

**Linuxha.net**  
**Step-by-step Guide:**  
**Oracle 10g Express Edition**

## Introduction

The purpose of this document is to suggest procedures for creating a clustered Oracle 10g Express Edition configuration with Linuxha.net. These procedures were tested with the following configuration.

- Fedora Core 5
- DRBD version 0.7.20
- Linuxha.net version 1.2
- Oracle 10g Express Edition version 10.2.0.1

The following conventions are used throughout this document:

# ls	Single-line command entered as root user.
# useradd \ > --home-dir /home2/xyz \ > --gid abc \ > xyz	Multi-line command entered as root user.
\$ ls	Single-line command entered as non-root user
\$ cut \ > --delimiter=":" \ > --fields=1,5 \ / etc/passwd	Multi-line command entered as non-root user.
127.0.0.1 localhost 192.168.1.32 fc5s1 192.168.1.33 fc5s2 192.168.100.32 fc5s1b 192.168.100.33 fc5s2b	Command output or file contents

## User Account Creation

The Oracle processes will run with the **oracle** user credentials. The **oracle** user and **dba** group must exist on both nodes and have the same user and groups ids.

All commands in this section are executed as **root** on both nodes, unless otherwise indicated.

1. Check whether the **oracle** user exists and that the user and group ids are the same on both nodes.

```
# grep oracle /etc/passwd
```

If there is no result, continue to step 2 since the **oracle** user does not exist. Otherwise, compare the user and group ids on both nodes. These are the third and fourth fields respectively of the user account record as shown below:

```
oracle:x:501:501::/usr/lib/oracle/xe:/bin/bash
```

If the user and group ids are not the same on both nodes, proceed to step 6. Otherwise go to the next section.

2. Identify next available group id

```
# expr `cut --delimiter=":" --fields=3 /etc/group | \  
> grep --regexp="^[5-9][0-9][0-9]$" | sort --numeric-sort|tail -1` + 1
```

3. Create **dba** group and assign the larger of the group ids identified in step 2, e.g. 504.

```
# groupadd --gid 504 dba
```

4. Identify next available user id

```
# expr `cut --delimiter=":" --fields=3 /etc/passwd | \
> grep --regexp="^[5-9][0-9][0-9]$" | sort --numeric-sort|tail -1` + 1
```

5. Create **oracle** user and assign the larger of the user ids identified in step 4, e.g. 504.

```
# useradd --uid 504 --gid dba --home-dir /usr/lib/oracle-xe -M oracle
```

6. If the **dba** group ids are different, change the group id on **one of the nodes** to be the same as the other, ensure that the new group id is not already in use.

```
# grep 504 /etc/group # check that group id 504 is not in use
# groupmod -g 504 dba # change group id to 504
```

7. If the **oracle** user ids are different, change the user id on **one of the nodes** to be the same as the other, ensure that the new user id is not already in use.

```
# grep 504 /etc/passwd # check that user id 504 is not in use
# usermod --uid 504 --gid dba oracle # change user id to 504
```

8. Change the user and group ownership of files owned by **oracle**.

The following commands will find all files whose user / group ownership is 501 and change their user / group ownership to **oracle / dba**.

```
# find /usr -gid 501 -exec chgrp dba {} \;
# find /usr -uid 501 -exec chown oracle {} \;
```

## Installation

Carry out these instructions on both nodes, all commands are executed as **root**.

1. Check whether Oracle 10g Express Edition is installed, by executing:

```
# rpm --query --all|grep oracle
```

If the following is displayed then Oracle 10g Express Edition is installed and you may proceed to step 4, otherwise continue to step 2.

```
oracle-xe-10.2.0.1-1.0
```

2. Download Oracle 10g Express Edition i386 RPM package (this is a **single-line** command).

```
# wget http://download.oracle.com/otn/linux/oracle10g/xe/10201/oracle-xe-10.2.0.1-1.0.i386.rpm
```

3. Install Oracle 10g Express Edition

Execute the following:

```
# rpm --install --hash oracle-xe-10.2.0.1-1.0.i386.rpm
```

which should generate the following output

```
##### [100%]
##### [100%]
Executing Post-install steps...
```

```
You must run '/etc/init.d/oracle-xe configure' as the root user to
configure the database.
```

4. Configure Oracle 10g Express Edition

If Oracle 10g Express Edition was just installed then configure it by executing

```
# /etc/init.d/oracle-xe configure
```

accept the default ports, provide a password for the SYS and SYSTEM, **enter n when prompted whether to start Oracle 10g Express Edition at boot.**

OR, if this is an existing Oracle 10g Express Edition installation that is configured to start at boot then disable automatic startup by executing:

```
# /etc/init.d/oracle-xe disable
```

#### 5. Edit **/etc/init.d/oracle-xe**

By default, the **oracle-xe** script will start or stop the application only if it is enabled at boot. This behaviour has to be changed in order to use this script to start and stop the application from the cluster.

Replace lines 597-628 in **/etc/init.d/oracle-xe** with the following:

```
case "$1" in
  start)
    start
    ;;
  configure)
    configure
    ;;
  stop)
    stop
    ;;
```

#### 6. Start the Oracle database

```
# /etc/init.d/oracle-xe start
```

#### 7. Delete the automatic startup and shutdown scripts

```
# rm /etc/rc0.d/K??oracle-xe /etc/rc3.d/S??oracle-xe
```

#### 8. Configure firewall

If the firewall is running, it must be configured to permit access to the listener and HTTP server ports defined in step 4. By default, these are ports 1521 and 8080 respectively.

Insert the following before the **COMMIT** command in **/etc/sysconfig/iptables**.

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j ACCEPT
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 1521 -j ACCEPT
```

Restart the firewall, by executing:

```
# /etc/init.d/iptables restart
```

## **Directories/Files for Replication**

All paths are relative to **/usr/lib/oracle/xe**

Directory/File	Comment
oradata	Database file directory
app/oracle/admin	Database parameter file and alert log directory
app/oracle/flash_recovery_area	Redo log, archive log, and RMAN backup directory.

Directory/File	Comment
app/oracle/product/10.2.0/server/dbs/spfileXE.ora	Server parameter file for XE database
app/oracle/product/10.2.0/server/dbs/orapwXE	File that stores names and passwords of XE database users with the SYSDBA privilege
app/oracle/product/10.2.0/server/network/admin	Network configuration script directory

## Configuration

Steps 2 to 15 are performed as **oracle**, the rest as **root**.

### Primary Node

1. Login as **oracle**

```
# su - oracle
```

2. Set the Oracle environment variables.

```
$ . $HOME/app/oracle/product/10.2.0/server/bin/oracle_env.sh
```

Append the command to **~/.bash\_profile** so that the environment variables are set automatically.

3. Connect to the database as SYSDBA.

```
$ sqlplus "/" as sysdba"
```

4. Change the address of the default local listener to the virtual IP address (192.168.1.39).

```
SQL> ALTER SYSTEM
  2 SET LOCAL_LISTENER='(ADDRESS = (PROTOCOL = TCP)
  3 (HOST = 192.168.1.39) (PORT = 1521))';
SQL> ALTER SYSTEM REGISTER;
```

5. (Optional) Allow remote access to the database server.

```
SQL> EXEC DBMS_XDB.SETLISTENERLOCALACCESS(FALSE);
```

6. (Optional) Enable the HR account and assign a password.

```
SQL> ALTER USER HR IDENTIFIED BY HR ACCOUNT UNLOCK;
```

7. Exit SQL\*Plus

```
SQL> EXIT
```

8. Create directory **~/app/oracle/admin/network** for storage of listener configuration scripts.

```
$ mkdir ~/app/oracle/admin/network
```

9. Copy the contents of **\$ORACLE\_HOME/network/admin** to **~/app/oracle/admin/network**

```
$ cp -r $ORACLE_HOME/network/admin/* ~/app/oracle/admin/network
```

10. Configure the Oracle listener to use the virtual IP address assigned to the clustered application.

Replace the host name in line 16 of **~/app/oracle/admin/network/listener.ora** with

the virtual IP address (192.168.1.39) as shown:

```
(ADDRESS = (PROTOCOL = TCP) (HOST = 192.168.1.39) (PORT = 1521))
```

11. Configure the database alias to point to the virtual IP address.

Replace the host name in line 5 of `~/app/oracle/admin/network/tnsnames.ora` with the virtual IP address as shown:

```
(ADDRESS = (PROTOCOL = TCP) (HOST = 192.168.1.39) (PORT = 1521))
```

12. Copy the initialization parameter file - **spfileXE.ora** - and the password file - **orapwXE** - from **\$ORACLE\_HOME/dbs** to **~/app/oracle/admin/XE/dbs**.

```
$ cp $ORACLE_HOME/dbs/spfileXE.ora $ORACLE_HOME/dbs/orapwXE
> ~/app/oracle/admin/XE/dbs
```

13. Create **\$ORACLE\_HOME/server/dbs/initXE.ora** as follows:

```
SPFILE=/usr/lib/oracle/xe/app/oracle/admin/XE/dbs/spfileXE.ora
```

14. Create a symbolic link to **~/app/oracle/admin/XE/dbs/orapwXE**.

```
$ rm $ORACLE_HOME/dbs/orapwXE
$ ln -s ~/app/oracle/admin/XE/dbs/orapwXE \
> ~/app/oracle/product/10.2.0/server/dbs/orapwXE
```

15. Logout the **oracle** user.

```
$ exit
```

16. Shut down Oracle 10g Express Edition

```
# /etc/init.d/oracle-xe stop
```

17. Append the following to **etc/profile**:

```
TNS_ADMIN=/usr/lib/oracle/xe/app/oracle/admin/network;export TNS_ADMIN
```

## Secondary Node

Steps 2 to 5 are performed as **oracle**, the rest as **root**.

1. Login as **oracle**

```
# su - oracle
```

2. Append the following to **~/bash\_profile**:

```
. $HOME/app/oracle/product/10.2.0/server/bin/oracle_env.sh
```

3. Create **~/app/oracle/product/10.2.0/server/dbs/initXE.ora** as follows:

```
SPFILE=/usr/lib/oracle/xe/app/oracle/admin/XE/dbs/spfileXE.ora
```

4. Create a symbolic link to **~/app/oracle/admin/XE/dbs/orapwXE**.

```
$ rm $ORACLE_HOME/dbs/orapwXE
$ ln -s ~/app/oracle/admin/XE/dbs/orapwXE \
> ~/app/oracle/product/10.2.0/server/dbs/orapwXE
```

5. Logout the **oracle** user.

```
$ exit
```

6. Shut down Oracle 10g Express Edition

```
# /etc/init.d/oracle-xe stop
```

7. Append the following line to **/etc/profile**:

```
TNS_ADMIN=/usr/lib/oracle/xe/app/oracle/admin/network;export TNS_ADMIN
```

## Create Replicated File Systems

All commands are to be executed as **root**. Steps 1 to 3 must be executed on both nodes, the rest on the primary node.

1. Use **fdisk** to create a partition (*/dev/sdb5*) to be used by the application volume group. This should be at least 2 GB.

2. Initialize partition

```
# pvcreate /dev/sdb5
```

3. Create application volume group (*oraclevg*)

```
# vgcreate oraclevg /dev/sdb5
```

4. Create logical volume (*adminlv*) to store parameter files and alert logs, this should be large enough to cater for growth of the alert logs.

```
# lvcreate --size 64M --name adminlv oraclevg
```

5. Create logical volume (*flashlv*) for flash recovery area, this needs to be large enough to hold the database backups and redo logs (active and archived).

```
# lvcreate --size 512M --name flashlv oraclevg
```

6. Create logical volume (*oradata1v*) to store database files, this should be at least 1 GB.

```
# lvcreate --size 1200M --name oradata1v oraclevg
```

7. Create file systems on *adminlv*, *flashlv* and *oradata1v*.

```
# mkfs -t ext3 /dev/oraclevg/adminlv
# mkfs -t ext3 /dev/oraclevg/flashlv
# mkfs -t ext3 /dev/oraclevg/oradata1v
```

## Populate Replicated File Systems

These instructions are to be carried out on the primary node only.

1. Copy contents of **/usr/lib/oracle/xe/app/oracle/admin** to *adminlv*.

```
# mount -t ext3 /dev/oraclevg/adminlv /mnt
# cd /usr/lib/oracle/xe/app/oracle/admin
# find . -print | cpio --pass-through /mnt
# umount /mnt
```

2. Copy contents of **/usr/lib/oracle/xe/app/oracle/flash\_recovery\_area** to *flashlv*.

```
# mount -t ext3 /dev/oraclevg/flashlv /mnt
# cd /usr/lib/oracle/xe/app/oracle/flash_recovery_area
# find . -print | cpio --pass-through /mnt
```

```
# umount /mnt
```

3. Copy contents of **/usr/lib/oracle/xe/oradata** to *oradata1v*.

```
# mount -t ext3 /dev/oraclevg/adminlv /mnt
# cd /usr/lib/oracle/xe/oradata
# find . -print | cpio --pass-through /mnt
# umount /mnt
```

## Build Application

These instructions are to be carried out on the primary node ONLY.

1. Create directory **/etc/cluster/oracle-xe**

```
# mkdir /etc/cluster/oracle-xe
```

2. Create **/etc/cluster/oracle-xe/appconf.xml** as shown:

```
<?xml version="1.0"?>
<appconf>
  <global>
    <version>0.1</version>
    <name>oracle-xe</name>
    <takeover>normal</takeover>
    <syncrate>2000</syncrate>
    <preferred_node>fc5s1</preferred_node>
  </global>
  <networks>
    <network net="main"
      ip="192.168.1.39" netmask="255.255.255.0"
    />
  </networks>
  <vg>
    <name>oraclevg</name>
    <type>filesystems</type>
  </vg>
  <application>
    <startscript>/etc/init.d/oracle-xe start</startscript>
    <stopscript>/etc/init.d/oracle-xe stop</stopscript>
    <maxstoptime>30</maxstoptime>
    <maxstarttime>60</maxstarttime>
  </application>
</appconf>
```

The bolded values are configuration-specific, as described in the following table:

Entry	Value
global/preferred_node	Host name of one of the nodes in the cluster, or LEAST_CPU_LOAD
networks/network.net	Same as one of <i>node/network.name</i> in <b>/etc/cluster/clconf.xml</b>
networks/network.ip	Virtual IP address of application
networks/network.netmask	Netmask corresponding to virtual IP address
vg/name	Name of application volume group

3. Create `/etc/cluster/oracle-xe/lems.local.xml` as shown:

```
<?xml version="1.0"?>
<lems_config>
  <globals
    modules="/sbin/cluster/lems/modules"
    programs="/sbin/cluster/lems/programs"
    logs="/var/log/cluster/lems"
  />

  <check>
    <name>flag_check</name>
    <type>internal</type>
    <module>flag_check oracle-xe</module>
    <interval>5</interval>
    <action_list>
      <action rc="0" action="NOP"/>
      <action rc="1" action="%RCDATA%"/>
      <action rc="2" action="ABORT"/>
    </action_list>
  </check>
  <check>
    <name>db</name>
    <type>internal</type>
    <module>procmon /etc/cluster/oracle-xe/db.xml</module>
    <interval>10</interval>
    <action_list>
      <action rc="0" action="NOP"/>
      <action rc="1" action="STOP"/>
      <action rc="2" action="FAILOVER"/>
    </action_list>
  </check>
  <check>
    <name>listener</name>
    <type>internal</type>
    <module>procmon /etc/cluster/oracle-xe/listener.xml</module>
    <interval>15</interval>
    <action_list>
      <action rc="0" action="NOP"/>
      <action rc="1" action="STOP"/>
      <action rc="2" action="FAILOVER"/>
    </action_list>
  </check>
  <check>
    <name>fsmonitor</name>
    <type>internal</type>
    <module>fsmon oracle-xe</module>
    <interval>10</interval>
    <action_list>
      <action rc="0" action="NOP"/>
      <action rc="1" action="PAUSE 30"/>
      <action rc="2" action="STOP"/>
      <action rc="3" action="FAILOVER"/>
      <action rc="10" action="PAUSE 60"/>
    </action_list>
  </check>
</lems_config>
```

4. Create **/etc/cluster/oracle-xe/db.xml** as shown:

```
<?xml version="1.0"?>
<procmon>
  <global>
    <logdir>/var/log/cluster</logdir>
    <restarts>5</restarts>
    <resetwindow>3600</resetwindow>
    <restartcmd>/etc/init.d/oracle-xe restart </restartcmd>
  </global>
  <process>
    <label>Database Writer Process</label>
    <user>oracle</user>
    <process_string>xe_dbw?_XE</process_string>
    <min_count>1</min_count>
    <max_count>10</max_count>
  </process>
  <process>
    <label>Other Server Processes</label>
    <user>oracle</user>
    <process_string>xe_(lgwr|ckpt|pmon|reco|smon)_XE</process_string>
    <min_count>5</min_count>
    <max_count>5</max_count>
  </process>
  <!-- Uncomment this section if archive log is enabled -->
  <!--
    <process>
      <label>Database Archiver Process</label>
      <user>oracle</user>
      <process_string>xe_arc?_XE</process_string>
      <min_count>1</min_count>
      <max_count>10</max_count>
    </process>
  -->
</procmon>
```

5. Create **/etc/cluster/oracle-xe/listener.xml** as shown:

```
<?xml version="1.0"?>
<procmon>
  <global>
    <logdir>/var/log/cluster</logdir>
    <restarts>5</restarts>
    <resetwindow>3600</resetwindow>
    <restartcmd>
      /bin/su - oracle -c "lsnrctl stop; lsnrctl start"
    </restartcmd>
  </global>
  <process>
    <label>Oracle 10g Express Edition Listener</label>
    <user>oracle</user>
    <process_string>
      /usr/lib/oracle/xe/app/oracle/product/10.2.0/server/bin/tnslsnr*
    </process_string>
    <min_count>1</min_count>
    <max_count>1</max_count>
  </process>
</procmon>
```

## 6. Verify LEMS configuration by executing:

```
# lems.pl --config /etc/cluster/oracle-xe/lems.local.xml \  
> --application oracle-xe --verbose --check --file /dev/tty
```

If there are no errors, a result similar to the following will be returned:

```
INFO 09/09/2006 00:39:42 Using modules from : /sbin/cluster/lems/modules  
INFO 09/09/2006 00:39:42 Using programs from : /sbin/cluster/lems/programs  
INFO 09/09/2006 00:39:42 Writing logs to : /var/log/cluster/lems  
INFO 09/09/2006 00:39:42 Listening on port : 9904  
INFO 09/09/2006 00:39:42 Global initialisation complete.  
INFO 09/09/2006 00:39:42 Started local server on port 9904  
INFO 09/09/2006 00:39:42 Validating monitor entry fsmonitor...  
INFO 09/09/2006 00:39:42 Validated monitor entry fsmonitor successfully.  
INFO 09/09/2006 00:39:42 Validating monitor entry flag_check...  
INFO 09/09/2006 00:39:42 Validated monitor entry flag_check successfully.  
INFO 09/09/2006 00:39:42 Validating monitor entry oracle-xe...  
INFO 09/09/2006 00:39:42 Validated monitor entry oracle-xe successfully.  
INFO 09/09/2006 00:39:42 Check mode - transferring validated config to remote node.  
INFO 09/09/2006 00:39:42 Configuration transferred successfully.  
INFO 09/09/2006 00:39:42 Calculated a check interval of 2.5 seconds.
```

## 7. Mount the replicated file systems

```
# mount /dev/oraclevg/adminlv /usr/lib/oracle/xe/app/oracle/admin  
# mount /dev/oraclevg/flashlv /usr/lib/oracle/xe/app/oracle/flash_recovery_area  
# mount /dev/oraclevg/oradata1lv /usr/lib/oracle/xe/oradata
```

## 8. Build **oracle-xe**.

```
# clbuildapp --application oracle-xe --sync
```

The following is the output of a successful build.

```
INFO 21/07/2006 23:37:06 Backups directory defaulted to /clbackup  
INFO 21/07/2006 23:37:06  
INFO 21/07/2006 23:37:06 Validation of Application 'oracle-xe' started.  
INFO 21/07/2006 23:37:06 ['var/log/cluster/build/oracle-xe-check-  
300607212337.log']  
INFO 21/07/2006 23:37:07 Initial Validation of Application successful.  
INFO 21/07/2006 23:37:08  
INFO 21/07/2006 23:37:08 NOTE: Build of new application is being performed.  
INFO 21/07/2006 23:37:08  
INFO 21/07/2006 23:37:08 Host Environment Validation started.  
INFO 21/07/2006 23:37:08 ['var/log/cluster/build/oracle-xe-envcheck-  
300607212337.log']  
INFO 21/07/2006 23:37:12 Host Environment Validation successful.  
INFO 21/07/2006 23:37:12  
INFO 21/07/2006 23:37:12 Cluster state : DOWN  
INFO 21/07/2006 23:37:12 Application state: UNDEFINED  
INFO 21/07/2006 23:37:12  
INFO 21/07/2006 23:37:12 Volume Group Configuration started.  
INFO 21/07/2006 23:37:12 ['var/log/cluster/build/oracle-xe-lvm-  
300607212337.log']  
INFO 21/07/2006 23:37:20 Volume Group Configuration successful.  
INFO 21/07/2006 23:37:20  
INFO 21/07/2006 23:37:20 Application Resource Allocation started.  
INFO 21/07/2006 23:37:20 ['var/log/cluster/build/oracle-xe-build-  
300607212337.log']  
INFO 21/07/2006 23:37:33 Application Resource Allocation successful.  
INFO 21/07/2006 23:37:33  
INFO 21/07/2006 23:37:33 Application Data Synchronisation started.  
INFO 21/07/2006 23:37:33 ['var/log/cluster/build/oracle-xe-syncdata-  
300607212337.log']  
Storage Syncing: 1200Mb/ 1Mb [0.1 % Complete]  
Storage Syncing: 0Mb/ 0Mb [100 % Complete]  
INFO 21/07/2006 23:49:46 Application Data Synchronisation successful.  
INFO 21/07/2006 23:49:47
```

## Run Application

For the commands used in this section, the host name of the primary node is *fc5s1*, and the secondary node is *fc5s2*. All commands are executed as **root** on either node unless indicated otherwise.

1. Form the cluster, if necessary.

```
# clform
```

2. Run **oracle-xe** on the primary node

```
# clrunapp --application oracle-xe --node fc5s1
```

3. Verify state of **oracle-xe** using **clstat**

```
# clstat --application oracle-xe
```

If **oracle-xe** is running, a result similar to the following will be displayed:

```
Cluster: cluster1 - UP

Application      Node      State  Runnig  Monitor  Stale  Fail-over?
oracle-xe        fc5s1     STARTED 0:00:00 Running    0      Yes

File Systems

Mount Point      Valid  Type      State  % Complete  Completion
/usr/lib/oracle/xe/app/oracle/admin      both  drbd      Sync
/usr/lib/oracle/xe/app/oracle/flash_recovery_area  both  drbd      Sync
/usr/lib/oracle/xe/oradata      both  drbd      Sync

Process Monitors

Name      Status  Restarts  Current  Reset at
oracle-xe  Running    5         0        N/A

General Monitors

Type      Name      Status
Flag Check  flag_check  Running
FS Monitor  fsmonitor  Running
```

4. On the node where **oracle-xe** is running, list the Oracle processes.

```
# ps -u oracle -f|sort --key=8
```

The following process list should be displayed:

```
UID      PID  PPID  C  STIME  TTY      TIME  CMD
oracle   10991  1    0  11:57  ?        00:00:00
/usr/lib/oracle/xe/app/oracle/product/10.2.0/server/bin/tnslsnr LISTENER -inherit
oracle   11012  1    0  11:57  ?        00:00:00 xe_cjq0_XE
oracle   11006  1    0  11:57  ?        00:00:00 xe_ckpt_XE
oracle   11019  1    0  11:57  ?        00:00:00 xe_d000_XE
oracle   11002  1    0  11:57  ?        00:00:00 xe_dbw0_XE
oracle   11004  1    0  11:57  ?        00:00:00 xe_lgwr_XE
oracle   11000  1    0  11:57  ?        00:00:00 xe_mman_XE
oracle   11016  1    0  11:57  ?        00:00:00 xe_mmn1_XE
oracle   11014  1    0  11:57  ?        00:00:01 xe_mmon_XE
oracle   10996  1    0  11:57  ?        00:00:00 xe_pmon_XE
oracle   10998  1    0  11:57  ?        00:00:00 xe_psp0_XE
oracle   11040  1    0  11:57  ?        00:00:00 xe_q000_XE
oracle   11042  1    0  11:57  ?        00:00:00 xe_q001_XE
oracle   11031  1    0  11:57  ?        00:00:00 xe_qmnc_XE
oracle   11010  1    0  11:57  ?        00:00:00 xe_reco_XE
oracle   11021  1    0  11:57  ?        00:00:00 xe_s000_XE
```

```

oracle 11023 1 0 11:57 ? 00:00:00 xe_s001_XE
oracle 11025 1 0 11:57 ? 00:00:00 xe_s002_XE
oracle 11027 1 0 11:57 ? 00:00:00 xe_s003_XE
oracle 11008 1 0 11:57 ? 00:00:00 xe_smon_XE

```

5. Check listener status on the node where **oracle-xe** is running.

```
$ su - oracle -c "lsnrctl stat"
```

The listener should return a status similar to the following:

```

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=EXTPROC_FOR_XE)))
STATUS of the LISTENER
-----
Alias                LISTENER
Version              TNSLSNR for Linux: Version 10.2.0.1.0 - Production
Start Date           24-JUL-2006 12:05:19
Uptime               0 days 1 hr. 11 min. 37 sec
Trace Level          off
Security             ON: Local OS Authentication
SNMP                 OFF
Default Service      XE
Listener Parameter File /usr/lib/oracle/xe/app/oracle/product/10.2.0/server
/network/admin/listener.ora
Listener Log File    /usr/lib/oracle/xe/app/oracle/product/10.2.0/server
/network/log/listener.log
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=EXTPROC_FOR_XE)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.168.1.39) (PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=fc5s1) (PORT=8080)) (Presentation=HTTP) (
Session=RAW))
a
Services Summary...
Service "PLSExtProc" has 1 instance(s).
  Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "XE" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
Service "XEXDB" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
Service "XE_XPT" has 1 instance(s).
  Instance "XE", status READY, has 1 handler(s) for this service...
b
The command completed successfully
c

```

Referring to the bolded lines:

- The listener should be running on the cluster application virtual IP address "HOST=**192.168.1.39**"
- The address for the HTTP service should be either the node name (HOST=**fc5s1**) or the virtual IP address (HOST=**192.168.1.39**)
- These six lines are displayed if the database has registered with the listener.

6. Verify database connection using SQL\*Plus on either node, by executing the following:

```

# su - oracle -c "sqlplus -S /nolog" << EOF
> CONNECT HR/HR@XE
> COL REGION_NAME FOR A30
> SELECT * FROM REGIONS;
> EXIT
> EOF

```

The following output would be generated if the connection was successful:

```

REGION_ID REGION_NAME
-----
1 Europe
2 Americas
3 Asia
4 Middle East and Africa

```

7. Verify remote access via HTTP from any networked computer by connecting to the apex application on the virtual IP address (192.168.1.39) at port 8080 from a web browser or by executing:

```
# curl http://192.1.168.39:8080/apex
```

If the HTTP server is accepting connections on the virtual IP address, **curl** will produce no output, otherwise an error message will be returned.

8. Stop **oracle-xe** by executing the following on the node where it is running:

```
# clhaltapp --application oracle-xe
```

9. Start **oracle-xe** on the secondary node (*fc5s2*)

```
# clrunapp --application oracle-xe --node fc5s2
```

10. Repeat steps 3-8.