# 在Linux下安装配置Oracle 11g R2详细过程

**1.Linux环境配置准备**

环境：Linux：Redhat Enterprise 5.4，DB：Oracle 11g R2 X64，Oracle安装到/home/oralce\_11目录下。

配置过程如下：（大部分是网上的内容，个人也是按照网上步骤来的）

1)         检查基本需求（对于牛逼的服务器来说，这个肯定是满足要求的，可以跳过）   
内存大小   
grep MemTotal /proc/meminfo   
交换分区大小   
grep SwapTotal /proc/meminfo   
增加交换分区   
创建/home/swap这么一个分区文件。文件的大小是5120000个block，一般情况下1个block为1K，所以这里空间是5G。   
dd if=/dev/zero of=/home/swap bs=1024 count=5120000   
接着再把这个分区变成swap分区。   
mkswap /home/swap   
然后把它加到fstab里面   
echo "/home/swap swap swap defaults 0 0" >> /etc/fstab   
2)         检查软件需求，没有的yum install安装   
rpm -q binutils compat-libstdc++-33 elfutils-libelf elfutils-libelf-devel glibc glibc-common glibc-devel gcc- gcc-c++ libaio-devel libaio libgcc libstdc++ libstdc++-devel make sysstat unixODBC unixODBC-devel pdksh

（这里如果出现没有安装的，那么就要把Linux光盘放入，进入Server文件夹，找到对应的RPM包，运行RPM命令来进行安装，如果喜欢图形界面的话，在图像界面也可以通过双击对应的RPM包来安装。需要注意的是在安装libaio-devel unixODBC unixODBC-devel 这3个软件包的时候，除了安装X64的包以为，还要安装i386的包，不能因为是X64的系统，就不装i386的包了。 ）   
3)         配置内核。   
编辑 /etc/sysctl.conf。   
将以下内容添加至文件   
# Kernel paramaters required by Oracle 11gR1   
fs.file-max = 6815744   
fs.aio-max-nr = 1048576   
kernel.shmall = 2097152   
kernel.shmmax = 2147483648   
kernel.shmmni = 4096   
kernel.sem = 250 32000 100 128   
net.ipv4.ip\_local\_port\_range = 9000 65500   
net.core.rmem\_default = 4194304   
net.core.rmem\_max = 4194304   
net.core.wmem\_default = 262144   
net.core.wmem\_max = 1048576   
4)         要使 /etc/sysctl.conf 更改立即生效，执行以下命令。   
sysctl –p   
5)         编辑 /etc/security/limits.conf。   
添加以下行：   
oracle           soft    nproc           2047   
oracle           hard    nproc           16384   
oracle           soft    nofile          1024   
oracle           hard    nofile          65536   
6)         编辑 /etc/pam.d/login。   
添加以下行：   
session    required     /lib/security/pam\_limits.so   
session    required     pam\_limits.so

【注意：如果是X64的系统，那么这里就要用 /lib64/security/pam\_limits.so，不然系统可能会在服务器上无法登录！】  
7)         编辑 /etc/profile。   
添加以下行：   
if [ $USER = "oracle" ]; then   
   if [ $SHELL = "/bin/ksh" ]; then   
      ulimit -p 16384   
      ulimit -n 65536   
   else   
      ulimit -u 16384 -n 65536   
   fi   
fi   
8)         创建操作系统用户，作为软件安装和支持组的拥有者。   
groupadd oinstall   
groupadd dba   
useradd -g oinstall -G dba oracle   
passwd oracle   
9)         创建软件安装目录。   
mkdir –p /home/oracle\_11/app/   
chown -R oracle.oinstall /home/oracle\_11/   
chmod -R 755 /home/oracle\_11/   
10)     编辑 /home/oracle/.bash\_profile。   
添加以下行：   
umask 022   
ORACLE\_BASE=/home/oracle\_11/app   
ORACLE\_HOME=$ORACLE\_BASE/oracle/product/11.2.0/db\_1   
ORACLE\_SID=MYTEST --实例名，根据需要进行修改   
PATH=$PATH:$HOME/bin:$ORACLE\_HOME/bin   
LD\_LIBRARY\_PATH=$ORACLE\_HOME/lib:/usr/lib   
LANG=C                      --设定此参数避免安装过程出现乱码   
export ORACLE\_BASE ORACLE\_HOME ORACLE\_SID PATH LD\_LIBRARY\_PATH LANG   
11)    在图形界面下（在本机操作或者通过VNC远程操作） 以root身份打开另一个终端窗口   
--这步至关重要，不执行这步在以oracle用户运行 # ./runInstaller时会启动不了图型安装界面   
#xhost +   
#xhost + localhost   
12)   将下载的Oracle安装文件解压到Linux中，然后安装数据库   
#su - oracle   
cd database/   
./runInstaller

13)接下来会出现图形化的安装向导，按照向导一步一步的操作，和Windows下安装Oracle的向导一样，没有太大区别，直到最后安装完成。

**2.确认安装成功**

Linux下的Oracle在安装结束后是处于运行状态的。运行top –u oracle可以看到以Oracle用户运行的进程。在图形化界面下，运行$ORACLE\_HOME/sqldeveloper/sqldeveloper.sh可以出现Oracle自带的免费Oracle管理客户端SQL Developer。试着连接刚安装的Oracle，连接成功。

这时需要重启服务器，看服务器重启后，Oracle是否可以正常启动。以Root用户运行reboot命令可以重启Linux，重启完成后，默认情况下Oracle并不像Windows中的Oracle那样注册成服务，随机器一起启动的，所以需要手动敲入命令来启动。启动方式：以oracle用户登录运行以下语句：

1，进入sqlplus：

sqlplus /nolog

2，以sysdba的身份连接到数据库，并启动Oracle数据库引擎：

SQL> conn /as sysdba

SQL> startup

3，退出sqlplus，运行Listener

SQL> exit

$ lsnrctl start

这样数据库的TNS也启动了，可以通过网络连接数据库了。一般情况下就启动这两个就够了，如果想用Oracle提供的EM来管理Oracle的话还需要启动EM控制台，运行如下命令：

$ emctl start dbconsole

这样就可以通过<https://servername:1158/em/console>来访问EM控制台了。

这里我在实际操作中运行sqlplus出现了错误，大致如下：

sqlplus / as sysdba

sqlplus: error while loading shared libraries: /u01/app/oracle/product/11.1.0/db\_1/lib/libnnz11.so: cannot restore segment prot after reloc: Permission denied

网上也有很多人遇到了这个问题，原来是SELinux在作怪，我采用的解决办法是禁止掉SELinux：

更改/etc/sysconfig/selinux 文件的内容为 SELINUX=disabled，然后重启服务器。

好像这个办法不是很好，也有说的解决办法是使用chcon 命令

示例: chcon -t texrel\_shlib\_t /usr/local/rsi/idl\_6.1/bin/bin.linux.x86/\*.so

另外我在启动Listener时也报错TNS-12555: TNS:permission denied

后来网上搜了下，发现是oracle用户就无法进入/var/tmp/.oracle文件夹，方法很简单，执行：chmod 777 /var/tmp/.oracle 之后就ok了.

还有，我在启动EM控制台的时候又遇到了错误：Environment variable ORACLE\_UNQNAME not defined. Please set ORACLE\_UNQNAME to database unique name.

设置了这个变量后（运行export ORACLE\_UNQNAME=xxxx实例名）后，再启动EM控制台，又出现了新的错误：OC4J Configuration issue. /u01/app/oracle/product/11.2.0/db\_1/oc4j/j2ee/OC4J\_DBConsole\_xxxx not found.没有这个文件？去查了一下，确实是没有这样一个文件存在，怎么办呢？其实很简单，运行：

emca -config dbcontrol db

根据向导重新配置一下就可以了。

**3.配置Oracle为开机自启动**

虽然说Oracle作为服务器不会经常关机，但是偶尔重启一下，每次都要去手动启动Oracle还是太麻烦了，希望的是能够像在Windows中一样，每次开机的时候自动启动Oracle，只需要将Oracle配置为Linux的服务就可以了。操作如下：

1,以root用户建立/etc/rc.d/init.d/oracle11g脚本文件，内容如下（环境变量设置根据实际情况进行修改）：

#!/bin/bash   
#   
# chkconfig: 35 95 1   
# description: init script to start/stop oracle database 11g, TNS listener, EM   
#   
#   
#   
# match these values to your environment:

ORACLE\_BASE=/home/oracle\_11/app   
ORACLE\_HOME=$ORACLE\_BASE/oracle/product/11.2.0/db\_1   
ORACLE\_SID=BRDWDEV   
ORACLE\_UNQNAME=BRDWDEV   
PATH=$PATH:$HOME/bin:$ORACLE\_HOME/bin   
LD\_LIBRARY\_PATH=$ORACLE\_HOME/lib:/usr/lib   
LANG=C

export ORACLE\_BASE ORACLE\_HOME ORACLE\_SID ORACLE\_UNQNAME PATH LD\_LIBRARY\_PATH LANG

# see how we are called:   
case $1 in   
start)   
su - oracle<<EOO   
lsnrctl start   
sqlplus /nolog<<EOS   
connect / as sysdba   
startup   
EOS   
emctl start dbconsole   
EOO   
;;

stop)   
su - oracle<<EOO   
lsnrctl stop   
sqlplus /nolog<<EOS   
connect / as sysdba   
shutdown immediate   
EOS   
emctl stop dbconsole   
EOO   
;;

\*)   
echo "Usage: $0 {start|stop}"   
;;   
esac

2，以root用户执行以下命令

chmod 755 /etc/rc.d/init.d/oracle11g

chkconfig --add oracle11g

3， 重启服务

service oracle11g stop

service oracle11g start

这样在开关机的时候oracle自动启动和停止。

另外，还有说，需要修改/etc/oratab这个文件，将最后的N改为Y，如下：

xxxxTEST:/home/oracle\_11/app/oracle/product/11.2.0/db\_1:Y

最后，我们要测试一下这个自启动是否真的有效，重启一下服务器吧，过几分钟后，可以看到客户端可以正常连接到Oracle了。Oracle环境配置成功！