

Installing Oracle 11g on OS4 Enterprise Linux

By The OS4 Documentation Team

Prepared by: Roberto J Dohnert

Copyright © 2013, PC/OpenSystems LLC



Downloading Oracle 11g 64 bits

First, download Oracle 11gR2 Enterprise/Standard Edition for Linux x86-64 from Oracle website.

Then extract the downloaded file:

```
unzip linux.x64_11gR2_database_1of2.zip
unzip linux.x64_11gR2_database_2of2.zip
```

A new folder named *database* will be extracted.

Preparing the system

Second, read the HTML documentation of the quick install guide. You can find this in the extracted folder at this location: *database/doc/install.112/e10860/toc.htm*

For the rest of this howto, we support Oracle being installed at */opt/oracle* in a fresh install of OS4 Enterprise Linux.

Next in preparation we will have to:

- Create required users and groups accounts
- Set required system parameters in */etc/sysctl.conf*
- Installing complementary packages

Creating system accounts

Creating Oracle Inventory group:

```
sudo groupadd oinstall
```

Creating Oracle DBA group:

```
sudo groupadd dba
```

Creating Oracle user home and account:

```
sudo mkdir -p /opt/oracle
sudo useradd -g oinstall -G dba -d /opt/oracle -s /bin/bash oracle
sudo passwd oracle
sudo chown -R oracle:oinstall /opt/oracle
```

Updating kernel parameters

This is one of the longest tasks of this installation. Because you will have to check your current configuration.

Edit the file */etc/sysctl.conf* as described below:

```
sudo gedit /etc/sysctl.conf
```

Add the following lines to get the minimal values required by Oracle, adjusting for your requirements

```

#
# Oracle 11g
#
kernel.sem = 250 32000 100 128
kernel.shmall = 2097152
kernel.shmmni = 4096
# Replace kernel.shmmax with the half of your memory in bytes
# if lower than 4Go minus 1
# 1073741824 is 1 GigaBytes
kernel.shmmax=1073741824

# Try sysctl -a | grep ip_local_port_range to get real values
net.ipv4.ip_local_port_range = 9000 65500

net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576

# Max value allowed, should be set to avoid IO errors
fs.aio-max-nr = 1048576
# 512 * PROCESSES / what really means processes ?
fs.file-max = 6815744

# To allow dba to allocate hugetlbfs pages
# 1001 is your oinstall group, id. grep oinstall /etc/group will give this value
vm.hugetlb_shm_group = 1001

```

Then, run the command below to set these kernel parameters:

```
sudo sysctl -p
```

Now, update */etc/security/limits.conf*:

```
sudo gedit /etc/security/limits.conf
```

And add the following parameters:

```

# Oracle
oracle      soft    nproc   2047
oracle      hard    nproc   16384
oracle      soft    nofile  1024
oracle      hard    nofile  65536
oracle      soft    stack   10240

```

Check that */etc/pam.d/login* contains a line like this one:

```
session    required    pam_limits.so
```

Now, update */etc/profile*:

```
sudo gedit /etc/profile
```

And add the following lines:

```

if [ $USER = "oracle" ]; then
    if [ $SHELL = "/bin/ksh" ]; then

```

```
        ulimit -p 16384
        ulimit -n 65536
    else
        ulimit -u 16384 -n 65536
    fi
fi
```

Installing additional packages

If not already done, update OS4 Enterprise Linux with the latest packages:

```
sudo apt-get update
sudo apt-get dist-upgrade
```

Install the following additional packages:

```
sudo apt-get install alien
sudo apt-get install autoconf
sudo apt-get install automake
sudo apt-get install autotools-dev
sudo apt-get install binutils
sudo apt-get install bzip2
sudo apt-get install doxygen
sudo apt-get install elfutils
sudo apt-get install expat
sudo apt-get install gawk
sudo apt-get install gcc
sudo apt-get install gcc-multilib
sudo apt-get install g++-multilib
```

```
# This will help preventing prevent from runInstaller terrific ./runInstaller:
line 54: ./install/.oui: No such file or directory error message
sudo apt-get install ia32-libs
```

```
sudo apt-get install ksh
sudo apt-get install less
sudo apt-get install lesstif2
sudo apt-get install lesstif2-dev
sudo apt-get install lib32z1
sudo apt-get install libaio1
sudo apt-get install libaio-dev
sudo apt-get install libc6-dev
sudo apt-get install libc6-dev-i386
sudo apt-get install libc6-i386
sudo apt-get install libelf-dev
sudo apt-get install libltdl-dev
sudo apt-get install libmotif4
sudo apt-get install libodbcinstq4-1 libodbcinstq4-1:i386
sudo apt-get install libpth-dev
sudo apt-get install libpthread-stubs0
sudo apt-get install libpthread-stubs0-dev
sudo apt-get install libstdc++5
sudo apt-get install lsb-cxx
sudo apt-get install make
sudo apt-get install openssh-server
sudo apt-get install pdksh
sudo apt-get install rlwrap
```

```
sudo apt-get install rpm
sudo apt-get install sysstat
sudo apt-get install unixodbc
sudo apt-get install unixodbc-dev
sudo apt-get install unzip
sudo apt-get install x11-utils
sudo apt-get install zlibc
```

Preventing installation errors

To prevent from error **genclntsh: Failed to link libclntsh.so.11.1 in make file for rdbms/lib/ins_rdbms.mk**

because of missing library: **/usr/bin/ld: cannot find /usr/lib64/libpthread_nonshared.a inside** we need to create a symlink for /usr/lib64:

```
sudo ln -s /usr/lib/x86_64-linux-gnu /usr/lib64
```

To help preventing from error **lib//libagts.so: undefined reference to `nnfyboot' in make: rdbms/lib/dg4odbc] Error 1**

```
sudo ln -sf /bin/bash /bin/sh
sudo ln -s /usr/bin/awk /bin/awk
sudo ln -s /usr/bin/rpm /bin/rpm
sudo ln -s /usr/bin/basename /bin/basename
```

To prevent from **# /lib64/libgcc_s.so.1: File or directory does not exists, while creating lib/liborasdkbase.so.11.1 in ins_rdbms.mk**, execute the command:

```
cd /lib64
sudo ln -s /lib/x86_64-linux-gnu/libgcc_s.so.1 .
```

```
sudo dpkg -i libstdc++5_3.3.6-17054 Enterprise Linux1_amd64.deb
```

Reboot your system to be sure these parameters will be active for the next part of the installation.

Others configurations

Connect as Oracle user and set some default startup file

```
sudo su - oracle
cd
touch .bashrc
ln -s .bashrc .bash_profile
```

Edit *.bashrc* file:

```
vi .bashrc
```

And add the following line:

```
umask 022
```

Execute the *.bashrc* file or restart a bash shell

```
source .bashrc
```

Installing Oracle

Now we can start the installation of the Oracle 11g archive.

Allow others users to connect to your X session, by entering in a bash shell:

```
xhost +
```

Then, login as *oracle* user:

```
su - oracle
```

Go in the extracted *database* folder and run the following commands:

```
export DISPLAY=:0
./runInstaller
```

Then, Oracle installer should start in graphic mode.

For the rest of the installation, You need choose the following options:

- Installing Oracle 11g Enterprise edition in advanced mode
- Installing server class
- Oracle base directory: /opt/oracle/Oracle11gee
- Oracle home: /opt/oracle/Oracle11gee/product/11.2.0/dbhome_1
- Oracle inventory: /opt/oracle/oraInventory
- Oracle database global name and SID: orcl
- Character set: Unicode AL32UTF8
- Memory: Uncheck activate automatic memory management to prevent from ORA-00845: MEMORY_TARGET not supported on this system. Another solution could be to remove /dev/shm symbolic link and mount it as a bind to /run/shm: mount --bind /run/shm /dev/shm.
- Oradata directory: /opt/oracle/Oracle11g/oradata

Once you will reach the "pre-reqs" screen, check "Ignore all" to continue the installation.

Then, you will encounter issues.

Solving compilation/link errors

Some issues should appear during installation.

If you have not created the symbolic links above, you will have to solve error **lib//libagtsh.so: undefined reference to `nnfyboot' in make: rdbms/lib/dg4odbc] Error 1**. For this, create the symbolic links and execute the commands:

```
export ORACLE_HOME=/opt/oracle/Oracle11gee/product/11.2.0/dbhome_1
cd $ORACLE_HOME/lib
ln -s libclient11.a libagtsh.a
$ORACLE_HOME/bin/genagtsh $ORACLE_HOME/lib/libagtsh.so 1.0
```

Now we will encounter many errors due to indirect library linking:

To solve error # **libnzz11.so: could not read symbols: Invalid operation /sysman/lib/ins_emagent.mk**, enter the command:

```
export ORACLE_HOME=/opt/oracle/Oracle11gee/product/11.2.0/dbhome_1
```

```
sed -i 's/^\(\\s*\$(MK_EMAGENT_NMECTL)\\)\s*$/\1 -lnnz11/g'  
$ORACLE_HOME/sysman/lib/ins_emagent.mk
```

To solve error # **nsglsn.c:(.text+0xc29): undefined reference to `ons_subscriber_close`** /**network/lib/ins_net_server.mk**, enter the command:

```
sed -i 's/^\(TNSLSNR_LINKLINE.*\$(TNSLSNR_OFILES)\\) \(\$(LINKTTLIBS)\\)/\1 -Wl,--no-as-needed \2/g' $ORACLE_HOME/network/lib/env_network.mk
```

To solve error # **libocrutl11.so: undefined reference to `lfifcp`** **rdbms/lib/ins_rdbms.mk**, enter the commands:

```
sed -i 's/^\(ORACLE_LINKLINE.*\$(ORACLE_LINKER)\\) \(\$(PL_FLAGS)\\)/\1 -Wl,--no-as-needed \2/g' $ORACLE_HOME/rdbms/lib/env_rdbms.mk
```

You also need to execute the following commands to prevent from later errors:

```
sed -i 's/^\(\\$LD \\$LD_RUNTIME\\) \(\$(LD_OPT)\\)/\1 -Wl,--no-as-needed \2/g'  
$ORACLE_HOME/bin/genorasdksh  
sed -i 's/^\(\\s*\)(\\$(OCRLIBS_DEFAULT)\\)/\1 -Wl,--no-as-needed \2/g'  
$ORACLE_HOME/srvm/lib/ins_srvm.mk
```

Finishing the installation

The install process will finally ask you to execute scripts as root user, just do so:

```
# sudo /opt/oracle/oraInventory/orainstRoot.sh  
sudo /opt/oracle/Oracle11gee/product/11.2.0/dbhome_1/root.sh
```

You should be able to access database web interface through the URL <https://localhost:1158/em>

Now, add the following variable declaration in your oracle user *.profile*, *.bashrc* or *.bash_profile* scripts:

```
export ORACLE_HOME=/opt/oracle/Oracle11gee/product/11.2.0/dbhome_1  
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib  
export PATH=$PATH:$ORACLE_HOME/bin
```

Then start a new shell or execute one of the updated files:

```
source .bashrc
```

Starting and stopping Oracle

We should now be able to start Oracle database if you had chosen to create one during installation. If you have not yet created any oracle instance and database use the ***\$ORACLE_HOME/bin/dbca*** assistant to do so.

Starting up the database

Once your server have been restarted, your database may not start. To solve this issue, first check in */etc/oratab* that it has the 'Y' flag, if not, set it.

```
sudo gedit /etc/oratab
```


And replace *N* by *Y*

```
orcl:/opt/oracle/Oracle11gee/product/11.2.0/dbhome_1:Y
```

Finally, to manually start Oracle, run these commands as oracle user

```
dbstart $ORACLE_HOME
```

Connecting to the database

Now, just try to connect to the freshly started database:

```
oracle@makina-oracle:~$ sqlplus system@orcl
```

```
SQL*Plus: Release 11.2.0.1.0 Production on Wed Aug 8 11:18:50 2012
```

```
Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

```
Enter password:
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production  
With the Partitioning, OLAP, Data Mining and Real Application Testing options
```

```
SQL> select 'Oracle 11g is running on OS4 Enterprise Linux 4.1.1 64 bit' as Welcome from dual;
```

```
WELCOME
```

```
-----  
Oracle 11g is running on OS4 Enterprise Linux 4.1.1 64 bit
```

```
SQL>
```

Shutting down the database

1. Normal Shutdown

During normal shutdown, before the oracle database is shut down, oracle will wait for all active users to disconnect their sessions. As the parameter name (normal) suggest, use this option to shutdown the database under normal conditions.

```
SQL> shutdown
```

```
Database closed.
```

```
Database dismounted.
```

```
ORACLE instance shut down.
```

```
SQL>
```

2. Shutdown Immediate

During immediate shutdown, before the oracle database is shut down, oracle will rollback active transaction and disconnect all active users. Use this option when there is a problem with your database and you don't have enough time to request users to log-off.

```
SQL> shutdown immediate;
```

```
Database closed.
```

```
Database dismounted.
```

```
ORACLE instance shut down.
```

```
SQL>
```

3. Shutdown Abort

During shutdown abort, before the oracle database is shutdown, all user sessions will be terminated immediately. Uncommitted transactions will not be rolled back. Use this option only during emergency situations when the "shutdown" and "shutdown immediate" doesn't work.

```
SQL> shutdown abort
```

```
ORACLE instance shut down.
```

```
SQL>
```