Oracle Database HOWTO

Table of Contents

Table of Contents

Oracle Database HOWTO

Paul Haigh, `paul@nailed.demon.co.uk`

v1.2, 4 August 1998

A guide to installing and configuring the Oracle Database Server on a Linux system

---

1.Introduction

2.Installing the Oracle Software

3.Creating a Database

4.Configuring SQL*Net on the Server

5.Client Configuration

6.Automatic Startup and Shutdown

7.Other Bits

8.Troubleshooting

9.Credits

1.Introduction

1.1 Version History

- `v0.1 - 21 Feb 1998 - Paul Haigh - Original Version.`
- `v0.2 - 01 Mar 1998 - Paul Haigh - Comments From Proofreaders Added.`
- `v1.0 - 10 Mar 1998 - Paul Haigh - First Release to LDP.`
- `v1.1 - 20 Jun 1998 - Paul Haigh - Added troubleshooting section & general tidyup.`
- `v1.2 - 04 Aug 1998 - Paul Haigh - Added Oracle Corp News & Removed Section on Future Enhancements.`

1.2 Copyright

1.3 Disclaimer

While I have tried to include the most correct and up to date information available to me, I cannot guarantee that usage of information in this document does not result in loss of data or equipment. I provide NO WARRANTY about the information in the HOWTO and I cannot be made liable for any consequences resulting from using the information in this HOWTO.

1.4 Aim of the HOWTO

In this HOWTO I will attempt to cover installation and basic admin of an Oracle database running on a Linux machine. In particular I will cover Oracle server installation, SQL*Net configuration and client configuration.

This document is not an in depth tutorial on using or administering an Oracle database, if that is what you are looking for there are great books on those subjects published by O'Reilly and others.

I am also not going to cover the development of Oracle programs under UNIX. If this is absolutley necessary to you then I would recommend that you purchase the SCO development system (with OpenServer 5.x) from SCO, which I am told can be obtained for a very reasonable US $19, from `www.sco.com`.

1.5 Requirements

I am assuming a number of items that you will need for following the HOWTO.

- **Oracle Server CD for SCO Openserver (Version 7.3.3.0.0.)**

  This **must** be a legal copy. Remember that Oracle are a profit making company and charge for their products. If you want a free SQL compliant database use PostgresSQL or similar.

  It is also possible to install oracle, using a 60 day evaluation licence, from a downloadable `tar` file from the Oracle web site. I have not personally tried this and it is completely unverified.
- **A Linux Server**

  You wouldn't be reading this without one...would you?
- **Kernel 2.0.30+**

  I cannot guarantee that these instructions will be accurate for any other Kernel. (Not that I am guaranteeing it for 2.0.30 either...).
- **iBCS**

  It is very important to have this installed and working with the latest possible version for your platform. (I am using iBCS−2.0−10.i386.rpm from Redhat Linux).
- **Lots of disc space**

  600 Mb+ is a reasonable amount. It is possible to install with less but you need to make some sacrifices, and I never like starting with those. However, I will attempt to point out areas in which space can be freed up.
- **32Mb+ Ram**

  I know that this sounds like a lot, especially in Linux terms, but remember that Oracle is a complex piece of software. You wouldn't have the same reservations on SCO!

  I am not saying that Oracle doesn't work with less, just that it is less than Oracle recommend and I wouldn't suggest it.
- **Licenses from Oracle**

  I know that I have already mentioned this but I want to be clear that this is important. Using software from Oracle without a license is illegal.

1.6 News From Oracle Corporation

Oracle have bowed into pressure from the Linux community. Oracle Corporation have decided to **officially** support Oracle 8 on the Linux (i386) platform. This should be released in December 1998, according to the Oracle website.

Better still Oracle will also be porting `Oracle Applications` to the linux platform. This should be available in the first half of 1999, according to the Oracle website.

References:

- `http://www.oracle.com/html/linux.html`
- `http://www.news.com/News/Item/0,4,24436,00.html`
- `http://www.zdnet.com/pcweek/news/0720/20morac.html`

2.Installing the Oracle Software

2.1 Server Preparation

Creating an Oracle User

Unsurprisingly we require a user to hold the Oracle database. Since we have no desire to relink the Oracle kernel (more about that later) we have to accept the Oracle defaults for user name and group name. This includes the user ORACLE and the group DBA.

1. Login as Root
2. Create the oracle user and the group dba.

```
$ groupadd dba
$ useradd  oracle
```

3. Ensure a home directory is created for the user oracle.

```
$ mkdir /home/oracle
$ mkdir /home/oracle/7.3.3.0.0 (Version of Oracle)
$ chown -R oracle.dba /home/oracle
```

2.2 Installing from CDROM

Unfortunately the Oracle Installer on the SCO disc will not work. A variety of problems can be experienced, from core dumps to hangs. As a result we need to copy the files from the CDROM manually and uncompress them:

(Ensure the CDROM is mounted on the system).

1. Log on as Oracle
2. Change directory to `/home/oracle/7.3.3.0.0.`
3. Copy all install files from CDROM

```
$ cp −a /mnt/cdrom/* .
```

4. Un−compress all Oracle files on CDROM.

```
$ find . −name *_ −exec ~/7.3.3.0.0/orainst/oiuncomp {} \;
```

2.3 Post Installation Tasks

Tasks for Root

Add the following lines to `/etc/profile` or add to the `.profile` for each user who is going to use Oracle.

```
# Oracle Specific
ORACLE_HOME=/home/oracle/7.3.3.0.0
ORACLE_SID=orcl
ORACLE_TERM=vt100
export ORACLE_HOME ORACLE_SID ORACLE_TERM

# Alter path for Oracle
PATH="$PATH:$ORACLE_HOME/bin"
```

We also need to change the owner and permissions of the Oracle ulimit increase utility.

```
$ chown root.root $ORACLE_HOME/bin/osh
```

```
$ chmod u+s $ORACLE_HOME/bin/osh
```

Tasks for Oracle

Change permissions for the Oracle files to ensure correct operation.

```
$ chmod +x $ORACLE_HOME/bin/*
$ chmod u+s $ORACLE_HOME/bin/oracle
```

Oracle tools require the messages to be in the `$ORACLE_HOME/tool_name/mesg` directory. So, move the `msb` files from the `msg_ship` directories to the `mesg` directories.

```
$ mv $ORACLE_HOME/plsql/mesg/mesg_ship/* $ORACLE_HOME/plsql/mesg/.
$ mv $ORACLE_HOME/rdbms/mesg/mesg_ship/* $ORACLE_HOME/rdbms/mesg/.
$ mv $ORACLE_HOME/svrmgr/mesg/mesg_ship/* $ORACLE_HOME/svrmgr/mesg/.
```

Create the following directories if they do not exist:

```
$ mkdir $ORACLE_HOME/rdbms/log
$ mkdir $ORACLE_HOME/rdbms/audit
$ mkdir $ORACLE_HOME/network/log
```

Things you can remove

The following directories can safely be removed:

- `$ORACLE_HOME/guicommon2/`
- `$ORACLE_HOME/ctx/`
- `$ORACLE_HOME/md/`
- `$ORACLE_HOME/mlx/`
- `$ORACLE_HOME/precomp/`
- `$ORACLE_HOME/slax/`

3.Creating a Database

Now the Oracle server is installed we need to create a database to test the installation.

```
If you are using Oracle 7.2.x or earlier, please read the
troubleshooting section below.
```

3.1 Create the Initialisation File

Copy the `$ORACLE_HOME/dbs/init.ora` to `$ORACLE_HOME/dbs/initorcl.ora`:

```
$ cd $ORACLE_HOME/dbs
$ cp init.ora initorcl.ora
```

Modify it by adding the following lines:

```
db_name = orcl
COMPATIBLE=7.3.3.0.0
```

3.2 Creating the Database Install Script

Create a script file called makedb.sql in the `$ORACLE_HOME/dbs` directory:

```
connect internal
startup nomount
set echo on
spool makedb.log
create database orcl
        maxinstances 1
        maxlogfiles  8
        datafile '$ORACLE_HOME/dbs/orcl_syst_01.dbf' size 40M reuse
        logfile
                '$ORACLE_HOME/dbs/orcl_redo_01.dbf' size 1M reuse,
                '$ORACLE_HOME/dbs/orcl_redo_02.dbf' size 1M reuse,
                '$ORACLE_HOME/dbs/orcl_redo_03.dbf' size 1M reuse;
@$ORACLE_HOME/rdbms/admin/catalog.sql
create tablespace rollback
        datafile '$ORACLE_HOME/dbs/orcl_roll_01.dbf' size 8.5M reuse;
create tablespace temp
        datafile '$ORACLE_HOME/dbs/orcl_temp_01.dbf' size 5M reuse
        temporary;
create tablespace users
        datafile '$ORACLE_HOME/dbs/orcl_user_01.dbf' size 10M reuse;
create rollback segment r1 tablespace rollback
```

```
        storage ( optimal 5M );
alter rollback segment r1 online;
connect system/manager
@$ORACLE_HOME/rdbms/admin/catdbsyn.sql
connect internal
@$ORACLE_HOME/rdbms/admin/catproc.sql
connect system/manager
@$ORACLE_HOME/sqlplus/admin/pupbld.sql
spool off
exit
```

3.3 Running the Database Installation Script

Start `svrmgrl` and run the script:

```
$ cd $ORACLE_HOME/dbs
$ svrmgrl

Oracle Server Manager Release 2.3.3.0.0 - Production

Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.

Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production

SVRMGR> connect internal
Connected.
SVRMGR> startup nomount
ORACLE instance started.
Total System Global Area       4313312 bytes
Fixed Size                       41876 bytes
Variable Size                  4140364 bytes
Database Buffers                122880 bytes
Redo Buffers                      8192 bytes
SVRMGR> @makedb
<loads of messages>
SVRMGR> exit
Server Manager complete.
```

3.4 Starting the Database

Firstly, we need to bring up the database by hand (we will automate this later on). To startup an Oracle database we need to issue the `startup` command when connected internally:

```
$ svrmgrl

Oracle Server Manager Release 2.3.3.0.0 - Production
```

```
Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.

Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production

SVRMGR> connect internal
Connected.
SVRMGR> startup
ORACLE instance started.
Total System Global Area       4313316 bytes
Fixed Size                       41876 bytes
Variable Size                  4140368 bytes
Database Buffers                122880 bytes
Redo Buffers                      8192 bytes
Database mounted.
Database opened.
SVRMGR> exit
Server Manager complete.
```

3.5 Stopping the Database

It is worth mentioning here that restarting a Linux server without shutting down the Oracle database first there is a high risk of corrupting the database.

So, before we issue the Linux `shutdown` command it is wise to bring down the database:

```
$ svrmgrl

Oracle Server Manager Release 2.3.3.0.0 - Production

Copyright (c) Oracle Corporation 1994, 1995. All rights reserved.

Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production

SVRMGR> connect internal
Connected.
SVRMGR> shutdown
Database closed.
Database dismounted.
ORACLE instance shut down.
SVRMGR> exit
Server Manager complete.
```

3.6 Create a Default User

The database, as created, has a two special users which are automatically created. These are:

```
Username                Password

SYSTEM                  MANAGER
SYS                     change_on_install
```

These users are typically used to hold the standard data dictionary information for the database. It is a good idea to change the passwords from the defaults as soon as possible.

This can be achieved by:

```
sqlplus system/manager

SQL*Plus: Release 3.3.3.0.0 - Production on Sat Feb 21 12:43:33 1998

Copyright (c) Oracle Corporation 1979, 1996.  All rights reserved.


Connected to:
Oracle7 Server Release 7.3.3.0.0 - Production Release

SQL> alter user system identified by <newpassword>;

User altered.

SQL> alter user sys identified by <newpassword>;

User altered.

SQL> exit;
Disconnected from Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production
```

Since the user `system/manager` is similar to using `root` on a UNIX machine, we need to create a user with less ability to cause damage. (remember to bring up the database before attempting to create a user)

Connect to SQL*Plus and create a user:

```
$ sqlplus system/manager

SQL*Plus: Release 3.3.3.0.0 - Production on Sat Feb 21 12:43:33 1998

Copyright (c) Oracle Corporation 1979, 1996.  All rights reserved.


Connected to:
Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production

SQL> create user <user> identified by <psw>
```

```
  2  default tablespace users
  3  temporary tablespace temp;

User created.

SQL> grant connect, resource to <user>

Grant succeeded.

SQL> exit
Disconnected from Oracle7 Server Release 7.3.3.0.0 - Production Release
PL/SQL Release 2.3.3.0.0 - Production
```

Now that you have a new user on the system you can play with the new system. To login to the Oracle database:

```
$ sqlplus <user>/<password>
```

If this completes with no error messages then you have a working Oracle database. If you never want to connect to this database from anywhere but this server then the job is complete, enjoy!

If, however, like most people you want to configure the networking software so that you can connect from other machines, keep on reading.

4.Configuring SQL*Net on the Server

All of these files configure the Oracle networking software (SQL*Net, aka Net8 for Oracle8). These files should all be created on the server in the `$ORACLE_HOME/network/admin` directory.

4.1 `tnsnames.ora`

The `TNSNAMES.ORA` file identifies services available from the machine. On our instance here we will describe all databases that the server has mounted. For each database instance on your server add a section like below:

```
orcl.world =
  (DESCRIPTION =
    (ADDRESS_LIST =
        (ADDRESS =
          (COMMUNITY = tcp.world)
          (PROTOCOL = TCP)
          (Host = <INSERT HOST NAME OF SERVER HERE> )
          (Port = 1521)
        )
        (ADDRESS =
```

```
          (COMMUNITY = tcp.world)
          (PROTOCOL = TCP)
          (Host = <INSERT HOST NAME OF SERVER HERE> )
          (Port = 1526)
        )
    )
    (CONNECT_DATA = (SID = ORCL)
    )
  )
```

4.2 `listener.ora`

The `listener.ora` file contains the descriptions of the services that other machines are allowed to connect to and any configuration that is required for the server listener.

It contains sections for the listener name, listener address, databases served by the listener and configuration parameters.

Here is an example:

```
# Name of listener and addresses to listen on
LISTENER =
        ( ADDRESS_LIST =
                (ADDRESS =
                        (PROTOCOL=tcp)
                        (HOST=<INSERT HOST>)
                        (PORT=1521)
                        (COMMUNITY=UK_SUP_TCPIP)
                )
                (ADDRESS =
                        (PROTOCOL=ipc)
                        (KEY=700)
                        (COMMUNITY=UK_SUP_TCPIP)
                )
        )

# List of services served by this listener
SID_LIST_LISTENER=
        (SID_LIST=
                (SID_DESC=
                        (SID_NAME=orcl)
                        (ORACLE_HOME=/home/oracle/7.3.3.0.0)
                )
        )

# Start of configuration parameters.
TRACE_LEVEL_LISTENER=OFF
TRACE_FILE_LISTENER = "listener"
LOG_FILE_LISTENER = "listener"
CONNECT_TIMEOUT_LISTENER = 10
STOP_LISTENER = YES
DBA_GROUP = dba
```

4.3 `sqlnet.ora`

The `sqlnet.ora` file contains configuration for the particular node of the network. This is independent of the number of databases or the number of listeners. The most important thing in this file is the `Dead Connection Timeout` configuration variable.

Dead connection timeout checks every incoming process to a database instance and ensures that the client end of it is still responding. If the client (of whatever type) is not responding then the Oracle server shadow process is killed.

This is very useful if you have many clients accessing a database, especially during a developmental phase when those clients are more likely to be failing to exit cleanly from the Oracle database.

Below is a copy of my own `sqlnet.ora` file for you to peruse:

```
TRACE_LEVEL_CLIENT = OFF
sqlnet.expire_time = 30         # The number of seconds between client checks.
names.default_domain = world
name.default_zone = world
```

4.4 Starting and Stopping the Listeners

Now that the configuration of the listeners and SQL*Net is complete we can attempt to connect to the database using the networking software. (Before we were using direct links to the database, whereas here we are simulating a connection from a remote client machine).

To start the listener using the above configuration:

```
$ lsnrctl

LSNRCTL for SCO System V/386: Version 2.3.3.0.0 - Production on 23-FEB-98 20:38:25

Copyright (c) Oracle Corporation 1994.  All rights reserved.

Welcome to LSNRCTL, type "help" for information.

LSNRCTL> start
Starting /home/oracle/7.3.3.0.0/bin/tnslsnr: please wait...

TNSLSNR for SCO System V/386: Version 2.3.3.0.0 - Production
System parameter file is /home/oracle/7.3.3.0.0/network/admin/listener.ora
Log messages written to /home/oracle/7.3.3.0.0/network/log/listener.log
Listening on: (ADDRESS=(PROTOCOL=tcp)(DEV=6)(HOST=192.168.1.1)(PORT=1521))
Listening on: (ADDRESS=(PROTOCOL=ipc)(DEV=10)(KEY=700))

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=magic.com)(PORT=1521)(COMMUNITY=UK_SUP_TCPIP))
STATUS of the LISTENER
------------------------
```

```
Alias                     LISTENER
Version                   TNSLSNR for SCO System V/386: Version 2.3.3.0.0 - Production
Start Date                23-FEB-98 20:38:50
Uptime                    0 days 0 hr. 0 min. 0 sec
Trace Level               off
Security                  OFF
SNMP                      ON
Listener Parameter File   /home/oracle/7.3.3.0.0/network/admin/listener.ora
Listener Log File         /home/oracle/7.3.3.0.0/network/log/listener.log
Services Summary...
  orcl          has 1 service handler(s)
The command completed successfully
LSNRCTL> exit
```

To stop the listeners:

```
$ lsnrctl

LSNRCTL for SCO System V/386: Version 2.3.3.0.0 - Production on 23-FEB-98 20:43:20

Copyright (c) Oracle Corporation 1994.  All rights reserved.

Welcome to LSNRCTL, type "help" for information.

LSNRCTL> stop
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=magic.com)(PORT=1521)(COMMUNITY=UK_SUP_TCPIP))
The command completed successfully
LSNRCTL> exit
```

If you have a DNS setup which doesn't return the IP address for the hostname specified then starting and stopping the listener can take some time (2–3 mins. dependant on the DNS timeout variable). If this is the case, don't worry, be patient.

5.Client Configuration

5.1 Windows Clients

SQL*Net configuration on the PC using newer versions of the Oracle Client Software is very easy. The best (and easiest) way of achiving a fully working client installation is to use the `SQL*Net Easy Configuration` tool supplied by Oracle.

This toolhas a wizard type interface to take you through the installation of the `tnsnames.ora` and `sqlnet.ora` files.

Select "Add Database Alias" and enter a name for the alias when prompted. This alias is the name you will refer to the database instance as, and as such should be the same as the instance name (orcl in this case).

Select TCP/IP as the protocol, and when prompted the hostname of the machine hosting the database and the instance name of the database.

That's it.

However, if you do not have the `SQL*Net Easy Configuration` Tool then don't worry. You can simply create the `tnsnames.ora` and the `sqlnet.ora` files in the `$ORACLE_HOME/network/admin` directory on the client exactly as they are on the server. This will provide an alias the same as on the server (always a good idea anyway).

5.2 Unix Clients

UNIX clients are not very different that windows clients. If you have the `Network Manager` from Oracle then user that in the same way as above, if not then you can, again, just use the same configuration files as the server in the `$ORACLE_HOME/network/admin` directory.

6.Automatic Startup and Shutdown

6.1 `dbstart` & `dbstop`

The automatic startup and shutdown of the Oracle database can be achieved (in 7.3.3.0.0) with the files `dbstart` and `dbshut` both provided by Oracle. These files rely on the existance of the file `/etc/oratab` to work (although by altering the `dbshut` and `dbstart` files this can be moved).

The format of the `/etc/oratab` file is as follows:

```
SID:ORACLE_HOME:AUTO
```

An example:

```
orcl:/home/oracle/7.3.3.0.0:Y
leaveup:/home/oracle/7.3.2.1.0:N
```

6.2 `init.d` & `rc.d`

To start and stop the database when the machine comes up and goes down by modifying the startup routines for the Linux machine. This is quite easy, although I should point out here that this may change depending on which flavour of Linux (slackware, debian, redhat, etc). I will show examples which work for Redhat Linux 5.0. To modify these for your own flavour of Linux, please see your Linux documentation sets. (Although it should hold true for any Sys V type UNIX).

Firstly, we need to create the script which will run `dbshut` and `dbstart` in the `/etc/rc.d/init.d` directory. Create the following file as `/etc/rc.d/init.d/oracle`:

```
#!/bin/sh
#
# /etc/rc.d/init.d/oracle
# Description: Starts and stops the Oracle database and listeners
# See how we were called.
case "$1" in
  start)
        echo -n "Starting Oracle Databases: "
        echo "----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Starting Oracle Databases as part of system up." >> /var/log/oracle
        echo "----------------------------------------------------" >> /var/log/oracle
        su - oracle -c dbstart >> /var/log/oracle
        echo "Done."
        echo -n "Starting Oracle Listeners: "
        su - oracle -c "lsnrctl start" >> /var/log/oracle
        echo "Done."
        echo ""
        echo "----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Finished." >> /var/log/oracle
        echo "----------------------------------------------------" >> /var/log/oracle
        touch /var/lock/subsys/oracle
        ;;
  stop)
        echo -n "Shutting Down Oracle Listeners: "
        echo "----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Shutting Down Oracle Databases as part of system down." >> /var/log/oracle
        echo "----------------------------------------------------" >> /var/log/oracle
        su - oracle -c "lsnrctl stop" >> /var/log/oracle
        echo "Done."
        rm -f /var/lock/subsys/oracle
        echo -n "Shutting Down Oracle Databases: "
        su - oracle -c dbshut >> /var/log/oracle
        echo "Done."
        echo ""
        echo "----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Finished." >> /var/log/oracle
        echo "----------------------------------------------------" >> /var/log/oracle
        ;;
  restart)
        echo -n "Restarting Oracle Databases: "
        echo "----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Restarting Oracle Databases as part of system up." >> /var/log/oracle
        echo "----------------------------------------------------" >> /var/log/oracle
        su - oracle -c dbstop >> /var/log/oracle
        su - oracle -c dbstart >> /var/log/oracle
        echo "Done."
        echo -n "Restarting Oracle Listeners: "
```

```
        su - oracle -c "lsnrctl stop" >> /var/log/oracle
        su - oracle -c "lsnrctl start" >> /var/log/oracle
        echo "Done."
        echo ""
        echo "-----------------------------------------------------" >> /var/log/oracle
        date +"! %T %a %D : Finished." >> /var/log/oracle
        echo "-----------------------------------------------------" >> /var/log/oracle
        touch /var/lock/subsys/oracle
        ;;
  *)
        echo "Usage: oracle {start|stop|restart}"
        exit 1
esac
```

It is worth checking that this file actually correctly stops and starts the databases for your system. Check the log file, `/var/log/oracle` for error messages.

Once this script is working we need to create start and kill symbolic links in the appropriate runlevel directories `/etc/rc.d/rcX.d`.

The following commands will ensure that the databases will come up in runlevels 2,3 and 4:

```
$ ln -s ../init.d/oracle /etc/rc.d/rc2.d/S99oracle
$ ln -s ../init.d/oracle /etc/rc.d/rc3.d/S99oracle
$ ln -s ../init.d/oracle /etc/rc.d/rc4.d/S99oracle
```

To stop the databases on reboot or restart we need the following links:

```
$ ln -s ../init.d/oracle /etc/rc.d/rc0.d/K01oracle          # Halting
$ ln -s ../init.d/oracle /etc/rc.d/rc6.d/K01oracle          # Rebooting
```

7.Other Bits

7.1 Intelligent Agent

If you have a need for the `Oracle Intelligent Agent`, I found that you can run it without any configuration changes. To start the IA try:

```
$ lsnrctl dbsnmp_start
```

To stop the IA try:

```
$ lsnrctl dbsnmp_stop
```

There do not appear to be any messages indicating a sucessful or otherwise start or stop of the intelligent agent. However, the IA responded to Enterprise manager on the client side so I can only assume that it is working.

8.Troubleshooting

See below for various troubleshooting hints.

8.1 I cannot create a database when using Oracle 7.2.x.

The files shipped by Oracle in the 7.2.x product are incorrect in assuming that you want to setup a parallel server configuration. The shipped `init.ora` file has the following line in it:

```
# define parallel server (multi-instance) parameters
ifile = ora_system:initps.ora
```

To fix the problem simply comment it out:

```
# define parallel server (multi-instance) parameters
#ifile = ora_system:initps.ora
```

8.2 I'm getting segmentation faults in `svrmgrl` under version 7.3.4.x.

I've had this problem reported to me by a number of people. Gerald Weber `gerald_weber@master.co.at` solved it:

```
Hi Paul,
```

```
first of all thanks for your help,but none of the possible problems you are
thinking about were responsible for my problem.
The problem is the iBCS-emulator itself.
It seems that Oracle performs an sysconf-calls which isn't supported in the
current version of iBCS.
Look at the trace :

<7>[22]615 sysconf(34)
<7>iBCS2 unsupported sysconf call 34
<7>[22]615 sysconf error return linux=-22 -> ibcs=22 <Invalid argument>
<7>[24]615 sysconf(34)
<7>iBCS2 unsupported sysconf call 34
<7>[24]615 sysconf error return linux=-22 -> ibcs=22 <Invalid argument>

Solution:  patching the iBCS-source.apply the following diff-pach :


--- sysconf.c   Sun Apr 19 19:19:15 1998
+++ sysconf.c.ori       Sun Apr 19 19:28:45 1998
@@ -60,7 +60,6 @@
 #define _SC_JOB_CONTROL        5
 #define _SC_SAVED_IDS  6
 #define _SC_VERSION    7
-#define _SC_HACK_FOR_ORACLE 34

 #define _SC_PAGESIZE   11

@@ -97,11 +96,6 @@
                case _SC_SAVED_IDS: {
                        return (1);
                }
-
-               case _SC_HACK_FOR_ORACLE: {
-                 return (1);
-               }
-

                case _SC_PAGESIZE: {
                  return PAGE_SIZE;
```

9.Credits

This document was based on a document written by Bob Withers, `bwit@pobox.com`. Additional information taken from documents written by Georg Rehfeld, `rehfeld@wmd.de` and David Mansfield, `david@claremont.com`.

Additional proof reading done by Bob Withers, Mark Watling, `mwatling@mjw-ltd.demon.co.uk`, Peter Sodhi, `petersodhi@unn.unisys.com` and Greg Hankins, `greg.hankins@cc.gatech.edu`.

My thanks go to the tremendous support from all the people involved in this document and the research that has gone into it. Particular thanks to Bob Withers and Mark Watling for the additional comments and help they have provided.