Oracle 9i under RedHat Linux 8.x and 9.x – Simple Installation HOWTO

Table of Contents

Table of Contents

Oracle 9i under RedHat Linux 8.x and 9.x − Simple Installation HOWTO

Evgueni Tzvetanov

v1.1, 23 June 2003

This document describes an example of an Oracle 9i Standard Edition database installation under RedHat Linux 8.x and 9.x.

---

6. Clients

7. Final Words

---

1. Introduction

I assume that the reader is familiar with Linux file systems, commands and tools, utilities, shells and shell scripts.

1.1 Overview

This document intends to show a simplified installation sequence for Oracle 9i database on RedHat Linux 8.x and 9.x. It is meant only as an example and if one intends to follow it, he/she should have in mind their own version of Linux, Oracle distribution and specific parameters of their own PC system.

This document also limits the installation information to Linux only, and is not intended to be used for other UNIX brands. To install Oracle 9i in other operating environments, use the Oracle Installation Guide.

I am not a DBA, nor I intend to show high skills in Oracle Administration, so please, do not ask me DBA questions. I will leave this task to the experienced DBAs :)

1.2 Copyright

This document is Copyright 2003 by Evgueni Tzvetanov (Speedy). Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation with no Invariant Sections, no Front−Cover Texts, and no Back−Cover Texts.

For the full text of the license, please visit the GNU Free Documentation License.

1.3 Disclaimer

Use the information in this document at your own risk. I disavow any potential liability for the contents of this document. Use of the concepts, examples, and/or other content of this document is entirely at your own risk.

All copyrights are owned by their owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Naming of particular products or brands should not be seen as endorsements.

You are strongly recommended to take a backup of your system before major installation and backups at regular intervals.

1.4 Credits

This document was created, using knowledge, gained from different sources −− books and web sites.

- Oracle 9i Installation Guide
- Oracle 9i Database Administrator's Guide
- DBA's guide to databases on Linux
- Oracle 8i: A beginner's guide
- Oracle web site (http://www.oracle.com)

I want to express my sincere thanks to their authors!

I also want to thank the people who gave me hints and led this update to wider range of solutions.

- Steve Schladel, a colleague and DBA in UMDNJ, who gave me a hint about a variable which solved the blocking of the installation process under RedHat 9.x.
- Rene Bauer, who sent a detailed explanation on the problems encountered with JRE version differences, related to gcc 2.9.x vs. gcc 3.2 distributions, and Legato Single Server Version instalation.

Some of the experiences listed in the examples were gained with a lot of tests and experiments. Please feel free to experiment; that's the main key to finding a desired solution. Even you crush a system or two, it helps you in your creativity and you learn from your mistakes.

Feel free to send any comments or suggestions to

---

2. Getting started

2.1 Downloading Oracle 9i database

Obtaining a copy of the Oracle 9i database is easy. Just go to the Oracle web site, click on **Download** and follow the path based on the distribution. There are few pages you must read before you get to the download links. These pages contain copyright and agreement information, which I suggest you read carefully. All the other information is just statistics. If you are not a member, you have to register on the Oracle web site to be granted access to any distribution, which can be downloaded for free.

The server distributions are in 3 files, named something like:

- lnx_920_disk1.cpio.gz
- lnx_920_disk2.cpio.gz
- lnx_920_disk3.cpio.gz

Other versions might have different names, but the installation could be considered similar.

According to Oracle's installation instructions there are 2 sources, that can be used − CDs and Hard Disk Installation. I used the first one. Downloading the distribution files and creating the CDs (see next item) is a time consuming operation, so arm yourself with patience. This will also require a lot of hard disk space. The 3 distribution files are around 1.5 GB. You will need as much as 3 times this space if you intend to keep the archives, the directories and CD images on your hard disk until you are sure that the CDs are created correctly.

Assuming that you are in the directory where the distributions are downloaded, execute the following with *all* 3 distribution files from a shell:

```
gunzip lnx_920_disk1.cpio.gz
cpio −idmv < lnx_920_disk1.cpio
```

These commands will create 3 directories, called Disk1, Disk2 and Disk3.

2.2 Creating the CDs

Writing information to CD in Linux is covered completely in CD−Writing−HOWTO, so the following will be only specific steps for the Oracle distributions.

Creating the CD images

The first step is to create the CD images for the 3 CDs of the Oracle distribution. Just execute the following sequence for all 3 directories, changing the filenames and directory names for each command:

```
mkisofs −r −o ora9id1.iso Disk1/
```

This will create 3 ISO CD images, which will be used further in the installation process. The command may generate a lot of warnings for name substitutions. Don't worry. The images will contain the correct directory tree information. To check if this is true, mount the images to a directory by using the loopback device (your kernel should support loopback devices to perform this test) as described in CD−Writing−HOWTO:

```
mount −t iso9660 −o ro,loop=/dev/loop0 cd_image mount_dir
```

where cd_image is your resulting CD image from the **mkisofs** command. After mounting, you can change directory and check the mounted directory tree to ensure it looks okay. If everything is fine, you can move to the next step.

Burning the CDs

Burning the CD images on CDs can be done with any command line or GUI program that uses ISO CD images to create CDs. See CD−Writing−HOWTO for details. Make sure however, that after burning the CDs they contain exactly the same directory trees as directories Disk1, Disk2 and Disk3. This can be done by mounting the new CDs and using whatever means you find convenient to walk through the directory tree. The images can be written on any CD writer or computer, using any software that will support CD writing from ISO images.

The **mkisofs** command may generate empty directories in the root directory of the CD image itself. These are not a problem for the installation contents. Just disregard them.

---

3. Pre−installation

There are few steps to be performed before the installation process. They are divided into the following categories:

3.1 System Requirements

Hardware requirements

Oracle 9i is a tough bit, you better have a good machine. Here is what the Installation Guide says (summary):

```
Parameter       Required        Sufficient (tested)
---------------------------------------------------
CPU             PII or K6       K6/2 *(1)
RAM             512 MB          256 MB
HDD space       3.5 GB          2.5 GB *(2)
Swap space      1GB or RAMx2    512 MB or more
```

(1) Dual CPUs are recommended for large databases. Usually a high performance CPU couple (Xeon or Athlon MP) or even quad system is considered a good choice.

(2) Using SCSI HDD with RAID or separate HDD for each database, if multiple database instances will be installed, is recommended.

Software requirements

Oracle requires a minimal version of the kernel and JRE software:

```
Parameter       Required        Sufficient (tested)
---------------------------------------------------
Kernel Version  2.4.7           later is fine
JRE             1.1.8v3         1.1.8v3
```

Since (and including) version 8i, the installation processes are GUI−based and built with Java, so a main requirement is X server and a Window Manager. At least the JRE version 1.1.8v3 should be installed. This installation was done with version 1.1.8v3 of JRE. I can not guarantee that other versions can do the job. Test it at your own risk.

These requirements are valid if you decide to install the Enterprise Server. For the Standard Edition the sufficient values are fine. Even in some cases these parameters could be lower.

De−facto installation environment

Here is the installation environment I used for this example:

```
CPU             K6 III /400
RAM             512 MB
HDD space       2.3 GB
Swap space      512 MB
CD ROM          24x
Kernel version  2.4.18−24.8.0
JRE             1.1.8v3
```

It seems that this environment is enough for testing with small database sizes, but as recommended before, if you need to use a large or very large database, choose better hardware.

3.2 Environment variables and profiles

The installation requires a few environment variables to be set before it starts. These variables can be set into one's profile directly by putting the lines in the **.bash_profile** of every account, which will be used to access the database, or in a more generic approach –– in the file **/etc/profile** (as this installation). Here is a simple list and example values for the variables (current example):

```
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/9.2.0.1.0
ORACLE_SID=ORTD

PATH=$PATH:$ORACLE_HOME/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib:$ORACLE_HOME/network/lib

export PATH ORACLE_BASE ORACLE_HOME ORACLE_SID LD_LIBRARY_PATH
```

Everything so far is valid for both RedHat 8.x and 9.x, but it seems that Oracle has thought of another variable, which must be set in RedHat 9. It must be included with the previous ones and exported accordingly. It is necessary to set it if you don't want to be stuck into an infinite loop during installation:

```
LD_ASSUME_KERNEL=2.4.1
```

Thanks to Steve Schladel, a colleague and a DBA, the installation continued smoothly without a problem.

The ORACLE_SID variable I set by using the abbreviation of ORacle Test Database. It is up to you what you will set as SID. This will be your default database name and part of the global database name.

For more information on the meanings of the variables, please refer to the Oracle Administrator's Manual or Reference.

3.3 JRE

The JRE (Java Runtime Environment) can be downloaded from Blackdown website. Pick 1.1.8v3 if you don't know what to download and unpack the tarball in the /usr/local directory. This will create a directory, named after the version of the JRE. Make the following link in /usr/local/:

```
ln -s jre118_v3 jre
```

for the JRE 1.1.8v3 distribution. If a newer version is used, change the command correspondingly. Now add to the PATH variable the bin directory of your JRE:

```
PATH=$PATH:/usr/local/jre/bin
export $PATH
```

or mix the lines here with those in the profile as you find efficient.

3.4 Setup tasks to perform as root user

Before starting the installation, the Linux environment and directory structure should be prepared. There are few steps and they are listed below with a description of each step. Remember, these steps are the minimum required steps for installation of the Standard Edition. If one intends to install Enterprise and additional services, I suggest a thorough study of the Oracle 9i Installation Guide.

Log in as root and follow the list.

Kernel Tune−up

If the following section is not quite clear, please refer to IPC programmers reference for Linux.

>From version 2.2.1 on, the kernel has dynamically set parameters through the /proc file system and does not need to be re−compiled. For this purpose the /proc file system can be used with great flexibility. Here are the steps needed to update the kernel and set up scripts to run on boot, so that the parameters are set every time the system is booted.

Change current directory to /proc/sys/kernel.

Listed below, are the minimum required sets to be updated (if necessary).

Read and set the semaphore parameters

The values of the semaphore parameters SEMMSL, SEMMNS, SEMOPM, SEMMNI can be read with the following command:

```
cat sem
```

The output should be similar to:

```
250 32000 32 128
```

which is the list of values of the parameters in the above list order. Now set the values with the following command:

```
echo 100 32000 100 100 > sem
```

Read and set the shared memory parameters

Shared memory parameters SHMMAX, SHMMIN, SHMMNI, SHMSEG can be read by using the following command:

```
cat shm_param
```

where shm_param is one of shmmax, shmmin, shmmni, shmall, etc. Now use the following commands to set values for the necessary parameters:

```
echo 2147483648 > shmmax
echo 4096 > shmmni
echo 2097152 > shmall
```

The above set of commands can be combined in a shell script and can be run on boot in the run level, to decide which Oracle database will be used, but keep in mind these parameters should be set *before* the database instance is started up. Please refer to the Linux Administrator's guide for more information on run−levels and startup init scripts.

File handles

Set the file handles using the following commands:

```
echo 65536 > /proc/sys/fs/file-max
ulimit -n 65536
```

Sockets

Set the sockets using the following command:

```
echo 1024 65000 > /proc/sys/net/ipv4/ip_local_port_range
```

Process Limit

Set the process limit using the following command:

```
ulimit -u 16384
```

Groups and accounts

As with all Oracle databases, 9i requires an account for the owner, traditionally named **oracle**. The requirements for the groups are a bit unusual; there must be a group for OSDBA, which by default is named **dba**. Oracle requires also second group (usually called **oinstall**) to assign authority to install software. Another group for operators should also be present (but is optional), used to grant operator access to certain groups of users. So here is the sequence of commands to create the groups:

```
groupadd dba
groupadd oinstall
groupadd oper
```

After the groups are created, we can add an account for the Oracle owner:

```
useradd -c DBA -g oinstall -G dba oracle
```

which will create the **oracle** account, with primary group **oinstall** and secondary – **dba**.

In addition, users who will use this database instance can be added to the dba group in the /etc/group file.

Mount points and permissions

Oracle 9i installation requires a minimum of two mount points –– one for the software and one for the database files. If OFA (Optimal Flexible architecture) is used, then the minimum mount points are four –– one for the software and three (or more) for the database files.

Oracle documentation recommends each mount point to be on a separate physical disk, but for a minimal use as in this case, the mount point could be directories. The names of the mount points are traditionally /u01, /u02, /u03, etc. They can be created as subdirectories anywhere on the system and linked with root directory links. In my case I created the directories (only 2 of them – /u01 and /u02) under the root directory.

Here is a short list of commands to create and prepare the mount point (as root):

```
mkdir /u01 /u02
chown oracle.dba /u01 /u02
chmod 755 /u01 /u02
```

The last two commands set the ownership of the installation directories to the user authorized to install Oracle 9i, but after the installation by administrator's choice the topmost directories can be switched back to root ownership.

Permissions for file creation

In most cases the umask should be set to 0022. When installing Oracle databases it is recommended that the umask is checked and set to 0022 if not set properly. This will ensure that all the files during the installation are created with proper attributes. Use the following commands:

```
umask
```

to test the mask, and:

```
umask 0022
```

to set the mask. This command can also be set on the profile in case it isn't the default value of 0022.

3.5 Setup tasks to perform as oracle user

Before starting the installation log in as user **oracle** in an X terminal and make sure that the DISPLAY and PATH variables contain the proper values. The DISPLAY variable should specify the name, server number and the screen number of the system, and where the Oracle Universal Installer will be run. This step is only necessary if you are using remote X server. In case of local installation, if you can run X Window Manager you will be able to run the installer without a problem. If there is a problem running the installer because of the DISPLAY variable, execute:

```
xhost +localhost
export DISPLAY=localhost:0.0
```

This should do the trick.

In addition as **oracle** user execute:

```
env
```

and check for all the previously mentioned environment variables. If there is a suspicion that one or more are not set properly, they should be corrected and the environment reloaded.

---

4. Installation

4.1 Mounting the CDs (before or during the installation)

Usually RedHat Linux has automount, which mounts the CD after the CD−ROM door is closed, but in case this has to be done manually, here it is briefly.

Invoke a terminal window and become superuser. Leave the terminal open during the course of the installation. It will become handy in time.

Place the first CD in your CD−ROM device. Check if you have a directory to mount the CDs in. Usually it should be /mnt/cdrom, but if this is not your favorite, just name it as you wish. Now execute the following command (assuming that the mount directory is /mnt/cdrom):

```
mount /dev/cdrom /mnt/cdrom
```

This will mount your CD onto the /mnt/cdrom directory. This will be your source installation directory.

During the installation, Oracle Universal Installer will ask for a CD change, so if the CD can't be unmounted automatically, go to your root terminal and do the following:

```
cd /mnt
eject
```

This will unmount the CD and open your CD−ROM. Now place the next CD and close the CD−ROM. Use the mount command again if the CD is not mounted automatically and switch to your installation screen to continue with the process.

4.2 Running the installer

After mounting the first disk, invoke a new terminal window and change directory to /mnt/cdrom. This is your installation mount point. From here run:

```
./runInstaller &
```

Additionally the file index.html from the same directory can be opened in a browser and from there you can read and follow Oracle documentation during the installation procedure.

4.3 Installation screens and steps

Oracle Universal Installer 2.2 has a simple and intuitive interface. You may choose to skip this section of the Installation item and follow your instincts, which might be a very good decision. The screens are self−explanatory and very nicely ordered in a good, logical queue.

In case you want to verify where you are in the installation, just compare your process with the following descriptions of screens and steps.

Welcome Screen

>From the welcome screen you may select to install or uninstall products. If this is the first installation, click Next to continue.

A new dialog window (Inventory Location) appears. It prompts you for an inventory directory for the installation(s) now and in the future. Click OK if you agree, otherwise enter the desired inventory directory.

UNIX Group Name Screen

This screen asks for your Oracle Installation Group, which by default is **ointsall**. If you wish to change this name or use another (special) group for further installation, now is the time to tweak the groups. Otherwise enter ointsall and click Next. A dialog appears, showing that you have to use your root terminal (remember when we opened it before). Go to the root terminal and follow the instructions on the screen, which are simply to execute the script in /tmp/ called orainstRoot.sh

If successfully run, you should see on the screen:

```
Creating Oracle Inventory pointer file (/etc/oraInst.loc)
Changing groupname of /u01/app/oracle/oraInventory to
oinstall.
```

Now switch back to the dialog and click Continue.

File Locations Screen

This screen shows the source and destination file locations for the installation process. If your environment variables are okay you should not change anything, but if you wish to do so, do it carefully and know what you are doing! You can simply click Next.

What follows is a long reading of the disk and a progress bar on the top right of this screen. Just be patient.

Available Products Screen

This screen shows the available products in this installation set. Since we are installing only the database, the first option (which is the default) is the one that is interesting to us. It actually includes some of the products listed in the next sections. Click Next.

Installation Types Screen

Since our example shows Standard Edition Installation, click Standard Edition. You may select Enterprise, but the path from this point on might be different and you might need to select other options. Anyway, it is the choice of the person who is installing.

Click Next. Another pause (not long) and then you will go to the next step.

Database Configuration Screen

This screen allows the operator to select what purpose the database will serve, so that the installation will be able to adjust parameters accordingly. I've chosen General Purpose and clicked Next. There is another wait period (short also).

Database Identification Screen

Here the operator is prompted to enter a global database name. The text on this screen is quite explanatory. For our example I entered:

```
ORTD.zeus
```

since my database server's hostname is zeus. Use your hostname in its place and click Next.

Database File Location Screen

This screen prompts you to enter the location where the physical database files will be situated. It also recommends you put the data files on a separate disk (directory). That's why I entered:

```
/u02/oradata
```

instead of the default value. Click Next when done.

Database Character Set Screen

This screen allows you to choose different character sets for your database. I have left it as it is: Use the Default Character Set. Click Next.

Summary Screen

If you see this screen, your preparation for the installation process with the OUI is done. Now you may browse the contents of the installation tree and see what you have chosen. This might make you go back and fix stuff, if you don't agree with the list. Otherwise click Install, and the installation process will start.

Install Screen

This is a very long−lasting, boring screen, which requires CD changing from time to time. It displays a progress bar and counts the percentage to display the overall progress by items.

The installation process is very long. First the software components are installed and then everything is linked.

Just find something to kill some time with and from time to time peek at the screen to see if the CD needs changing.

IMPORTANT! In my case (I did the installation twice) there was an Error screen around 79% in the linkage process. The error dialog showed the following information:

```
Error in invoking target install of makefile
/u01/app/oracle/product/9.2.0.1.0/ctx/lib/ins_ctx.mk
```

I clicked Ignore and in both installations it seemed there was no problem to finish the linkage.

A dialog is popped up when the linkage is finished. It prompts for some terminal commands that have to be executed before the installation process completes. Basically, there is a script called **root.sh** in the $ORACLE_HOME directory, which should be executed as root, from your root terminal. The execution itself takes some time and a lot of HDD activity, so be patient and don't worry.

Configuration Tools Screen

The configuration tools screen tries to execute 3 tasks and reports failures for all of them. Here is how I solved the problem. In the $ORACLE_HOME directory there is a link, named JRE, which links to the Oracle JRE directory. Delete this link and create one to your own JRE distribution, which was described earlier. Here is the list of commands:

```
rm -f JRE
ln -s /usr/local/jre/ JRE
```

After executing these commands, try selecting one by one the items in the Tool Name column of the table and clicking Retry. If this does not work, don't worry. These are optional in the installation session and you can do them later.

Troubleshooting

If the last (optional) tasks do not execute, I have a special set of instructions on how to do at least two of them. But first, I want to stress on having Oracle documentation handy, and that you try to read between the lines sometimes when you troubleshoot your installation. If the conditions are right, you should not get any serious errors in your session, but in case you get one, check all the initial conditions, file and directory permission,s and disk space.

You can easily delete everything and start from scratch if you just purge the contents of the Oracle mount point directories (/u01, /u02, etc.), and delete the file **oratab** in the /etc directory. There is one more set of files you may need to delete so that there is no trace of Oracle on your system. They are in the /usr/local/bin directory and the names are **oraenv**, **coraenv** and **dbhome**. However you may leave them there and on your next installation the installer will ask you if you want to keep them.

Now the optional tasks:

Net Configuration Assistant

To run this, go to the $ORACLE_HOME/bin directory and from there just execute:

```
./netca &
```

This application will setup one of four tasks. Listener is what you really want to set. The others are not as important, but try setting them as well. Before setting the listener, you might have a listener already running. Check this by executing the following command:

```
ps afx | grep LISTEN | grep -v grep
```

If you get a line similar to:

```
15922 ? S 0:00 /u01/app/oracle/product/9.2.0.1.0/bin/tnslsnr
LISTENER
```

you've got a listener running. Stop the listener with the following command from the $ORACLE_HOME/bin directory:

```
./lsnrctl stop
```

You might need to also delete the existing listener and create a new one. This depends on you completely. Just follow the natural flow of the application. At the final round the new listener will be started if created a new one of course.

Database Configuration Assistant

To run this, go to $ORACLE_HOME/bin and execute:

```
./dbca &
```

This application is probably the most important in the set. It helps you create and configure your database. If a database was not created during the installation session or a restart of the applications, as mentioned before, this is the one you would want to run to create your database.

While you have not used the database yet, you might want to delete and recreate it. You may do it from here. This document can not include all the functionality of the application, but since it is a GUI, it is fairly easy to perform tasks with it.

Personally I prefer to create it from here, so that I have control of all the parameters needed, especially the directories where the data files will be created. Choose Typical when prompted for the database parameters. The next screen, with the tree−like structure allows you to set the location of your database files. Instead of {ORACLE_BASE}, you can put /u02 for example. This will put the database files to be used in your /u02/oradata/SID directory. This is just an example. It is up to you to decide how to spread your database(s) directory tree.

At the end of the database creation you will be prompted for passwords for the users SYS and SYSTEM. Enter them (with confirmations), remember them, and exit (this works as OK if you haven't guessed it already). That should be it. Now execute:

```
ps afx
```

and the screen should contain lines similar to the following:

```
PID     TTY      STAT   TIME COMMAND
    1   ?        S      0:04 init
    2   ?        SW     0:01 [keventd]

..... Lots of other lines we are not interested in...

11665   ?        S      0:00 ora_pmon_ORTD
11667   ?        S      0:00 ora_dbw0_ORTD
11671   ?        S      0:00 ora_lgwr_ORTD
11673   ?        S      0:00 ora_ckpt_ORTD
11675   ?        S      0:00 ora_smon_ORTD
11677   ?        S      0:00 ora_reco_ORTD
11679   ?        S      0:00 ora_cjq0_ORTD
11681   ?        S      0:02 ora_qmn0_ORTD
```

```
11683   ?        S      0:00 ora_s000_ORTD
11685   ?        S      0:00 ora_d000_ORTD
11812   pts/1    S      0:00 /u01/app/oracle/product/9.2.0.1.0/bin/tnslsnr LISTENER -inherit
```

If you see this picture, this is your database instance running. The last line in the above example is the listener (it might be somewhere else on your screen).

Congratulations! You've successfully finished your installation!

Troubleshooting updates

The following is an update, courtesy of Rene Bauer. You may encounter problems with the following items:

- JRE version compatibility
- Missing database configuration file
- Legato Single Server Version installation failure

The JRE compatibility problem was not reported only by Rene, but I've got few more emails about it. The problem exists because RedHat 8 and 9 come with gcc version 3.2.x, but some older versions of JRE are compiled with gcc 2.9.x. The solution is to download the latest JRE version. At the time of this update the version is 1.4.1, and since there is no jre executable, run the following command to create the proper sym link:

```
ln -s $JRE_DIR/bin/java $JRE_DIR/bin/jre
```

where $JRE_BIN is your jre bin directory.

The second update is in regard to a missing configuration file after database creation. The file init{ORACLE_SID}.ora should be created by default in your $ORACLE_HOME/dbs/ directory.

Rene's solution is: The installer creates a file called spfile{ORACLE_SID}.ora, so a sym link can be created to mimic the missing file. Change your working directory to be $ORACLE_HOME/dbs/ and execute:

```
ln -s spfile{ORACLE_SID}.ora init{ORACLE_SID}.ora
```

My solution: There is always a file after the installation with database creation called init.ora. I simply copied this file to init{ORACLE_SID}.ora. Change your working directory to be $ORACLE_HOME/dbs/ and execute:

```
cp init.ora initORTD.ora
```

The third problem is in regard to Oracle's installation process, installing Legato Single Server Version. I did not have this sort of problem, but here is Rene's version, which I'll try to make as short as possible: The last root.sh file, executed on Oracle installation's request had caused an error while attempting to install LSSV. According to Rene, RedHat 8 and 9 come with ncurses5, but by default the ncurses4 compatibility package is not installed, so after it is installed, the LSSV installation problem disappears.

These may be not the last problems to be solved, so I encourage any comments and additions, which I can add to the Troubleshooting section. I'd appreciate any feedback, which can improve quality of this document.

5. Post−installation

The Oracle Installation Guide describes a very long list of procedures to be done after the installation process. Some of them are necessary, some of them probably in your instance you will never need, but most importantly this document will not repeat them.

Please refer to the Installation Guide for this procedures. Here I will describe steps that are very helpful for the minimal installation that I used, and seem enough for the basic database operations.

5.1 Editing the oratab file

There is one step in particular that is important, so that you may start and shutdown the database from the command prompt any time, or to allow the database to be autostarted during boot. Go to the /etc directory and edit the file **oratab**. In the line:

`ORTD:/u01/app/oracle/product/9.2.0.1.0:N`

set the last value to Y and it should read:

`ORTD:/u01/app/oracle/product/9.2.0.1.0:Y`

Now save the file and go back to your Oracle home directory.

5.2 Script for automatic startup on boot

It is generally a good idea to automate the database startup at boot time on your database server. In this case you will need a script, which can be included in the run−level sets. Here is an example only. Please, tune it to your own machine and run−levels!

```
#!/bin/bash
#
# Run-level Startup script for the Oracle Instance and Listener
#
# chkconfig: 345 91 19
# description: Startup/Shutdown Oracle listener and instance

ORA_HOME="/u01/app/oracle/product/9.2.0.1.0"
ORA_OWNR="oracle"

# if the executables do not exist -- display error

if [ ! -f $ORA_HOME/bin/dbstart -o ! -d $ORA_HOME ]
then
        echo "Oracle startup: cannot start"
        exit 1
fi

# depending on parameter -- startup, shutdown, restart
# of the instance and listener or usage display

case "$1" in
    start)
        # Oracle listener and instance startup
```

```
        echo -n "Starting Oracle: "
        su - $ORA_OWNR -c "$ORA_HOME/bin/lsnrctl start"
        su - $ORA_OWNR -c $ORA_HOME/bin/dbstart
        touch /var/lock/subsys/oracle
        echo "OK"
        ;;
    stop)
        # Oracle listener and instance shutdown
        echo -n "Shutdown Oracle: "
        su - $ORA_OWNR -c "$ORA_HOME/bin/lsnrctl stop"
        su - $ORA_OWNR -c $ORA_HOME/bin/dbshut
        rm -f /var/lock/subsys/oracle
        echo "OK"
        ;;
    reload|restart)
        $0 stop
        $0 start
        ;;
    *)
        echo "Usage: $0 start|stop|restart|reload"
        exit 1
esac
exit 0
```

You can simply copy and paste this file into your system and change it according to your settings.

Once you copy and edit the file, move it to the /etc/rc.d/init.d directory and change it's ownership and the attributes to be the same as the other files in the directory.

To do that and the following operations, you will need to be logged as root user.

The fifth line of this code reads:

```
# chkconfig: 345 91 19
```

It is used by the chkconfig command when setting the scripts in the run−levels. The value 345 lists the run−levels in which we allow Oracle to run, so if you want to be able to run an Oracle instances only in levels 3 and 5, then change this value to 35. The next value (91) describes the order number in the startup sequence. This means that if there are 100 startup processes, the Oracle startup will be the 91st to run. The next number is the shutdown number, meaning that when the system is shutdown and there are for example 100 processes to be shutdown, the Oracle process will be 19 in order. These numbers are used to set the relative position of the Oracle startup and shutdown processes. They are set here as an example. You can change them, but keep in mind that Oracle instancse can not be run without some processes in the system running beforehand.

Now execute the following command:

```
chkconfig -add oracle
```

This will add the corresponding links in the run−level directories as listed in the beginning of the script.

To test the newly created automation, you have to reboot your system, but remember that before this automation runs, the kernel parameters should be set. In fact you may include the kernel tune commands in your **start)** section of the above script.

5.3 Testing your installation with SQL*Plus

If you have installed a typical database, there should be a user created called **scott**. The password is **tiger**. Just execute the following:

```
sqlplus scott/tiger
```

which should run the SQL*Plus interpreter and log in as user scott. You should see the screen like the following:

```
SQL*Plus: Release 9.2.0.1.0 - Production on Fri Feb 21 10:55:45 2003

Copyright (c) 1982, 2002, Oracle Corporation.  All rights reserved.

Connected to:
Oracle9i Release 9.2.0.1.0 - Production
JServer Release 9.2.0.1.0 - Production

SQL>quit
Disconnected from Oracle9i Release 9.2.0.1.0 - Production
JServer Release 9.2.0.1.0 - Production
```

Before exiting you can do some SQL on your own to just see the tables in this database schema or whatever you have in mind.

>From this point on it is a matter of your own creativity and purpose.

---

6. Clients

The Client Software for the Oracle 9i database can be run on any platform, so you may download the client for the platform you desire (whichever is available of course). I personally like to have client software on both DB server and on a MS Windows workstation. This makes me experience multi−platform database control, or so I think. :)

Download and prepare the CD with the client on the same platform you will run it from. It is not necessary, but it is recommended you have a CD with it. Although you may run the installation process from your HDD where you downloaded the client, I recommend you spend few pennies on a CD burner. It always becomes handy.

The client is also using the Oracle Universal Installer, so this might seem pretty easy after all the previous hassle.

When you first run the Enterprise Management Console, it will lead you to connect to the database. Provide your full database and your SID names and try to connect as user SYS with SYSDBA privileges. At some point you have to enter the IP address of the database server and chose a port the client should connect to. I left the default values (as with the network setup on the server as well).

If you don't see any error messages, that's it. You have done it.

Enjoy your server and client playgrounds!

7. Final Words

Oracle 9i is a huge complex with very useful, and most of all very secure applications. This document slightly touches the surface and is not intended to be a comprehensive guide. I hope that it can help solving some problems during the initial installation, but it is important to analyze your own situation and turn to more reliable sources.

Some of the operations could be executed manually using SQL*Plus. This could be a more flexible way to tune and set up the installation. I will leave this in more experience hands.

I haven't mentioned so many things, that are described in details in Oracle documentation itself. So explore the genuine source in depth!

Good luck!