Installing Oracle 10g Enterprise Edition on Red Hat Enterprise Linux 5

Ahmet Emre Aladağ

12-07-2008

Contents

Chapter 1

Preparation

1.1 Introduction

We used Gnome as our session manager and logged in with root user on Gnome. Be sure to have installed developer tools before starting the installation.

1.2 Requirements

1.2.1 Hardware Requirements

Memory requirements:

- 1 GB for the instance with Database Control

Disk space requirements:

- 1.5 GB of swap space
- 400 MB of disk space in the /tmp directory
- Between 1.5 GB and 3.5 GB for the Oracle software
- 1.2 GB for the preconfigured database (optional)
- 2.4 GB for the flash recovery area (optional)

1.2.2 Software Requirements

Make sure that you have the following packages (libs): libXp, compat-libstdc++

It is required to have the following packages which you can check with the following command:

```
rpm -q binutils compat-db control-center gcc gcc-c++ \
glibc glibc-common gnome-libs libstdc++ libstdc++-devel \
make pdksh sysstat xscreensaver
```

1.3 Changing the Red Hat Release

As RHEL5 has been released after Oracle 10g was, Oracle 10g does not support RHEL5 officially. The latest version 10g supports in RHEL4. So we have to get round it by changing the release configuration:

```
echo "rhel4" > /etc/redhat-release
```

1.4 Creating the oracle user account

We need to create oinstall and dba groups and an oracle account belonging to them. When logged as root user,

```
/usr/sbin/groupadd oinstall
/usr/sbin/groupadd dba
/usr/sbin/useradd -m -g oinstall -G dba oracle
```

Set the password for the oracle user:

```
passwd oracle
```

You can set "oracle" as the password too

1.5 Creating the Installation Path and Setting the Permissions

```
mkdir -p /u01/app/oracle
chown -R oracle:oinstall /u01/app/oracle
chmod -R 775 /u01/app/oracle
```

1.6 Configuring Kernel Parameters

We can set the kernel parameters with the following command:

```
cat >> /etc/sysctl.conf << EOF
kernel.shmall = 2097152
kernel.shmmax = 536870912
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 658576
net.ipv4.ip_local_port_range = 1024 65000
net.core.rmem_default = 262144
net.core.wmem_default = 262144
net.core.rmem_max = 1048536
net.core.wmem_max = 1048536
EOF
```

Activate the modification with the following command:

```
/sbin/sysctl -p
```

1.7 Configuring Shell Limits

```
cat >> /etc/security/limits.conf << EOF
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
EOF
cat >> /etc/pam.d/login << EOF
session required /lib/security/pam_limits.so
EOF
cat >> /etc/profile << EOF
if [ \$USER = "oracle" ]; then
 if [ \$SHELL = "/bin/ksh" ]; then
  ulimit -p 16384
  ulimit -n 65536
 else
  ulimit -u 16384 -n 65536
 fi
 umask 022
fi
EOF
cat >> /etc/csh.login << EOF
if ( \$USER == "oracle" ) then
 limit maxproc 16384
 limit descriptors 65536
 umask 022
endif
EOF
```

1.8 Configuring Hangcheck Timer

```
modprobe hangcheck-timer hangcheck_tick=30 hangcheck_margin=180
cat >> /etc/rc.d/rc.local << EOF
modprobe hangcheck-timer hangcheck_tick=30 hangcheck_margin=180
EOF
```

1.9 Environment Settings for Oracle User

In this section, we'll use oracle user.

```
su - oracle
```

To set the environment variables, apply the following command to append Oracle-specific contents to /home/oracle/.bash_profile:

```
cat >> /home/oracle/.bash_profile << EOF
# User specific environment and startup programs
umask 022
PATH=/bin:/usr/bin:/usr/local/bin:/usr/X11R6/bin
export LD_LIBRARY_PATH=/usr/lib:/usr/X11R6/lib
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=\$ORACLE_BASE/product/10.2.0/db_1
export ORACLE_SID=orcl
PATH=\$ORACLE_HOME/bin:\$ORACLE_HOME/jdk/jre/lib/i386/server:\
\$ORACLE_HOME/rdbms/lib:\$ORACLE_HOME/lib:\$LD_LIBRARY_PATH:\
/usr/kerberos/sbin:/usr/kerberos/bin:/usr/local/sbin:/sbin:\
/usr/sbin:/root/bin:\$PATH
PATH=\$PATH:\$HOME/bin
export PATH
unset USERNAME
EOF
```

Activate these variables and the paths with:

```
source ~/.bash_profile
```

Chapter 2

Installation

2.1 Running the Installer

In a root terminal,

```
xhost +
su - oracle
```

Extract the 10201_database_linux32.zip file into /u01, or somewhere else where oracle user has read right:

```
unzip 10201_database_linux32.zip -d /u01
cd /u01/database ./runInstaller
```

The installation dialog will now start. If you face errors about some libs are not found, consider installing the following packages: libXp compat-libstdc++ from http://www.rpmfind.net

2.2 Installation Steps

1. Select "Advanced Installation" and click Next.
2. Leave "Inventory directory path" and "group name" as it is and click Next.
3. Leave "Enterprise Edition" selected and click Next.
4. Verify that the following information is true:
   Name: OraDb10g_home1
   Path: /u01/app/oracle/product/10.2.0/db_1
   Be careful that it is not /u01/app/oracle/oracle/product/10.2.0/db_1
5. It will check the prerequisites. Do not start if there are any errors and be careful about memory and environment variable warnings.

6. Leave "Create a database" selected and click Next.
7. Select Advanced and click Next.
8. Click Install and it will install the database system. Then DBCA starts automatically, configuring the components you installed so far.
9. Select General Purpose and click Next.
10. Set orcl for Database Name and SID and click Next.
11. Leave "Configure the Database with Enterprise" checkbox selected and click Next.
12. Assisgn a common password for SYS account, such as "oracle" and click next.
13. Select Filesystem and click Next.
14. Select "Oracle-Managed Files" and click Next.
15. Leave recovery area as it is and click Next.
16. If you face a "Database Content" dialog, click Next with no additional setting.
17. Set Use Unicode (AL32UTF8) as Database Character set and UTF16 as National Character Set on the "Character Sets" tab.
18. Click Next and OK until you get a confirmation to create the database.
19. Close the information after the database is created. Click Exit.
20. The "Configuration assistant"s will start the components you installed earlier.
21. You will face a dialog which wants you to execute two sh scripts. Open a new terminal (as root) and run those scripts:
    sh /u01/app/oracle/oraInventory/orainstRoot.sh
    sh /u01/app/oracle/product/10.2.0/db_1/root.sh
22. Click OK **after** you run the scripts.
23. Click Exit.

Chapter 3

Post-Installation

3.1 Starting the services

Now, the Oracle instance should be installed and working. Let's check it:

```
su - oracle
sqlplus / as sysdba
```

If it says "Connected to an idle instance", it means that the instance is not started. Do the following:

```
lsnrct start
emctl start dbconsole
sqlplus / as sysdba
startup (in sqlplus session)
```

You will get some feedback about the current status of the services and the results of the commands.

3.2 Starting Oracle Instance up on System Startup

3.2.1 Creating the startup bash script

Open a text editor (as root) and put the content below in it, saving the file as /etc/init.d/dbora :

```
#!/bin/sh
# chkconfig: 345 99 10
# description: Oracle auto start-stop script.
#
# Set ORA_HOME to be equivalent to the $ORACLE_HOME
# from which you wish to execute dbstart and dbshut;
```

```
#
# Set ORA_OWNER to the user id of the owner of the
# Oracle database in ORA_HOME.
ORA_HOME=/u01/app/oracle/product/10.2.0/db_1
ORA_OWNER=oracle
if [ ! -f $ORA_HOME/bin/dbstart ]
then
    echo "Oracle startup: cannot start"
    exit
fi
case "$1" in
    'start')
        # Start the Oracle databases:
        # The following command assumes that the oracle login
        # will not prompt the user for any values
        su - $ORA_OWNER -c "$ORA_HOME/bin/lsnrctl start"
        su - $ORA_OWNER -c $ORA_HOME/bin/dbstart
        su - $ORA_OWNER -c "$ORA_HOME/bin/emctl start dbconsole"
        ;;
    'stop')
        # Stop the Oracle databases:
        # The following command assumes that the oracle login
        # will not prompt the user for any values
        su - $ORA_OWNER -c "$ORA_HOME/bin/emctl stop dbconsole"
        su - $ORA_OWNER -c $ORA_HOME/bin/dbshut
        su - $ORA_OWNER -c "$ORA_HOME/bin/lsnrctl stop"
        ;;
esac
```

Then, open a root terminal and give the file executable rights:

```
chmod 755 /etc/init.d/dbora
```

Then link this script to the files inside /etc/rc3.d:

```
ln -s /etc/init.d/dbora /etc/rc3.d/S99dbora
ln -s /etc/init.d/dbora /etc/rc4.d/S99dbora
ln -s /etc/init.d/dbora /etc/rc5.d/S99dbora
ln -s /etc/init.d/dbora /etc/rc0.d/K10dbora
ln -s /etc/init.d/dbora /etc/rc6.d/K10dbora
```

3.2.2 Fixing the dbstart script

The dbstart script, which is called inside the dbora script, has some errors. Open the file /u01/app/oracle/product/10.2.0/db_1/bin/dbstart as oracle user and change

```
ORACLE_HOME_LISTNER=/ade/vikrkuma_new/oracle
```

to

```
ORACLE_HOME_LISTNER=$ORACLE_HOME
```

Otherwise, your listener might not get started.

3.2.3 Flagging the orcl instance to be started

We open the /etc/oratab file and change the last letter from N to Y:

```
orcl:/u01/app/oracle/product/10.2.0/db_1:N
```

to

```
orcl:/u01/app/oracle/product/10.2.0/db_1:Y
```

3.3 Setting the IP Address

If the instance is running, shut it down with (as oracle user):

```
sqlplus / as sysdba
shutdown immediate (might cause data loss)
```

Then, open the /u01/app/oracle/product/10.2.0/db_1/network/admin/tnsnames.ora file and set your hostname/static IP Address:

```
HOST = oracle10
```

Start the instance in sqlplus session:

```
startup
```

Chapter 4

Useful Resources

- libXp package: http://rpm2html.osmirror.nl/fedora/core/5/i386/os/Fedora/RPMS/libXp-1.0.0-2.2.i386.html
- libstc++ package: http://rpm2html.osmirror.nl/fedora/core/3/i386/os/Fedora/RPMS/compat-libstdc++-8-3.3.4.2.i386.html
- Startup script: http://listweb.bilkent.edu.tr/Linux/linux-sunucu/2005/Nov/0095.html
- Linking startup script: http://www.idevelopment.info/data/Oracle/DBA_tips/Linux/LINUX_8.shtml
- Fixing dbstart: http://forums.oracle.com/forums/thread.jspa?threadID=336184
- Flagging Instance to start up: http://download-uk.oracle.com/docs/html/B10812_01/chapter2.htm#sthref210