Tips and Techniques: Install and Configure Oracle9i on Red Hat Linux Advanced Server

An Oracle White Paper
August 2002

ORACLE®

Tips and Techniques: Install and Configure Oracle9i on Red Hat Linux
Advanced Server

EXECUTIVE OVERVIEW

Red Hat Linux Advanced Server is an enterprise ready platform, borne out of feedback and close collaboration between the engineering teams from Oracle and Red Hat. With features like large memory support, lower mapped base support for process, shared memory file system (shmfs), high page table entry (pte) fix, asynchronous I/O support, bigpages feature and greedy swap patch, Red Hat Linux Advanced Server is a robust product.

The key to successful installation and deployment of Oracle9i Database Release is proper installation of the operating system. Careful installation and tuning of Red Hat Linux Advanced Server would make deployment of Oracle9i database an easier task.

The objective of this paper is to help resolve common issues in the installation of Oracle9i Database Release 2, and to provide tips on various kernel parameters for Red Hat Linux Advanced Server, that would eventually help in tuning database server. Various debugging tools and techniques are discussed.

TERMINOLOGY

The underlying terms are used in this paper, with specific meanings as indicated.

Name	Explanation
VLM	Very large Memory (> 4GB memory)
shmfs	Shared memory file system on /dev/shm
highmem	Higher memory addresses above 1 GB
mapped_base	Per process value for setting address to search memory chunk during mmap
VLM window	Virtual memory range to dynamically map in the buffer cache
pte	Page table entry

highpte	Kernel patch which allows putting pte-s into highmem
enterprise kernel	Kernel that can support VLM
SMP kernel	Kernel that can support only up to 4GB VM
OEM	Oracle Enterprise Manager
SGA	Shared Global Area for oracle instance
RH	Red Hat
AS	Advanced Server
OS	Operating System
bigpages	Kernel feature to enable large page size for shared memory pages

INSTALLATION OF RED HAT LINUX ADVANCED SERVER 2.1

The primary task in the installation of Oracle9i Database on Advanced Server 2.1 is a proper install of Advanced Server 2.1. Hence, this section focuses on the install of Red Hat Advanced Server 2.1.

Installation steps for Red Hat Advanced Server 2.1

The installation of Red Hat Advanced Server 2.1 is started on a certified hardware. After booting off the install CD, one of the installation modes is chosen based on the type of hardware.

1. The following installation modes are listed.

- Install/upgrade in Graphical mode: <ENTER>

- Install/upgrade in text mode: text <ENTER>

- Low resolution mode: lowres <ENTER>

- Disable framebuffer mode: nofb <ENTER>

- Expert mode: expert <ENTER>

- Rescue mode: linux rescue <ENTER>

- Have a driver disk mode: linux dd <ENTER>

The first three modes are for older systems, which have lower video card memory and lower resolution monitors. The Expert mode is used when hardware autoprobing fails. This mode would take complete control over the installation process. Rescue mode is used when the machine doesn't boot off the hard disk and no bootable floppy disk is available. Linux dd mode is used when the machine has newer hardware, for which drivers are not available on Advanced Server CD. The driver for the newer hardware is available on a disk.

2. The next step is to select the language to use during installation.

3. Followed by the selection of keyboard and mouse setup ("Simulate 3 button mouse" is chosen, if the system has a 2 button mouse).

4. Next step is to choose the installation type,

- Advanced Server (It is the recommended choice for 9i R2 database installation)

- Custom (This option provides a more detailed and flexible package group selection. The downside is, it is very time consuming. If "install all packages" is selected, /usr partition requires around 6 GB of disk space.)

5. Partition setup, the next step, is the most critical step in the whole installation process. Poor partition selection may result in very poor overall performance. The following options would be listed.

- Automatic. (Use of this option is not recommended, as only / and /boot partitions are created and also very little swap is used.)

- Disk Druid. (This is the recommended option, because of the ease of use and flexibility)

- fdisk. (This option is suited for expert fdisk users. It provides the same features and flexibility as the preceding option, but without the naive user-friendly interface)

On a system, where a single instance Oracle9i R2 database is to be deployed, the following partitions are recommended on the OS disk (18GB in size),

/	2 GB
/boot	64 MB
/tmp	1 GB
/usr	3 GB (3 GB Advanced Server install, 7 GB for all package install)
/var	512 MB
swap	(at least size of physical memory)
/home	2 GB (Can be more if required)

6. The following choices are available for Boot Loader. LILO or GRUB could be chosen, based on the expertise of the administrator.

   - GRUB

   - LILO

   - no boot loader

7. The next step is to configure the network. Generally, Static IP address is assigned to the server on which the Oracle9i R2 database is deployed. If DHCP is chosen, the install automatically gathers the network information to configure the same.

   - DHCP

   - Static IP (please include domainname in hostname)

If static IP address is selected, it is advised that a fully qualified hostname be provided during the hostname interview (e.g. foo.us.acme.com). The install process would interview for information on the Gateway, netmask, DNS etc.

8. Based on the requirement, high or medium security firewall can be configured.

   - High

   - Medium

   - No firewall

9. Additional Language Support - This option is to select additional languages, over the one chosen for installation.

10. Timezone - Selection of appropriate time zone.

11. Account Configuration (Recommendation is to create only "root" during install time. If other users are created during install time, Advanced Server install creates groups with same usernames, which might be undesirable.)

12. Package Group Selection – The later part of the install depends on the selection made in this step. The following are presented for selection.

   - GNOME

   - KDE

   - Software Development

   - Check "select individual package" option.

For an Oracle9i R2 install, selection of "Software Development" package is a must, for a couple of reasons. It is needed for relinking Oracle at install time, as well as, to take advantage of the Advanced Server features like asynchronous I/O, lower mapped base.

Selection of GNOME and KDE package group selection is helpful.

To select additional RPM packages that are not part of standard Advanced Server install, the "select individual package" option needs to be chosen.

13. Graphical Interface (X) Configuration – It is suggested to skip this configuration, if the server being installed is to be used as a database server and is to be hosted in data centers, with limited accessibility to console. Selection of appropriate video card is necessary to complete this step, if X needs to be configured on the system.

14. Boot Floppy Creation: It is always good idea to choose "Create" from the selection list. The boot floppy created would help during recovery of the system.

- Create

- Skip

15. Monitor Configuration: Appropriate monitor information is selected here.

16. Login Console – Either of these options could be selected.

- Graphical

- Text

At this stage, installation of Advanced Server is completed. Check "Install Tips for Red Hat Advanced Server 2.1" section for additional requirements or installation issues.

Reboot the system after setting up all required OS/kernel parameters. The Advanced Server is now ready for database install.

Install Tips for Red Hat Linux Advanced Server 2.1

The previous section, "Installation of Red Hat Linux Advanced Server 2.1", outlined steps for installing Red Hat Advanced Server. If the installation still runs into issues, the following list might be helpful in resolving the same.

- If the loading of driver fails at install time, the latest or certified driver is procured on a floppy and install is retried with linux dd option.

- If the system has two SCSI controllers, it might attempt to boot from any one of them. It is suggested to turn off the boot-capability at the firmware, for the SCSI controller from which boot is not desired.

- If the system has a raid controller, sometime /dev/ida/* may not get displayed by fdisk. This is an issue with the array drivers. It is recommended to use full device name with fdisk commands (example fdisk /dev/ida/c0d0).

- If machine has <= 4 GB memory SMP kernel will be installed and if machine has > 4 GB memory enterprise kernel will be installed. If the install of debug kernel is desired in a testing environment, the debug kernel is available only for SMP version. (i.e. highmem is enabled, but PAE36 is not, so it ignores memory over 4GB.)

- It is highly recommended to use disk druid for disk layouts.

- When running into issues with packages selection, it is advantageous and easiest to select "Advanced Server install".

- ftp is not installed by default. (wu-ftpd package). Telnet, ftp services are disabled when installed for security purpose. These services could be enabled by invoking, "/sbin/chkconfig <service name>".

- Telnet, ftp, rsh, rexec, rlogin services are insecure. Use of secure services (openssh packages), like ssh, is advised.

- Highly secure authentication configuration is available with the Custom installation option. Users with MD5 password and shadow password could be created.

- Additional services like NIS, LDAP, kerbero5 and SMB authentication could be selected with the Custom install option.

- Network could be configured using "neat" tool.

- For data storage, internal storage (local disks) or external storage - Network attached storage or fiber channel attached off host storage could be used. Availability of proper controllers and drivers needs to be ensured.

- It is suggested that the system be allocated with enough disk space for database. That would make the installation of Oracle9i R2 easier.

TIPS TO INSTALL ORACLE9I DATABASE RELEASE 2

It is advisable to go through this section before starting the Oracle9i R2 install.

Relink issues during Oracle9i Release 1 database install

Relinking errors occur during the installation of third CD of the Oracle9i R2 DB on Red Hat Advanced Server. This problem has been resolved with binutils-2.11.90.0.8-13.i386.rpm. Solution is as follows and available on Metalink and OTN.

1) It is suggested that the existing version information of binutils on the system is backed up.

#rpm -qa|grep binutils   (binutils-2.11.90.0.8-12)

If the system is installed with the Red Hat Advanced Server, then binutils-2.11.90.0.8-12 version is installed on it.

 2) The binutils-2.11.90.0.8-13.i386.rpm could be downloaded from Oracle Metalink (ARU) or OTN site. The binutils on the system is upgraded using the following command.

#rpm -Uvh --force binutils-2.11.90.0.8-13.i386.rpm

This patch only fixes problem during install. Therefore, once oracle product is installed, it is advised to revert back to the saved version of binutils. For e.g., to revert to the 2.11.90.0.8-12 version, the following command could be executed.

#rpm -Uvh --force binutils-2.11.90.0.8-12.i386.rpm

The following files need to be modified to avoid the relink errors in future. The "-z defs" flag needs to be removed in all of them.

$ORACLE_HOME/bin/genclntsh

$ORACLE_HOME/ldap/lib/ins_ldap.mk

$ORACLE_HOME/ldap/lib/env_ldap.mk

$ORACLE_HOME/sqlplus/lib/env_sqlplus.mk

Cannot install using CDROM

If there are issues with using CDROM, like read operation succeeds, but what is read from CDROM is corrupted, following workaround may help.

The workaround is to disable DMA on the CD-ROM, by executing,

hdparm -d0 /dev/cdrom

Sun JDK - Oracle9i R2 and Advanced Server 2.1

IBM's JDK 1.3.1 is packaged with Red Hat Linux Advanced Server 2.1. But, Oracle9i R2 has been certified with Sun's JDK 1.3.1_02 and Blackdown's JDK 1.1.8. Therefore, it is suggested that these JDKs are installed on the system, prior to installing Oracle9i R2. It would be a good idea to remove IBM jdk, as there are performance issues with use of IBM jdk with Oracle9i R2. If there is a need to retain IBM jdk, then it should be ensured that Sun's jdk is included in PATH, before IBM JDK.

Creating SGA of more than 8 GB, using bigpages and shmfs

Creating SGA of more than 8GB on a 16GB machine, using bigpages in Shared memory file system (shmfs), would fail on Red Hat Advanced Server 2.1. The system returns the error, *"write: No space left on device wrote 8192000 blocks."* This only

happens with bigpages in shared memory file system (shmfs). But, SGA of more than 8GB can be created using bigpages in SYSV. This problem is fixed in errata e.8 kernel.

Database creation failure

If database creation fails, either during install or using dbca, the potential error/warning is due to some default kernel parameters. The following error generally indicates a wrong setting for SHMMAX and semaphore parameters.

"WARNING: EINVAL creating segment of size 0x000000000c400000"

Correct values needs to be set for shared memory parameters.

The SHMMAX is set by executing,

    echo "<number>"> /proc/sys/kernel/shmmax

    where the value of <number> is at least half of physical RAM.

The semaphore values are set by the following command.

    echo "100 32000 100 100" > /proc/sys/kernel/sem

It is recommended to visit the "Tuning of Red Hat Advanced Server for Oracle" section for more information on tuning the kernel parameters.

ORACLE DATABASE INSTALL

It is strongly recommended that the user go through the install documentation and readme shipped with Oracle9i R2 CD. This section would guide the user, screen by screen, in installing a Single Instance Oracle database, using OUI. Users familiar with OUI may skip this section.

The user needs to execute *runInstaller* script from Oracle9i R2 CD 1.

The screen that appears is the Welcome screen. The user is given the option to uninstall any existing oracle product. The user could click next to continue, after done with uninstalling any products.

The next screen that appears is the Inventory Locations screen. The user is suggested to choose a proper Oracle Base directory, which is the base for all Oracle products and then continue the install.

Following the inventory location screen, is the Unix group Name screen. The user is requested to select unix group name which will have permissions to update

Oracle Software. After choosing to continue the install, Installer would prompt the user to run a script as root, to proceed further with interviewing for file locations.

The File Locations screen queries the user for name and location of Oracle home. The Oracle home is the location where the Oracle9i R2 binaries and configuration files would be installed. The user is prompted to proceed with the next screen to browse the list of Available products.

The Available Product screen lists the four options available. The user is advised to choose the first option, "Oracle9i DB 9.2.0.1.0 ". The user could change product language also on this screen. With the installation language chosen, the next step is to choose the Installation type.

The Installation Types screen lets the user choose between the Enterprise Edition, Standard Edition, Or Custom install. The user is suggested to choose Enterprise Edition, as this paper focuses on install of Oracle9i R2 for an Enterprise server.

The next screen is for Database Configuration, which allows the user to create the database of his needs. The user could choose to create database at a later point in time, by selecting "software only" install option. This paper details only the "software only" option.

The installer having gathered all the required information, displays the Summary screen. The user is strongly suggested to review all the selections at this point of time. It would take approximately 1/2 hour to install the binaries, based on machine speed.

The installer guides the user to change to CD 2 and 3. It would prompt the user to run another script as root. Execution of that script completes the install.

When the user exits from the Exit install screen, the Installer automatically launch OEM console. If the database was created as part of the install, the user could configure database, listener and other configuration related to the database, at this point of time. Else, the user is suggested to exit from the OEM console.

If the user chose to defer database creation during OUI install, the user could use tools like dbca, netca and OEM console to complete the configuration. Alternately, the user could write scripts to do the same. The following section on "Tuning of Red Hat Advanced Server for Oracle" the Red Hat Advanced Server would let the user identify parameters that could be fine tuned for successful operation of Oracle9i R2 database.

TUNING OF RED HAT LINUX ADVANCED SERVER FOR ORACLE

In this section, tuning of various Red Hat Advanced Server 2.1 kernel parameters is explained. Many of the parameters related to the new features introduced in the Red Hat Advanced Server are discussed. The significance of these kernel parameters to Oracle and the range of workable values that could be set are detailed.

It is suggested that those parameters that require to be set at every boot be saved in /etc/rc.local. The same parameters could also be set in /etc/sysctl.conf, but the syntax is different.

Swap:

It is recommended that the minimum swap size be atleast the same as physical memory size.

File handles:

The recommendation for minimum value of File Handles is 65536. It could be set using ulimit -n or /proc/sys/fs/file-max. It could be set to higher values as needed.

    echo "65536" > /proc/sys/fs/file-max

    ulimit -n 65536

Shared memory:

SHMMAX: This parameter sets the largest shared memory segment size allowed by the kernel, in bytes. Default value for this parameter is 33554432. With this value, database creation may fail. Therefore it is advisable to increase the value of SHMMAX, such that, Oracle will allocate the SGA in one segment.

For a system with 4 GB RAM, the suggested value of SHMMAX is 2 GB. Maximum value of SHMMAX that could be set is 4 GB - 1.

    echo "2147483648" > /proc/sys/kernel/shmmax

SHMMNI: This parameter sets the maximum number of shared memory segments for the whole system.

    echo "4096" > /proc/sys/kernel/shmmni

SHMALL: This parameter sets the total amount of shared memory that can be used at one time on the system, in bytes.

    echo "2097152" > /proc/sys/kernel/shmall

Shared memory file system on /dev/shm (shmfs):

To create large SGA size > 1.7 GB Shared memory file system (shmfs) is used. Following commands could be used to create Shared memory file system (shmfs) and these need to be added to /etc/rc.local. User could also create Shared memory file system (shmfs) using /etc/fstab. It would be a good idea to remove or comment out existing entry for /dev/shm from /etc/fstab before adding entry for /dev/shm.

    mount -t shm shmfs -o size=32g /dev/shm

    (m and k also works fine) OR

    mount -t shm shmfs -o nr_blocks=8388608 /dev/shm

Using Shared memory file system (shmfs) on 8 GB system, SGA of upto 7 GB could be created. To take advantage of Shared memory file system (shmfs), the following parameter needs to be set in init<sid>.ora

    use_indirect_data_buffers=true

It has to be ensured that SHMMAX is set to a proper value, before increasing SGA using Shared memory file system (shmfs) solution.

Lower mapped base:

Lowering mapped base helps increase SGA size from 1.7 GB to 2.7 GB without using Shared memory file system (shmfs). This is true for any system with atleast 4GB of memory.

To lower mapped base, any of following techniques could be used.

The pid of the process (shell) that started oracle needs to be identified. This could be done using ps command. The pid could also be identified using "echo $$". As root, the following command is executed to lower mapped_base to 0x10000000.

    echo 268435456 > /proc/$pid/mapped_base

where $pid is the pid of the shell from which Oracle was started.

Lowering mapped base works with both VLM and non-VLM installations. This requires relinking of Oracle. To relink Oracle to make use of Lower mapped base, the following commands need to be run from $ORACLE_HOME/rdbms/lib.

    genksms -s 0x15000000 > ksms.s

    make -f ins_rdbms.mk ksms.o

    make -f ins_rdbms.mk ioracle

To take advantage of lower mapped base solution, proper value needs to be set for SHMMAX, as described in the Shared memory section of this paper.

VLM_WINDOW_SIZE

This is an undocumented feature and works only with lowering mapped base. This variable could be set in Oracle user profile file. Increasing VLM_WINDOW_SIZE has reflected very little performance improvement. It could be set by executing,

export VLM_WINDOW_SIZE=1073741824 (size in bytes, default is 512 MB)

Bigpages

This feature of Advanced Server generates pages of size 4MB Vs the regular 4KB. And these pages do not get swapped out of memory. The intent is to lock entire SGA in the memory. Therefore it generates smaller number of page table entries (pte-s) and results in support of more users. Typically this feature is very useful with larger SGA size and more number of users. This feature is tuned to perform with Enterprise kernel.

The number of bigpages used can be displayed by,

cat /proc/meminfo.

To enable bigpages feature, the following steps are to be performed.

1) set one following values in /etc/rc.local.

echo 2 > /proc/sys/kernel/shm-use-bigpages (bigpages in shmfs)

echo 1 > /proc/sys/kernel/shm-use-bigpages (bigpages in SYSV memory)

echo 0 > /proc/sys/kernel/shm-use-bigpages (disable bigpages feature, default)

Use bigpages either in shmfs or SYSV. To disable bigpages use last entry.

2) The following line is added to /etc/lilo.conf in the boot section and LILO is reconfigured and system is rebooted.

    append = "bigpages=xxxxMB"

Where xxxx is value of big pages in MB.

The suggested values for bigpages / SGA combinations on a 8GB machine are listed.

For a SGA of 4GB, bigpages of size 4100MB could be set and for SGA of 2GB, bigpages of size 2100MB could be set.

The Maximum size of 5.4 GB of SGA can be created on 8 GB machine.

Sockets and Processes

Sockets : Defines the local port range that is used by TCP and UDP to choose the local port. The first number is the first, the second the last local port number.)

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

Process: Maximum number of 16384 processes, per user, can be set by executing

  ulimit -u 16384

Semaphore

Recommended values for various semaphore variables are as follows:

SEMMSL - The largest PROCESSES parameter of any Oracle database on the system plus ten. The minimum recommended value is 100

SEMMNS - Sum of the PROCESSES parameter for each Oracle database, adding the largest one twice, and then adding an additional 10 for each db. The minimum recommended value is 256

SEMOPM - Defines the maximum number of operations per semop call. Recommended value is 100

SEMMNI - Defines the maximum number of semaphore sets in the entire system. Minimum - recommended value is 100

All four values are set together using following command.

  echo "100 32000 100 100" > /proc/sys/kernel/sem

Asynchronous I/O

To use asynchronous I/O and synchronous I/O, Oracle is relinked with the following options. The default I/O mode for Oracle is Synchronous I/O.

  make -f ins_rdbms.mk async_on

  make -f ins_rdbms.mk async_off

If Oracle binary linked with async_on option, then following parameters turn on/off asynchronous I/O operations.

  disk_asynch_io = true (init.ora parameter default)

  tape_asynch_io = true (init.ora parameter default)

  disk_asynch_io=false (with the Oracle binary linked with async_on, will result in
     synchronous I/O)

Asynchronous I/O feature is only available on Red Hat Advanced Server release and not on other Linux platforms. To get better I/O throughput with DSS workloads, the /proc/sys/fs/aio-max-size is increased from the default 131072 bytes to >=1M. For OLTP workloads, the default size of 131072 would suffice, as these workloads perform very small writes.

DNS options

In Red Hat Advanced Server 2.1, the default number of attempts to resolve hostname is 2. If all the DNS servers listed in /etc/resolv.conf are very busy, the

hostname resolution might fail, which could potentially result in loss of Oracle sqlnet connection.

The solution is to increase the number of attempts and timeout to 5 and 15 Sec respectively, in the network configuration. The following lines are added to /etc/resolv.conf:

options attempts:5

options timeout:15

MONITORING

For fine-tuning database parameters in init.ora, it is essential to monitor the kernel parameters periodically. The data monitored are the vital statistics of the system and need to be collected at different points of operation, ranging from medium to peak loads. Monitoring and tuning of system and database parameters go hand in hand.

Listed below are some of the data points that could be monitored and tools that could effectively help in collecting the same.

To monitor memory, /proc/meminfo and /proc/slabinfo information could be collected. These two would give a complete picture of memory.

Other interesting stat to monitor is network information. /proc/net/dev, /proc/net/snmp, /proc/net/softnet_stat data is collected for that purpose. Users could use the tool "***netstat***".

Also utilities like ***vmstat*** for reporting virtual memory information, ***sar*** for measuring system activity information, ***iostat*** for measuring io and cpu stats for devices and partitions and ***top*** for measuring top cpu processes are available.

DEBUGGING AND TRACING

Debugging and tracing facilities are very critical in the case of crash or hang of a system, during database install time or at runtime. By looking at trace or logs, problem could be easily analyzed and resolved. It is highly advisable to have the following parameter always set, to help in debugging.

    echo 1 > /proc/sys/kernel/sysrq

Certain files and tools provide good starting point for analyzing the state of the system. /var/log/messages is one of them.

For problems related to loading drivers, **dmesg**, **lspci** and **lsmod** are very useful. Utilities like **strace** and **ltrace** are also available.

Other than these small utilities, there are the other debugging tools and techniques available, that would help in analyzing the system, even after it has hung or crashed completely. They are

    a)   use of magic sysrq keys

    b)   serial console.

    c)   Netdump utility

In all these cases, as was discussed before, the value of /proc/sys/kernel/sysrq should be set to 1.

Magic sysrq keys

Even in case of hang or crash, user could still collect system information. User could use magic sysrq keys. A key sequence is to be pressed for collecting some information or performing a specific task. The sequence is "alt", "sysrq" and any of following magic keys.

A table follows that lists the Magic sysrq key and the corresponding functionality.

Functionality	Magic sysrq key
sync data to disks	s
remount partitions read only	u
reboot the machine	b
power the machine off	o
turns off keyboard raw mode and puts it in XLATE	r
dump the cpu registers and flags	p
list tasks and some useful information about them	t
kills a process on the current console	k
send SIGTERM to everything except init	e
send SIGKILL to everything except init	i

send SIGKILL to everything including init	l
change the console loglevel	0-9

Serial console and network based debugging become significant when the local logging mechanisms, like klogd and syslogd fail. This could happen when there is a device driver or filesystem problem which has affected the /var filesystem.

Serial Console

To debug using serial console, /etc/lilo.conf needs an entry, as listed below, in the boot section. LILO is reconfigured for the new configuration to take effect.

    append="console=ttyS0,115200 console=tty0"

The other requirement is, to establish a physical or network connection, from the Linux system, to serial console (serial cable, terminal server etc..)

Oops, crash or hang output can be saved using serial console.

Netdump utility

Netdump is a network based debugging utility. The Netdump server and client are installed by default, with the Advanced Server install option. The services are enabled using

chkconfig <netdump_server/netdump> on

When a system crashes, the whole memory image is dumped on the Netdump server. Because the memory image is the size of physical memory of the system, the Netdump server needs lots of disk space to operate. A single Netdump server can handle multiple clients.

There is another feature of this package, by which the user can use **netconsole** to remotely log the debug / trace information, without using the space-consuming dump of the memory image.

CONCLUSION

The features and the facilities present in the Red Hat Advanced Server make it an Enterprise ready solution. The new features of the Red Hat Advanced Server have been designed and integrated to work with Oracle9i Database Release 2 database. Together, these solutions provide

a highly reliable, robust, scalable and enterprise ready "Unbreakable" Oracle on Linux.

The key to successful deployment of Oracle on Linux is to install and configure the Advanced Server to provide an environment for Oracle to work well. The tips and techniques highlighted and discussed in this paper, ranging from installing the OS, installing and configuring the database, workarounds and common issues in installing these software, fine tuning the kernel parameters affecting Oracle's runtime and continuous monitoring and debugging of the system, enable the Linux community to deploy a highly performant Oracle9i Database Release 2 on Red Hat Linux Advanced Server 2.1.

Tips and Techniques: Install and Configure Oracle9i on Red Hat Linux
Advanced Server

ORACLE