2 ls -l /etc/hosts.equiv
-rw-------    1 root     root           68 Jan 31 00:25 /etc/hosts.equiv

$ rsh int-linux2 ls -l /etc/hosts.equiv
-rw-------    1 root     root           68 Jan 31 00:25 /etc/hosts.equiv
```

7> Disable secure linux by editing the "/etc/selinux/config" file, making sure the SELINUX flag is set as follows.

```
vi /etc/selinux/config
SELINUX=disabled

echo "session required /lib/security/pam_limits.so" >> /etc/pam.d/login
```

#disable the firewall on all nodes, on Linux this can be done by running the following command(s) as the root user on each node of the cluster:

```
service iptables stop
service ip6tables stop

To permanently disable the firewall, use:
chkconfig iptables off
chkconfig ip6tables off
```

8> NTP – Network time protocol

Using NTP (Network Time Protocol).
This is simple process. Just adding the time server to /etc/ntp.conf and to /etc/ntp/step-tickers Here I use time server as 192.168.1.91.

```
service ntpd stop
rm /var/run/ntpd.pid

#vi /etc/ntp.conf
server 192.168.1.91
#vi /etc/ntp/step-tickers
192.168.1.91
Then of course make sure that ntp will start at boot time by,
chkconfig --level 2345 ntpd on
chkconfig --list ntpd
```

If you are using NTP, you must add the "-x" option into the following line in the "/etc/sysconfig/ntpd" file.

```
OPTIONS="-x -u ntp:ntp -p /var/run/ntpd.pid"
```

```
And let's start the service now by,
service ntpd start

It will display output as
ntpd: Synchronizing with time server: [ OK ]
Starting ntpd: [ OK ]

ps -ef |grep ntp
root     11538     1  0 13:38 ?        00:00:00 ntpd -x -u ntp:ntp -p
/var/run/ntpd.pid


OR


Either configure NTP, or make sure it is not configured so the Oracle Cluster
Time Synchronization Service (ctssd) can synchronize the times of the RAC
nodes. In this case we will deconfigure NTP.
```

```
# service ntpd stop
Shutting down ntpd:                                        [  OK  ]
# chkconfig ntpd off
# mv /etc/ntp.conf /etc/ntp.conf.org
# rm /var/run/ntpd.pid
```

If you are using NTP, you must add the "-x" option into the following line in the "/etc/sysconfig/ntpd" file.

```
OPTIONS="-x -u ntp:ntp -p /var/run/ntpd.pid"
```

Then restart NTP.

```
# service ntpd restart
```

Start the Name Service Cache Daemon (nscd).

```
chkconfig --level 35 nscd on
service nscd start
```

```
9> Configure Openfiler on linux1 and linux2

Install openfiler by following the below steps

http://www.appsdba.info/docs/oracle_apps/RAC/Install_Open_Filer.pdf

vi /etc/iscsi.conf

DiscoveryAddress=192.168.1.6
chkconfig iscsi on


service iscsi restart

fdisk -l

-- Partition the mount points
```

```
[root@linux1 ~]# fdisk /dev/sdb
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF
disklab
el
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.

Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4):
Value out of range.
Partition number (1-4): 1
First cylinder (1-1009, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1009, default 1009):
Using default value 1009

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

Calling ioctl() to re-read partition table.
Syncing disks.
[root@linux1 ~]# fdisk /dev/sdc
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF
disklab
el
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.


The number of cylinders for this disk is set to 9137.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite)

Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-9137, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-9137, default 9137):
Using default value 9137

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

```
Calling ioctl() to re-read partition table.
Syncing disks.
[root@linux1 ~]# fdisk /dev/sdd
Device contains neither a valid DOS partition table, nor Sun, SGI or OSF
disklab
el
Building a new DOS disklabel. Changes will remain in memory only,
until you decide to write them. After that, of course, the previous
content won't be recoverable.
```

Author – *A.Kishore*
http:/www.appsdba.info

10. Install and Configure Automatic Storage Management and Disks

Installing ASMLib Packages

This installation needs to be performed on all nodes as the root user account:

```
$ su -

Single processor
rpm -Uvh oracleasm-support-2.0.3-2.i386.rpm
rpm -Uvh oracleasm-2.6.9-67.EL-2.0.3-1.i686.rpm – I have download this package
rpm -Uvh oracleasmlib-2.0.2-1.i386.rpm
```

Configuring and Loading the ASMLib Packages

Now that we downloaded and installed the ASMLib Packages for Linux, we need to configure and load the ASM kernel module. This task needs to be run on all nodes as root:

```
$ su -
/etc/init.d/oracleasm configure

Default user to own the driver interface []: oracle
Default group to own the driver interface []: dba
Start Oracle ASM library driver on boot (y/n) [n]: y
Fix permissions of Oracle ASM disks on boot (y/n) [y]: y
Writing Oracle ASM library driver configuration            [  OK  ]
Creating /dev/oracleasm mount point                        [  OK  ]
Loading module "oracleasm"                                 [  OK  ]
Mounting ASMlib driver filesystem                          [  OK  ]
Scanning system for ASM disks                              [  OK  ]

/etc/init.d/oracleasm enable
Writing Oracle ASM library driver configuration            [  OK  ]
Scanning system for ASM disks                              [  OK  ]

Create ASM disks. Create the ASM disks on any one node as the root user.
/etc/init.d/oracleasm createdisk VOL1 /dev/sdc5
/etc/init.d/oracleasm createdisk VOL2 /dev/sdb6
/etc/init.d/oracleasm createdisk VOL3 /dev/sdb7
Verify that the ASM disks are visible from every node.
# /etc/init.d/oracleasm scandisks
Scanning system for ASM disks: [ OK ]
# /etc/init.d/oracleasm listdisks
VOL1
VOL2
VOL3


[root@linux1]# /etc/init.d/oracleasm scandisks
Scanning system for ASM disks                              [  OK  ]
```