

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

## Create and Configure the Second Virtual Machine

```
shutdown rac1 - init 0
```

```
cd /d02/oracle/vm/rac
```

```
[root@erp rac]# pwd
/d02/oracle/vm/rac
[root@erp rac]# cp -R rac1 rac2 &
[1] 11120
```

Start rac2 and change the name

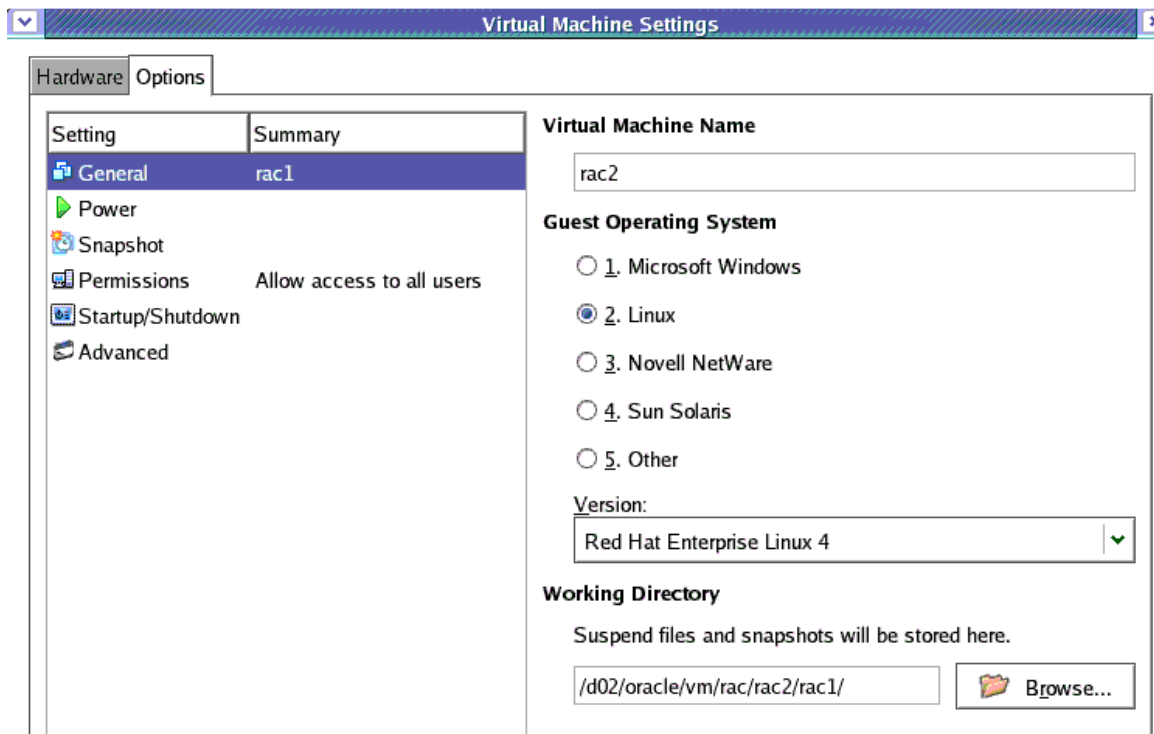
The screenshot shows the Oracle VM VirtualBox interface. At the top, there are browser tabs for 'erp.com', 'rac1', and 'rac1'. The main window displays the configuration for a virtual machine named 'rac1'. The state is 'Powered Off', the guest OS is 'Red Hat Enterprise Linux 4', and the configuration file is '/d02/oracle/vm/rac/rac2/rac1/rac1.vmx'. The version is 'Current virtual machine'.

On the left, there are buttons for 'Commands', 'Power on this virtual machine', 'Edit virtual machine settings', and 'Notes'. The 'Notes' section contains the text: 'Type here to enter notes for this virtual machine.'

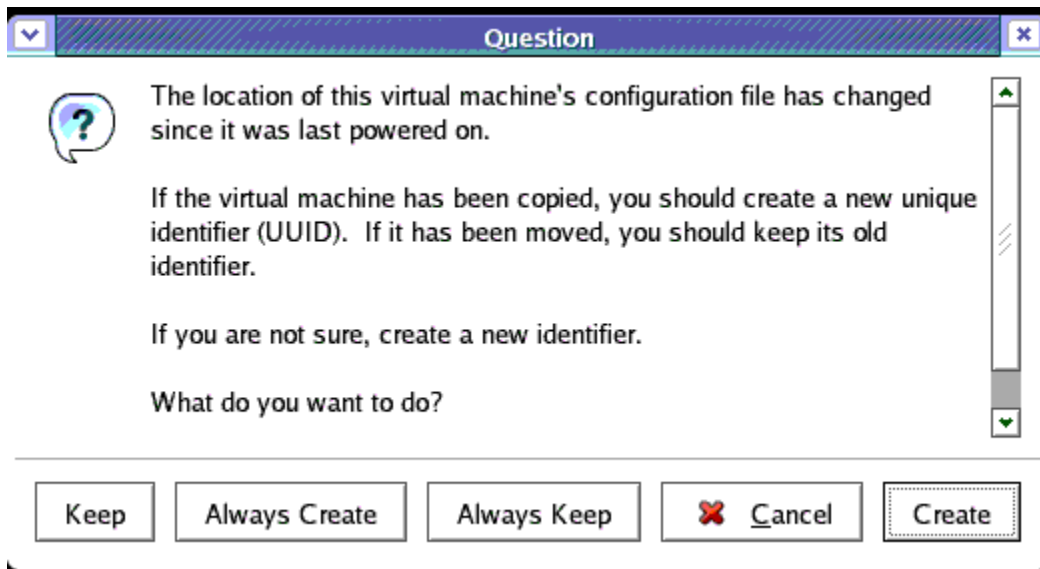
On the right, there is a 'Devices' section with the following configuration:

Device	Configuration
Memory	700MB
Hard Disk (SCSI 0:0)	20.0GB
Hard Disk (SCSI 1:0)	102MB (Persistent)
Hard Disk (SCSI 1:1)	3.0GB (Persistent)
Hard Disk (SCSI 1:2)	3.0GB (Persistent)
Hard Disk (SCSI 1:3)	2.0GB (Persistent)
CD-ROM 1 (IDE 1:0)	Using file /software/OEL4/Enterprise-R4-U5-i386-disc3
Ethernet 1	Bridged
Ethernet 2	Host-only
Mouse	Auto detect
Processors	1

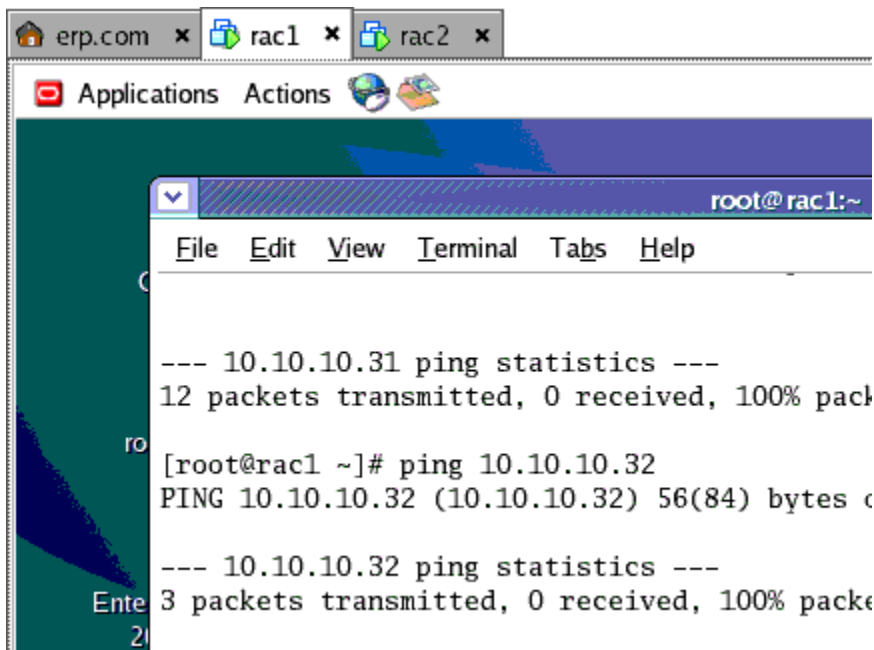
Author – A.Kishore  
<http://www.appsdba.info>



Start rac1 and rac2



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



Check that the communication between rac1 and rac2

```
ping rac2  
ping rac2-priv
```

```
-- Change the ORACLE_SID on rac2  
su - oracle  
vi .bash_profile  
ORACLE_SID=devdb2
```

```
-- change the host name on rac2  
vi /etc/sysconfig/network
```

```
reboot
```

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

Establish user equivalence with RSH

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
```

```
service shell
{
    socket_type           = stream
    wait                  = no
    user                  = root
    log_on_success        += USERID
    log_on_failure        += USERID
    server                = /usr/sbin/in.rshd
    disable               = no
}
```

```
/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
+rac1 oracle
+rac2 oracle
+rac1-priv oracle
+rac2-priv oracle
```

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

```
# su -  
  
# which rsh  
/usr/kerberos/bin/rsh  
  
cd /usr/kerberos/bin  
mv rsh rsh.original  
  
which rsh  
/usr/bin/rsh
```

Perform the above activities on rac1 and rac2

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 rac1 to perform the install, so run the following commands from that node:

```
# su - oracle  
  
$ rsh rac1 ls -l /etc/hosts.equiv
```

```
rac1-> rsh rac1 ls -l /etc/hosts.equiv  
-rw----- 1 root root 63 Sep 16 03:35 /etc/hosts.equiv
```