Oracle® Database

Database Installation Guide

21c for Linux

F31591-01
August 2021

Oracle Database Database Installation Guide, 21c for Linux

F31591-01

Copyright © 2019, 2021, Oracle and/or its affiliates.

Primary Author: Prakash Jashnani

Contributing Authors: Douglas Williams, Subhash Chandra

Contributors: Jean-Francois Verrier, David Austin, Neha Avasthy, Sampath Ravindhran, Prasad Bagal, Subhranshu Banerjee, Gerald Venzl, Tammy Bednar, Sivaselvam Narayanasamy, Sangeeth Jose, Pushkar Punit, Darcy Christensen, Kiran Chamala, Jonathan Creighton, Benoit Dageville, Logeshwaran Rajan, Rajesh Dasari, Angad Gokakkar , Anu Natarajan, Girdhari Ghantiyala, Bernard Clouse, Chandrasekharan Iyer, David Jimenez, Sivaram Soma, Lisa Vaz, Vishal Saxena, Vasu Venkatasubramanian, Suman Palavalli, Sameer Joshi, Malai Stalin, Markus Michalewicz, Subrahmanyam Kodavaluru, Sudip Datta, Madhu Hunasigi, Jim Erickson, Marcus Fallen, Joseph Francis, Allan Graves, Barbara Glover, Thirumaleshwara Hasandka, Putta Ramesh, Sergio Leunissen, Aneesh Khandelwal, Joel Kallman, Eugene Karichkin, Jai Krishnani, Prasad K Kulkarni, Kevin Jernigan, Christopher Jones, Simon Law, Bryn Llewellyn, Saar Maoz, Gopal Mulagund, Ankur Kemkar, Sue Lee, Rich Long, Raunak Rungta, Barb Lundhild, Robert Achacoso, Rudregowda Mallegowda, Prasad Kuruvadi Nagaraj, Krishna Mohan, John McHugh, Satish Panchumarthy , Rajesh Prasad, Rajendra Pingte, Ramesh Chakravarthula, Srinivas Poovala, David Price, Hanlin Qian, Michael Coulter, Hema Ramamurthy, Mark Richwine, Dipak Saggi, Rodrigo Gonzalez Alba, Naveen Ramamurthy, Mohit Singhal, Dharma Sirnapalli, Akshay Shah, James Spiller, Roy Swonger, Binoy Sukumaran, Ravi Thammaiah, Pablo Sainz Albanez, Hector Vieyra, Peter Wahl, John Haxby, Sergiusz Wolicki

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government’s use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface

Audience xi
Documentation Accessibility xi
Diversity and Inclusion xi
Set Up Java Access Bridge to Implement Java Accessibility xii
Command Syntax xii
Conventions xiii

1 Oracle Database Installation Checklist

Server Hardware Checklist for Oracle Database Installation 1-1
Operating System Checklist for Oracle Database Installation on Linux 1-2
Server Configuration Checklist for Oracle Database Installation 1-3
Oracle User Environment Configuration Checklist for Oracle Database Installation 1-5
Storage Checklist for Oracle Database Installation 1-6
Installer Planning Checklist for Oracle Database 1-7
Deployment Checklist for Oracle Database 1-10

2 Checking and Configuring Server Hardware for Oracle Database

Logging In to a Remote System Using X Window System 2-1
Checking Server Hardware and Memory Configuration 2-2

3 Automatically Configuring Oracle Linux with Oracle Database Preinstallation RPM

About the Oracle Database Preinstallation RPM 3-1
Overview of Oracle Linux Configuration with Oracle Database Preinstallation RPM 3-2
Installing the Oracle Database Preinstallation RPM Using ULN 3-3
Installing Oracle Database Preinstallation RPM During an Oracle Linux Installation 3-5
Installing Oracle Oracle Database Preinstallation RPM Using the Oracle Linux yum Server 3-6
Configure Additional Operating System Features 3-7

4 Configuring Operating Systems for Oracle Database on Linux

Guidelines for Linux Operating System Installation 4-2
- Completing a Minimal or Default Linux Installation 4-2
- About Oracle Linux with the Unbreakable Enterprise Kernel 4-3
- About the Oracle Database Preinstallation RPM 4-3
- Restrictions for HugePages and Transparent HugePages Configurations 4-4

Reviewing Operating System and Software Upgrade Best Practices 4-5
- General Upgrade Best Practices 4-5
- New Server Operating System Upgrade Option 4-6

Reviewing Operating System Security Common Practices 4-6

About Installation Fixup Scripts 4-7

About Operating System Requirements 4-7

Operating System Requirements for x86-64 Linux Platforms 4-8
- General Guidelines for Operating System Packages 4-8
- Supported Oracle Linux 8 Distributions for x86-64 4-9
- Supported Oracle Linux 7 Distributions for x86-64 4-11
- Supported Red Hat Enterprise Linux 8 Distributions for x86-64 4-13
- Supported SUSE Linux Enterprise Server 15 Distributions for x86-64 4-14
- Installing Operating System Packages 4-16

Additional Drivers and Software Packages for Linux 4-16
- Installing PAM for Login Authentication on Linux 4-17
- Installation Requirements for OCFS2 4-17
  - About OCFS2 and Shared Storage 4-17
  - Installing OCFS2 4-17
- Installing Oracle Messaging Gateway 4-18
- Installation Requirements for ODBC and LDAP 4-18
  - About ODBC Drivers and Oracle Database 4-18
  - Installing ODBC Drivers for Linux x86-64 4-19
  - About LDAP and Oracle Plug-ins 4-19
  - Installing the LDAP Package 4-19
- Installation Requirements for Programming Environments for Linux 4-19
  - Installation Requirements for Programming Environments for Linux x86-64 4-19
- Installation Requirements for Web Browsers 4-20

Checking Kernel and Package Requirements for Linux 4-21

Setting Clock Source for VMs on Linux x86-64 4-21

Installing the cvuqdisk RPM for Linux 4-22

Confirming Host Name Resolution 4-23

Disabling Transparent HugePages 4-23

Using Automatic SSH Configuration During Installation 4-25

Verifying the Disk I/O Scheduler on Linux 4-26

5 Configuring Users, Groups and Environments for Oracle Grid Infrastructure and Oracle Database

Required Operating System Groups and Users 5-1
Determining If an Oracle Inventory and Oracle Inventory Group Exist 5-2
Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist 5-3
About Oracle Installation Owner Accounts 5-3
Identifying an Oracle Software Owner User Account 5-3
Oracle Installations with Standard and Job Role Separation Groups and Users 5-4
About Oracle Installations with Job Role Separation 5-5
Standard Oracle Database Groups for Database Administrators 5-6
Extended Oracle Database Groups for Job Role Separation 5-6
Creating an ASMSNMP User 5-7
Oracle Automatic Storage Management Groups for Job Role Separation 5-7
Creating Operating System Privileges Groups 5-8
Creating the OSDBA for ASM Group 5-9
Creating the OSOPER for ASM Group 5-9
Creating the OSDBA Group for Database Installations 5-9
Creating an OSOPER Group for Database Installations 5-10
Creating the OSBACKUPDBA Group for Database Installations 5-10
Creating the OSDGDBA Group for Database Installations 5-10
Creating the OSKMDBA Group for Database Installations 5-11
Creating the OSRACDBA Group for Database Installations 5-11
Creating Operating System Oracle Installation User Accounts 5-11
Creating an Oracle Software Owner User 5-12
Environment Requirements for Oracle Software Owners 5-12
Procedure for Configuring Oracle Software Owner Environments 5-13
Modifying Oracle Owner User Groups 5-15
Checking Resource Limits for Oracle Software Installation Users 5-16
Setting Remote Display and X11 Forwarding Configuration 5-17
Preventing Installation Errors Caused by Terminal Output Commands 5-18
Creating Oracle Database Vault User Accounts 5-19
Unsetting Oracle Installation Owner Environment Variables 5-19

6 Configuring Networks for Oracle Database

About Oracle Database Network Configuration Options 6-1
About Assigning Global Database Names During Installation 6-2
Network Configuration for Computers Completed After Installation 6-3

Network Configuration for Multihome Computers 6-3
Setting the ORACLE_HOSTNAME Environment Variable 6-4
Network Configuration for Computers with Multiple Aliases 6-4

7 Supported Storage Options for Oracle Database and Oracle Grid Infrastructure

Supported Storage Options for Oracle Database 7-1
About Oracle Grid Infrastructure for a Standalone Server 7-2
Oracle ACFS and Oracle ADVM 7-3
- Oracle ACFS and Oracle ADVM Support on Linux x86–64 7-4
- Restrictions and Guidelines for Oracle ACFS 7-4

File System Options for Oracle Database 7-5
Guidelines for Placing Oracle Database Files On a File System or Logical Volume 7-6
About NFS Storage for Data Files 7-7
About Direct NFS Client Mounts to NFS Storage Devices 7-7

8 Configuring File System Storage for Oracle Database

Deploying Oracle Direct NFS 8-1
- Configuring NFS Buffer Size Parameters for Oracle Database 8-2
- Checking TCP Network Protocol Buffer for Direct NFS Client 8-2
- Creating an oranfstab File for Direct NFS Client 8-3
- Enabling and Disabling Direct NFS Client Control of NFS 8-6
- Enabling Hybrid Columnar Compression on Direct NFS Client 8-6

Deploying a DAX-Enabled File System for Persistent Memory Database 8-7
- About Using a DAX-Enabled File System for Persistent Memory Database 8-7
- Creating a DAX-Enabled File System for Persistent Memory Database 8-7

9 Configuring Storage for Oracle Grid Infrastructure for a Standalone Server

Configuring Storage for Oracle Automatic Storage Management 9-2
- Identifying Storage Requirements for Oracle Automatic Storage Management 9-2
- Oracle ASM Disk Space Requirements 9-5
- ASM Disk Group Options for Installation 9-6
- Using an Existing Oracle ASM Disk Group 9-6

Configuring Storage Device Path Persistence Using Oracle ASMFD 9-7
- About Oracle ASM with Oracle ASM Filter Driver 9-7
- Configuring Oracle ASM Filter Driver 9-8
  - Guidelines for Configuring Oracle ASM Filter Driver 9-8

Configuring Oracle ASM Filter Driver During Installation 9-8
Configuring Oracle ASM Filter Driver After Installation 9-9
Migrating Oracle ASM Disk Groups From ASMLIB to Oracle ASM Filter Driver 9-10
Determining If Oracle ASM Filter Driver Is Configured 9-11
Creating DAS or SAN Disk Partitions for Oracle Automatic Storage Management 9-11
Creating Directories for Oracle Database Files 9-12
Creating Files on a NAS Device for Use with Oracle Automatic Storage Management 9-13

10 Installing and Configuring Oracle Grid Infrastructure for a Standalone Server

About Image-Based Oracle Grid Infrastructure Installation 10-2
Setup Wizard Installation Options for Creating Images 10-3
Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation 10-4
Installing Oracle Grid Infrastructure for a Standalone Server for an Existing Database 10-6
Installing Oracle Grid Infrastructure for a Standalone Server Using a Software-Only Installation 10-8
About Oracle Grid Infrastructure Software-Only Installations 10-8
Installing Software Binaries for Oracle Grid Infrastructure for a Standalone Server 10-8
Configuring Software Binaries for Oracle Grid Infrastructure for a Standalone Server 10-9
Testing the Oracle Automatic Storage Management Installation 10-10
Modifying Oracle Grid Infrastructure for a Standalone Server Binaries After Installation 10-11
Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA 10-13
Enabling Oracle ACFS on Oracle Restart Configurations 10-13
Applying Patches During an Oracle Grid Infrastructure Installation or Upgrade 10-13
Patching and Switching Oracle Grid Infrastructure Homes 10-14

11 Installing Oracle Database

About Image-Based Oracle Database Installation 11-2
About Deploying Oracle Database Using Oracle Fleet Patching and Provisioning 11-2
Downloading Oracle Software 11-3
Downloading the Installation Archive Files from the Oracle Database Website 11-3
Downloading the Software from Oracle Software Delivery Cloud Portal 11-4
About Character Set Selection During Installation 11-4
About Automatic Memory Management Installation Options 11-5
Running the Installer in a Different Language 11-6
Installing the Oracle Database Software 11-7
Setup Wizard Installation Options for Creating Images 11-7
Applying Patches During an Oracle Database Installation or Upgrade 11-8
Running Oracle Database Setup Wizard to Install Oracle Database 11-8

Installing Oracle Database Using RPM Packages 11-10
About RPM-Based Oracle Database Installation 11-10
Restrictions and Guidelines for RPM-Based Installations 11-11
RPM Packages Naming Convention 11-11
Running RPM Packages to Install Oracle Database 11-12
Installing Standard Edition High Availability 11-14
About Standard Edition High Availability 11-14
Requirements for Installing Standard Edition High Availability 11-14
Deploying Standard Edition High Availability 11-15
Installing Standard Edition High Availability Database Software on Local File System 11-16
Installing Standard Edition High Availability Database Software on Oracle ACFS 11-18
About Creating Databases Using Oracle DBCA 11-20

12 Configuring Oracle Homes

Evolution of Oracle Homes 12-1
About Read-Only Oracle Homes 12-1
About Oracle Base Home 12-2
About Oracle Base Config 12-3
About orabasetab 12-3
Copying demo Directories to Oracle Base Home 12-4
File Path and Directory Changes in Read-Only Oracle Homes 12-7

13 Oracle Database Postinstallation Tasks

Required Postinstallation Tasks 13-2
Downloading Release Update Patches 13-2
Unlocking and Resetting Oracle Database User Passwords 13-3
Requirements for Database Passwords 13-3
Oracle Database System Privileges Accounts and Passwords 13-3
Guidelines for Changing System Privileges Account Passwords 13-6
Locking and Unlocking User Accounts 13-7
Using SQL*Plus to Unlock Accounts and Reset Passwords 13-7
Recommended Postinstallation Tasks 13-8
Creating a Backup of the root.sh Script 13-8
Setting Language and Locale Preferences for Client Connections 13-8
Recompiling All Invalid Objects 13-10
About Installing Oracle Autonomous Health Framework 13-11
About Changes in Default SGA Permissions for Oracle Database 13-11
Checking Installed Oracle Database Contents and Directory Location 13-12
Enabling and Disabling Oracle Database Options After Installation 13-12

Chopt Tool 13-13
Starting Oracle Enterprise Manager Database Express 13-14
Creating a Fast Recovery Area 13-14
About the Fast Recovery Area and the Fast Recovery Area Disk Group 13-14
Creating the Fast Recovery Area Disk Group 13-15
Cloning Oracle Database 13-15
Cloning an Oracle Database Home Using an Image File 13-16
Cloning an Oracle Database Home Using clone.pl 13-17

14 Removing Oracle Database Software

About Oracle Deinstallation Options 14-1
Oracle Deinstallation (Deinstall) 14-3
Deinstallation Examples for Oracle Database 14-5
Downgrading Oracle Restart 14-5
Removing an RPM-based Oracle Database Installation 14-9
Deinstalling Previous Release Grid Home 14-9

A Completing Preinstallation Tasks Manually

Configuring Kernel Parameters for Linux A-1
Minimum Parameter Settings for Installation A-1
Changing Kernel Parameter Values A-3
Configuring Additional Kernel Settings for SUSE Linux A-4
Setting UDP and TCP Kernel Parameters Manually A-5
Configuring Default Thread Limits Value for SUSE Linux A-6
Configuring Storage Device Path Persistence Using Oracle ASMLIB A-6
About Oracle ASM with Oracle ASMLIB A-7
Installing and Configuring Oracle ASMLIB Software A-7
Configuring Disk Devices to Use Oracle ASMLIB A-9
Administering Oracle ASMLIB and Disks A-11
Deinstalling Oracle ASMLIB On Oracle Database A-13
Configuring Storage Device Path Persistence Manually A-13
Configuring Device Persistence Manually for Oracle ASM A-14

B Installing and Configuring Oracle Database Using Response Files

Modes for Oracle DBCA Noninteractive (Silent) Configuration B-1
Reasons for Using Silent Mode or Response File Mode B-2
Using Response Files B-2
Preparing Response Files B-3
Editing a Response File Template B-3

Recording Response Files B-4
Running Oracle Universal Installer Using a Response File B-5
Running Configuration Assistants Using Response Files B-7
Running Net Configuration Assistant Using Response Files B-7
Running Oracle DBCA Using Response Files B-8
Postinstallation Configuration Using Response File Created During Installation B-9
Using the Installation Response File for Postinstallation Configuration B-9
Running Postinstallation Configuration Using Response File B-11

C Optimal Flexible Architecture

About the Optimal Flexible Architecture Standard C-1
About Multiple Oracle Homes Support C-2
About the Oracle Inventory Directory and Installation C-3
Oracle Base Directory Naming Convention C-4
Oracle Home Directory Naming Convention C-5
Optimal Flexible Architecture File Path Examples C-5

D Managing Oracle Database Port Numbers

About Managing Ports D-1
Oracle Database Component Port Numbers and Protocols D-1

Preface

This guide explains how to install and configure single-instance Oracle Database.

This guide also provides information about Optimal Flexible Architecture, cloning an Oracle home, and how to remove the database software.

- Audience
  This guide is intended for anyone responsible for installing Oracle Database 21c.
- Documentation Accessibility
- Diversity and Inclusion
- Set Up Java Access Bridge to Implement Java Accessibility
  Install Java Access Bridge so that assistive technologies on Microsoft Windows systems can use the Java Accessibility API.
- Command Syntax
  Refer to these command syntax conventions to understand command examples in this guide.
- Conventions

Audience

This guide is intended for anyone responsible for installing Oracle Database 21c.

Additional installation guides for Oracle Database, Oracle Real Application Clusters, Oracle Clusterware, Oracle Database Examples, and Oracle Enterprise Manager Cloud Control are available at the following URL:

http://docs.oracle.com

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to

build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Set Up Java Access Bridge to Implement Java Accessibility

Install Java Access Bridge so that assistive technologies on Microsoft Windows systems can use the Java Accessibility API.

Java Access Bridge is a technology that enables Java applications and applets that implement the Java Accessibility API to be visible to assistive technologies on Microsoft Windows systems.

Refer to *Java Platform, Standard Edition Accessibility Guide* for information about the minimum supported versions of assistive technologies required to use Java Access Bridge. Also refer to this guide to obtain installation and testing instructions, and instructions for how to use Java Access Bridge.

Related Topics

- *Java Platform, Standard Edition Java Accessibility Guide*

Command Syntax

Refer to these command syntax conventions to understand command examples in this guide.

Convention	Description
`$`	Bourne or BASH shell prompt in a command example. Do not enter the prompt as part of the command.
`%`	C Shell prompt in a command example. Do not enter the prompt as part of the command.
`#`	Superuser (root) prompt in a command example. Do not enter the prompt as part of the command.
`monospace`	UNIX command syntax

backslash \	A backslash is the UNIX and Linux command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash:
	
	`dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \`
	`count=10000`

braces { }	Braces indicate required items: `.DEFINE {macro1}`

Convention	Description
brackets [ ]	Brackets indicate optional items: `cvtcrt termname [outfile]`
ellipses ...	Ellipses indicate an arbitrary number of similar items: `CHKVAL fieldname value1 value2 ... valueN`
italic	Italic type indicates a variable. Substitute a value for the variable: *`library_name`*
vertical line |	A vertical line indicates a choice within braces or brackets: `FILE filesize [K|M]`

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1
Oracle Database Installation Checklist

Use checklists to review system requirements, and to plan and carry out Oracle Database installation.

Oracle recommends that you use checklists as part of your installation planning process. Using checklists can help you to confirm that your server hardware and configuration meet minimum requirements for this release, and can help you to ensure you carry out a successful installation.

- Server Hardware Checklist for Oracle Database Installation
  Use this checklist to check hardware requirements for Oracle Database.
- Operating System Checklist for Oracle Database Installation on Linux
  Use this checklist to check minimum operating system requirements for Oracle Database.
- Server Configuration Checklist for Oracle Database Installation
  Use this checklist to check minimum server configuration requirements for Oracle Database installations.
- Oracle User Environment Configuration Checklist for Oracle Database Installation
  Use this checklist to plan operating system users, groups, and environments for Oracle Database management.
- Storage Checklist for Oracle Database Installation
  Use this checklist to review storage minimum requirements and assist with configuration planning.
- Installer Planning Checklist for Oracle Database
  Use this checklist to assist you to be prepared before starting Oracle Universal Installer.
- Deployment Checklist for Oracle Database
  Use this checklist to decide the deployment method for a single-instance Oracle Database.

Server Hardware Checklist for Oracle Database Installation

Use this checklist to check hardware requirements for Oracle Database.

Table 1-1 Server Hardware Checklist for Oracle Database Installation

Check	Task
Server Make and Architecture	Confirm that server make, model, core architecture, and host bus adaptors (HBA) or network interface controllers (NICs) are supported to run with Oracle Database and Oracle Grid Infrastructure.
Runlevel	3 or 5
Server Display Cards	At least 1024 x 768 display resolution, which Oracle Universal Installer requires.
Minimum network connectivity	Server is connected to a network

Table 1-1 (Cont.) Server Hardware Checklist for Oracle Database Installation

Check	Task
Minimum RAM	• At least 1 GB RAM for Oracle Database installations. 2 GB RAM recommended. • At least 8 GB RAM for Oracle Grid Infrastructure installations.

Operating System Checklist for Oracle Database Installation on Linux

Use this checklist to check minimum operating system requirements for Oracle Database.

Table 1-2 Operating System General Checklist for Oracle Database on Linux

Item	Task
Operating system general requirements	OpenSSH installed manually, if you do not have it installed already as part of a default Linux installation. A Linux kernel in the list of supported kernels and releases listed in this guide.
Linux x86-64 operating system requirements	The following Linux x86-64 kernels are supported: • Oracle Linux 8.2 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.1.2.el8uek.x86_64 or later Oracle Linux 8.2 with the Red Hat Compatible Kernel: 4.18.0-193.19.1.el8_2.x86_64 or later • Oracle Linux 7.6 with the Unbreakable Enterprise Kernel 5: 4.14.35-2025.404.1.el7uek.x86_64 or later Oracle Linux 7.4 with the Unbreakable Enterprise Kernel 4: 4.1.12-124.53.1.el7uek.x86_64 or later • Red Hat Enterprise Linux 8.2: 4.18.0-193.19.1.el8_2.x86_64 or later • SUSE Linux Enterprise Server 15 SP1: 4.12.14-197.29-default or later Review the system requirements section for a list of minimum package requirements.
Oracle Database Preinstallation RPM for Oracle Linux	If you use Oracle Linux, then Oracle recommends that you run the Oracle Database Preinstallation RPM for your Linux release to configure your operating system for Oracle Database and Oracle Grid Infrastructure installations.
Disable Transparent HugePages	Oracle recommends that you disable Transparent HugePages and use standard HugePages for enhanced performance.

Related Topics

- Automatically Configuring Oracle Linux with Oracle Database Preinstallation RPM
  Use Oracle Database Preinstallation RPM to simplify operating system configuration in preparation for Oracle software installations.
- Operating System Requirements for x86-64 Linux Platforms
  The Linux distributions and packages listed in this section are supported for this release on x86-64.

- Disabling Transparent HugePages
  Oracle recommends that you disable Transparent HugePages before you start installation.

Server Configuration Checklist for Oracle Database Installation

Use this checklist to check minimum server configuration requirements for Oracle Database installations.

Table 1-3 Server Configuration Checklist for Oracle Database

Check	Task
Disk space allocated to the `/tmp` directory	At least 1 GB of space in the `/tmp` directory.
Swap space allocation relative to RAM (Oracle Database)	Between 1 GB and 2 GB: 1.5 times the size of the RAM Between 2 GB and 16 GB: Equal to the size of the RAM More than 16 GB: 16 GB **Note:** If you enable HugePages for your Linux servers, then you should deduct the memory allocated to HugePages from the available RAM before calculating swap space.
Swap space allocation relative to RAM (Oracle Restart)	Between 8 GB and 16 GB: Equal to the size of the RAM More than 16 GB: 16 GB **Note:** If you enable HugePages for your Linux servers, then you should deduct the memory allocated to HugePages from the available RAM before calculating swap space.
Oracle Inventory (oraInventory) and OINSTALL Group Requirements	• For upgrades, the installer detects an existing oraInventory directory from the `/etc/oraInst.loc` file, and uses the existing oraInventory. • For new installs, if you have not configured an oraInventory directory, then you can specify the oraInventory directory during the software installation and Oracle Universal Installer will set up the software directories for you. The Oracle inventory is one directory level up from the Oracle base for the Oracle software installation and designates the installation owner's primary group as the Oracle inventory group. Ensure that the oraInventory path that you specify is in compliance with the Oracle Optimal Flexible Architecture recommendations. The Oracle Inventory directory is the central inventory of Oracle software installed on your system. Users who have the Oracle Inventory group as their primary group are granted the OINSTALL privilege to write to the central inventory. The OINSTALL group must be the primary group of all Oracle software installation owners on the server. It should be writable by any Oracle installation owner.
Groups and users	Oracle recommends that you create groups and user accounts required for your security plans before starting installation. Installation owners have resource limits settings and other requirements. Group and user names must use only ASCII characters.
Mount point paths for the software binaries	Oracle recommends that you create an Optimal Flexible Architecture configuration as described in the appendix "Optimal Flexible Architecture" in *Oracle Database Installation Guide* for your platform.

Table 1-3 (Cont.) Server Configuration Checklist for Oracle Database

Check	Task
Ensure that the Oracle home (the Oracle home path you select for Oracle Database) uses only ASCII characters	The ASCII character restriction includes installation owner user names, which are used as a default for some home paths, as well as other directory names you may select for paths.
Unset Oracle software environment variables	If you have an existing Oracle software installation, and you are using the same user to install this installation, then unset the following environment variables: `$ORACLE_HOME`,`$ORA_NLS10`, and `$TNS_ADMIN`. If you have set `$ORA_CRS_HOME` as an environment variable, then unset it before starting an installation or upgrade. Do not use `$ORA_CRS_HOME` as a user environment variable, except as directed by Oracle Support.
Set locale (if needed)	Specify the language and the territory, or locale, in which you want to use Oracle components. A locale is a linguistic and cultural environment in which a system or program is running. NLS (National Language Support) parameters determine the locale-specific behavior on both servers and clients. The locale setting of a component determines the language of the user interface of the component, and the globalization behavior, such as date and number formatting.
Check Shared Memory File System Mount	By default, your operating system includes an entry in `/etc/fstab` to mount `/dev/shm`. However, if your Cluster Verification Utility (CVU) or installer checks fail, ensure that the `/dev/shm` mount area is of type `tmpfs` and is mounted with the following options: • `rw` and `exec` permissions set on it • Without `noexec` or `nosuid` set on it **Note:** These options may not be listed as they are usually set as the default permissions by your operating system.
Symlinks	Oracle home or Oracle base cannot be `symlinks`, nor can any of their parent directories, all the way to up to the `root` directory.

Related Topics

- Checking and Configuring Server Hardware for Oracle Database
  Verify that servers where you install Oracle Database meet the minimum requirements for installation.
- Configuring Users, Groups and Environments for Oracle Grid Infrastructure and Oracle Database
  Before installation, create operating system groups and users, and configure user environments.
- *Oracle Database Globalization Support Guide*

Oracle User Environment Configuration Checklist for Oracle Database Installation

Use this checklist to plan operating system users, groups, and environments for Oracle Database management.

Table 1-4 User Environment Configuration for Oracle Database

Check	Task
Review Oracle Inventory (oraInventory) and OINSTALL Group Requirements	The physical group you designate as the Oracle Inventory directory is the central inventory of Oracle software installed on your system. It should be the primary group for all Oracle software installation owners. Users who have the Oracle Inventory group as their primary group are granted the OINSTALL privilege to read and write to the central inventory. • If you have an existing installation, then OUI detects the existing oraInventory directory from the `/etc/oraInst.loc` file, and uses this location. • If you are installing Oracle software for the first time, then you can specify the Oracle inventory directory and the Oracle base directory during the Oracle software installation, and Oracle Universal Installer will set up the software directories for you. Ensure that the directory paths that you specify are in compliance with the Oracle Optimal Flexible Architecture recommendations. Ensure that the group designated as the OINSTALL group is available as the primary group for all planned Oracle software installation owners.
Create operating system groups and users for standard or role-allocated system privileges	Create operating system groups and users depending on your security requirements, as described in this install guide. Set resource limits settings and other requirements for Oracle software installation owners. Group and user names must use only ASCII characters.
Unset Oracle Software Environment Variables	If you have had an existing installation on your system, and you are using the same user account for this installation, then unset the ORACLE_HOME, ORACLE_BASE, ORACLE_SID, TNS_ADMIN environment variables and any other environment variable set for the Oracle installation user that is connected with Oracle software homes.
Configure the Oracle Software Owner Environment	Configure the environment of the `oracle` or `grid` user by performing the following tasks: • Set the default file mode creation mask (`umask`) to 022 in the shell startup file. • Set the `DISPLAY` environment variable.

Related Topics

- Optimal Flexible Architecture
  Oracle Optimal Flexible Architecture (OFA) rules are a set of configuration guidelines created to ensure well-organized Oracle installations, which simplifies administration, support and maintenance.

Related Topics

- Configuring Users, Groups and Environments for Oracle Grid Infrastructure and Oracle Database
  Before installation, create operating system groups and users, and configure user environments.

Storage Checklist for Oracle Database Installation

Use this checklist to review storage minimum requirements and assist with configuration planning.

Table 1-5 Storage Checklist for Oracle Database

Check	Task
Minimum local disk storage space for Oracle software	For Linux x86-64: At least 6.0 GB for an Oracle Grid Infrastructure for a standalone server installation. At least 7.8 GB for Oracle Database Enterprise Edition. At least 7.8 GB for Oracle Database Standard Edition 2. **Note:** Oracle recommends that you allocate approximately 100 GB to allow additional space for applying any future patches on top of the existing Oracle home. For specific patch-related disk space requirements, please refer to your patch documentation.
Select Database File Storage Option	Ensure that you have one of the following storage options available: • File system mounted on the server. Oracle recommends that the file system you select is separate from the file system used by the operating system or the Oracle software. Options include the following: – A file system on a logical volume manager (LVM) volume or a RAID device – A network file system (NFS) mounted on a certified network-attached storage (NAS) device • Oracle Automatic Storage Management (Oracle ASM). Oracle ASM is installed as part of an Oracle Grid Infrastructure installation. If you plan to use Oracle ASM for storage, then you should install Oracle Grid Infrastructure before you install and create the database.
Determine your recovery plan	If you want to enable recovery during installation, then be prepared to select one of the following options: • **File system**: Configure a fast recovery area on a file system during installation • **Oracle Automatic Storage Management**: Configure a fast recovery area disk group using Oracle ASMCA. Review the storage configuration sections of this document for more information about configuring recovery.

Related Topics

- Configuring File System Storage for Oracle Database
  Complete these procedures to use file system storage for Oracle Database.

Installer Planning Checklist for Oracle Database

Use this checklist to assist you to be prepared before starting Oracle Universal Installer.

Table 1-6 Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Read the Release Notes	Review release notes for your platform, which are available for your release at the following URL: http://docs.oracle.com/en/database/database.html
Review Oracle Support Certification Matrix	New platforms and operating system software versions may be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions: https://support.oracle.com/ You must register online before using My Oracle Support. After logging in, from the menu options, select the **Certifications** tab. On the Certifications page, use the **Certification Search** options to search by Product, Release, and Platform. You can also search using the **Certification Quick Link** options such as Product Delivery, and Lifetime Support.
Review the Licensing Information	You are permitted to use only those components in Oracle Database for which you have purchased licenses. For more information about licenses, refer to the following URL: *Oracle Database Licensing Information*
Run OUI with CVU and use fixup scripts	Oracle Universal Installer is fully integrated with Cluster Verification Utility (CVU), automating many CVU prerequisite checks. Oracle Universal Installer runs all prerequisite checks and creates fixup scripts when you run the installer. You can run OUI up to the Summary screen without starting the installation. You can also run CVU commands manually to check system readiness. For more information, see: *Oracle Clusterware Administration and Deployment Guide*

Table 1-6 (Cont.) Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task

Update and run Oracle ORAchk for runtime and upgrade checks, or runtime health checks	Oracle recommends that you update to the latest version of Oracle ORAchk by installing the latest version of Autonomous Health Framework (AHF).
	The Oracle ORAchk utility provides system checks that can help to prevent issues before and after installation. These checks include kernel requirements, operating system resource allocations, and other system requirements.
	Use the Oracle ORAchk Upgrade Readiness Assessment to obtain an automated upgrade-specific system health check for upgrades to 21c. For example:
	• Before you perform a fresh database installation:
	`./orachk -profile preinstall`
	• To upgrade your existing database to a higher version or release:
	`./orachk -pregupgrade -targetversion 21.0.0.0.0`
	The Oracle ORAchk Upgrade Readiness Assessment automates many of the manual pre- and post-upgrade checks described in Oracle upgrade documentation. Check My Oracle Support Note 2550798.1 for more information about Oracle ORAchk support.
	https://support.oracle.com/epmos/faces/DocContentDisplay?id=2550798.1&parent=DOCUMENTATION&sourceId=USERGUIDE

Verify if Oracle Grid Infrastructure is installed	For Oracle Real Application Clusters (Oracle RAC) installations, ensure that you have installed and configured Oracle Grid Infrastructure for a cluster. If you want to use Oracle ASM or Oracle Restart, then install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, to use Oracle ASM, you must complete an Oracle Grid Infrastructure installation, and then manually register the database with Oracle Restart.
Check running Oracle processes, and shut down if necessary	• On a standalone database not using Oracle ASM: You do not need to shut down the database while you install Oracle Grid Infrastructure. • On a standalone database using Oracle ASM: The Oracle ASM instances are restarted during installation. • On an Oracle RAC Database node: This installation requires an upgrade of Oracle Clusterware, as Oracle Clusterware is required to run Oracle RAC. As part of the upgrade, you must shut down the database one node at a time as the rolling upgrade proceeds from node to node.
Ensure `cron` jobs do not run during installation	If the installer is running when daily `cron` jobs start, then you may encounter unexplained installation problems if your `cron` job is performing cleanup, and temporary files are deleted before the installation is finished. Oracle recommends that you complete installation before daily `cron` jobs are run, or disable daily `cron` jobs that perform cleanup until after the installation is completed.

Table 1-6 (Cont.) Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Obtain your My Oracle Support account information.	During installation, you may require a My Oracle Support user name and password to configure updates, download software updates, and other installation tasks. You can register for My Oracle Support at the following URL: https://support.oracle.com/
Decide Oracle Database management tool	By default, Oracle Database is managed by Oracle Enterprise Manager Database Express. If you have an existing Oracle Management Agent, and decide to use Oracle Enterprise Manager Cloud Control to centrally manage your database, then obtain the following information to enter during the database installation: • OMS host • OMS port • EM admin username • EM admin password • Specify password of ASMSNMP user For more information, see: • *Oracle Enterprise Manager Cloud Control Administrator's Guide*
Review memory allocation and Automatic Memory Management feature	You can enable automatic memory management either during, or after Oracle Database installation. If you enable automatic memory management after installation, then you must shut down and restart the database. If the total physical memory of your database instance is greater than 4 GB, then you cannot select the Oracle Automatic Memory Management option during database installation and creation. Instead, use automatic shared memory management. Automatic shared memory management automatically distributes the available memory among the various components as required, allowing the system to maximize the use of all available SGA memory. For more information, see: *Oracle Database Administrator's Guide*

Table 1-6 (Cont.) Oracle Universal Installer Planning Checklist for Oracle Database Installation

Check	Task
Determine superuser (`root`) privilege delegation option for installation	During a database or grid infrastructure installation, you are asked to run configuration scripts as the `root` user. You can either run these scripts manually as `root` when prompted, or you can provide configuration information and passwords using a root privilege delegation option. To run `root` scripts automatically, select **Automatically run configuration scripts** during installation. • **Use root user credentials** Provide the superuser password for cluster member node servers. • **Use Sudo** Sudo is a UNIX and Linux utility that allows members of the sudoers list privileges to run individual commands as `root`. Provide the username and password of an operating system user that is a member of sudoers, and is authorized to run Sudo on each cluster member node. To enable Sudo, have a system administrator with the appropriate privileges configure a user that is a member of the sudoers list, and provide the username and password when prompted during installation.
Oracle Database Client and Oracle Database interoperability	For information about interoperability between Oracle Database Client and Oracle Database releases, see My Oracle Support Note 207303.1: https://support.oracle.com/rs?type=doc&id=207303.1

Related Topics

- Installing Oracle Database
  Oracle Database and Oracle Grid Infrastructure installation software is available as image-based zip files and can be installed using several options.
- Installing and Configuring Oracle Grid Infrastructure for a Standalone Server
  Oracle Grid Infrastructure for a standalone server includes Oracle Restart and Oracle Automatic Storage Management.

Deployment Checklist for Oracle Database

Use this checklist to decide the deployment method for a single-instance Oracle Database.

Table 1-7 Deployment Checklist for Oracle Database (single-instance)

Item	Task
To deploy single-instance Oracle Database software	Use one of the following deployment methods: • Install Oracle Database software using Oracle Universal Installer (OUI). • Provision Oracle Database software using Oracle Fleet Patching and Provisioning. • Clone Oracle Database.

Table 1-7 (Cont.) Deployment Checklist for Oracle Database (single-instance)

Item	Task
To deploy single-instance Oracle Database software and create databases	Use one of the following deployment methods: • Install Oracle Database software using Oracle Universal Installer (OUI). • Provision Oracle Database software using Oracle Fleet Patching and Provisioning. • Clone Oracle Database.
To create single-instance Oracle Database in an already-installed Oracle home	• Use Oracle Database Configuration Assistant (Oracle DBCA). • Use Oracle Fleet Patching and Provisioning

Related Topics

- Installing Oracle Database
  Oracle Database and Oracle Grid Infrastructure installation software is available as image-based zip files and can be installed using several options.
- About Deploying Oracle Database Using Oracle Fleet Patching and Provisioning
  You can use Oracle Fleet Patching and Provisioning (Oracle FPP) to provision Oracle Database software.
- *Oracle Database Administrator's Guide*

> **See Also:**
>
> *Oracle Clusterware Administration and Deployment Guide* for more information about Oracle Fleet Patching and Provisioning

2

Checking and Configuring Server Hardware for Oracle Database

Verify that servers where you install Oracle Database meet the minimum requirements for installation.

This section provides minimum server requirements to complete installation of Oracle Database. It does not provide system resource guidelines, or other tuning guidelines for particular workloads.

- Logging In to a Remote System Using X Window System
  Use this procedure to run Oracle Universal Installer (OUI) by logging on to a remote system where the runtime setting prohibits logging in directly to a graphical user interface (GUI).
- Checking Server Hardware and Memory Configuration
  Use this procedure to gather information about your server configuration.

Logging In to a Remote System Using X Window System

Use this procedure to run Oracle Universal Installer (OUI) by logging on to a remote system where the runtime setting prohibits logging in directly to a graphical user interface (GUI).

OUI is a graphical user interface (GUI) application. On servers where the runtime settings prevent GUI applications from running, you can redirect the GUI display to a client system connecting to the server.

> **Note:**
>
> If you log in as another user (for example, `oracle` or `grid`), then repeat this procedure for that user as well.

1. Start an X Window System session. If you are using an X Window System terminal emulator from a PC or similar system, then you may need to configure security settings to permit remote hosts to display X applications on your local system.
2. Enter a command using the following syntax to enable remote hosts to display X applications on the local X server:

   ```
   # xhost + RemoteHost
   ```

   RemoteHost is the fully qualified remote host name. For example:

   ```
   # xhost + somehost.example.com
   somehost.example.com being added to the access control list
   ```

3. If you are not installing the software on the local system, then use the `ssh` command to connect to the system where you want to install the software:

   ```
   # ssh -Y  RemoteHost
   ```

   RemoteHost is the fully qualified remote host name. The `-Y` flag ("yes") enables remote X11 clients to have full access to the original X11 display. For example:

   ```
   # ssh -Y somehost.example.com
   ```

4. If you are not logged in as the `root` user, and you are performing configuration steps that require `root` user privileges, then switch the user to `root`.

> **Note:**
>
> For more information about remote login using X Window System, refer to your X server documentation, or contact your X server vendor or system administrator. Depending on the X server software that you are using, you may have to complete the tasks in a different order.

Checking Server Hardware and Memory Configuration

Use this procedure to gather information about your server configuration.

1. Use the following command to determine physical RAM size on the server:

   ```
   # grep MemTotal /proc/meminfo
   ```

   If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.

2. Determine the size of the configured swap space:

   ```
   # grep SwapTotal /proc/meminfo
   ```

   If necessary, see your operating system documentation for information about how to configure additional swap space.

3. Determine the amount of space available in the `/tmp` directory:

   ```
   # df -h /tmp
   ```

   If the free space available in the `/tmp` directory is less than what is required, then complete one of the following steps:

   - Delete unused files from the `/tmp` directory to meet the disk space requirement.

> **Note:**
>
> If you perform this step after installing Oracle software, then do not remove `/tmp/.oracle` or `/var/tmp/.oracle` directories or their files.

- When you set the Oracle user's environment, also set the `TMP` and `TMPDIR` environment variables to the directory you want to use instead of `/tmp`.

4. Determine the amount of free RAM and disk swap space on the system:

   ```
   # free
   ```

5. Determine if the system architecture can run the software:

   ```
   # uname -m
   ```

   Verify that the processor architecture matches the Oracle software release to install. For example, you should see the following for a x86-64 bit system:

   ```
   x86_64
   ```

   If you do not see the expected output, then you cannot install the software on this system.

6. Verify that shared memory (`/dev/shm`) is mounted properly with sufficient size:

   ```
   df -h /dev/shm
   ```

   The `df-h` command displays the filesystem on which `/dev/shm` is mounted, and also displays in GB the total size and free size of shared memory.

Related Topics

- Server Hardware Checklist for Oracle Database Installation
  Use this checklist to check hardware requirements for Oracle Database.

3

Automatically Configuring Oracle Linux with Oracle Database Preinstallation RPM

Use Oracle Database Preinstallation RPM to simplify operating system configuration in preparation for Oracle software installations.

Oracle recommends that you install Oracle Linux and use Oracle Database Preinstallation RPM to configure your operating systems for Oracle Database and Oracle Grid Infrastructure installations.

- About the Oracle Database Preinstallation RPM
  If your Linux distribution is Oracle Linux, or Red Hat Enterprise Linux, and you are an Oracle Linux support customer, then you can complete most preinstallation configuration tasks by using the Oracle Database Preinstallation RPM for your release.
- Overview of Oracle Linux Configuration with Oracle Database Preinstallation RPM
  Use Oracle Database Preinstallation RPM to simplify operating system configuration, and to ensure that you have required kernel packages.
- Installing the Oracle Database Preinstallation RPM Using ULN
  Use this procedure to subscribe to Unbreakable Linux Network (ULN) Oracle Linux channels for your Oracle software.
- Installing Oracle Database Preinstallation RPM During an Oracle Linux Installation
  Use this procedure to install a new Oracle Linux installation and to perform system configuration with the Oracle Preinstallation RPM:
- Installing Oracle Oracle Database Preinstallation RPM Using the Oracle Linux yum Server
  Install Oracle Linux and configure your Oracle Linux installation for security errata or bug fix updates using the Oracle Linux yum server.
- Configure Additional Operating System Features
  Oracle recommends that you configure your operating system before starting installation with additional features, such as IPMI or additional programming environments.

About the Oracle Database Preinstallation RPM

If your Linux distribution is Oracle Linux, or Red Hat Enterprise Linux, and you are an Oracle Linux support customer, then you can complete most preinstallation configuration tasks by using the Oracle Database Preinstallation RPM for your release.

Oracle Database Preinstallation RPMs are available from the Oracle Linux Network or available on the Oracle Linux DVDs. Using the Oracle Database Preinstallation RPM is not required, but Oracle recommends you use it to save time in setting up your cluster servers.

When installed, the Oracle Database Preinstallation RPM does the following:

- Automatically downloads and installs any additional RPM packages needed for installing Oracle Grid Infrastructure and Oracle Database, and resolves any dependencies

- Creates an `oracle` user, and creates the oraInventory (`oinstall`) and OSDBA (`dba`) groups for that user
- As needed, sets `sysctl.conf` settings, system startup parameters, and driver parameters to values based on recommendations from the Oracle Database Preinstallation RPM program
- Sets hard and soft resource limits
- Sets other recommended parameters, depending on your kernel version
- Sets `numa=off` in the kernel for Linux x86_64 machines.

Configure Oracle Database Preinstallation RPM only once on your operating system when you install Oracle Database or Oracle Grid Infrastructure for the first time on your system. For subsequent installations on the same system, do not install Oracle Database Preinstallation RPM again.

Do not install Oracle Database Preinstallation RPM on Oracle Engineered Systems, such as Oracle Exadata Database Machine. Oracle Engineered Systems include integrated system software that contain the required version of the operating system kernel and all software packages.

Note:

- The Oracle Database Preinstallation RPM designated for each Oracle Database release sets kernel parameters and resource limits only for the user account `oracle`. To use multiple software account owners, you must perform system configuration for other accounts manually.
- The Oracle Database Preinstallation RPM installs the X11 client libraries, but it does not install the X Window System server packages. To use graphical user interfaces such as OUI, configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

Overview of Oracle Linux Configuration with Oracle Database Preinstallation RPM

Use Oracle Database Preinstallation RPM to simplify operating system configuration, and to ensure that you have required kernel packages.

Oracle recommends that you install Oracle Linux and use Oracle Database Preinstallation RPM to configure your operating systems for Oracle Database and Oracle Grid Infrastructure installations.

Configure Oracle Database Preinstallation RPM only once on your operating system when you install Oracle Database or Oracle Grid Infrastructure for the first time on your system. For subsequent installations on the same system, do not install Oracle Database Preinstallation RPM again.

Do not install Oracle Database Preinstallation RPM on Oracle Engineered Systems, such as Oracle Exadata Database Machine. Oracle Engineered Systems include

integrated system software that contain the required version of the operating system kernel and all software packages.

The Oracle Database Preinstallation RPM for your Oracle Linux distributions and database release automatically installs any additional packages needed for installing Oracle Grid Infrastructure and Oracle Database, and configures your server operating system automatically, including setting kernel parameters and other basic operating system requirements for installation. For more information about Oracle Linux and Oracle Database Preinstallation RPM, refer to:

http://docs.oracle.com/en/operating-systems/linux.html

Configuring a server using Oracle Linux and the Oracle Database Preinstallation RPM consists of the following steps:

1. Install Oracle Linux.
2. Register your Linux distribution with Oracle Unbreakable Linux Network (ULN) or download and configure the yum repository for your system using the Oracle Linux yum server for your Oracle Linux release.
3. Install the Oracle Database Preinstallation RPM with the RPM for your Oracle Grid Infrastructure and Oracle Database releases, and update your Linux release.
4. Create role-allocated groups and users with identical names and ID numbers.
5. Complete network interface configuration for each cluster node candidate.
6. Complete system configuration for shared storage access as required for each standard or core node cluster candidate.

After these steps are complete, you can proceed to install Oracle Grid Infrastructure and Oracle Database.

Installing the Oracle Database Preinstallation RPM Using ULN

Use this procedure to subscribe to Unbreakable Linux Network (ULN) Oracle Linux channels for your Oracle software.

To obtain Unbreakable Linux Network (ULN) support, subscribe to Oracle Linux channels, and to add the Oracle Linux channel that distributes the Oracle Database Preinstallation RPM:

1. Download the Oracle Linux ISO from one of the following websites:
   - Oracle yum

     https://yum.oracle.com/oracle-linux-isos.html

   - Oracle Software Delivery Cloud website:

     https://edelivery.oracle.com/linux

> **Note:**
>
> Ensure that you use the latest available update release for Oracle Linux.

2. Register your server with Unbreakable Linux Network (ULN). By default, you are registered for the Oracle Linux Latest channel for your operating system and hardware.
   - Oracle Linux 7
     https://docs.oracle.com/en/operating-systems/oracle-linux/uln-user/
   - Oracle Linux 8
     https://docs.oracle.com/en/operating-systems/oracle-linux/8/software-management/
3. Log in to Unbreakable Linux Network:

   https://linux.oracle.com

4. Start a terminal session and enter the following command as `root`, depending on your platform. For example:
   - Oracle Linux 7

     ```
     # yum install oracle-database-preinstall-21c
     ```

   - Oracle Linux 8

     ```
     # dnf install oracle-database-preinstall-21c
     ```

     > **Note:**
     >
     > Use the `-y` option if you want `yum` to skip the package confirmation prompt.

   You should see output indicating that you have subscribed to the Oracle Linux channel, and that packages are being installed.

   The Oracle Database Preinstallation RPM automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.
5. Check the RPM log file to review the system configuration changes. For example:

   ```
   /var/log/oracle-database-preinstall-21c/backup/timestamp/
   orakernel.log
   ```

6. Repeat steps 1 through 4 on all other servers in your cluster.

If you have a premier support subscription, you can enable Ksplice to provide zero downtime patching. Refer to the Ksplice User's Guide for installation instructions:

https://docs.oracle.com/en/operating-systems/oracle-linux/ksplice-user/

Installing Oracle Database Preinstallation RPM During an Oracle Linux Installation

Use this procedure to install a new Oracle Linux installation and to perform system configuration with the Oracle Preinstallation RPM:

1. Obtain Oracle Linux by downloading disk images
   - Oracle yum

     https://yum.oracle.com/oracle-linux-isos.html

   - Oracle Software Delivery Cloud website:

     https://edelivery.oracle.com/linux

   > **Note:**
   >
   > Ensure that you use the latest available update release for Oracle Linux.

2. Start the Oracle Linux installation and respond to installation screens with values appropriate for your environment.
3. Review the first software selection screen, which lists task-specific software options. At the bottom of the screen, there is an option to customize now or customize later. Select **Customize now**, and click **Next**.
4. On Oracle Linux, select **Servers** on the left of the screen and **System administration tools** on the right of the screen (options may vary between releases).

   The Packages in System Tools window opens.
5. Select the Oracle Database Preinstallation RPM package box from the package list. For example, select a package similar to the following:

   For Oracle Linux 7:

   ```
   oracle-database-preinstall-21c-1.0-1.el7.x86_64.rpm
   ```

   For Oracle Linux 8:

   ```
   oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
   ```

   If you do not have an Oracle Database Preinstallation RPM package option that is current for your Oracle Database release, because you are using an Oracle Linux installation that is previous to your Oracle Database and Oracle Grid Infrastructure release, then install the RPM for your release manually after completing the operating system installation.
6. Close the optional package window and click **Next**.
7. Complete the other screens to finish the Oracle Linux installation.

Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

8. Repeat steps 2 through 6 on all other cluster member nodes.

Installing Oracle Oracle Database Preinstallation RPM Using the Oracle Linux yum Server

Install Oracle Linux and configure your Oracle Linux installation for security errata or bug fix updates using the Oracle Linux yum server.

1. Refer to the official Oracle Linux documentation to install Oracle Linux:
   - Oracle Linux 7
     Oracle Linux 7 Installation Guide

     Oracle Linux 7: Managing Software
   - Oracle Linux 8
     Oracle Linux 8 Installation Guide

     Oracle Linux 8: Managing Software
2. After you have completed your Oracle Linux installation, run the command `yum update` as needed to obtain the most current security errata and bug fixes for your Oracle Linux installation.
3. Restart your system.
4. Start a terminal session and enter the following command as `root`:
   - Oracle Linux 7

     ```
     # yum install oracle-database-preinstall-21c
     ```

     > **Note:**
     >
     > Use the `-y` option if you want `yum` to skip the package confirmation prompt.
   - Oracle Linux 8

     ```
     # dnf install oracle-database-preinstall-21c
     ```

     > **Note:**
     >
     > Use the `-y` option if you want `yum` to skip the package confirmation prompt.

     You should see output indicating that you have subscribed to the Oracle Linux channel, and that packages are being installed.

The Oracle Database Preinstallation RPM automatically creates a standard (not role-allocated) Oracle installation owner and groups and sets up other kernel configuration settings as required for Oracle installations. If you plan to use job-role separation, then create the extended set of database users and groups depending on your requirements.

Configure Additional Operating System Features

Oracle recommends that you configure your operating system before starting installation with additional features, such as IPMI or additional programming environments.

Review the preinstallation chapters to ensure that you have completed configuration as needed for the features you want.

4

Configuring Operating Systems for Oracle Database on Linux

Complete operating system configuration requirements and checks for Linux operating systems before you start installation.

You must meet the minimum requirements such as the operating system distribution, packages, and other software requirements for this release before you start an Oracle Database or Oracle Grid Infrastructure installation.

- Guidelines for Linux Operating System Installation
  Operating system guidelines to be aware of before proceeding with an Oracle installation.
- Reviewing Operating System and Software Upgrade Best Practices
  These topics provide general planning guidelines and platform-specific information about upgrades and migration.
- Reviewing Operating System Security Common Practices
  Secure operating systems are an important basis for general system security.
- About Installation Fixup Scripts
  Oracle Universal Installer detects when the minimum requirements for an installation are not met, and creates shell scripts, called fixup scripts, to finish incomplete system configuration steps.
- About Operating System Requirements
  Depending on the products that you intend to install, verify that you have the required operating system kernel and packages installed.
- Operating System Requirements for x86-64 Linux Platforms
  The Linux distributions and packages listed in this section are supported for this release on x86-64.
- Additional Drivers and Software Packages for Linux
  Information about optional drivers and software packages.
- Checking Kernel and Package Requirements for Linux
  Verify your kernel and packages to see if they meet minimum requirements for installation.
- Setting Clock Source for VMs on Linux x86-64
  Oracle recommends that you set the clock source to `tsc` for better performance in virtual environments (VM) on Linux x86-64.
- Installing the cvuqdisk RPM for Linux
  If you do not use an Oracle Database Preinstallation RPM, and you want to use the Cluster Verification Utility, then you must install the `cvuqdisk` RPM.
- Confirming Host Name Resolution
  Check to ensure that the host name for your server is resolvable.
- Disabling Transparent HugePages
  Oracle recommends that you disable Transparent HugePages before you start installation.

- Using Automatic SSH Configuration During Installation
  To install Oracle software, configure secure shell (SSH) connectivity between all cluster member nodes.
- Verifying the Disk I/O Scheduler on Linux
  For best performance for Oracle ASM, Oracle recommends that you use the Deadline I/O Scheduler.

Guidelines for Linux Operating System Installation

Operating system guidelines to be aware of before proceeding with an Oracle installation.

This section provides information about installing a supported Linux distribution. Complete the minimum hardware configuration before you install the operating system.

- Completing a Minimal or Default Linux Installation
  You must perform either a minimal or default Linux installation.
- About Oracle Linux with the Unbreakable Enterprise Kernel
  The Unbreakable Enterprise Kernel for Oracle Linux provides the latest innovations from upstream development to customers who run Oracle Linux in the data center.
- About the Oracle Database Preinstallation RPM
- Restrictions for HugePages and Transparent HugePages Configurations
  Review the HugePages and Transparent HugePages guidelines discussed in this section.

Related Topics

- Oracle Linux Operating System Documentation

Completing a Minimal or Default Linux Installation

You must perform either a minimal or default Linux installation.

To complete a minimal Linux installation, select one of the minimal installation options.

A minimal Linux installation option is either a custom installation where you select the **Minimal** option from Package Group Selection, or where you deselect all packages except for the Base pack. A minimal Linux installation lacks many RPMs required for database installation, so you must use an RPM package for your Oracle Linux release to install the required packages. The package you use depends on your Linux release, and your support status with Unbreakable Linux Network (ULN).

If you do not install the Oracle Database Preinstallation RPM, then Oracle recommends that you install your Linux operating system with the default software packages (RPMs).

A default Linux installation includes most of the required packages and helps you limit manual verification of package dependencies. Oracle recommends that you do not customize the RPMs during installation.

Refer to the official Oracle Linux documentation for more information about installing Oracle Linux:

- *Oracle Linux 7 Installation Guide*
- *Oracle Linux 7: Managing Software*
- *Oracle Linux 8 Installation Guide*
- *Oracle Linux 8: Managing Software*

About Oracle Linux with the Unbreakable Enterprise Kernel

The Unbreakable Enterprise Kernel for Oracle Linux provides the latest innovations from upstream development to customers who run Oracle Linux in the data center.

The Unbreakable Enterprise Kernel is included and enabled by default in Oracle Linux kernels. It is based on a recent stable mainline development Linux kernel, and also includes optimizations developed in collaboration with Oracle Database, Oracle middleware, and Oracle hardware engineering teams to ensure stability and optimal performance for the most demanding enterprise workloads.

Oracle highly recommends deploying the Unbreakable Enterprise Kernel in your Oracle Linux environment, especially if you run enterprise applications. However, using Unbreakable Enterprise Kernel is optional. If you require strict Red Hat Enterprise Linux (RHEL) kernel compatibility, then Oracle Linux also includes a kernel compatible with the RHEL Linux kernel, compiled directly from the RHEL source code.

You can obtain more information about the Unbreakable Enterprise Kernel for Oracle Linux at the following URL:

https://www.oracle.com/linux/

The Unbreakable Enterprise Kernel for Oracle Linux is the standard kernel used with Oracle products. The build and QA systems for Oracle Database and other Oracle products use the Unbreakable Enterprise Kernel for Oracle Linux exclusively. The Unbreakable Enterprise Kernel for Oracle Linux is also the kernel used in Oracle Exadata and Oracle Exalogic systems. Unbreakable Enterprise Kernel for Oracle Linux is used in all benchmark tests on Linux in which Oracle participates, as well as in the Oracle Database Preinstallation RPM program for x86-64.

Oracle Ksplice, which is part of Oracle Linux, updates the Linux operating system (OS) kernel, while it is running, without requiring restarts or any interruption. Ksplice is available only with Oracle Linux.

About the Oracle Database Preinstallation RPM

If your Linux distribution is Oracle Linux, or Red Hat Enterprise Linux, and you are an Oracle Linux support customer, then you can complete most preinstallation configuration tasks by using the Oracle Database Preinstallation RPM for your release.

Oracle Database Preinstallation RPMs are available from the Oracle Linux Network or available on the Oracle Linux DVDs. Using the Oracle Database Preinstallation RPM is not required, but Oracle recommends you use it to save time in setting up your cluster servers.

When installed, the Oracle Database Preinstallation RPM does the following:

- Automatically downloads and installs any additional RPM packages needed for installing Oracle Grid Infrastructure and Oracle Database, and resolves any dependencies
- Creates an `oracle` user, and creates the oraInventory (`oinstall`) and OSDBA (`dba`) groups for that user

- As needed, sets `sysctl.conf` settings, system startup parameters, and driver parameters to values based on recommendations from the Oracle Database Preinstallation RPM program
- Sets hard and soft resource limits
- Sets other recommended parameters, depending on your kernel version
- Sets `numa=off` in the kernel for Linux x86_64 machines.

Configure Oracle Database Preinstallation RPM only once on your operating system when you install Oracle Database or Oracle Grid Infrastructure for the first time on your system. For subsequent installations on the same system, do not install Oracle Database Preinstallation RPM again.

Do not install Oracle Database Preinstallation RPM on Oracle Engineered Systems, such as Oracle Exadata Database Machine. Oracle Engineered Systems include integrated system software that contain the required version of the operating system kernel and all software packages.

Note:

- The Oracle Database Preinstallation RPM designated for each Oracle Database release sets kernel parameters and resource limits only for the user account `oracle`. To use multiple software account owners, you must perform system configuration for other accounts manually.
- The Oracle Database Preinstallation RPM installs the X11 client libraries, but it does not install the X Window System server packages. To use graphical user interfaces such as OUI, configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

Restrictions for HugePages and Transparent HugePages Configurations

Review the HugePages and Transparent HugePages guidelines discussed in this section.

Oracle recommends that you disable Transparent HugePages, because they may cause delays in accessing memory that can result in node restarts in Oracle RAC environments, or performance issues or delays for Oracle Database single instances. Oracle continues to recommend using standard HugePages for Linux.

Transparent HugePages memory is enabled by default with Oracle Linux 6 and later, Red Hat Enterprise Linux 6 and later, SUSE 11 and later, kernels.

If you install Oracle Database Preinstallation RPM, then it disables Transparent HugePages.

Transparent Hugepages are similar to standard HugePages. However, while standard HugePages allocate memory at startup, Transparent Hugepages memory uses the `khugepaged` thread in the kernel to allocate memory dynamically during runtime, using swappable HugePages.

HugePages allocates non-swappable memory for large page tables using memory-mapped files. HugePages are not enabled by default. If you enable HugePages, then you should deduct the memory allocated to HugePages from the available RAM before calculating swap space. Refer to your distribution documentation and to Oracle Technology Network and My Oracle Support for more information.

During Oracle Grid Infrastructure installation, the Grid Infrastructure Management Repository (GIMR) is configured to use HugePages. Because the Grid Infrastructure Management Repository database starts before all other databases installed on the cluster, if the space allocated to HugePages is insufficient, then the System Global Area (SGA) of one or more databases may be mapped to regular pages, instead of Hugepages, which can adversely affect performance. Configure the HugePages memory allocation to a size large enough to accommodate the sum of the SGA sizes of all the databases you intend to install on the cluster, as well as the Grid Infrastructure Management Repository.

Reviewing Operating System and Software Upgrade Best Practices

These topics provide general planning guidelines and platform-specific information about upgrades and migration.

- General Upgrade Best Practices
  Be aware of these guidelines as a best practice before you perform an upgrade.
- New Server Operating System Upgrade Option
  You can upgrade your operating system by installing a new operating system on a server, and then migrating your database either manually, or by using Export/Import method.

General Upgrade Best Practices

Be aware of these guidelines as a best practice before you perform an upgrade.

If you have an existing Oracle Database installation, then do the following:

- Record the version numbers, patches, and other configuration information
- Review upgrade procedures for your existing installation
- Review Oracle Database upgrade documentation before proceeding with installation, to decide how you want to proceed

> **Caution:**
>
> Always create a backup of existing databases before starting any configuration change.

Refer to *Oracle Database Upgrade Guide* for more information about required software updates, pre-upgrade tasks, post-upgrade tasks, compatibility, and interoperability between different releases.

Related Topics

- *Oracle Database Upgrade Guide*

New Server Operating System Upgrade Option

You can upgrade your operating system by installing a new operating system on a server, and then migrating your database either manually, or by using Export/Import method.

> **Note:**
>
> Confirm that the server operating system is supported, and that kernel and package requirements for the operating system meet or exceed the minimum requirements for the Oracle Database release to which you want to migrate.

Manual, Command-Line Copy for Migrating Data and Upgrading Oracle Database

You can copy files to the new server and upgrade it manually. If you use this procedure, then you cannot use Oracle Database Upgrade Assistant. However, you can revert to your existing database if you encounter upgrade issues.

1. Copy the database files from the computer running the previous operating system to the one running the new operating system.
2. Re-create the control files on the computer running the new operating system.
3. Manually upgrade the database using command-line scripts and utilities.

> **See Also:**
>
> *Oracle Database Upgrade Guide* to review the procedure for upgrading the database manually, and to evaluate the risks and benefits of this option

Export/Import Method for Migrating Data and Upgrading Oracle Database

You can install the operating system on the new server, install the new Oracle Database release on the new server, and then use Oracle Data Pump Export and Import utilities to migrate a copy of data from your current database to a new database in the new release. Oracle Data Pump Export and Import are recommended for higher performance and to ensure support for new data types.

> **See Also:**
>
> *Oracle Database Upgrade Guide* to review the Export/Import method for migrating data and upgrading Oracle Database

Reviewing Operating System Security Common Practices

Secure operating systems are an important basis for general system security.

Ensure that your operating system deployment is in compliance with common security practices as described in your operating system vendor security guide.

About Installation Fixup Scripts

Oracle Universal Installer detects when the minimum requirements for an installation are not met, and creates shell scripts, called fixup scripts, to finish incomplete system configuration steps.

If Oracle Universal Installer detects an incomplete task, then it generates fixup scripts (`runfixup.sh`). You can run the fixup script and click **Fix and Check Again**. The fixup script modifies both persistent parameter settings and parameters in memory, so you do not have to restart the system.

The Fixup script does the following tasks:

- Sets kernel parameters, if necessary, to values required for successful installation, including:
  - Shared memory parameters.
  - Open file descriptor and UDP send/receive parameters.
- Creates and sets permissions on the Oracle Inventory (central inventory) directory.
- Creates or reconfigures primary and secondary group memberships for the installation owner, if necessary, for the Oracle Inventory directory and the operating system privileges groups.
- Sets shell limits, if necessary, to required values.

> **Note:**
>
> Using fixup scripts does not ensure that all the prerequisites for installing Oracle Database are met. You must still verify that all the preinstallation requirements are met to ensure a successful installation.

Oracle Universal Installer is fully integrated with Cluster Verification Utility (CVU) automating many prerequisite checks for your Oracle Grid Infrastructure or Oracle Real Application Clusters (Oracle RAC) installation. You can also manually perform various CVU verifications by running the `cluvfy` command.

Related Topics

- Completing Preinstallation Tasks Manually
  You can complete the preinstallation configuration tasks manually.
- Minimum Parameter Settings for Installation
  Use this table to set parameters manually if you cannot use the fixup scripts.
- *Oracle Clusterware Administration and Deployment Guide*

About Operating System Requirements

Depending on the products that you intend to install, verify that you have the required operating system kernel and packages installed.

Requirements listed in this document are current as of the date listed on the title page.

Oracle Universal Installer performs checks on your system to verify that it meets the listed operating system package requirements. To ensure that these checks complete successfully, verify the requirements before you start OUI.

> **Note:**
>
> Oracle does not support running different operating system versions on cluster members, unless an operating system is being upgraded. You cannot run different operating system version binaries on members of the same cluster, even if each operating system is supported.

Operating System Requirements for x86-64 Linux Platforms

The Linux distributions and packages listed in this section are supported for this release on x86-64.

Identify the requirements for your Linux distribution, and ensure that you have a supported kernel and required packages installed before starting installation.

The platform-specific hardware and software requirements included in this guide were current when this guide was published. However, because new platforms and operating system software versions may be certified after this guide is published, review the certification matrix on the My Oracle Support website for the most up-to-date list of certified hardware platforms and operating system versions:

https://support.oracle.com/

- General Guidelines for Operating System Packages
  Review these guidelines before you proceed with installing operating system packages.
- Supported Oracle Linux 8 Distributions for x86-64
  Use the following information to check supported Oracle Linux 8 distributions:
- Supported Oracle Linux 7 Distributions for x86-64
  Use the following information to check supported Oracle Linux 7 distributions:
- Supported Red Hat Enterprise Linux 8 Distributions for x86-64
  Use the following information to check supported Red Hat Enterprise Linux 8 distributions:
- Supported SUSE Linux Enterprise Server 15 Distributions for x86-64
  Use the following information to check supported SUSE Linux Enterprise Server 15 distributions:
- Installing Operating System Packages
  Learn how to install the latest version of your Oracle Linux and SUSE Linux Enterprise Server operating system packages listed earlier.

General Guidelines for Operating System Packages

Review these guidelines before you proceed with installing operating system packages.

- Oracle recommends that you install Oracle Linux and use Oracle Database Preinstallation RPM to configure your operating systems for Oracle Database and Oracle Grid Infrastructure installations.
- The Oracle Database installation requires an X Window System (for example, `libX`). The `libX` packages are part of a default Linux installation. If you perform an installation on a system with a reduced set of packages, then you must ensure that `libx` or a similar X Window System package is installed.
- The Unbreakable Enterprise Kernel for Oracle Linux can be installed on x86-64 servers running either Oracle Linux or Red Hat Enterprise Linux. As of Oracle Linux 5 Update 6, the Unbreakable Enterprise Kernel is the default system kernel. An x86 (32-bit) release of Oracle Linux including the Unbreakable Enterprise Kernel is available with Oracle Linux 5 update 7 and later.
- 32-bit packages in these requirements lists are needed only if you intend to use 32-bit client applications to access 64-bit servers that have the same operating system.
- Oracle Database 12c Release 2 (12.2) and later does not require the compiler packages `gcc` and `gcc-c++` on for Oracle Database or Oracle Grid Infrastructure installations.
- These operating system requirements do not apply to Oracle Engineered Systems, such as Oracle Exadata Database Machine. Oracle Engineered Systems include integrated system software that contain the required version of the operating system kernel and all software packages. Please verify that you have the minimum required Exadata image. Refer My Oracle Support note 888828.1 for more information.

Related Topics

- My Oracle Support Note 888828.1

Supported Oracle Linux 8 Distributions for x86-64

Use the following information to check supported Oracle Linux 8 distributions:

Table 4-1 x86-64 Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Oracle Linux 8	Minimum supported versions: • Oracle Linux 8.2 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.1.2.el8uek.x86_64 or later • Oracle Linux 8.2 with the Red Hat Compatible Kernel: 4.18.0-193.19.1.el8_2.x86_64 or later **Note:** Oracle recommends that you update Oracle Linux to the latest available version and release level.

Table 4-1 (Cont.) x86-64 Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
Packages for Oracle Linux 8	Subscribe to the Oracle Linux 8 channel on the Unbreakable Linux Network, or configure a yum repository from the Oracle Linux yum server website, and then install the Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`. The Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`, automatically installs all required packages listed in the table below, their dependencies for Oracle Grid Infrastructure and Oracle Database installations, and also performs other system configuration. If you install the Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. bc binutils compat-openssl10 elfutils-libelf glibc glibc-devel ksh libaio libXrender libX11 libXau libXi libXtst libgcc libnsl libstdc++ libxcb libibverbs make policycoreutils policycoreutils-python-utils smartmontools sysstat **Note:** If you intend to use 32-bit client applications to access 64-bit servers, then you must also install (where available) the latest 32-bit versions of the packages listed in this table.
Optional Packages for Oracle Linux 8	Based on your requirement, install the latest released versions of the following packages: ipmiutil (for Intelligent Platform Management Interface) libnsl2 (for Oracle Database Client only) libnsl2-devel (for Oracle Database Client only) libvirt-libs (for KVM) net-tools (for Oracle RAC and Oracle Clusterware) nfs-utils (for Oracle ACFS)

Table 4-1 (Cont.) x86-64 Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
KVM virtualization	Kernel-based virtual machine (KVM), also known as KVM virtualization, is certified on Oracle Database 21c for all supported Oracle Linux 8 distributions. For more information on supported virtualization technologies for Oracle Database, refer to the virtualization matrix: https://www.oracle.com/database/technologies/virtualization-matrix.html

Supported Oracle Linux 7 Distributions for x86-64

Use the following information to check supported Oracle Linux 7 distributions:

Table 4-2 x86-64 Oracle Linux 7 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Oracle Linux 7	Minimum supported versions: • Oracle Linux 7.4 with the Unbreakable Enterprise Kernel 4: 4.1.12-124.53.1.el7uek.x86_64 or later • Oracle Linux 7.6 with the Unbreakable Enterprise Kernel 5: 4.14.35-2025.404.1.el7uek.x86_64 or later **Note:** Oracle recommends that you use update Oracle Linux to the latest available version and release level.

Table 4-2 (Cont.) x86-64 Oracle Linux 7 Minimum Operating System Requirements

Item	Requirements
Packages for Oracle Linux 7	Subscribe to the Oracle Linux 7 channel on the Unbreakable Linux Network, or configure a yum repository from the Oracle Linux yum server website, and then install the Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`. The Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`, automatically installs all required packages listed in the table below, their dependencies for Oracle Grid Infrastructure and Oracle Database installations, and also performs other system configuration. If you install the Oracle Database Preinstallation RPM, `oracle-database-preinstall-21c`, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. bc binutils elfutils-libelf glibc glibc-devel ksh libaio libXrender libX11 libXau libXi libXtst libgcc libstdc++ libxcb make policycoreutils policycoreutils-python smartmontools sysstat **Note:** If you intend to use 32-bit client applications to access 64-bit servers, then you must also install (where available) the latest 32-bit versions of the packages listed in this table.
Optional Packages for Oracle Linux 7	Based on your requirement, install the latest released versions of the following packages: ipmiutil (for Intelligent Platform Management Interface) libvirt-libs (for KVM) net-tools (for Oracle RAC and Oracle Clusterware) nfs-utils (for Oracle ACFS)
KVM virtualization	Kernel-based virtual machine (KVM), also known as KVM virtualization, is certified on Oracle Database 21c for all supported Oracle Linux 7 distributions. For more information on supported virtualization technologies for Oracle Database, refer to the virtualization matrix: https://www.oracle.com/database/technologies/virtualization-matrix.html

Related Topics

- Installing Operating System Packages
  Learn how to install the latest version of your Oracle Linux and SUSE Linux Enterprise Server operating system packages listed earlier.

Related Topics

- Installing Operating System Packages
  Learn how to install the latest version of your Oracle Linux and SUSE Linux Enterprise Server operating system packages listed earlier.

Supported Red Hat Enterprise Linux 8 Distributions for x86-64

Use the following information to check supported Red Hat Enterprise Linux 8 distributions:

Table 4-3 x86-64 Red Hat Enterprise Linux 8 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Red Hat Enterprise Linux 8	Minimum supported versions: • Red Hat Enterprise Linux 8.2: 4.18.0-193.19.1.el8_2.x86_64 or later
Packages for Red Hat Enterprise Linux 8	Install the latest released versions of the following packages: bc binutils compat-openssl10 elfutils-libelf glibc glibc-devel ksh libaio libXrender libX11 libXau libXi libXtst libgcc libnsl libstdc++ libxcb libibverbs make smartmontools sysstat **Note:** If you intend to use 32-bit client applications to access 64-bit servers, then you must also install (where available) the latest 32-bit versions of the packages listed in this table.

Table 4-3 (Cont.) x86-64 Red Hat Enterprise Linux 8 Minimum Operating System Requirements

Item	Requirements
Optional Packages for Red Hat Enterprise Linux 8	Based on your requirement, install the latest released versions of the following packages: ipmiutil (for Intelligent Platform Management Interface) libnsl2 (for Oracle Database Client only) libnsl2-devel (for Oracle Database Client only) net-tools (for Oracle RAC and Oracle Clusterware) nfs-utils (for Oracle ACFS)

Supported SUSE Linux Enterprise Server 15 Distributions for x86-64

Use the following information to check supported SUSE Linux Enterprise Server 15 distributions:

Table 4-4 x86-64 SUSE Linux Enterprise Server 15 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
SUSE Linux Enterprise Server	Minimum supported version: SUSE Linux Enterprise Server 15 SP1: 4.12.14-197.29-default or later

Table 4-4 (Cont.) x86-64 SUSE Linux Enterprise Server 15 Minimum Operating System Requirements

Item	Requirements
Packages for SUSE Linux Enterprise Server 15	Install the latest released versions of the following packages: bc binutils glibc glibc-devel insserv-compat libaio-devel libaio1 libX11-6 libXau6 libXext-devel libXext6 libXi-devel libXi6 libXrender-devel libXrender1 libXtst6 libcap-ng-utils libcap-ng0 libcap-progs libcap1 libcap2 libelf1 libgcc_s1 libjpeg8 libpcap1 libpcre1 libpcre16 libpng16 libstdc++6 libtiff5 libgfortran4 mksh make pixz smartmontools sysstat xorg-x11-libs **Note:** If you intend to use 32-bit client applications to access 64-bit servers, then you must also install (where available) the latest 32-bit versions of the packages listed in this table.
Optional Packages for SUSE Linux Enterprise Server 15	Based on your requirement, install the latest released versions of the following packages: ipmiutil (for Intelligent Platform Management Interface) net-tools (for Oracle RAC and Oracle Clusterware) nfs-kernel-server (for Oracle ACFS)

Installing Operating System Packages

Learn how to install the latest version of your Oracle Linux and SUSE Linux Enterprise Server operating system packages listed earlier.

You must install the latest version of your operating system packages from the respective operating system vendor repository by using a package manager like `YUM` or `YaST` depending on your operating system.

Ensure that the appropriate channel or repository is enabled to include these packages.

For example:

On Oracle Linux or Red Hat Enterprise Linux, to install the latest `bc` package using `YUM`, run the following command:

```
$ yum install bc
```

On SUSE Linux Enterprise Server, to install the latest `bc` package using `YaST`, run the following command:

```
$ yast --install bc
```

Additional Drivers and Software Packages for Linux

Information about optional drivers and software packages.

You are not required to install additional drivers and packages, but you may choose to install or configure these drivers and packages.

- Installing PAM for Login Authentication on Linux
  Pluggable Authentication Modules (PAM) is a system of libraries that handle user authentication tasks for applications.
- Installation Requirements for OCFS2
  Review the following sections to install OCFS2.
- Installing Oracle Messaging Gateway
  Oracle Messaging Gateway is installed with Enterprise Edition of Oracle Database. However, you may require a CSD or Fix Packs.
- Installation Requirements for ODBC and LDAP
  Review these topics to install Open Database Connectivity (ODBC) and Lightweight Directory Access Protocol (LDAP).
- Installation Requirements for Programming Environments for Linux
  Review the following section to install programming environments:
- Installation Requirements for Web Browsers
  Web browsers are required only if you intend to use Oracle Enterprise Manager Database Express and Oracle Enterprise Manager Cloud Control. Web browsers must support JavaScript, and the HTML 4.0 and CSS 1.0 standards.

Installing PAM for Login Authentication on Linux

Pluggable Authentication Modules (PAM) is a system of libraries that handle user authentication tasks for applications.

On Linux, external scheduler jobs require PAM. Oracle strongly recommends that you install the latest Linux-PAM library for your Linux distribution.

Use a package management system (`yum`, `up2date`, `YaST`) for your distribution to install the latest `pam` (Pluggable Authentication Modules for Linux) library.

Installation Requirements for OCFS2

Review the following sections to install OCFS2.

- About OCFS2 and Shared Storage
  You can use Oracle Cluster File System 2 (OCFS2) with Oracle Grid Infrastructure. However, you are not required to use OCFS2.
- Installing OCFS2
  An OCFS2 installation consists of two parts, the kernel module and the tools module.

About OCFS2 and Shared Storage

You can use Oracle Cluster File System 2 (OCFS2) with Oracle Grid Infrastructure. However, you are not required to use OCFS2.

OCFS2 is a POSIX-compliant general purpose shared disk cluster file system for Linux. OCFS2 is supported for this release only with Oracle Linux 7.

Installing OCFS2

An OCFS2 installation consists of two parts, the kernel module and the tools module.

The supported version of the OCFS2 kernel module depends on the version of Unbreakable Enterprise Kernel available with Oracle Linux 7 and Oracle Linux 8.

1. Run the following command to install the latest version of the OCFS2 kernel module:

   ```
   # yum install kernel-uek ocfs2
   ```

2. Run the following command to install the OCFS2 tools module after you install the OCFS2 kernel module:

   ```
   # yum install ocfs2-tools
   ```

> **Note:**
>
> Each cluster node should run the same version of OCFS2 modules and a compatible version of Unbreakable Enterprise Kernel.

https://oss.oracle.com/projects/ocfs2/

Installing Oracle Messaging Gateway

Oracle Messaging Gateway is installed with Enterprise Edition of Oracle Database. However, you may require a CSD or Fix Packs.

If you require a CSD or Fix Packs for IBM WebSphere MQ, then see the following website for more information:

https://www.ibm.com/support/

> **Note:**
>
> Oracle Messaging Gateway does not support the integration of Advanced Queuing with TIBCO Rendezvous on IBM: Linux on System z.

Related Topics

- *Oracle Database Advanced Queuing User's Guide*

Installation Requirements for ODBC and LDAP

Review these topics to install Open Database Connectivity (ODBC) and Lightweight Directory Access Protocol (LDAP).

- About ODBC Drivers and Oracle Database
  Open Database Connectivity (ODBC) is a set of database access APIs that connect to the database, prepare, and then run SQL statements on the database.
- Installing ODBC Drivers for Linux x86-64
  If you intend to use ODBC, then install the most recent ODBC Driver Manager for Linux.
- About LDAP and Oracle Plug-ins
  Lightweight Directory Access Protocol (LDAP) is an application protocol for accessing and maintaining distributed directory information services over IP networks.
- Installing the LDAP Package
  LDAP is included in a default Linux operating system installation.

About ODBC Drivers and Oracle Database

Open Database Connectivity (ODBC) is a set of database access APIs that connect to the database, prepare, and then run SQL statements on the database.

An application that uses an ODBC driver can access non-uniform data sources, such as spreadsheets and comma-delimited files.

Installing ODBC Drivers for Linux x86-64

If you intend to use ODBC, then install the most recent ODBC Driver Manager for Linux.

Download and install the ODBC Driver Manager and Linux RPMs from the following website:

http://www.unixodbc.org

Review the minimum supported ODBC driver releases, and install ODBC drivers of the following or later releases for all Linux distributions:

```
unixODBC-2.3.4 or later
```

About LDAP and Oracle Plug-ins

Lightweight Directory Access Protocol (LDAP) is an application protocol for accessing and maintaining distributed directory information services over IP networks.

You require the LDAP package if you want to use features requiring LDAP, including the Oracle Database scripts `odisrvreg` and `oidca` for Oracle Internet Directory, or `schemasync` for third-party LDAP directories.

Installing the LDAP Package

LDAP is included in a default Linux operating system installation.

If you did not perform a default Linux installation, and you intend to use Oracle scripts requiring LDAP, then use a package management system (`up2date`, `YaST`) for your distribution to install a supported LDAP package for your distribution, and install any other required packages for that LDAP package.

Installation Requirements for Programming Environments for Linux

Review the following section to install programming environments:

- Installation Requirements for Programming Environments for Linux x86-64
  Ensure that your system meets the requirements for the programming environment you want to configure:

Installation Requirements for Programming Environments for Linux x86-64

Ensure that your system meets the requirements for the programming environment you want to configure:

Table 4-5 Requirements for Programming Environments for Linux X86–64

Programming Environments	Support Requirements
Java Database Connectivity (JDBC) / JDBC Oracle Call Interface (JDBC OCI)	JDK 8 (Java SE Development Kit) with the JNDI extension with Oracle Java Database Connectivity.

Table 4-5 (Cont.) Requirements for Programming Environments for Linux X86–64

Programming Environments	Support Requirements
Oracle Call Interface (OCI) Oracle C++ Call Interface Pro*C/C++	Intel C/C++ Compiler 19.0.8.324 or later, or the GNU C and C++ compilers listed in this table. Oracle C++ Call Interface (OCCI) applications can be built with g++ and Intel C++ Compiler 17.0.8.066 used with the standard template libraries of the `gcc` compilers listed in this table. **Note:** For OCCI applications on Linux-x86-64 using `g++5.3`, `g++6.3`, `g++7.3` and `g++8`, instead of the `libocci.so.21.1` library, link to the `$ORACLE_HOME/lib/libocci_gcc53.so.21.1` library. For the `g++` compilation command, use the `-Wno-narrowing` option to build the applications using the `g++5.3`, `g++6.3`, `g++7.3` and `g++8` compiler versions.
`gcc` compiler packages	Install the latest released versions of the `gcc` packages listed here. gcc gcc-c++ gcc-info gcc-locale gcc48 gcc48-info gcc48-locale gcc48-c++ **Note:** If you intend to use 32-bit client applications to access 64-bit servers, then you must also install the latest 32-bit versions of the packages listed in this table.
Oracle XML Developer's Kit (XDK)	Oracle XML Developer's Kit is supported with the same compilers as OCCI.
Pro*COBOL	Micro Focus Visual COBOL for Eclipse 2.3 - Update 2
Unzip and Tar	Install the latest versions of Unzip and Tar. Unzip and Tar utlities are required to extract the image files for Oracle Database and Oracle Grid Infrastructure installations and patches.

Installation Requirements for Web Browsers

Web browsers are required only if you intend to use Oracle Enterprise Manager Database Express and Oracle Enterprise Manager Cloud Control. Web browsers must support JavaScript, and the HTML 4.0 and CSS 1.0 standards.

https://support.oracle.com

Related Topics

- *Oracle Enterprise Manager Cloud Control Basic Installation Guide*

Checking Kernel and Package Requirements for Linux

Verify your kernel and packages to see if they meet minimum requirements for installation.

1. To determine the distribution and version of Linux installed, enter one of the following commands:

   ```
   # cat /etc/oracle-release
   # cat /etc/redhat-release
   # cat /etc/os-release
   # lsb_release -id
   ```

2. To determine if the required kernel errata is installed, enter the following command:

   ```
   # uname -r
   ```

   The following is an example of the output this command displays on an Oracle Linux 8 system:

   ```
   5.4.17-2011.1.2.el8uek.x86_64
   ```

   Review the required errata level for your distribution. If the errata level is previous to the required minimum errata update, then obtain and install the latest kernel update from your Linux distributor.

3. To determine whether the required packages are installed, enter commands similar to the following:

   ```
   # rpm -q package_name
   ```

   Alternatively, if you require specific system architecture information, then enter the following command:

   ```
   # rpm -qa --queryformat "%{NAME}-%{VERSION}-%{RELEASE} (%{ARCH})\n" | grep
   package_name
   ```

   You can also combine a query for multiple packages, and review the output for the correct versions. For example:

   ```
   # rpm -q binutils compat-libstdc++ gcc glibc libaio libgcc libstdc++ \
   make sysstat unixodbc
   ```

   If a package is not installed, then install it from your Linux distribution media or download the required package version from your Linux distributor's website.

Setting Clock Source for VMs on Linux x86-64

Oracle recommends that you set the clock source to `tsc` for better performance in virtual environments (VM) on Linux x86-64.

1. As the `root` user, check if the `tsc` clock source is available on your system.

   ```
   # cat /sys/devices/system/clocksource/clocksource0/
   available_clocksource
   xen tsc acpi_pm
   ```

2. If the `tsc` clock source is available, then set `tsc` as the current clock source.

   ```
   # echo "tsc">/sys/devices/system/clocksource/clocksource0/
   current_clocksource
   ```

3. Verify that the current clock source is set to `tsc`.

   ```
   # cat /sys/devices/system/clocksource/clocksource0/
   current_clocksource
   tsc
   ```

4. Using any text editor, append the `clocksource` directive to the `GRUB_CMDLINE_LINUX` line in the `/etc/default/grub` file to retain this clock source setting even after a reboot.

   ```
   GRUB_CMDLINE_LINUX="rd.lvm.lv=ol/root rd.lvm.lv=ol/swap rhgb quiet
   numa=off transparent_hugepage=never clocksource=tsc"
   ```

Installing the cvuqdisk RPM for Linux

If you do not use an Oracle Database Preinstallation RPM, and you want to use the Cluster Verification Utility, then you must install the `cvuqdisk` RPM.

Without `cvuqdisk`, Cluster Verification Utility cannot discover shared disks, and you receive the error message "Package cvuqdisk not installed" when you run Cluster Verification Utility. Use the `cvuqdisk` RPM for your hardware (for example, `x86_64`).

1. Locate the `cvuqdisk` RPM package, which is located in the directory *Grid_home*`/cv/rpm`. Where *Grid_home* is the Oracle Grid Infrastructure home directory.
2. Copy the `cvuqdisk` package to each node on the cluster. You should ensure that each node is running the same version of Linux.
3. Log in as `root`.
4. Use the following command to find if you have an existing version of the `cvuqdisk` package:

   ```
   # rpm -qi cvuqdisk
   ```

   If you have an existing version of `cvuqdisk`, then enter the following command to deinstall the existing version:

   ```
   # rpm -e cvuqdisk
   ```

5. Set the environment variable `CVUQDISK_GRP` to point to the group that owns `cvuqdisk`, typically `oinstall`. For example:

```
# CVUQDISK_GRP=oinstall; export CVUQDISK_GRP
```

6. In the directory where you have saved the `cvuqdisk` RPM, use the command `rpm -iv package` to install the `cvuqdisk` package. For example:

```
# rpm -iv cvuqdisk-1.0.10-1.rpm
```

Confirming Host Name Resolution

Check to ensure that the host name for your server is resolvable.

Typically, the computer on which you want to install Oracle Database is connected to a network. Ensure that the computer host name is resolvable, either through a Domain Name System (DNS), a network information service (NIS), or a centrally-maintained TCP/IP host file, such as `/etc/hosts`. Use the `ping` command to ensure that your computer host name is resolvable. For example:

```
ping myhostname
pinging myhostname.example.com [192.0.2.2] with 32 bytes of data:
Reply from 192.0.2.2: bytes=32 time=138ms TTL=56
```

Related Topics

- Configuring Networks for Oracle Database
  If you install Oracle Databases on servers with multiple Oracle homes, multiple aliases, or without a static IP address, then review these network configuration topics.

Disabling Transparent HugePages

Oracle recommends that you disable Transparent HugePages before you start installation.

Transparent HugePages memory differs from standard HugePages memory because the kernel `khugepaged` thread allocates memory dynamically during runtime. Standard HugePages memory is pre-allocated at startup, and does not change during runtime.

> **Note:**
>
> Although Transparent HugePages is disabled on UEK2 and later UEK kernels, Transparent HugePages may be enabled by default on your Linux system.

Transparent HugePages memory is enabled by default with Oracle Linux 6 and later, Red Hat Enterprise Linux 6 and later, SUSE 11 and later, kernels.

Transparent HugePages can cause memory allocation delays during runtime. To avoid performance issues, Oracle recommends that you disable Transparent HugePages on all Oracle Database servers. Oracle recommends that you instead use standard HugePages for enhanced performance.

To check if Transparent HugePages is enabled, run one of the following commands as the `root` user:

Red Hat Enterprise Linux kernels:

```
# cat /sys/kernel/mm/redhat_transparent_hugepage/enabled
```

Other kernels:

```
# cat /sys/kernel/mm/transparent_hugepage/enabled
```

The following is a sample output that shows Transparent HugePages are being used as the [always] flag is enabled.

```
[always] never
```

> **Note:**
>
> If Transparent HugePages is removed from the kernel, then neither `/sys/kernel/mm/transparent_hugepage` nor `/sys/kernel/mm/redhat_transparent_hugepage files` exist.

To disable Transparent HugePages:

1. For Oracle Linux 7 and Red Hat Enterprise Linux 7, add or modify the `transparent_hugepage=never` parameter in the `/etc/default/grub` file:

   ```
   transparent_hugepage=never
   ```

   For example:

   ```
   GRUB_TIMEOUT=5
   GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
   GRUB_DEFAULT=saved
   GRUB_DISABLE_SUBMENU=true
   GRUB_TERMINAL_OUTPUT="console"
   GRUB_CMDLINE_LINUX="crashkernel=auto rhgb quiet numa=off
   transparent_hugepage=never"
   GRUB_DISABLE_RECOVERY="true"
   ```

   > **Note:**
   >
   > The file name may vary for your operating systems. Check your operating system documentation for the exact file name and the steps to disable Transparent HugePages.

2. Run the `grub2-mkconfig` command to regenerate the `grub.cfg` file.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

3. Restart the system to make the changes permanent.

Using Automatic SSH Configuration During Installation

To install Oracle software, configure secure shell (SSH) connectivity between all cluster member nodes.

Oracle Universal Installer (OUI) uses the `ssh` and `scp` commands during installation to run remote commands on and copy files to the other cluster nodes. You must configure SSH so that these commands do not prompt for a password.

> **Note:**
>
> Oracle configuration assistants use SSH for configuration operations from local to remote nodes. Oracle Enterprise Manager also uses SSH. RSH is no longer supported.

You can configure SSH from the OUI interface during installation for the user account running the installation. The automatic configuration creates passwordless SSH connectivity between all cluster member nodes. Oracle recommends that you use the automatic procedure if possible.

To enable the script to run, you must remove `stty` commands from the profiles of any existing Oracle software installation owners you want to use, and remove other security measures that are triggered during a login, and that generate messages to the terminal. These messages, mail checks, and other displays prevent Oracle software installation owners from using the SSH configuration script that is built into OUI. If they are not disabled, then SSH must be configured manually before an installation can be run.

In rare cases, Oracle Clusterware installation can fail during the "AttachHome" operation when the remote node closes the SSH connection. To avoid this problem, set the timeout wait to unlimited by setting the following parameter in the SSH daemon configuration file `/etc/ssh/sshd_config` on all cluster nodes:

```
LoginGraceTime 0
```

Related Topics

- Preventing Installation Errors Caused by Terminal Output Commands
  During an Oracle Grid Infrastructure installation, OUI uses SSH to run commands and copy files to the other nodes. During the installation, hidden files on the system (for example, `.bashrc` or `.cshrc`) can cause `makefile` and other installation errors if they contain terminal output commands.

Verifying the Disk I/O Scheduler on Linux

For best performance for Oracle ASM, Oracle recommends that you use the Deadline I/O Scheduler.

Disk I/O schedulers reorder, delay, or merge requests for disk I/O to achieve better throughput and lower latency. Linux has multiple disk I/O schedulers available, including Deadline, Noop, Anticipatory, and Completely Fair Queuing (CFQ).

On each cluster node, enter the following command to verify that the Deadline disk I/O scheduler is configured for use:

```
# cat /sys/block/${ASM_DISK}/queue/scheduler
noop [deadline] cfq
```

In this example, the default disk I/O scheduler is Deadline and `ASM_DISK` is the Oracle Automatic Storage Management (Oracle ASM) disk device.

On some virtual environments (VM) and special devices such as fast storage devices, the output of the above command may be `none`. The operating system or VM bypasses the kernel I/O scheduling and submits all I/O requests directly to the device. Do not change the I/O Scheduler settings on such environments.

If the default disk I/O scheduler is not Deadline, then set it using a rules file:

1. Using a text editor, create a UDEV rules file for the Oracle ASM devices:

   ```
   # vi /etc/udev/rules.d/60-oracle-schedulers.rules
   ```

2. Add the following line to the rules file and save it:

   ```
   ACTION=="add|change", KERNEL=="sd[a-z]", ATTR{queue/
   rotational}=="0", ATTR{queue/scheduler}="deadline"
   ```

3. On clustered systems, copy the rules file to all other nodes on the cluster. For example:

   ```
   $ scp 60-oracle-schedulers.rules root@node2:/etc/udev/rules.d/
   ```

4. Load the rules file and restart the UDEV service. For example:

   a. Oracle Linux and Red Hat Enterprise Linux

      ```
      # udevadm control --reload-rules
      ```

   b. SUSE Linux Enterprise Server

      ```
      # /etc/init.d boot.udev restart
      ```

5. Verify that the disk I/O scheduler is set as Deadline.

5

Configuring Users, Groups and Environments for Oracle Grid Infrastructure and Oracle Database

Before installation, create operating system groups and users, and configure user environments.

- Required Operating System Groups and Users
  Oracle software installations require an installation owner, an Oracle Inventory group, which is the primary group of all Oracle installation owners, and at least one group designated as a system privileges group.
- Oracle Installations with Standard and Job Role Separation Groups and Users
  A job role separation configuration of Oracle Database and Oracle ASM is a configuration with groups and users to provide separate groups for operating system authentication.
- Creating Operating System Privileges Groups
  The following sections describe how to create operating system groups for Oracle Grid Infrastructure and Oracle Database:
- Creating Operating System Oracle Installation User Accounts
  Before starting installation, create Oracle software owner user accounts, and configure their environments.
- Creating Oracle Database Vault User Accounts
  If you intend to use Oracle Database Vault by default, then you must create an Oracle Database Vault user account, and configure that user.
- Unsetting Oracle Installation Owner Environment Variables
  Unset Oracle installation owner environment variables before you start the installation.

Required Operating System Groups and Users

Oracle software installations require an installation owner, an Oracle Inventory group, which is the primary group of all Oracle installation owners, and at least one group designated as a system privileges group.

Review group and user options with your system administrator. If you have system administration privileges, then review the topics in this section and configure operating system groups and users as needed.

- Determining If an Oracle Inventory and Oracle Inventory Group Exist
  Determine if you have an existing Oracle central inventory, and ensure that you use the same Oracle Inventory for all Oracle software installations. Also, ensure that all Oracle software users you intend to use for installation have permissions to write to this directory.
- Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist
  Create an Oracle Inventory group manually as part of a planned installation, particularly where more than one Oracle software product is installed on servers.

- About Oracle Installation Owner Accounts
  Select or create an Oracle installation owner for your installation, depending on the group and user management plan you want to use for your installations.
- Identifying an Oracle Software Owner User Account
  You must create at least one software owner user account the first time you install Oracle software on the system. Either use an existing Oracle software user account, or create an Oracle software owner user account for your installation.

Determining If an Oracle Inventory and Oracle Inventory Group Exist

Determine if you have an existing Oracle central inventory, and ensure that you use the same Oracle Inventory for all Oracle software installations. Also, ensure that all Oracle software users you intend to use for installation have permissions to write to this directory.

When you install Oracle software on the system for the first time, OUI creates the `oraInst.loc` file. This file identifies the name of the Oracle Inventory group (by default, `oinstall`), and the path of the Oracle central inventory directory. If you have an existing Oracle central inventory, then ensure that you use the same Oracle Inventory for all Oracle software installations, and ensure that all Oracle software users you intend to use for installation have permissions to write to this directory.

```
oraInst.loccentral_inventory_locationgroup

inventory_loc=central_inventory_location
inst_group=group
```

Use the `more` command to determine if you have an Oracle central inventory on your system. For example:

```
# more /etc/oraInst.loc

inventory_loc=/u01/app/oraInventory
inst_group=oinstall
```

Use the command `grep groupname /etc/group` to confirm that the group specified as the Oracle Inventory group still exists on the system. For example:

```
$ grep oinstall /etc/group
oinstall:x:54321:grid,oracle
```

> **Note:**
>
> Do not put the `oraInventory` directory under the Oracle base directory for a new installation, because that can result in user permission errors for other installations.

Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist

Create an Oracle Inventory group manually as part of a planned installation, particularly where more than one Oracle software product is installed on servers.

By default, if an oraInventory group does not exist, then the installer uses the primary group of the installation owner for the Oracle software being installed as the oraInventory group. Ensure that this group is available as a primary group for all planned Oracle software installation owners.

```
oraInst.loc

# /usr/sbin/groupadd -g 54321 oinstall
```

About Oracle Installation Owner Accounts

Select or create an Oracle installation owner for your installation, depending on the group and user management plan you want to use for your installations.

You must create a software owner for your installation in the following circumstances:

- If an Oracle software owner user does not exist; for example, if this is the first installation of Oracle software on the system.
- If an Oracle software owner user exists, but you want to use a different operating system user, with different group membership, to separate Oracle Grid Infrastructure administrative privileges from Oracle Database administrative privileges.

In Oracle documentation, a user created to own only Oracle Grid Infrastructure software installations is called the Grid user (`grid`). This user owns both the Oracle Clusterware and Oracle Automatic Storage Management binaries. A user created to own either all Oracle installations, or one or more Oracle database installations, is called the Oracle user (`oracle`). You can have only one Oracle Grid Infrastructure installation owner, but you can have different Oracle users to own different installations.

Oracle software owners must have the Oracle Inventory group as their primary group, so that each Oracle software installation owner can write to the central inventory (oraInventory), and so that OCR and Oracle Clusterware resource permissions are set correctly. The database software owner must also have the OSDBA group and (if you create them) the OSOPER, OSBACKUPDBA, OSDGDBA, OSRACDBA, and OSKMDBA groups as secondary groups.

Identifying an Oracle Software Owner User Account

You must create at least one software owner user account the first time you install Oracle software on the system. Either use an existing Oracle software user account, or create an Oracle software owner user account for your installation.

To use an existing user account, obtain the name of an existing Oracle installation owner from your system administrator. Confirm that the existing owner is a member of the Oracle Inventory group.

For example, if you know that the name of the Oracle Inventory group is `oinstall`, then an Oracle software owner should be listed as a member of `oinstall`:

```
$ grep "oinstall" /etc/group
oinstall:x:54321:grid,oracle
```

You can then use the ID command to verify that the Oracle installation owners you intend to use have the Oracle Inventory group as their primary group. For example:

```
$ id oracle
uid=54321(oracle) gid=54321(oinstall)
groups=54321(oinstall),54322(dba),
54323(oper),54324(backupdba),54325(dgdba),54326(kmdba),54327(asmdba),543
30(racdba)


$ id grid
uid=54331(grid) gid=54321(oinstall) groups=54321(oinstall),54322(dba),
54327(asmdba),54328(asmoper),54329(asmadmin),54330(racdba)
```

For Oracle Restart installations, to successfully install Oracle Database, ensure that the `grid` user is a member of the `racdba` group.

After you create operating system groups, create or modify Oracle user accounts in accordance with your operating system authentication planning.

Related Topics

- Creating an Oracle Software Owner User
  If the Oracle software owner user (`oracle` or `grid`) does not exist, or if you require a new Oracle software owner user, then create it as described in this section.
- Modifying Oracle Owner User Groups
  If you have created an Oracle software installation owner account, but it is not a member of the groups you want to designate as the OSDBA, OSOPER, OSDBA for ASM, ASMADMIN, or other system privileges group, then modify the group settings for that user before installation.

Oracle Installations with Standard and Job Role Separation Groups and Users

A job role separation configuration of Oracle Database and Oracle ASM is a configuration with groups and users to provide separate groups for operating system authentication.

Review the following sections to understand more about a Job Role Separation deployment:

- About Oracle Installations with Job Role Separation
  Job role separation requires that you create different operating system groups for each set of system privileges that you grant through operating system authorization.

- Standard Oracle Database Groups for Database Administrators
  Oracle Database has two standard administration groups: OSDBA, which is required, and OSOPER, which is optional.
- Extended Oracle Database Groups for Job Role Separation
  Oracle Database provides an extended set of database groups to grant task-specific system privileges for database administration.
- Creating an ASMSNMP User
  The `ASMSNMP` user is an Oracle ASM user with privileges to monitor Oracle ASM instances. You are prompted to provide a password for this user during installation.
- Oracle Automatic Storage Management Groups for Job Role Separation
  Oracle Grid Infrastructure operating system groups provide their members task-specific system privileges to access and to administer Oracle Automatic Storage Management.

About Oracle Installations with Job Role Separation

Job role separation requires that you create different operating system groups for each set of system privileges that you grant through operating system authorization.

With Oracle Grid Infrastructure job role separation, Oracle ASM has separate operating system groups that provide operating system authorization for Oracle ASM system privileges for storage tier administration. This operating system authorization is separated from Oracle Database operating system authorization. In addition, the Oracle Grid Infrastructure installation owner provides operating system user authorization for modifications to Oracle Grid Infrastructure binaries.

With Oracle Database job role separation, each Oracle Database installation has separate operating system groups to provide authorization for system privileges on that Oracle Database. Multiple databases can, therefore, be installed on the cluster without sharing operating system authorization for system privileges. In addition, each Oracle software installation is owned by a separate installation owner, to provide operating system user authorization for modifications to Oracle Database binaries. Note that any Oracle software owner can start and stop all databases and shared Oracle Grid Infrastructure resources such as Oracle ASM or Virtual IP (VIP). Job role separation configuration enables database security, and does not restrict user roles in starting and stopping various Oracle Clusterware resources.

You can choose to create one administrative user and one group for operating system authentication for all system privileges on the storage and database tiers. For example, you can designate the `oracle` user to be the installation owner for all Oracle software, and designate `oinstall` to be the group whose members are granted all system privileges for Oracle Clusterware; all system privileges for Oracle ASM; all system privileges for all Oracle Databases on the servers; and all `OINSTALL` system privileges for installation owners. This group must also be the Oracle Inventory group.

If you do not want to use role allocation groups, then Oracle strongly recommends that you use at least two groups:

- A system privileges group whose members are granted administrative system privileges, including OSDBA, OSASM, and other system privileges groups.
- An installation owner group (the `oraInventory` group) whose members are granted Oracle installation owner system privileges (the `OINSTALL` system privilege).

> **Note:**
>
> To configure users for installation that are on a network directory service such as Network Information Services (NIS), refer to your directory service documentation.

Related Topics

- *Oracle Database Administrator's Guide*
- *Oracle Automatic Storage Management Administrator's Guide*

Standard Oracle Database Groups for Database Administrators

Oracle Database has two standard administration groups: OSDBA, which is required, and OSOPER, which is optional.

- The OSDBA group (typically, `dba`)

  You must create this group the first time you install Oracle Database software on the system. This group identifies operating system user accounts that have database administrative privileges (the `SYSDBA` privilege).

  If you do not create separate OSDBA, OSOPER, and OSASM groups for the Oracle ASM instance, then operating system user accounts that have the `SYSOPER` and `SYSASM` privileges must be members of this group. The name used for this group in Oracle code examples is `dba`. If you do not designate a separate group as the OSASM group, then the OSDBA group you define is also by default the OSASM group.

- The OSOPER group for Oracle Database (typically, `oper`)

  OSOPER grants the OPERATOR privilege to start up and shut down the database (the `SYSOPER` privilege). By default, members of the OSDBA group have all privileges granted by the `SYSOPER` privilege.

Extended Oracle Database Groups for Job Role Separation

Oracle Database provides an extended set of database groups to grant task-specific system privileges for database administration.

The extended set of Oracle Database system privileges groups are task-specific and less privileged than the OSDBA/SYSDBA system privileges. They are designed to provide privileges to carry out everyday database operations. Users granted these system privileges are also authorized through operating system group membership.

You do not have to create these specific group names, but during interactive and silent installation, you must assign operating system groups whose members are granted access to these system privileges. You can assign the same group to provide authorization for these privileges, but Oracle recommends that you provide a unique group to designate each privilege.

The subset of OSDBA job role separation privileges and groups consist of the following:

- OSBACKUPDBA group for Oracle Database (typically, `backupdba`)

Create this group if you want a separate group of operating system users to have a limited set of database backup and recovery related administrative privileges (the SYSBACKUP privilege).

- OSDGDBA group for Oracle Data Guard (typically, `dgdba`)

  Create this group if you want a separate group of operating system users to have a limited set of privileges to administer and monitor Oracle Data Guard (the SYSDG privilege). To use this privilege, add the Oracle Database installation owners as members of this group.

- The OSKMDBA group for encryption key management (typically, `kmdba`)

  Create this group if you want a separate group of operating system users to have a limited set of privileges for encryption key management such as Oracle Wallet Manager management (the SYSKM privilege). To use this privilege, add the Oracle Database installation owners as members of this group.

- The OSRACDBA group for Oracle Real Application Clusters Administration (typically, `racdba`)

  Create this group if you want a separate group of operating system users to have a limited set of Oracle Real Application Clusters (RAC) administrative privileges (the SYSRAC privilege). To use this privilege:

  – Add the Oracle Database installation owners as members of this group.

  – For Oracle Restart configurations, if you have a separate Oracle Grid Infrastructure installation owner user (`grid`), then you must also add the `grid` user as a member of the OSRACDBA group of the database to enable Oracle Grid Infrastructure components to connect to the database.

Related Topics

- *Oracle Database Administrator's Guide*
- *Oracle Database Security Guide*

Creating an ASMSNMP User

The `ASMSNMP` user is an Oracle ASM user with privileges to monitor Oracle ASM instances. You are prompted to provide a password for this user during installation.

In addition to the OSASM group, whose members are granted the SYSASM system privilege to administer Oracle ASM, Oracle recommends that you create a less privileged user, `ASMSNMP`, and grant that user SYSDBA privileges to monitor the Oracle ASM instance. Oracle Enterprise Manager uses the `ASMSNMP` user to monitor Oracle ASM status.

During installation, you are prompted to provide a password for the `ASMSNMP` user. You can create an operating system authenticated user, or you can create an Oracle Database user called `asmsnmp`. In either case, grant the user SYSDBA privileges.

Oracle Automatic Storage Management Groups for Job Role Separation

Oracle Grid Infrastructure operating system groups provide their members task-specific system privileges to access and to administer Oracle Automatic Storage Management.

- The OSASM group for Oracle ASM Administration (typically, `asmadmin`)

  Create this group as a separate group to separate administration privileges groups for Oracle ASM and Oracle Database administrators. Members of this group are granted the

SYSASM system privileges to administer Oracle ASM. In Oracle documentation, the operating system group whose members are granted privileges is called the OSASM group, and in code examples, where there is a group specifically created to grant this privilege, it is referred to as `asmadmin`.

Oracle ASM can support multiple databases. If you have multiple databases on your system, and use multiple OSDBA groups so that you can provide separate SYSDBA privileges for each database, then you should create a group whose members are granted the OSASM/SYSASM administrative privileges, and create a grid infrastructure user (grid) that does not own a database installation, so that you separate Oracle Grid Infrastructure SYSASM administrative privileges from a database administrative privileges group.

Members of the OSASM group can use SQL to connect to an Oracle ASM instance as SYSASM using operating system authentication. The SYSASM privileges permit mounting and dismounting disk groups, and other storage administration tasks. SYSASM privileges provide no access privileges on an RDBMS instance.

If you do not designate a separate group as the OSASM group, but you do define an OSDBA group for database administration, then by default the OSDBA group you define is also defined as the OSASM group.

- The OSOPER group for Oracle ASM (typically, `asmoper`)

  This is an optional group. Create this group if you want a separate group of operating system users to have a limited set of Oracle instance administrative privileges (the SYSOPER for ASM privilege), including starting up and stopping the Oracle ASM instance. By default, members of the OSASM group also have all privileges granted by the SYSOPER for ASM privilege.

Creating Operating System Privileges Groups

The following sections describe how to create operating system groups for Oracle Grid Infrastructure and Oracle Database:

- Creating the OSDBA for ASM Group
  You must designate a group as the OSDBA for ASM (`asmdba`) group during installation. Members of this group are granted access privileges to Oracle Automatic Storage Management.
- Creating the OSOPER for ASM Group
  You can choose to designate a group as the OSOPER for ASM group (`asmoper`) during installation. Members of this group are granted startup and shutdown privileges to Oracle Automatic Storage Management.
- Creating the OSDBA Group for Database Installations
  Each Oracle Database requires an operating system group to be designated as the OSDBA group. Members of this group are granted the SYSDBA system privileges to administer the database.
- Creating an OSOPER Group for Database Installations
  Create an OSOPER group only if you want to identify a group of operating system users with a limited set of database administrative privileges (SYSOPER operator privileges).

- Creating the OSBACKUPDBA Group for Database Installations
  You must designate a group as the OSBACKUPDBA group during installation. Members of this group are granted the SYSBACKUP privileges to perform backup and recovery operations using RMAN or SQL*Plus.
- Creating the OSDGDBA Group for Database Installations
  You must designate a group as the OSDGDBA group during installation. Members of this group are granted the SYSDG privileges to perform Data Guard operations.
- Creating the OSKMDBA Group for Database Installations
  You must designate a group as the OSKMDBA group during installation. Members of this group are granted the SYSKM privileges to perform Transparent Data Encryption keystore operations.
- Creating the OSRACDBA Group for Database Installations
  You must designate a group as the OSRACDBA group during database installation. Members of this group are granted the SYSRAC privileges to perform day–to–day administration of Oracle databases on an Oracle RAC cluster.

Creating the OSDBA for ASM Group

You must designate a group as the OSDBA for ASM (`asmdba`) group during installation. Members of this group are granted access privileges to Oracle Automatic Storage Management.

Create an OSDBA for ASM group using the group name `asmdba` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54327 asmdba
```

Creating the OSOPER for ASM Group

You can choose to designate a group as the OSOPER for ASM group (`asmoper`) during installation. Members of this group are granted startup and shutdown privileges to Oracle Automatic Storage Management.

If you want to create an OSOPER for ASM group, use the group name `asmoper` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54328 asmoper
```

Creating the OSDBA Group for Database Installations

Each Oracle Database requires an operating system group to be designated as the OSDBA group. Members of this group are granted the SYSDBA system privileges to administer the database.

You must create an OSDBA group in the following circumstances:

- An OSDBA group does not exist, for example, if this is the first installation of Oracle Database software on the system
- An OSDBA group exists, but you want to give a different group of operating system users database administrative privileges for a new Oracle Database installation

Create the OSDBA group using the group name `dba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54322 dba
```

Creating an OSOPER Group for Database Installations

Create an OSOPER group only if you want to identify a group of operating system users with a limited set of database administrative privileges (SYSOPER operator privileges).

For most installations, it is sufficient to create only the OSDBA group. However, to use an OSOPER group, create it in the following circumstances:

- If an OSOPER group does not exist; for example, if this is the first installation of Oracle Database software on the system
- If an OSOPER group exists, but you want to give a different group of operating system users database operator privileges in a new Oracle installation

If the OSOPER group does not exist, or if you require a new OSOPER group, then create it. Use the group name `oper` unless a group with that name already exists. For example:

```
# groupadd -g 54323 oper
```

Creating the OSBACKUPDBA Group for Database Installations

You must designate a group as the OSBACKUPDBA group during installation. Members of this group are granted the SYSBACKUP privileges to perform backup and recovery operations using RMAN or SQL*Plus.

Create the OSBACKUPDBA group using the group name `backupdba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54324 backupdba
```

Creating the OSDGDBA Group for Database Installations

You must designate a group as the OSDGDBA group during installation. Members of this group are granted the SYSDG privileges to perform Data Guard operations.

Create the OSDGDBA group using the group name `dgdba`, unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54325 dgdba
```

Creating the OSKMDBA Group for Database Installations

You must designate a group as the OSKMDBA group during installation. Members of this group are granted the SYSKM privileges to perform Transparent Data Encryption keystore operations.

If you want a separate group for Transparent Data Encryption, then create the OSKMDBA group using the group name `kmdba` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54326 kmdba
```

Creating the OSRACDBA Group for Database Installations

You must designate a group as the OSRACDBA group during database installation. Members of this group are granted the SYSRAC privileges to perform day–to–day administration of Oracle databases on an Oracle RAC cluster.

Create the OSRACDBA group using the groups name `racdba` unless a group with that name already exists:

```
# /usr/sbin/groupadd -g 54330 racdba
```

Creating Operating System Oracle Installation User Accounts

Before starting installation, create Oracle software owner user accounts, and configure their environments.

Oracle software owner user accounts require resource settings and other environment configuration. To protect against accidents, Oracle recommends that you create one software installation owner account for each Oracle software program you install.

- Creating an Oracle Software Owner User
  If the Oracle software owner user (`oracle` or `grid`) does not exist, or if you require a new Oracle software owner user, then create it as described in this section.
- Environment Requirements for Oracle Software Owners
  You must make the following changes to configure Oracle software owner environments:
- Procedure for Configuring Oracle Software Owner Environments
  Configure each Oracle installation owner user account environment:
- Modifying Oracle Owner User Groups
  If you have created an Oracle software installation owner account, but it is not a member of the groups you want to designate as the OSDBA, OSOPER, OSDBA for ASM, ASMADMIN, or other system privileges group, then modify the group settings for that user before installation.
- Checking Resource Limits for Oracle Software Installation Users
  For each installation software owner user account, check the resource limits for installation.
- Setting Remote Display and X11 Forwarding Configuration
  If you are on a remote terminal, and the local system has only one visual (which is typical), then use the following syntax to set your user account `DISPLAY` environment variable:

- Preventing Installation Errors Caused by Terminal Output Commands
  During an Oracle Grid Infrastructure installation, OUI uses SSH to run commands and copy files to the other nodes. During the installation, hidden files on the system (for example, `.bashrc` or `.cshrc`) can cause `makefile` and other installation errors if they contain terminal output commands.

Creating an Oracle Software Owner User

If the Oracle software owner user (`oracle` or `grid`) does not exist, or if you require a new Oracle software owner user, then create it as described in this section.

The following example shows how to create the user `oracle` with the user ID 54321; with the primary group `oinstall`; and with secondary groups `dba`, `asmdba`, `backupdba`, `dgdba`, `kmdba`, and `racdba`:

```
# /usr/sbin/useradd -u 54321 -g oinstall -G
dba,asmdba,backupdba,dgdba,kmdba,racdba oracle
```

The following example shows how to create the user `grid` with the user ID 54331; with the primary group `oinstall`; and with secondary groups `dba`, `asmdba`, `backupdba`, `dgdba`, `kmdba`, and `racdba`:

```
# /usr/sbin/useradd -u 54331 -g oinstall -G
dba,asmdba,backupdba,dgdba,kmdba,racdba grid
```

You must note the user ID number for installation users, because you need it during preinstallation.

For Oracle Grid Infrastructure installations, user IDs and group IDs must be identical on all candidate nodes.

Environment Requirements for Oracle Software Owners

You must make the following changes to configure Oracle software owner environments:

- Set the installation software owner user (`grid`, `oracle`) default file mode creation mask (umask) to 022 in the shell startup file. Setting the mask to 022 ensures that the user performing the software installation creates files with 644 permissions.
- Set ulimit settings for file descriptors and processes for the installation software owner (`grid`, `oracle`).
- Set the DISPLAY environment variable in preparation for running an Oracle Universal Installer (OUI) installation.

> **Caution:**
>
> If you have existing Oracle installations that you installed with the user ID that is your Oracle Grid Infrastructure software owner, then unset all Oracle environment variable settings for that user.

Procedure for Configuring Oracle Software Owner Environments

Configure each Oracle installation owner user account environment:

1. Start an X terminal session (`xterm`) on the server where you are running the installation.
2. Enter the following command to ensure that X Window applications can display on this system, where hostname is the fully qualified name of the local host from which you are accessing the server:

   ```
   $ xhost + hostname
   ```

3. If you are not logged in as the software owner user, then switch to the software owner user you are configuring. For example, with the user `grid`:

   ```
   $ su - grid
   ```

   On systems where you cannot run `su` commands, use `sudo` instead:

   ```
   $ sudo -u grid -s
   ```

4. To determine the default shell for the user, enter the following command:

   ```
   $ echo $SHELL
   ```

5. Open the user's shell startup file in any text editor:
   - Bash shell (`bash`):

     ```
     $ vi .bash_profile
     ```

   - Bourne shell (`sh`) or Korn shell (`ksh`):

     ```
     $ vi .profile
     ```

   - C shell (`csh` or `tcsh`):

     ```
     % vi .login
     ```

6. Enter or edit the following line, specifying a value of `022` for the default file mode creation mask:

   ```
   umask 022
   ```

7. If the `ORACLE_SID`, `ORACLE_HOME`, or `ORACLE_BASE` environment variables are set in the file, then remove these lines from the file.
8. Save the file, and exit from the text editor.
9. To run the shell startup script, enter one of the following commands:
   - Bash shell:

     ```
     $ . ./.bash_profile
     ```

- Bourne, Bash, or Korn shell:

  ```
  $ . ./.profile
  ```

- C shell:

  ```
  % source ./.login
  ```

10. Use the following command to check the PATH environment variable:

    ```
    $ echo $PATH
    ```

    Remove any Oracle environment variables.

11. Unset any Oracle environment variables.

    If you have an existing Oracle software installation, and you are using the same user to install this installation, then unset the $ORACLE_HOME, $ORA_NLS10, and $TNS_ADMIN environment variables.

    If you have set $ORA_CRS_HOME as an environment variable, then unset it before starting an installation or upgrade. Do not use $ORA_CRS_HOME as a user environment variable, except as directed by Oracle Support.

12. If you are not installing the software on the local system, then enter a command similar to the following to direct X applications to display on the local system:

    - Bourne, Bash, or Korn shell:

      ```
      $ export DISPLAY=local_host:0.0
      ```

    - C shell:

      ```
      % setenv DISPLAY local_host:0.0
      ```

    In this example, `local_host` is the host name or IP address of the system (your workstation, or another client) on which you want to display the installer.

13. If the `/tmp` directory has less than 1 GB of free space, then identify a file system with at least 1 GB of free space and set the `TMP` and `TMPDIR` environment variables to specify a temporary directory on this file system:

    > **Note:**
    >
    > You cannot use a shared file system as the location of the temporary file directory (typically `/tmp`) for Oracle RAC installations. If you place `/tmp` on a shared file system, then the installation fails.

    a. Use the `df -h` command to identify a suitable file system with sufficient free space.

b. If necessary, enter commands similar to the following to create a temporary directory on the file system that you identified, and set the appropriate permissions on the directory:

```
$ sudo -s
# mkdir /mount_point/tmp
# chmod 775 /mount_point/tmp
# exit
```

c. Enter commands similar to the following to set the `TMP` and `TMPDIR` environment variables:

Bourne, Bash, or Korn shell:

```
$ TMP=/mount_point/tmp
$ TMPDIR=/mount_point/tmp
$ export TMP TMPDIR
```

C shell:

```
% setenv TMP /mount_point/tmp
% setenv TMPDIR /mount_point/tmp
```

14. To verify that the environment has been set correctly, enter the following commands:

```
$ umask
$ env | more
```

Verify that the `umask` command displays a value of `22`, `022`, or `0022` and that the environment variables you set in this section have the correct values.

Modifying Oracle Owner User Groups

If you have created an Oracle software installation owner account, but it is not a member of the groups you want to designate as the OSDBA, OSOPER, OSDBA for ASM, ASMADMIN, or other system privileges group, then modify the group settings for that user before installation.

> **Warning:**
>
> Each Oracle software owner must be a member of the same central inventory group. Do not modify the primary group of an existing Oracle software owner account, or designate different groups as the OINSTALL group. If Oracle software owner accounts have different groups as their primary group, then you can corrupt the central inventory.

During installation, the user that is installing the software should have the OINSTALL group as its primary group, and it must be a member of the operating system groups appropriate for your installation. For example:

```
# /usr/sbin/usermod -g oinstall -G
dba,asmdba,backupdba,dgdba,kmdba,racdba[,oper] oracle
```

Checking Resource Limits for Oracle Software Installation Users

For each installation software owner user account, check the resource limits for installation.

On Oracle Linux systems, Oracle recommends that you install Oracle Database Preinstallation RPMs to meet preinstallation requirements like configuring your operating system to set the resource limits in the `limits.conf` file. Oracle Database Preinstallation RPM only configures the `limits.conf` file for the `oracle` user. If you are implementing Oracle Grid Infrastructure job role separation, then copy the values from the `oracle` user to the `grid` user in the `limits.conf` file.

Use the following ranges as guidelines for resource allocation to Oracle installation owners:

Table 5-1 Installation Owner Resource Limit Recommended Ranges

Resource Shell Limit	Resource	Soft Limit	Hard Limit
Open file descriptors	nofile	at least 1024	at least 65536
Number of processes available to a single user	nproc	at least 2047	at least 16384
Size of the stack segment of the process	stack	at least 10240 KB	at least 10240 KB, and at most 32768 KB
Maximum locked memory limit	memlock	at least 90 percent of the current RAM when HugePages memory is enabled and at least 3145728 KB (3 GB) when HugePages memory is disabled	at least 90 percent of the current RAM when HugePages memory is enabled and at least 3145728 KB (3 GB) when HugePages memory is disabled

To check resource limits:

1. Log in as an installation owner.
2. Check the soft and hard limits for the file descriptor setting. Ensure that the result is in the recommended range. For example:

```
$ ulimit -Sn
1024
$ ulimit -Hn
65536
```

3. Check the soft and hard limits for the number of processes available to a user. Ensure that the result is in the recommended range. For example:

```
$ ulimit -Su
2047
$ ulimit -Hu
16384
```

4. Check the soft limit for the stack setting. Ensure that the result is in the recommended range. For example:

```
$ ulimit -Ss
10240
$ ulimit -Hs
32768
```

5. Repeat this procedure for each Oracle software installation owner.

If necessary, update the resource limits in the `/etc/security/limits.conf` configuration file for the installation owner. However, the configuration file may be distribution specific. Contact your system administrator for distribution specific configuration file information.

> **Note:**
>
> If you make changes to an Oracle installation user account and that user account is logged in, then changes to the `limits.conf` file do not take effect until you log these users out and log them back in. You must do this before you use these accounts for installation.

Setting Remote Display and X11 Forwarding Configuration

If you are on a remote terminal, and the local system has only one visual (which is typical), then use the following syntax to set your user account `DISPLAY` environment variable:

Remote Display

```
Bourne, Korn, and Bash shells

$ export DISPLAY=hostname:0


C shell

% setenv DISPLAY hostname:0
```

For example, if you are using the Bash shell and if your host name is `local_host`, then enter the following command:

```
$ export DISPLAY=node1:0
```

X11 Forwarding

To ensure that X11 forwarding does not cause the installation to fail, use the following procedure to create a user-level SSH client configuration file for Oracle installation owner user accounts:

1. Using any text editor, edit or create the software installation owner's `~/.ssh/config file`.
2. Ensure that the `ForwardX11` attribute in the `~/.ssh/config` file is set to `no`. For example:

   ```
   Host *
       ForwardX11 no
   ```

3. Ensure that the permissions on `~/.ssh` are secured to the Oracle installation owner user account. For example:

   ```
   $ ls -al .ssh
   total 28
   drwx------  2 grid oinstall 4096 Jun 21 2020
   drwx------ 19 grid oinstall 4096 Jun 21 2020
   -rw-r--r--  1 grid oinstall 1202 Jun 21 2020 authorized_keys
   -rwx------  1 grid oinstall  668 Jun 21 2020 id_dsa
   -rwx------  1 grid oinstall  601 Jun 21 2020 id_dsa.pub
   -rwx------  1 grid oinstall 1610 Jun 21 2020 known_hosts
   ```

Preventing Installation Errors Caused by Terminal Output Commands

During an Oracle Grid Infrastructure installation, OUI uses SSH to run commands and copy files to the other nodes. During the installation, hidden files on the system (for example, `.bashrc` or `.cshrc`) can cause `makefile` and other installation errors if they contain terminal output commands.

To avoid this problem, you must modify hidden files in each Oracle installation owner user home directory to suppress all output on `STDOUT` or `STDERR` (for example, `stty`, `xtitle`, and other such commands) as in the following examples:

Bourne, Bash, or Korn shell:

```
if [ -t 0 ]; then
   stty intr ^C
fi
```

C shell:

```
test -t 0
if ($status == 0) then
   stty intr ^C
endif
```

> **Note:**
>
> If the remote shell can load hidden files that contain `stty` commands, then OUI indicates an error and stops the installation.

Creating Oracle Database Vault User Accounts

If you intend to use Oracle Database Vault by default, then you must create an Oracle Database Vault user account, and configure that user.

You must create the Database Vault Owner account before you can use Oracle Database Vault. You can also create a Database Vault Account Manager administrative account.

Oracle Database Vault installs a baseline database auditing policy. This policy covers the access control configuration information stored in Oracle Database Vault database tables, information stored in Oracle Catalog (rollback segments, tablespaces, and so on), the use of system privileges, and Oracle Label Security configuration. When you install Oracle Database Vault, the security specific database initialization parameters are initialized with default values.

Related Topics

- *Oracle Database Vault Administrator's Guide*

Unsetting Oracle Installation Owner Environment Variables

Unset Oracle installation owner environment variables before you start the installation.

The environment variables you have set for the Oracle installation owner account you use to run the installation can cause issues if they are set to values that conflict with the values needed for installation.

If you have set ORA_CRS_HOME as an environment variable, following instructions from Oracle Support, then unset it before starting an installation or upgrade. You should never use ORA_CRS_HOME as an environment variable except under explicit direction from Oracle Support.

If you have had an existing installation on your system, and you are using the same user account to install this installation, then unset the following environment variables: ORA_CRS_HOME, ORACLE_HOME, ORA_NLS10, TNS_ADMIN, and any other environment variable set for the Oracle installation user that is connected with Oracle software homes.

Also, ensure that the `$ORACLE_HOME/bin` path is removed from your PATH environment variable.

6
Configuring Networks for Oracle Database

If you install Oracle Databases on servers with multiple Oracle homes, multiple aliases, or without a static IP address, then review these network configuration topics.

If you are installing Oracle Database on a server with a static host name and IP address and at least one network interface, then no special network configuration is required.

- About Oracle Database Network Configuration Options
  You can enable database clients to connect to servers associated with multiple IP addresses, and you can install Oracle Database on servers with no network connections, and set up database services after installation.
- About Assigning Global Database Names During Installation
  The database name input field is used to set the `DB_NAME`, `DB_UNIQUE_NAME`, and `DB_DOMAIN` Oracle initialization parameter values.
- Network Configuration for Computers Completed After Installation
  You must confirm that a non-networked computer can connect to itself to ensure that you can configure client network resolution after installation. A non-networked computer is a computer that does not have a fixed network address, such as a computer using DHCP.
- Network Configuration for Multihome Computers
  You must set the `ORACLE_HOSTNAME` environment variable to install Oracle Database on a multihomed computer. A multihomed computer is associated with multiple IP addresses.
- Setting the ORACLE_HOSTNAME Environment Variable
  Run the commands shown in this example as the Oracle user account to set the `ORACLE_HOSTNAME` environment variable.
- Network Configuration for Computers with Multiple Aliases
  You must set the `ORACLE_HOSTNAME` environment variable to install Oracle Database on a multialias computer. A multialias computer is one to which multiple aliases resolve.

About Oracle Database Network Configuration Options

You can enable database clients to connect to servers associated with multiple IP addresses, and you can install Oracle Database on servers with no network connections, and set up database services after installation.

Typically, the computer on which you want to install Oracle Database is a server running a single database instance, with a single host name that is resolvable on a network. Oracle Universal Installer uses the host name and Oracle Database instance information to set up network services automatically. The database provides database services to clients using a connect descriptor that resolves to the host name where the database instance is running.

However, you can configure Oracle Database on servers with the following nonstandard configurations:

- **Multihomed Computers**: Servers with multiple Oracle Database installations
- **Multiple Alias Computers**: Servers with multiple aliases, so that more than one host name resolves to the computer

- **Non-Networked computers**: Servers that do not have network connectivity at the time of installation

About Assigning Global Database Names During Installation

The database name input field is used to set the `DB_NAME`, `DB_UNIQUE_NAME`, and `DB_DOMAIN` Oracle initialization parameter values.

The Oracle Database software identifies a database by its global database name. A global database name consists of the database name and database domain. Usually, the database domain is the same as the network domain, but it need not be. The global database name uniquely distinguishes a database from any other database in the same network. You specify the global database name when you create a database during the installation or using the Oracle Database Configuration Assistant.

```
sales.us.example.com
```

Here:

- `sales.us` is the name of the database. The database name, `DB_UNIQUE_NAME`, portion is a string of no more than 30 characters that can contain alphanumeric characters, underscore (_), dollar sign ($), and pound sign (#) but must begin with an alphabetic character. No other special characters are permitted in a database name.
- `sales.us` is also the `DB_NAME`. The `DB_NAME` initialization parameter specifies a database identifier of up to eight characters.
- `example.com` is the database domain in which the database is located. In this example, the database domain equals the network domain. Together, the database name and the database domain make the global database name unique. The domain portion is a string of no more than 128 characters that can contain alphanumeric characters, underscore (_), and pound sign (#). The `DB_DOMAIN` initialization parameter specifies the database domain name.

However, the `DB_NAME` parameter need not necessarily be the first eight characters of `DB_UNIQUE_NAME`.

The `DB_NAME` parameter and the `DB_DOMAIN` parameter combine to create the global database name value.

The system identifier (SID) identifies a specific database instance. The SID uniquely distinguishes the instance from any other instance on the same computer. Each database instance requires a unique SID and database name. In most cases, the SID equals the database name portion of the global database name.

Related Topics

- *Oracle Database Reference*
- *Oracle Database Administrator's Guide*

Network Configuration for Computers Completed After Installation

You must confirm that a non-networked computer can connect to itself to ensure that you can configure client network resolution after installation. A non-networked computer is a computer that does not have a fixed network address, such as a computer using DHCP.

You can install Oracle Database on a non-networked computer. If the computer, such as a laptop, is configured for DHCP and you plan to connect the computer to the network after the Oracle Database installation, then use the `ping` command on the computer on which you want to install the database to check if the computer can connect to itself. Perform this step by first using only the host name and then using the fully qualified name, which should be in the `/etc/hosts` file.

If you connect the computer to a network after installation, then the Oracle Database instance on the computer can work with other instances on the network. The computer can use a static IP or DHCP, depending on the network to which you are connected.

When you run the `ping` command on the computer itself, the ping command should return the IP address of the computer. If the `ping` command fails, then contact your network administrator.

Network Configuration for Multihome Computers

You must set the `ORACLE_HOSTNAME` environment variable to install Oracle Database on a multihomed computer. A multihomed computer is associated with multiple IP addresses.

Typically, a server configured to run multiple Oracle Database Oracle homes is configured with multiple network interface cards. A host name resolves to an IP address configured for one network card for each Oracle Database. You can also set up aliases for host names. By default, during installation, Oracle Universal Installer uses the value set for the environment variable `ORACLE_HOSTNAME` set for the Oracle installation user account running the installation to find the host name. If the user environment variable `ORACLE_HOSTNAME` is not set for the Oracle user, and you are installing on a computer that has multiple network cards, then Oracle Universal Installer determines the host name from the `/etc/hosts` file and the information you provide during the installation session.

Oracle Database clients connecting to the database must be able to access the computer by using either the alias for the host name, or by using the host name associated with that instance. To verify that the client can resolve to the database using both alias and host name, use the `ping` command to check connectivity to the host name both for the database on the server (host name only), and for the fully qualified domain name (host name and domain name).

> **Note:**
>
> Clients must be able to obtain a response using the `ping` command both for the host name and for the fully qualified domain name. If either test fails, then contact your network administrator to resolve the issue.

Setting the ORACLE_HOSTNAME Environment Variable

Run the commands shown in this example as the Oracle user account to set the `ORACLE_HOSTNAME` environment variable.

The following example shows the commands to run on the Oracle user account to set the ORACLE_HOSTNAME environment variable. In this example, the fully qualified host name is `somehost.example.com`.

Bourne, Bash or Korn Shell

```
$ ORACLE_HOSTNAME=somehost.example.com
$ export ORACLE_HOSTNAME
```

C Shell

```
% setenv ORACLE_HOSTNAME somehost.example.com
```

Network Configuration for Computers with Multiple Aliases

You must set the `ORACLE_HOSTNAME` environment variable to install Oracle Database on a multialias computer. A multialias computer is one to which multiple aliases resolve.

A computer with multiple aliases is a computer that is registered with the naming service under a single IP address, but which resolves multiple aliases to that address. The naming service resolves any of those aliases to the same computer. Before installing Oracle Database on such a computer, set the Oracle installation owner environment variable `ORACLE_HOSTNAME` to the computer whose host name you want to use.

7

Supported Storage Options for Oracle Database and Oracle Grid Infrastructure

Review supported storage options as part of your installation planning process.

- Supported Storage Options for Oracle Database
  The following table shows the storage options supported for Oracle Database binaries and files:
- About Oracle Grid Infrastructure for a Standalone Server
  If you plan to use Oracle Automatic Storage Management (Oracle ASM), then you must install Oracle Restart before installing your database.
- Oracle ACFS and Oracle ADVM
  Oracle Automatic Storage Management Cluster File System (Oracle ACFS) extends Oracle ASM technology to support of all of your application data in both single instance and cluster configurations.
- File System Options for Oracle Database
  If you install Oracle Database files on a file system, then Oracle Database Configuration Assistant creates the database files in a directory on a file system mounted on the computer.
- Guidelines for Placing Oracle Database Files On a File System or Logical Volume
  If you choose to place the Oracle Database files on a file system, then use the following guidelines when deciding where to place them:
- About NFS Storage for Data Files
  Review this section for NFS storage configuration guidelines.
- About Direct NFS Client Mounts to NFS Storage Devices
  Direct NFS Client integrates the NFS client functionality directly in the Oracle software to optimize the I/O path between Oracle and the NFS server. This integration can provide significant performance improvements.

Supported Storage Options for Oracle Database

The following table shows the storage options supported for Oracle Database binaries and files:

Table 7-1 Supported Storage Options for Oracle Database

Storage Option	Oracle Database Binaries	Oracle Database Data Files	Oracle Database Recovery Files
Oracle Automatic Storage Management (Oracle ASM)	No	Yes	Yes

Table 7-1 (Cont.) Supported Storage Options for Oracle Database

Storage Option	Oracle Database Binaries	Oracle Database Data Files	Oracle Database Recovery Files
Oracle Automatic Storage Management Cluster File System (Oracle ACFS)	Yes	Yes	Yes
Local file system	Yes	Yes, but not recommended	Yes, but not recommended
Network file system (NFS) on a certified network-attached storage (NAS) filer	Yes	Yes	Yes

Guidelines for Storage Options

Use the following guidelines when choosing storage options:

- Oracle strongly recommends that you use a dedicated set of disks for Oracle ASM.
- Loopback devices are not supported for use with Oracle ASM.
- You can choose any combination of the supported storage options for each file type provided that you satisfy all requirements listed for the chosen storage options.
- You can use Oracle ASM to store Oracle Clusterware files.
- Direct use of raw or block devices is not supported. You can only use raw or block devices under Oracle ASM.

Related Topics

- *Oracle Database Upgrade Guide*

About Oracle Grid Infrastructure for a Standalone Server

If you plan to use Oracle Automatic Storage Management (Oracle ASM), then you must install Oracle Restart before installing your database.

Oracle Grid Infrastructure for a standalone server is a version of Oracle Grid Infrastructure that supports single instance databases. This support includes volume management, file system, and automatic restart capabilities. Oracle Grid Infrastructure for a standalone server includes Oracle Restart and Oracle Automatic Storage Management. Oracle combined the two infrastructure products into a single set of binaries that is installed into an Oracle Restart home.

Oracle Restart is a feature provided as part of Oracle Grid Infrastructure. Oracle Restart monitors and can restart Oracle Database instances, Oracle Net Listeners, and Oracle ASM instances. Oracle Restart is currently restricted to manage single instance Oracle Databases and Oracle ASM instances only.

Oracle Automatic Storage Management is a volume manager and a file system for Oracle Database files that supports single-instance Oracle Database and Oracle Real Application Clusters (Oracle RAC) configurations. Oracle Automatic Storage

Management also supports a general purpose file system for your application needs, including Oracle Database binaries. Oracle Automatic Storage Management is Oracle's recommended storage management solution that provides an alternative to conventional volume managers, and file systems.

Oracle Restart improves the availability of your Oracle database by providing the following services:

- When there is a hardware or a software failure, Oracle Restart automatically starts all Oracle components, including the Oracle database instance, Oracle Net Listener, database services, and Oracle ASM.
- Oracle Restart starts components in the proper order when the database host is restarted.
- Oracle Restart runs periodic checks to monitor the status of Oracle components. If a check operation fails for a component, then the component is shut down and restarted.

Note the following restrictions for using Oracle Restart:

- You can neither install Oracle Restart on an Oracle Grid Infrastructure cluster member node, nor add an Oracle Restart server to an Oracle Grid Infrastructure cluster member node. Oracle Restart supports single-instance databases on one server, while Oracle Grid Infrastructure for a Cluster supports single-instance or Oracle RAC databases on a cluster.
- If you want to use Oracle ASM or Oracle Restart, then you should install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must install Oracle Restart, and then manually register the database with Oracle Restart.
- You can use the Oracle Restart implementation of Oracle Grid Infrastructure only in single-instance (nonclustered) environments. Use Oracle Grid Infrastructure with Oracle Clusterware for clustered environments.

Oracle ACFS and Oracle ADVM

Oracle Automatic Storage Management Cluster File System (Oracle ACFS) extends Oracle ASM technology to support of all of your application data in both single instance and cluster configurations.

Oracle Automatic Storage Management Dynamic Volume Manager (Oracle ADVM) provides volume management services and a standard disk device driver interface to clients. Oracle ACFS communicates with Oracle ASM through the Oracle ADVM interface.

- Oracle ACFS and Oracle ADVM Support on Linux x86–64
  Oracle ACFS and Oracle ADVM are supported on Oracle Linux, Red Hat Enterprise Linux, and SUSE Linux Enterprise Server.
- Restrictions and Guidelines for Oracle ACFS
  Review these topics as part of your storage plan for using Oracle ACFS for single instance and cluster configurations.

Related Topics

- *Oracle Automatic Storage Management Cluster File System Administrator's Guide*

Oracle ACFS and Oracle ADVM Support on Linux x86–64

Oracle ACFS and Oracle ADVM are supported on Oracle Linux, Red Hat Enterprise Linux, and SUSE Linux Enterprise Server.

Table 7-2 Platforms That Support Oracle ACFS and Oracle ADVM

Platform / Operating System	Kernel
Oracle Linux 7	All Oracle Linux 7 compatible kernels
Oracle Linux 8	All Oracle Linux 8 compatible kernels
Red Hat Enterprise Linux 8	All Red Hat Enterprise Linux 8 compatible kernels
SUSE Linux Enterprise Server 15	All SUSE Linux Enterprise Server 15 compatible kernels
Linux Containers	Not supported

Note:

If you use Security Enhanced Linux (SELinux) in enforcing mode with Oracle ACFS, then ensure that you mount the Oracle ACFS file systems with an SELinux default context. Refer to your Linux vendor documentation for information about the context mount option.

Important:

You must apply patches to some of the Linux kernel versions for successful Oracle Grid Infrastructure installation. Refer to the following notes for more information:

- My Oracle Support Note 1369107.1 for more information and a complete list of platforms and releases that support Oracle ACFS and Oracle ADVM:

  https://support.oracle.com/rs?type=doc&id=1369107.1

- Patch Set Updates for Oracle Products (My Oracle Support Note 854428.1) for current release and support information:

  https://support.oracle.com/rs?type=doc&id=854428.1

Restrictions and Guidelines for Oracle ACFS

Review these topics as part of your storage plan for using Oracle ACFS for single instance and cluster configurations.

- Oracle Automatic Storage Management Cluster File System (Oracle ACFS) provides a general purpose file system.
- You can only use Oracle ACFS when Oracle ASM is configured.

- Note the following general guidelines and restrictions for placing Oracle Database and Oracle Grid Infrastructure files on Oracle ACFS:
  - You can place Oracle Database binaries, data files, and administrative files (for example, trace files) on Oracle ACFS.
  - Oracle ACFS does not support encryption or replication with Oracle Database data files, tablespace files, control files, redo logs, archive logs, RMAN backups, Data Pump dumpsets, and flashback files.
  - You can place Oracle Database homes on Oracle ACFS only if the database release is Oracle Database 11g Release 2, or more recent releases. You cannot install earlier releases of Oracle Database on Oracle ACFS.
  - For installations on Oracle Clusterware, you cannot place Oracle Clusterware files on Oracle ACFS.
- Oracle Restart does not support root-based Oracle Clusterware resources. For this reason, the following restrictions apply if you run Oracle ACFS on an Oracle Restart Configuration:
  - Starting with Oracle Database 18c, configuration assistants do not allow the creation of Oracle Database homes on Oracle ACFS in an Oracle Restart configuration.
  - Oracle Restart does not support Oracle ACFS resources on all platforms.
  - Starting with Oracle Database 12c, Oracle Restart configurations do not support the Oracle ACFS registry.
  - On Linux, Oracle ACFS provides an automated mechanism to load and unload drivers and mount and unmount Oracle ACFS file systems on system restart and shutdown. However, Oracle ACFS does not provide automated recovery of mounted file systems when the system is running. Other than Linux, Oracle ACFS does not provide this automated mechanism on other operating systems.
  - Creating Oracle data files on an Oracle ACFS file system is not supported in Oracle Restart configurations. Creating Oracle data files on an Oracle ACFS file system is supported on Oracle Grid Infrastructure for a cluster configurations.
- Oracle ACFS and Oracle ADVM are not supported on IBM AIX Workload Partitions (WPARs).

Related Topics

- *Oracle Automatic Storage Management Administrator's Guide*

File System Options for Oracle Database

If you install Oracle Database files on a file system, then Oracle Database Configuration Assistant creates the database files in a directory on a file system mounted on the computer.

Oracle recommends that the file system be separate from the file systems used by the operating system or the Oracle Database software.

The file system can be any of the following:

Standard Oracle Database Creation Options

- A file system on a disk that is physically attached to the system.

If you are creating a database on basic disks that are not logical volumes or RAID devices, then Oracle recommends that you follow the Optimal Flexible Architecture (OFA) recommendations and distribute the database files over many disks.

- A file system on a logical volume manager (LVM) volume or a RAID device.

  If you are using multiple disks in an LVM or RAID configuration, then Oracle recommends that you use the Stripe and Mirror Everything (S.A.M.E) methodology to increase performance and reliability. Using this methodology, you do not have to specify multiple file system mount points for the database storage.

- A network file system (NFS) mounted from a certified network-attached storage (NAS) device. You also have the option to use Direct NFS Client, which simplifies the administration of NFS configurations and also improves performance.

  If the NAS device is certified by Oracle, then you can store the database files on them.

Advanced Oracle Database Creation Options

- The three file system options available to standard Oracle Database installations.
- With Oracle Managed Files, you specify file system directories in which the database automatically creates, names, and manages files at the database object level.

  If you use the Oracle Managed Files feature, then you must specify only the database object name instead of file names when creating or deleting database files.

Related Topics

- *Oracle Database Administrator's Guide*

Guidelines for Placing Oracle Database Files On a File System or Logical Volume

If you choose to place the Oracle Database files on a file system, then use the following guidelines when deciding where to place them:

- The default path suggested by Oracle Universal Installer for the database file directory is a subdirectory of the Oracle base directory.
- You can choose either a single file system or more than one file system to store the database files:
  - If you want to use a single file system, then choose a file system on a physical device that is dedicated to the database.

    For best performance and reliability, choose a RAID device or a logical volume on more than one physical device, and implement a stripe-and-mirror-everything (SAME) storage policy.
  - If you want to use more than one file system, then choose file systems on separate physical devices that are dedicated to the database.

    This method enables you to distribute physical input-output operations and create separate control files on different devices for increased reliability. It also enables you to fully implement Oracle Optimal Flexible Architecture (OFA)

guidelines. Choose the `Advanced` database creation option to implement this method.

- If you intend to create a preconfigured database during the installation, then the file system (or file systems) that you choose must have at least 2 GB of free disk space.

  For production databases, you must estimate the disk space requirement depending on the use of the database.
- For optimum performance, the file systems that you choose must be on physical devices that are used only by the database.
- The Oracle user running the Oracle Database installation must have write permissions to create the files in the path that you specify.

About NFS Storage for Data Files

Review this section for NFS storage configuration guidelines.

Network-Attached Storage and NFS Protocol

Network-attached storage (NAS) systems use the network file system (NFS) protocol to to access files over a network, which enables client servers to access files over networks as easily as to storage devices attached directly to the servers. You can store data files on supported NFS systems. NFS is a shared file system protocol, so NFS can support both single instance and Oracle Real Application Clusters databases.

> **Note:**
>
> The performance of Oracle software and databases stored on NAS devices depends on the performance of the network connection between the servers and the network-attached storage devices.For better performance, Oracle recommends that you connect servers to NAS devices using private dedicated network connections. NFS network connections should use Gigabit Ethernet or better.

Refer to your vendor documentation to complete NFS configuration and mounting.

Requirements for Using NFS Storage

Before you start installation, NFS file systems must be mounted and available to servers.

About Direct NFS Client Mounts to NFS Storage Devices

Direct NFS Client integrates the NFS client functionality directly in the Oracle software to optimize the I/O path between Oracle and the NFS server. This integration can provide significant performance improvements.

Direct NFS Client supports NFSv3, NFSv4, NFSv4.1, and pNFS protocols to access the NFS server. Direct NFS Client also simplifies, and in many cases automates, the performance optimization of the NFS client configuration for database workloads.

Starting with Oracle Database 12c Release 2, when you enable Direct NFS, you can also enable the Direct NFS dispatcher. The Direct NFS dispatcher consolidates the number of TCP connections that are created from a database instance to the NFS server. In large

database deployments, using Direct NFS dispatcher improves scalability and network performance. Parallel NFS deployments also require a large number of connections. Hence, the Direct NFS dispatcher is recommended with Parallel NFS deployments too.

Direct NFS Client can obtain NFS mount points either from the operating system mount entries, or from the `oranfstab` file.

Direct NFS Client Requirements

- NFS servers must have write size values (`wtmax`) of 32768 or greater to work with Direct NFS Client.
- NFS mount points must be mounted both by the operating system kernel NFS client and Direct NFS Client, even though you configure Direct NFS Client to provide file service.

  If Oracle Database cannot connect to an NFS server using Direct NFS Client, then Oracle Database connects to the NFS server using the operating system kernel NFS client. When Oracle Database fails to connect to NAS storage though Direct NFS Client, it logs an informational message about the Direct NFS Client connect error in the Oracle alert and trace files.
- Follow standard guidelines for maintaining integrity of Oracle Database files mounted by both operating system NFS and by Direct NFS Client.

Direct NFS Mount Point Search Order

Direct NFS Client searches for mount entries in the following order:

1. `ORACLE_BASE_HOME/dbs/oranfstab`
2. `ORACLE_BASE_CONFIG/dbs/oranfstab`
3. `/etc/oranfstab`
4. `/etc/mtab`

Direct NFS Client uses the first matching entry as the mount point.

> **Note:**
>
> You can have only one active NFS Client implementation for each instance. Enabling Direct NFS Client on an instance prevents you from using another NFS Client implementation, such as kernel NFS Client.

- To print the ORACLE_BASE_HOME path, run:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabasehome
```

- To print the ORACLE_BASE_CONFIG path, run:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabaseconfig
```

Related Topics

- About Oracle Base Home
  In a read-only ORACLE_HOME the user-specific files, instance-specific files, and log files reside in a location known as the ORACLE_BASE_HOME.
- About Oracle Base Config
  In a read-only ORACLE_HOME the configuration files reside in a location known as ORACLE_BASE_CONFIG.
- Creating an oranfstab File for Direct NFS Client
  Direct NFS uses a configuration file, `oranfstab`, to determine the available mount points.
- Configuring NFS Buffer Size Parameters for Oracle Database
  Set the values for the NFS buffer size parameters `rsize` and `wsize` to at least 32768.

See Also:

- *Oracle Database Reference* for information about setting the `enable_dnfs_dispatcher` parameter in the initialization parameter file to enable Direct NFS dispatcher
- *Oracle Database Performance Tuning Guide* for performance benefits of enabling Parallel NFS and Direct NFS dispatcher
- *Oracle Automatic Storage Management Administrator's Guide* for guidelines about managing Oracle Database data files created with Direct NFS Client or kernel NFS

8

Configuring File System Storage for Oracle Database

Complete these procedures to use file system storage for Oracle Database.

For optimal database organization and performance, Oracle recommends that you install data files and the Oracle Database software in different disks.

- Deploying Oracle Direct NFS
  If you plan to place storage on Network File System (NFS) protocol devices, then Oracle recommends that you use Oracle Direct NFS (dNFS) to take advantage of performance optimizations built into the Oracle Direct NFS client.
- Deploying a DAX-Enabled File System for Persistent Memory Database
  Learn how to deploy a DAX-enabled file system for Persistent Memory Database.

Deploying Oracle Direct NFS

If you plan to place storage on Network File System (NFS) protocol devices, then Oracle recommends that you use Oracle Direct NFS (dNFS) to take advantage of performance optimizations built into the Oracle Direct NFS client.

- Configuring NFS Buffer Size Parameters for Oracle Database
  Set the values for the NFS buffer size parameters `rsize` and `wsize` to at least 32768.
- Checking TCP Network Protocol Buffer for Direct NFS Client
  Check your TCP network buffer size to ensure that it is adequate for the speed of your servers.
- Creating an oranfstab File for Direct NFS Client
  Direct NFS uses a configuration file, `oranfstab`, to determine the available mount points.
- Enabling and Disabling Direct NFS Client Control of NFS
  By default, Direct NFS Client is installed in a disabled state with single-instance Oracle Database installations. Before enabling Direct NFS, you must configure an `oranfstab` file.
- Enabling Hybrid Columnar Compression on Direct NFS Client
  Perform these steps to enable Hybrid Columnar Compression (HCC) on Direct NFS Client:

Related Topics

- My Oracle Support note 1496040.1

Configuring NFS Buffer Size Parameters for Oracle Database

Set the values for the NFS buffer size parameters `rsize` and `wsize` to at least 32768.

For example, to use `rsize` and `wsize` buffer settings with the value 32768 for an Oracle Database data files mount point, set mount point parameters to values similar to the following:

```
nfs_server:/vol/DATA/oradata /home/oracle/netapp nfs\
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,vers=3,timeo=600
```

Direct NFS Client issues writes at `wtmax` granularity to the NFS server.

Related Topics

- My Oracle Support note 359515.1

Checking TCP Network Protocol Buffer for Direct NFS Client

Check your TCP network buffer size to ensure that it is adequate for the speed of your servers.

By default, the network buffer size is set to 1 MB for TCP, and 2 MB for UDP. The TCP buffer size can set a limit on file transfers, which can negatively affect performance for Direct NFS Client users.

To check the current TCP buffer size, enter the following command:

```
# sysctl -a |grep -e net.ipv4.tcp_[rw]mem
```

The output of this command is similar to the following:

```
net.ipv4.tcp_rmem = 4096        87380   1056768
net.ipv4.tcp_wmem = 4096        16384   1056768
```

Oracle recommends that you set the value based on the link speed of your servers. For example, perform the following steps:

1. As `root`, use a text editor to open `/etc/sysctl.conf`, and add or change the following:

   ```
   net.ipv4.tcp_rmem = 4096        87380   4194304
   net.ipv4.tcp_wmem = 4096        16384   4194304
   ```

2. Apply your changes by running the following command:

   ```
   # sysctl -p
   ```

3. Restart the network:

   ```
   # /etc/rc.d/init.d/network restart
   ```

Creating an oranfstab File for Direct NFS Client

Direct NFS uses a configuration file, `oranfstab`, to determine the available mount points.

Create an `oranfstab` file with the following attributes for each NFS server that you want to access using Direct NFS Client:

- `server`

  The NFS server name.

  For NFS setup with Kerberos authentication, the `server` attribute name must be the fully-qualified name of the NFS server. This `server` attribute name is used to create service principal for Ticket Granting Service (TGS) request from the Kerberos server. If you are configuring external storage snapshot cloning, then the NFS `server` name should be a valid host name. For all other scenarios, the NFS `server` name can be any unique name.

- `local`

  Up to four paths on the database host, specified by IP address or by name, as displayed using the `ifconfig` command run on the database host.

- `path`

  Up to four network paths to the NFS server, specified either by IP address, or by name, as displayed using the `ifconfig` command on the NFS server.

- `export`

  The exported path from the NFS server.

- `mount`

  The corresponding local mount point for the exported volume.

- `mnt_timeout`

  Specifies (in seconds) the time Direct NFS Client should wait for a successful mount before timing out. This parameter is optional. The default timeout is 10 minutes (600).

- `nfs_version`

  Specifies the NFS protocol version used by Direct NFS Client. Possible values are NFSv3, NFSv4, NFSv4.1, and pNFS. The default version is NFSv3. If you select NFSv4.x, then you must configure the value in `oranfstab` for `nfs_version`.

  Specify `nfs_version` as `pNFS`, if you want to use Direct NFS with Parallel NFS. Direct NFS supports only the default `sys` security authentication with Parallel NFS. Direct NFS does not support Parallel NFS when combined with any of the Kerberos authentication parameters.

- `security_default`

  Specifies the default security mode applicable for all the exported NFS server paths for a server entry. This parameter is optional. `sys` is the default value. See the description of the `security` parameter for the supported security levels for the `security_default` parameter.

- `security`

Specifies the security level, to enable security using Kerberos authentication protocol with Direct NFS Client. This optional parameter can be specified per export-mount pair. The supported security levels for the `security_default` and `security` parameters are:

`sys`: UNIX level security AUTH_UNIX authentication based on user identifier (UID) and group identifier (GID) values. This is the default value for security parameters.

`krb5`: Direct NFS runs with plain Kerberos authentication. Server is authenticated as the real server which it claims to be.

`krb5i`: Direct NFS runs with Kerberos authentication and NFS integrity. Server is authenticated and each of the message transfers is checked for integrity.

`krb5p`: Direct NFS runs with Kerberos authentication and NFS privacy. Server is authenticated, and all data is completely encrypted.

The `security` parameter, if specified, takes precedence over the `security_default` parameter. If neither of these parameters are specified, then `sys` is the default authentication.

For NFS server Kerberos security setup, review the relevant NFS server documentation. For Kerberos client setup, review the relevant operating system documentation.

- `dontroute`

  Specifies that outgoing messages should not be routed by the operating system, but instead sent using the IP address to which they are bound.

  > **Note:**
  >
  > The `dontroute` option is a POSIX option, which sometimes does not work on Linux systems with multiple paths in the same subnet.

- `management`

  Enables Direct NFS Client to use the management interface for SNMP queries. You can use this parameter if SNMP is running on separate management interfaces on the NFS server. The default value is the server parameter value.

- `community`

  Specifies the community string for use in SNMP queries. Default value is `public`.

The following examples show three possible NFS server entries in `oranfstab`. A single `oranfstab` can have multiple NFS server entries.

Example 8-1 Using Local and Path NFS Server Entries

The following example uses both local and path. Because they are in different subnets, you do not have to specify `dontroute`.

```
server: MyDataServer1
local: 192.0.2.0
path: 192.0.2.1
local: 192.0.100.0
path: 192.0.100.1
export: /vol/oradata1 mount: /mnt/oradata1
```

Example 8-2 Using Local and Path in the Same Subnet, with dontroute

Local and path in the same subnet, where `dontroute` is specified:

```
server: MyDataServer2
local: 192.0.2.0
path: 192.0.2.128
local: 192.0.2.1
path: 192.0.2.129
dontroute
export: /vol/oradata2 mount: /mnt/oradata2
```

Example 8-3 Using Names in Place of IP Addresses, with Multiple Exports, management and community

```
server: MyDataServer3
local: LocalPath1
path: NfsPath1
local: LocalPath2
path: NfsPath2
local: LocalPath3
path: NfsPath3
local: LocalPath4
path: NfsPath4
dontroute
export: /vol/oradata3 mount: /mnt/oradata3
export: /vol/oradata4 mount: /mnt/oradata4
export: /vol/oradata5 mount: /mnt/oradata5
export: /vol/oradata6 mount: /mnt/oradata6
management: MgmtPath1
community: private
```

Example 8-4 Using Kerberos Authentication with Direct NFS Export

The `security` parameter overrides `security_default`:

```
server:  nfsserver
 local: 192.0.2.0
 path: 192.0.2.2
 local: 192.0.2.3
 path: 192.0.2.4
 export: /private/oracle1/logs  mount: /logs  security: krb5
 export: /private/oracle1/data  mount: /data  security: krb5p
 export: /private/oracle1/archive mount: /archive security: sys
 export: /private/oracle1/data1 mount: /data1
 security_default: krb5i
```

Enabling and Disabling Direct NFS Client Control of NFS

By default, Direct NFS Client is installed in a disabled state with single-instance Oracle Database installations. Before enabling Direct NFS, you must configure an `oranfstab` file.

Use these procedures to enable or disable Direct NFS Client Oracle Disk Manager Control of NFS

Enabling Direct NFS Client Control of NFS

1. Change the directory to `$ORACLE_HOME/rdbms/lib`.
2. Enter the following command:

```
make -f ins_rdbms.mk dnfs_on
```

Disabling Direct NFS Client Control of NFS

1. Log in as the Oracle software installation owner, and disable Direct NFS Client using the following commands:

```
cd $ORACLE_HOME/rdbms/lib
make -f ins_rdbms.mk dnfs_off
```

2. Remove the `oranfstab` file.

> **Note:**
>
> If you remove an NFS path that an Oracle Database is using, then you must restart the database for the change to take effect.

Enabling Hybrid Columnar Compression on Direct NFS Client

Perform these steps to enable Hybrid Columnar Compression (HCC) on Direct NFS Client:

1. Ensure that SNMP is enabled on the ZFS storage server. For example:

```
$ snmpget -v1 -c public server_name .1.3.6.1.4.1.42.2.225.1.4.2.0
SNMPv2-SMI::enterprises.42.2.225.1.4.2.0 = STRING: "Sun Storage
7410"
```

2. If SNMP is enabled on an interface other than the NFS server, then configure `oranfstab` using the management parameter.
3. If SNMP is configured using a community string other than public, then configure `oranfstab` file using the community parameter.
4. Ensure that `libnetsnmp.so` is installed by checking if `snmpget` is available.

Deploying a DAX-Enabled File System for Persistent Memory Database

Learn how to deploy a DAX-enabled file system for Persistent Memory Database.

- About Using a DAX-Enabled File System for Persistent Memory Database
  You must mount XFS in DAX mode for the Persistent Memory (PMEM) database to leverage the byte addressability provided by fsdax.
- Creating a DAX-Enabled File System for Persistent Memory Database
  Learn how to create and use a DAX-enabled file system for your Persistent Memory Database.

About Using a DAX-Enabled File System for Persistent Memory Database

You must mount XFS in DAX mode for the Persistent Memory (PMEM) database to leverage the byte addressability provided by fsdax.

You can use a PMEM filestore for database data files and control files. For performance reasons, Oracle recommends that you store redo log files as independent files in the file system.

To use a PMEM device with Oracle Database, perform the following steps:

1. Configure a PMEM device in the `fsdax` mode.
2. Create a DAX-capable file system (XFS) on the PMEM device.
3. Mount the file system using the `dax` mount option.
4. Start the Oracle Database instance in `NOMOUNT` mode.
5. Run the `CREATE PMEM FILESTORE SQL` command to create and use a file in the DAX-capable file system as a backing file for the PMEM filestore.
6. Create a database (tablespace) on the PMEM filestore.

Requirements

- You must have an Intel motherboard and Intel persistent memory (PMEM) storage device configured on your system.

Related Topics

- Using Persistent Memory Database
- Configuring the PMEM Device

Creating a DAX-Enabled File System for Persistent Memory Database

Learn how to create and use a DAX-enabled file system for your Persistent Memory Database.

You must use an XFS-based DAX file system as the filestore for the PMEM database. Scalability and performance of the Persistent Memory Database is affected without the correct configuration, hence Oracle recommends that you use the XFS file system using the following configuration steps.

This configuration ensures that XFS extents are aligned to 2 MB and you can use 2 MB pages when mapping a file to an Oracle process.

1. Ensure that you have a namespace in FSDAX mode with a 2 MB alignment.

```
$ ndctl list
[
  {
    "dev":"namespace0.0",
    "mode":"fsdax",
    "map":"dev",
    "size":91953823744,
    "uuid":"e11456a7-b26a-4409-b31a-f11b8ee06e5a",
    "sector_size":512,
    "align":2097152,
    "blockdev":"pmem0"
  }
]
```

2. Create two partitions on your PMEM storage device. One small partition of approximately 2% for the metadata and the rest for the data.
3. Ensure that the starting sector of the data partition is aligned to 2 MB. Ensure that the data partition is configured as a real time device when creating the xfs filesystem to guarantee 2 MB aligned extents. The starting sector number of the data partition should be a multiple of 4096 when you run the `fdisk -l` command.

   In this example, the data partition `/dev/pmem0p2` starts at a sector offset that is a multiple of 4096.

```
$ sudo /sbin/fdisk -l /dev/pmem0

Disk /dev/pmem0: 92.0 GB, 91953823744 bytes, 179597312 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disk label type: dos
Disk identifier: 0x1ef2bc18

      Device Boot      Start         End      Blocks   Id  System
/dev/pmem0p1            2048     4196351     2097152   83  Linux
/dev/pmem0p2         4198400   179597311    87699456   83  Linux
```

4. Create an XFS file system with the data partition as the real time device and an extent size of 2 MB.

```
mkfs.xfs -m rmapbt=0,reflink=0 -d rtinherit=1 -r rtdev=/dev/
pmem0p2,extsize=2m /dev/pmem0p1
```

5. Mount the Persistent Memory (PMEM) device with DAX using the following real time device mount options.

```
mount /dev/pmem0p1 /mnt -o dax,rtdev=/dev/pmem0p2
```

9

Configuring Storage for Oracle Grid Infrastructure for a Standalone Server

Complete these procedures to use Oracle Grid Infrastructure for a standalone server, which includes Oracle Automatic Storage Management (Oracle ASM).

Oracle Grid Infrastructure for a standalone server, also known as Oracle Restart, provides system support for a single-instance Oracle Database. Oracle ASM is a volume manager and a file system for Oracle database files that supports single-instance Oracle Database and Oracle Real Application Clusters (Oracle RAC) configurations. Oracle Automatic Storage Management also supports a general purpose file system for your application needs, including Oracle Database binaries. Oracle Automatic Storage Management is Oracle's recommended storage management solution. It provides an alternative to conventional volume managers and file systems.

> **Note:**
>
> - If you want to use Oracle ASM or Oracle Restart, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.
> - You can neither install Oracle Restart on an Oracle Grid Infrastructure cluster member node, nor add an Oracle Restart server to an Oracle Grid Infrastructure cluster member node. Oracle Restart supports single-instance databases on one server, while Oracle Grid Infrastructure for a Cluster supports single-instance or Oracle RAC databases on a cluster.

- Configuring Storage for Oracle Automatic Storage Management
  Identify storage requirements and Oracle ASM disk group options.
- Configuring Storage Device Path Persistence Using Oracle ASMFD
  Oracle ASM Filter Driver (Oracle ASMFD) maintains storage file path persistence and helps to protect files from accidental overwrites.
- Creating DAS or SAN Disk Partitions for Oracle Automatic Storage Management
  You can use direct-attached storage (DAS) and storage area network (SAN) disks with Oracle ASM.
- Creating Directories for Oracle Database Files
  You can store Oracle Database and recovery files on a separate file system from the configuration files.
- Creating Files on a NAS Device for Use with Oracle Automatic Storage Management
  If you have a certified NAS storage device, then you can create zero-padded files in an NFS mounted directory and use those files as disk devices in an Oracle ASM disk group.

Configuring Storage for Oracle Automatic Storage Management

Identify storage requirements and Oracle ASM disk group options.

- Identifying Storage Requirements for Oracle Automatic Storage Management
  To identify the storage requirements for using Oracle ASM, you must determine the number of devices and the amount of free disk space that you require.
- Oracle ASM Disk Space Requirements
  Determine the total amount of Oracle Automatic Storage Management (Oracle ASM) disk space that you require for the database files and recovery files.
- ASM Disk Group Options for Installation
  Plan how you want to configure Oracle ASM disk groups for deployment.
- Using an Existing Oracle ASM Disk Group
  Use Oracle Enterprise Manager Cloud Control or the Oracle ASM command line tool (`asmcmd`) to identify existing disk groups, and to determine if sufficient space is available in the disk group.

Related Topics

- *Oracle Automatic Storage Management Administrator's Guide*

Identifying Storage Requirements for Oracle Automatic Storage Management

To identify the storage requirements for using Oracle ASM, you must determine the number of devices and the amount of free disk space that you require.

To complete this task, perform the following steps:

1. Determine whether you want to use Oracle ASM for Oracle Database files, recovery files, or both. Oracle Database files include data files, control files, redo log files, the server parameter file, and the password file.

   During the database installation, you have the option to select either a file system or Oracle ASM as the storage mechanism for Oracle Database files. Similarly, you also have the option to select either a file system or Oracle ASM as the storage mechanism for your recovery files.

   > **Note:**
   >
   > You do not have to use the same storage mechanism for both Oracle Database files and recovery files. You can use a file system for one file type and Oracle ASM for the other.

   If you select Oracle ASM as your storage option for Oracle Database files, then depending on your choice in the Specify Recovery Options screen, you have the following recovery options:

- If you select the Oracle ASM option for your recovery files, then Oracle Universal Installer provides you with only the option to use the same disk group for both Oracle Database files and recovery files.
- If you decide not to enable recovery during the database installation, then, after the database installation, you can modify the `DB_RECOVERY_FILE_DEST` parameter to enable the fast recovery area.

2. Choose the Oracle ASM redundancy level to use for each Oracle ASM disk group that you create.

   The redundancy level that you choose for the Oracle ASM disk group determines how Oracle ASM mirrors files in the disk group and determines the number of disks and amount of disk space that you require, as follows:

   - **External redundancy**

     This option does not allow Oracle ASM to mirror the contents of the disk group. Oracle recommends that you select this redundancy level either when the disk group contains devices, such as RAID devices, that provide their own data protection or when the database does not require uninterrupted access to data.

   - **Normal redundancy**

     To optimize performance and reliability in a normal redundancy disk group, Oracle ASM uses two-way mirroring for data files and three-way mirroring for control files, by default. In addition, you can choose the mirroring characteristics for individual files in a disk group.

     A normal redundancy disk group requires a minimum of two failure groups (or two disk devices) if you are using two-way mirroring. The effective disk space in a normal redundancy disk group is half the sum of the disk space of all of its devices.

     For most installations, Oracle recommends that you use normal redundancy disk groups. On Oracle Exadata, Oracle recommends that you use high redundancy disk groups for added protection against failure.

   - **High redundancy**

     The contents of the disk group are three-way mirrored by default. To create a disk group with high redundancy, you must specify at least three failure groups (a minimum of three devices).

     Although high-redundancy disk groups provide a high level of data protection, you must consider the higher cost of additional storage devices before deciding to use this redundancy level.

   - **Flex redundancy**

     A flex redundancy disk group is a new disk group type with features such as flexible file redundancy, mirror splitting, and redundancy change. A flex disk group can consolidate files with different redundancy requirements into a single disk group. It also provides the capability for databases to change the redundancy of its files.

     For database data, you can choose no mirroring (unprotected), two-way mirroring (mirrored), or three-way mirroring (high). A flex redundancy disk group requires a minimum of three disk devices (or three failure groups).

   - **Extended redundancy**
     Extended redundancy disk group has features similar to the flex redundancy disk group. Extended redundancy is available when you configure an Oracle Extended Cluster. Extended redundancy extends Oracle ASM data protection to cover failure of sites by placing enough copies of data in different failure groups of each site.

3. Determine the total amount of disk space that you require for the database files and recovery files.

   If an Oracle ASM instance is running on the system, then you can use an existing disk group to meet these storage requirements. If necessary, you can add disks to an existing disk group during the database installation.

   See, "Oracle ASM Disk Space Requirements" in *Oracle Database Installation Guide* for the Oracle ASM disk space requirements.

   > **Note:**
   >
   > - The disk devices must be owned by the user performing the grid installation.
   >
   > Check with your system administrator to determine if the disks used by Oracle ASM are mirrored at the storage level. If so, select External for the redundancy. If the disks are not mirrored at the storage level, then select Normal for the redundancy.
   > - Every Oracle ASM disk is divided into allocation units (AU). An allocation unit is the fundamental unit of allocation within a disk group. You can select the AU Size value from 1, 2, 4, 8, 16, 32 or 64 MB, depending on the specific disk group compatibility level. The default value is 4 MB for flex disk group and 1 MB for all other disk group types. On engineered systems, the default value is 4 MB.

4. Optionally, identify failure groups for the Oracle ASM disk group devices.

   If you intend to use a normal, high or flex redundancy disk group, then you can further protect your database against hardware failure by associating a set of disk devices in a custom failure group. By default, each device comprises its own failure group. However, if two disk devices in a normal redundancy disk group are attached to the same Host Bus Adapter (HBA), then the disk group becomes unavailable if the controller fails. The controller in this example is a single point of failure.

   To protect against failures of this type, use two HBAs, each with two disks, and define a failure group for the disks attached to each controller. This configuration enables the disk group to tolerate the failure of one HBA.

   Consider the following guidelines while defining custom failure groups:

   - You can specify custom failure groups in the **Create ASM Disk Group** screen during an Oracle Grid Infrastructure installation.
   - You can also define custom failure groups after installation, using the GUI tool ASMCA, the command line tool `asmcmd`, or SQL commands.
   - If you define custom failure groups, then for failure groups containing database files only, you must specify a minimum of two failure groups for normal redundancy disk groups and three failure groups for high redundancy disk groups.

5. If you are sure that a suitable disk group does not exist on the system, then install or identify appropriate disk devices to add to a new disk group.

   Use the following guidelines when identifying appropriate disk devices:

- The disk devices must be owned by the user performing the Oracle Grid Infrastructure installation.
- All the devices in an Oracle ASM disk group must be the same size and have the same performance characteristics.
- Do not specify multiple partitions on a single physical disk as a disk group device. Oracle ASM expects each disk group device to be on a separate physical disk.
- Although you can specify a logical volume as a device in an Oracle ASM disk group, Oracle does not recommend their use because it adds a layer of complexity that is unnecessary with Oracle ASM. Oracle recommends that if you choose to use a logical volume manager, then use the logical volume manager to represent a single logical unit number (LUN) without striping or mirroring, so that you can minimize the effect on storage performance of the additional storage layer.

Related Topics

- Oracle ASM Disk Space Requirements
  Determine the total amount of Oracle Automatic Storage Management (Oracle ASM) disk space that you require for the database files and recovery files.
- *Oracle Automatic Storage Management Administrator's Guide*

Oracle ASM Disk Space Requirements

Determine the total amount of Oracle Automatic Storage Management (Oracle ASM) disk space that you require for the database files and recovery files.

Table 9-1 Oracle ASM Disk Number and Minimum Space Requirements for a multitenant container database (CDB) with one pluggable database (PDB)

Redundancy Level	Minimum Number of Disks	Data Files	Recovery Files	Both File Types
External	1	5.1 GB	15.3 GB	20.4 GB
Normal or Flex with two-way mirroring	2	10.3 GB	30.9 GB	41.2 GB
High or Flex with three-way mirroring	3	15.6 GB	46.8 GB	62.4 GB

Guidelines for Oracle ASM Disks

- If an Oracle ASM instance is running on the system, then you can use an existing disk group to meet these storage requirements. If necessary, you can add disks to an existing disk group during the database installation.
- The disk devices must be owned by the user performing the grid installation.

  Check with your system administrator to determine if the disks used by Oracle ASM are mirrored at the storage level. If so, select External for the redundancy. If the disks are not mirrored at the storage level, then select Normal for the redundancy.
- Every Oracle ASM disk is divided into allocation units (AU). An allocation unit is the fundamental unit of allocation within a disk group. You can select the AU Size value from 1, 2, 4, 8, 16, 32 or 64 MB, depending on the specific disk group compatibility level. The default value is 4 MB for flex disk group and 1 MB for all other disk group types. On engineered systems, the default value is 4 MB.

ASM Disk Group Options for Installation

Plan how you want to configure Oracle ASM disk groups for deployment.

During Oracle Grid Infrastructure installation, you can create one Oracle ASM disk group. After the Oracle Grid Infrastructure installation, you can create additional disk groups using Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA), SQL*Plus, or Automatic Storage Management Command-Line Utility (ASMCMD).

Related Topics

- *Oracle Automatic Storage Management Administrator's Guide*

Using an Existing Oracle ASM Disk Group

Use Oracle Enterprise Manager Cloud Control or the Oracle ASM command line tool (`asmcmd`) to identify existing disk groups, and to determine if sufficient space is available in the disk group.

1. Connect to the Oracle ASM instance and start the instance if necessary:

   ```
   $ $ORACLE_HOME/bin/asmcmd
   ASMCMD> startup
   ```

2. Enter one of the following commands to view the existing disk groups, their redundancy level, and the amount of free disk space in each one:

   ```
   ASMCMD> lsdg
   ```

   or

   ```
   $ORACLE_HOME/bin/asmcmd -p lsdg
   ```

   The `lsdg` command lists information about mounted disk groups only.

3. From the output, identify a disk group with the appropriate redundancy level and note the free space that it contains.
4. If necessary, install or identify the additional disk devices required to meet the storage requirements for your installation.

> **Note:**
>
> If you are adding devices to an existing disk group, then Oracle recommends that you use devices that have the same size and performance characteristics as the existing devices in that disk group.

Configuring Storage Device Path Persistence Using Oracle ASMFD

Oracle ASM Filter Driver (Oracle ASMFD) maintains storage file path persistence and helps to protect files from accidental overwrites.

- About Oracle ASM with Oracle ASM Filter Driver
  During Oracle Grid Infrastructure installation, you can choose to install and configure Oracle Automatic Storage Management Filter Driver (Oracle ASMFD). Oracle ASMFD helps prevent corruption in Oracle ASM disks and files within the disk group.
- Configuring Oracle ASM Filter Driver
  You can configure Oracle ASM Filter Driver (Oracle ASMFD) for device persistence during the installation of Oracle Grid Infrastructure or after Oracle Grid Infrastructure has been installed.
- Migrating Oracle ASM Disk Groups From ASMLIB to Oracle ASM Filter Driver
  Follow this procedure to migrate Oracle ASM disk groups to Oracle ASMFD in an Oracle Grid Infrastructure for a standalone server environment.
- Determining If Oracle ASM Filter Driver Is Configured
  The value of the `AFD_STATE` parameter of `SYS_ASMFD_PROPERTIES` specifies whether Oracle ASMFD is configured for the Oracle ASM instance.

About Oracle ASM with Oracle ASM Filter Driver

During Oracle Grid Infrastructure installation, you can choose to install and configure Oracle Automatic Storage Management Filter Driver (Oracle ASMFD). Oracle ASMFD helps prevent corruption in Oracle ASM disks and files within the disk group.

Oracle ASM Filter Driver (Oracle ASMFD) rejects write I/O requests that are not issued by Oracle software. This write filter helps to prevent users with administrative privileges from inadvertently overwriting Oracle ASM disks, thus preventing corruption in Oracle ASM disks and files within the disk group. For disk partitions, the area protected is the area on the disk managed by Oracle ASMFD, assuming the partition table is left untouched by the user.

Oracle ASMFD simplifies the configuration and management of disk devices by eliminating the need to rebind disk devices used with Oracle ASM each time the system is restarted.

If Oracle ASMLIB exists on your Linux system, then deinstall Oracle ASMLIB before installing Oracle Grid Infrastructure, so that you can choose to install and configure Oracle ASMFD during an Oracle Grid Infrastructure installation.

> **Caution:**
>
> When you configure Oracle ASM, including Oracle ASMFD, do not modify or erase the contents of the Oracle ASM disks, or modify any files, including the configuration files.

> **Note:**
>
> Oracle ASMFD is supported on Linux x86–64 and Oracle Solaris operating systems.

Related Topics

- Configuring Storage Device Path Persistence Using Oracle ASMLIB
  To use Oracle ASMLIB to configure Oracle ASM devices, complete the following tasks:
- *Oracle Automatic Storage Management Administrator's Guide*

Configuring Oracle ASM Filter Driver

You can configure Oracle ASM Filter Driver (Oracle ASMFD) for device persistence during the installation of Oracle Grid Infrastructure or after Oracle Grid Infrastructure has been installed.

- Guidelines for Configuring Oracle ASM Filter Driver
  Review these guidelines before you configure Oracle ASM Filter Driver (Oracle ASMFD) for device persistence.
- Configuring Oracle ASM Filter Driver During Installation
  You can optionally enable automated installation and configuration of Oracle ASM Filter Driver for your system during the installation of Oracle Grid Infrastructure.
- Configuring Oracle ASM Filter Driver After Installation
  You can install and configure Oracle ASM Filter Driver for your system after installing Oracle Grid Infrastructure for a standalone server.

Guidelines for Configuring Oracle ASM Filter Driver

Review these guidelines before you configure Oracle ASM Filter Driver (Oracle ASMFD) for device persistence.

- If ASMLIB is installed and configured for an existing Oracle ASM installation, then you must explicitly deinstall the existing ASMLIB configuration before installing and configuring Oracle ASMFD.
- If ASMLIB has been previously installed on the system, then you must set the Oracle ASM disk string to `''` (with ASMCMD `dsset ''`) before running ASMCMD `afd_configure` to configure Oracle ASMFD.

Related Topics

- Administering Oracle ASM Filter Driver

Configuring Oracle ASM Filter Driver During Installation

You can optionally enable automated installation and configuration of Oracle ASM Filter Driver for your system during the installation of Oracle Grid Infrastructure.

If you do not use `udev` on the Oracle Grid Infrastructure server, then you can provision disks for Oracle ASMFD before the installer is launched. This procedure must be performed after the Oracle Grid Infrastructure installation zip files have been unzipped

in the Oracle Grid Infrastructure home, but before you enable the **Configure ASM Filter Driver** check box during the installation process when the setup script is run.

1. As the `root` user, set the environment variable `$ORACLE_HOME` to the location of the Grid home and the environment variable `$ORACLE_BASE` to a temporary location.

   ```
   $ su root
   # export ORACLE_HOME=/u01/app/oracle/product/21.0.0/grid
   # export ORACLE_BASE=/tmp
   ```

   The `ORACLE_BASE` environment variable is set to a temporary location to avoid creating diagnostic or trace files in the Grid home before the Oracle Grid Infrastructure installation.

2. Add the `ORACLE_HOME/bin` directory to the `PATH` environment variable.

   ```
   # export PATH=$ORACLE_HOME/bin:$PATH
   ```

3. Use the ASMCMD `afd_label` command to provision disk devices for use with Oracle ASM Filter Driver.

   ```
   # $ORACLE_HOME/bin/asmcmd afd_label DATA1 /dev/disk1a --init
   ```

4. Use the ASMCMD `afd_lslbl` command to verify the device has been marked for use with Oracle ASMFD.

   ```
   # $ORACLE_HOME/bin/asmcmd afd_lslbl /dev/disk1a
   ```

5. Unset the `ORACLE_BASE` environment variable when you have finished provisioning disks for Oracle ASMFD.

   ```
   # unset ORACLE_BASE
   ```

6. On the **Create ASM Disk Group** page during the installation of Oracle Grid Infrastructure, check the **Configure ASM Filter Driver** box to enable configuration of Oracle ASM Filter Driver and then add your provisioned disks to the initial disk group (typically named DATA).

Configuring Oracle ASM Filter Driver After Installation

You can install and configure Oracle ASM Filter Driver for your system after installing Oracle Grid Infrastructure for a standalone server.

1. As the `grid` user, update the Oracle ASM disk discovery string to enable Oracle ASMFD to discover devices in the future.

   a. Retrieve the current value of the Oracle ASM disk discovery string.

      ```
      $ $ORACLE_HOME/bin/asmcmd dsget
      ```

      The value of *`disk_string`* in the above command is the current Oracle ASM disk discovery string value retrieved by `dsget`. If the disks are in the default location for the operating system environment and disk discovery string has not been set, then the disk discovery string is empty.

b. Update the Oracle ASM disk discovery string. If the string is empty, then update the value of the disk discovery string to include the default disk location.

```
$ $ORACLE_HOME/bin/asmcmd dsset 'disk_string,AFD:*'
```

2. As the `root` user, stop Oracle Grid Infrastructure for a standalone server.

```
# $ORACLE_HOME/bin/crsctl stop has
```

3. Configure Oracle ASMFD.

```
# $ORACLE_HOME/bin/asmcmd afd_configure
```

4. As the `grid` user, verify the status of Oracle ASMFD.

```
$ $ORACLE_HOME/bin/asmcmd afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'ENABLED'
on host 'myhost'
```

5. As the `root` user, label all the Oracle ASM disks for Oracle ASMFD.

```
# $ORACLE_HOME/bin/asmcmd afd_label 'label_name' 'device_path' --
migrate
```

6. Start Oracle Grid Infrastructure for a standalone server.

```
# $ORACLE_HOME/bin/crsctl start has
```

7. As the `grid` user, set the Oracle ASMFD discovery disk string to the updated Oracle ASM disk discovery string value from Step 1.

```
$ $ORACLE_HOME/bin/asmcmd afd_dsset disk_string
```

Migrating Oracle ASM Disk Groups From ASMLIB to Oracle ASM Filter Driver

Follow this procedure to migrate Oracle ASM disk groups to Oracle ASMFD in an Oracle Grid Infrastructure for a standalone server environment.

1. As the `grid` user, list the existing disk groups.

```
$ $ORACLE_HOME/bin/asmcmd lsdg
```

2. List the disks associated with each disk group.

```
$ $ORACLE_HOME/bin/asmcmd lsdsk -G diskgroup
```

3. Check if Oracle ASM is active.

```
$ $ORACLE_HOME/bin/srvctl status asm
```

4. Stop all the databases and dismount all disk groups.

```
$ $ORACLE_HOME/bin/srvctl stop database -db db_unique_name
$ /$ORACLE_HOME/bin/srvctl stop diskgroup -diskgroup diskgroup_name -f
```

5. Label all existing disks in the disk group.

```
$ $ORACLE_HOME/bin/asmcmd afd_label label disk_path --migrate
```

6. Scan the disks.

```
$ $ORACLE_HOME/bin/asmcmd afd_scan
```

7. Start the database and mount the disk group.

```
$ $ORACLE_HOME/bin/srvctl start diskgroup -diskgroup diskgroup_name
```

Determining If Oracle ASM Filter Driver Is Configured

The value of the `AFD_STATE` parameter of `SYS_ASMFD_PROPERTIES` specifies whether Oracle ASMFD is configured for the Oracle ASM instance.

1. As the `grid` user, check the state of Oracle ASMFD with the ASMCMD `afd_state` command.

```
$ $ORACLE_HOME/bin/asmcmd afd_state
ASMCMD-9526: The AFD state is 'LOADED' and filtering is 'DEFAULT' on host
'myhost'
```

2. Run the following query on the Oracle ASM instance to determine if Oracles ASMFD is present on the host and display the value of `AFD_STATE` from `SYS_CONTEXT`.

```
SQL> SELECT SYS_CONTEXT('SYS_ASMFD_PROPERTIES', 'AFD_STATE') FROM DUAL;
SYS_CONTEXT('SYS_ASMFD_PROPERTIES','AFD_STATE')
```

   - If the value of `AFD_STATE` is equal to `NOT AVAILABLE`, then Oracle ASMFD is not configured.
   - If the value of `AFD_STATE` is equal to `CONFIGURED`, then Oracle ASMFD is set up and the Oracle ASM instance can register with the driver.

Creating DAS or SAN Disk Partitions for Oracle Automatic Storage Management

You can use direct-attached storage (DAS) and storage area network (SAN) disks with Oracle ASM.

To use a DAS or SAN disk in Oracle ASM, Oracle recommends that the disk have a partition table. Oracle recommends creating exactly one partition for each disk.

Creating Directories for Oracle Database Files

You can store Oracle Database and recovery files on a separate file system from the configuration files.

Perform this procedure to place the Oracle Database or recovery files on a separate file system from the Oracle base directory:

1. Use the following command to determine the free disk space on each mounted file system:

   ```
   # df -h
   ```

2. Identify the file systems to use, from the display:

Option	Description
Database Files	Select one of the following: • A single file system with at least 1.5 GB of free disk space • Two or more file systems with at least 3.5 GB of free disk space in total
Recovery Files	Choose a file system with at least 2 GB of free disk space

   If you are using the same file system for multiple file types, then add the disk space requirements for each type to determine the total disk space requirement.

3. Note the names of the mount point directories for the file systems that you identified.

4. If the user performing installation has permissions to create directories on the disks where you plan to install Oracle Database, then DBCA creates the Oracle Database file directory, and the Recovery file directory. If the user performing installation does not have write access, then you must create these directories manually.

   For example, given the user `oracle` and Oracle Inventory Group `oinstall`, and using the paths `/u03/oradata/wrk_area` for Oracle Database files, and `/u01/oradata/rcv_area` for the recovery area, these commands create the recommended subdirectories in each of the mount point directories and set the appropriate owner, group, and permissions on them:

   - Database file directory:

     ```
     # mkdir -p /u01/oradata/
     # chown oracle:oinstall /u01/oradata/
     # chmod 775 /u01/oradata
     ```

     The default location for the database file directory is `$ORACLE_BASE/oradata`.

- Recovery file directory (fast recovery area):

  ```
  # mkdir -p /u01/oradata/rcv_area
  # chown oracle:oinstall /u01/oradata/rcv_area
  # chmod 775 /u01/oradata/rcv_area
  ```

  The default fast recovery area is `$ORACLE_BASE/fast_recovery_area`.

  Oracle recommends that you keep the fast recovery area on a separate physical disk than that of the database file directory. This method enables you to use the fast recovery area to retrieve data if the disk containing oradata is unusable for any reason.

Creating Files on a NAS Device for Use with Oracle Automatic Storage Management

If you have a certified NAS storage device, then you can create zero-padded files in an NFS mounted directory and use those files as disk devices in an Oracle ASM disk group.

Ensure that you specify the ASM discovery path for Oracle ASM disks.

During installation of Oracle Grid Infrastructure, Oracle Universal Installer (OUI) can create files in the NFS mounted directory you specify. The following procedure explains how to manually create files in an NFS mounted directory to use as disk devices in an Oracle ASM disk group:

1. If necessary, create an exported directory for the disk group files on the NAS device.
2. Switch user to `root`.
3. Create a mount point directory on the local system.

   For example:

   ```
   # mkdir -p /mnt/oracleasm
   ```

4. To ensure that the NFS file system is mounted when the system restarts, add an entry for the file system in the mount file `/etc/fstab`.
5. Enter a command similar to the following to mount the NFS on the local system:

   ```
   # mount /mnt/oracleasm
   ```

6. Choose a name for the disk group to create, and create a directory for the files on the NFS file system, using the disk group name as the directory name.

   For example, if you want to set up a disk group for a sales database:

   ```
   # mkdir /mnt/oracleasm/sales1
   ```

7. Use commands similar to the following to create the required number of zero-padded files in this directory:

   ```
   # dd if=/dev/zero
   of=/mnt/oracleasm/sales1/disk1 bs=1024k
   count=1000
   ```

This example creates 1 GB files on the NFS file system. You must create one, two, or three files respectively to create an external, normal, or high redundancy disk group.

> **Note:**
>
> Creating multiple zero-padded files on the same NAS device does not guard against NAS failure. Instead, create one file for each NAS device and mirror them using the Oracle ASM technology.

8. Enter commands similar to the following to change the owner, group, and permissions on the directory and files that you created:

```
# chown -R grid:asmadmin /mnt/oracleasm
# chmod -R 660 /mnt/oracleasm
```

   In this example, the installation owner is `grid` and the OSASM group is `asmadmin`.

9. During Oracle Database installations, edit the Oracle ASM disk discovery string to specify a regular expression that matches the file names you created.

   For example:

```
/mnt/oracleasm/sales1/
```

Related Topics

- My Oracle Support Note 359515.1

10

Installing and Configuring Oracle Grid Infrastructure for a Standalone Server

Oracle Grid Infrastructure for a standalone server includes Oracle Restart and Oracle Automatic Storage Management.

If you install Oracle Grid Infrastructure for a standalone server and then create your database, then the database is automatically added to the Oracle Restart configuration. Oracle Restart automatically restarts the database when required.

If you install Oracle Grid Infrastructure for a standalone server on a host computer on which a database already exists, then you must manually add the database, the listener, the Oracle ASM instance, and other components to the Oracle Restart configuration before you are able to configure automatic database restarts.

> **Note:**
>
> Oracle Grid Infrastructure for a standalone server can support multiple single-instance databases on a single host computer.

- About Image-Based Oracle Grid Infrastructure Installation
  Installation and configuration of Oracle Grid Infrastructure software is simplified with image-based installation.
- Setup Wizard Installation Options for Creating Images
  Gold image-creation options to use with your Oracle Database or Oracle Grid Infrastructure installation setup wizards.
- Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation
  Complete these steps to install Oracle Grid Infrastructure for a standalone server and then create a database that is managed by Oracle Restart.
- Installing Oracle Grid Infrastructure for a Standalone Server for an Existing Database
  Follow the high-level instructions in this section to install Oracle Grid Infrastructure for a standalone server and configure it for an existing Oracle Database.
- Installing Oracle Grid Infrastructure for a Standalone Server Using a Software-Only Installation
  A software-only installation only installs the Oracle Grid Infrastructure for a standalone server binaries at the specified location. You must complete a few manual configuration steps to enable Oracle Restart after you install the software.
- Testing the Oracle Automatic Storage Management Installation
  After installing Oracle Grid Infrastructure for a single instance, use the ASMCMD command-line utility to test the Oracle ASM installation.

- Modifying Oracle Grid Infrastructure for a Standalone Server Binaries After Installation
  After installation, if you need to modify the Oracle Grid Infrastructure for a standalone server configuration, then you must unlock the Grid home and relink the binaries.
- Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA
  After installing Oracle Grid Infrastructure for a standalone server, you can also use Oracle ASMCA to create and configure disk groups, Oracle ADVM, and Oracle ACFS.
- Enabling Oracle ACFS on Oracle Restart Configurations
  You must enable root access to use Oracle ACFS.
- Applying Patches During an Oracle Grid Infrastructure Installation or Upgrade
  You can download and apply Release Updates (RUs) and one-off patches during an Oracle Grid Infrastructure installation or upgrade.
- Patching and Switching Oracle Grid Infrastructure Homes
  Use `switchGridHome` command to switch from the current Oracle Grid Infrastructure home to the patched Oracle Grid Infrastructure home.

Related Topics

- Upgrading and Patching Oracle Restart on Linux

About Image-Based Oracle Grid Infrastructure Installation

Installation and configuration of Oracle Grid Infrastructure software is simplified with image-based installation.

To install Oracle Grid Infrastructure, create the new Grid home with the necessary user group permissions, and then extract the image file into the newly-created Grid home, and run the setup wizard to register the Oracle Grid Infrastructure product.

Using image-based installation, you can do the following:

- Configure Oracle Grid Infrastructure for a new cluster.
- Configure Oracle Grid Infrastructure for a standalone server (Oracle Restart).
- Upgrade Oracle Grid Infrastructure.
- Setup software only.
- Add or remove nodes from your existing cluster, if the Oracle Grid Infrastructure software is already installed or configured.

This installation feature streamlines the installation process and supports automation of large-scale custom deployments. You can also use this installation method for deployment of customized images, after you patch the base-release software with the necessary Release Updates (RUs) or Release Update Revisions (RURs).

> **Note:**
>
> You must extract the image software into the directory where you want your Grid home to be located, and then run the `%ORACLE_HOME%\gridSetup.sh` script to start the Oracle Grid Infrastructure Setup Wizard. Ensure that the Grid home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations.

Setup Wizard Installation Options for Creating Images

Gold image-creation options to use with your Oracle Database or Oracle Grid Infrastructure installation setup wizards.

Options

In image-based installations, you can start your Oracle Database installation or Oracle Grid Infrastructure installations by running the setup wizards `runInstaller` and `gridSetup.sh` respectively. Both these wizards come with the following image-creation options.

Table 10-1 Image-Creation Options for Setup Wizard

Option	Description
`-createGoldImage`	Creates a gold image from the current Oracle home.
`-destinationLocation`	Specify the complete path, or location, where the gold image will be created.
`-exclFiles`	Specify the complete paths to the files to be excluded from the newly created gold image.
`–help`	Displays help for all the available options.

For example:

```
./runInstaller -createGoldImage -destinationLocation /tmp/my_db_images -
exclFiles /u01/app/oracle/product/21.0.0/dbhome_1/relnotes
```

```
./gridSetup.sh -createGoldImage -destinationLocation /tmp/my_grid_images -
exclFiles /u01/app/oracle/product/21.0.0/dbhome_1/relnotes
```

Where:

`/tmp/my_db_images` is a temporary file location where the image zip file is created.

`/tmp/my_grid_images` is a temporary file location where the image zip file is created.

`/u01/app/oracle/product/21.0.0/dbhome_1/relnotes` is the file to be excluded in the newly created gold image.

Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation

Complete these steps to install Oracle Grid Infrastructure for a standalone server and then create a database that is managed by Oracle Restart.

Install Oracle Grid Infrastructure for a standalone server, which installs Oracle Restart and Oracle ASM, and creates one disk group.

You should have your network information, storage information, and operating system users and groups available to you before you start the installation. You should also be prepared to run root scripts or provide information to automate root scripts.

1. Log in as the Oracle Restart software owner user (`oracle`).
2. If this is the first time you are installing Oracle software, then create the Oracle base and the Oracle inventory directories as per the Oracle Optimal Flexible Architecture (OFA) recommendations. Specify the correct owner, group, and permissions for these directories.

   ```
   # mkdir -p /u01/app/oracle
   # mkdir -p /u01/app/oraInventory
   # chown -R oracle:oinstall /u01/app/oracle
   # chown -R oracle:oinstall /u01/app/oraInventory
   # chmod -R 775 /u01/app
   ```

3. Download the Oracle Grid Infrastructure for a standalone server installation image files, create the grid home directory, and extract the image files in this grid home directory.

   For example:

   ```
   $ mkdir -p /u01/app/oracle/product/21.0.0/grid
   $ cd /u01/app/oracle/product/21.0.0/grid
   $ unzip -q /tmp/grid_home.zip
   ```

   > **Note:**
   >
   > - Ensure that the Grid home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations. Also, unzip the installation image files only in this Grid home directory that you created.
   > - Oracle home or Oracle base cannot be `symlinks`, nor can any of their parent directories, all the way to up to the `root` directory.

4. Configure and provision the shared disks for use with Oracle ASM Filter Driver (Oracle ASMFD).

a. Log in as `root` and set $ORACLE_HOME to the grid home location and $ORACLE_BASE to a temporary location.

```
su root
# set ORACLE_HOME=/u01/app/oracle/product/21.0.0/grid
# set ORACLE_BASE=/tmp
```

You set $ORACLE_BASE to a temporary location to avoid creating diagnostic or trace files in the Grid home before the Oracle Grid Infrastructure installation.

b. Use the ASMCMD `afd_label` command to provision disk devices for use with Oracle ASM Filter Driver. For example:

```
# cd /u01/app/oracle/product/21.0.0/grid/bin
# ./asmcmd afd_label DATA1 /dev/sdb --init
# ./asmcmd afd_label DATA2 /dev/sdc --init
# ./asmcmd afd_label DATA3 /dev/sdd --init
```

c. Use the ASMCMD `afd_lslbl` command to verify the device has been marked for use with Oracle ASMFD. For example:

```
# ./asmcmd afd_lslbl /dev/sdb
# ./asmcmd afd_lslbl /dev/sdc
# ./asmcmd afd_lslbl /dev/sdd
```

d. Unset $ORACLE_BASE.

```
# unset ORACLE_BASE
```

5. Log in as the Oracle Restart software owner user amd run `gridSetup.sh` to start the Oracle Grid Infrastructure setup wizard:

```
$ Grid_home/gridSetup.sh
```

Where *Grid_home* is the Oracle Grid Infrastructure home directory.

> **Note:**
>
> You can use the `gridSetup.sh` command with the `-applyRU` and `-applyOneOffs` options to install Release Updates (RUs) and one-off patches during an Oracle Grid Infrastructure installation or upgrade.

6. In the Select Configuration Option screen, select the **Configure Oracle Grid Infrastructure for a Standalone Server (Oracle Restart)** option to install and configure Oracle Restart and Oracle ASM. Click **Next.**
7. During installation, disk paths mounted on Oracle ASM and registered on Oracle ASMFD with the string `AFD:*` are listed as default database storage candidate disks.
8. Configure Oracle ASM as needed with additional disk groups.
   - The default Disk Group Name is DATA. You can enter a new name for the disk group, or use the default name.

- Any additional disk devices that you create must be owned by the user performing the grid installation.

9. If you want to use Oracle ASM Filter Driver (Oracle ASMFD) to manage your Oracle ASM disk devices, then select **Configure Oracle ASM Filter Driver** on the Create ASM Disk Group screen.

   On Linux, if you want to use Oracle ASM Filter Driver (Oracle ASMFD) to manage your Oracle ASM disk devices, then you must deinstall Oracle ASM library driver (Oracle ASMLIB) before you start the Oracle Grid Infrastructure installation.

10. Respond to the configuration prompts as needed to configure Oracle Grid Infrastructure. Click **Help** for information.

11. Provide information to automate root scripts, or run scripts as root when prompted by OUI.

    If you configure automation for running root scripts, and a root script fails, then you can fix the problem manually, and click **Retry** to run the root script again.

12. Start the Oracle Database installation, and select Oracle ASM disk groups for Oracle Database files storage. For assistance during installation, click **Help** on the Oracle Universal Installer page where you need more information.

Related Topics

- Optimal Flexible Architecture
  Oracle Optimal Flexible Architecture (OFA) rules are a set of configuration guidelines created to ensure well-organized Oracle installations, which simplifies administration, support and maintenance.
- Deinstalling Oracle ASMLIB On Oracle Database
  If Oracle ASM library driver (Oracle ASMLIB) is installed but you do not use it for device path persistence, then deinstall Oracle ASMLIB:

Related Topics

- Applying Patches During an Oracle Grid Infrastructure Installation or Upgrade
  You can download and apply Release Updates (RUs) and one-off patches during an Oracle Grid Infrastructure installation or upgrade.
- *Oracle Automatic Storage Management Administrator's Guide*

Installing Oracle Grid Infrastructure for a Standalone Server for an Existing Database

Follow the high-level instructions in this section to install Oracle Grid Infrastructure for a standalone server and configure it for an existing Oracle Database.

Oracle Restart can manage resources from the same release and releases up to one version lower than Oracle Restart. For instance, you can install Oracle Grid Infrastructure for a standalone server 19c (Oracle Restart) to provide services for Oracle Database 19c and Oracle Database 18c. Earlier Oracle Database releases can coexist on the same server without being managed by Oracle Restart.

1. Log in as the Oracle Database software owner user.
2. Set the ORACLE_HOME environment variable.

Bourne, Bash or Korn shell:

```
$ ORACLE_HOME=/u01/app/oracle/product/21.0.0/dbhome_1
$ export ORACLE_HOME
```

C shell:

```
% setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
```

3. Go to the Oracle home and stop any existing database listeners.

```
$ $ORACLE_HOME/lsnrctl stop listener_name
```

> **Note:**
>
> If this command throws an error, reload the listener using `lsnrctl reload LISTENER`

4. On the same host computer as the database, install Oracle Grid Infrastructure for a standalone server, and select **Configure Oracle Grid Infrastructure for a Standalone Server (Oracle Restart)** as the installation option. See, “Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation” in *Oracle Database Installation Guide*.

   The Oracle Grid Infrastructure for a standalone server components are installed in an Oracle Grid Infrastructure Oracle home (Grid home), which is in a different location from existing Oracle Database homes.

5. Go to the Oracle home and start any existing database listeners.

```
$ $ORACLE_HOME/lsnrctl start listener_name
```

6. Register the Oracle Database and its listeners for High Availability with Oracle Restart using the `srvctl` command:

```
$ cd $ORACLE_HOME/bin
$ srvctl add database -db dbname -o oracle_home_path
$ srvctl add listener -listener listener_name
```

Related Topics

- Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation
  Complete these steps to install Oracle Grid Infrastructure for a standalone server and then create a database that is managed by Oracle Restart.
- *Oracle Database Administrator’s Guide*

Installing Oracle Grid Infrastructure for a Standalone Server Using a Software-Only Installation

A software-only installation only installs the Oracle Grid Infrastructure for a standalone server binaries at the specified location. You must complete a few manual configuration steps to enable Oracle Restart after you install the software.

- About Oracle Grid Infrastructure Software-Only Installations
  Manually installing and configuring the software binaries for Oracle Grid Infrastructure.
- Installing Software Binaries for Oracle Grid Infrastructure for a Standalone Server
  Use this procedure to do a software-only installation of Oracle Grid Infrastructure for a standalone server.
- Configuring Software Binaries for Oracle Grid Infrastructure for a Standalone Server
  Use these steps to configure and activate a software-only Oracle Grid Infrastructure for a standalone server installation for Oracle Restart.

About Oracle Grid Infrastructure Software-Only Installations

Manually installing and configuring the software binaries for Oracle Grid Infrastructure.

Oracle recommends that only advanced users perform software-only installations, because this installation method provides no validation of the installation, and this installation option requires manual postinstallation steps to enable the Oracle Grid Infrastructure for a standalone server software.

Performing a software-only installation requires the following steps:

1. Installing the software binaries.
2. Configuring the software binaries.

Installing Software Binaries for Oracle Grid Infrastructure for a Standalone Server

Use this procedure to do a software-only installation of Oracle Grid Infrastructure for a standalone server.

1. Log in as the Oracle Restart software owner user (`oracle`).
2. Download the Oracle Grid Infrastructure for a standalone server installation image files, create the Grid home directory, and extract the image files in this Grid home directory.

   For example:

   ```
   $ mkdir -p /u01/app/oracle/product/21.0.0/grid
   $ chown oracle:oinstall /u01/app/oracle/product/21.0.0/grid
   $ cd /u01/app/oracle/product/21.0.0/grid
   $ unzip -q /tmp/grid_home.zip
   ```

> **Note:**
>
> Ensure that the Grid home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations. Also, unzip the installation image files only in this Grid home directory that you created.

3. Ensure that you complete all the storage and server preinstallation requirements. Verify that your server meets the installation requirements using the following command:

```
$ cd /u01/app/oracle/product/21.0.0/grid
$ runcluvfy.sh stage -pre hacfg
```

4. Run `gridSetup.sh` to start the Oracle Grid Infrastructure setup wizard:

```
$ Grid_home/gridSetup.sh
```

   Where, *Grid_home* is the Oracle Grid Infrastructure home directory.

5. In the Select Configuration Option screen, select the **Set Up Software Only** option to perform a software-only installation of Oracle Grid Infrastructure for a standalone server. Click **Next.**
6. Respond to the prompts as needed to set up Oracle Grid Infrastructure. Click **Help** for information.
7. The Oracle Grid Infrastructure setup wizard prompts you to run the `root.sh` script and, if required, the `orainstRoot.sh` script.
8. The `root.sh` script output provides information about how to proceed, depending on the configuration you plan to complete in this installation. Note this information.

Configuring Software Binaries for Oracle Grid Infrastructure for a Standalone Server

Use these steps to configure and activate a software-only Oracle Grid Infrastructure for a standalone server installation for Oracle Restart.

Configuring With Oracle Automatic Storage Management

1. Install the software binaries. See, “Installing Software Binaries for Oracle Grid Infrastructure for a Standalone Server” in *Oracle Database Installation Guide*.
2. Run `gridSetup.sh` to start the Oracle Grid Infrastructure setup wizard.

   See, “Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation” in *Oracle Database Installation Guide*.

Configuring Without Oracle Automatic Storage Management

1. Log in as `root` and run the `roothas.sh` script located in the *Grid_home* path, using the following syntax:

```
# cd Grid_home/crs/install
# ./roothas.sh
```

Where, *Grid_home* is the Oracle Grid Infrastructure home directory.

For example:

```
# cd /u01/app/oracle/product/21.0.0/grid/crs/install
# ./roothas.sh
```

2. Change directory to the path *Grid_home*/`oui/bin`.
3. Log in as the Oracle Restart software owner user and use the following command syntax, where Grid_home is the path of the Oracle Grid Infrastructure for a standalone server home.

```
$ ./runInstaller -updateNodeList ORACLE_HOME=Grid_home -
defaultHomeName CLUSTER_NODES= CRS=TRUE
```

For example:

```
$ ./runInstaller -updateNodeList ORACLE_HOME=/u01/app/oracle/
product/21.0.0/grid
-defaultHomeName CLUSTER_NODES= CRS=TRUE
```

4. Use the `SRVCTL` utility along with Oracle Network Configuration Assistant to add the listener to the Oracle Restart configuration.

> **Note:**
>
> This procedure does not work for Oracle Restart upgrades from previous releases.

Related Topics

- Installing Software Binaries for Oracle Grid Infrastructure for a Standalone Server
  Use this procedure to do a software-only installation of Oracle Grid Infrastructure for a standalone server.
- Installing Oracle Grid Infrastructure for a Standalone Server with a New Database Installation
  Complete these steps to install Oracle Grid Infrastructure for a standalone server and then create a database that is managed by Oracle Restart.

Testing the Oracle Automatic Storage Management Installation

After installing Oracle Grid Infrastructure for a single instance, use the ASMCMD command-line utility to test the Oracle ASM installation.

1. Open a shell window, and temporarily set the ORACLE_SID and ORACLE_HOME environment variables to specify the appropriate values for the Oracle ASM instance to use.

For example, if the Oracle ASM SID is named +ASM and the Oracle home is located in the grid subdirectory of the ORACLE_BASE directory, then enter the following commands to create the required settings:

Bourne, Bash or Korn shell:

```
$ ORACLE_SID=+ASM
$ export ORACLE_SID
$ ORACLE_HOME=/u01/app/oracle/product/21.0.0/grid
$ export ORACLE_HOME
```

C shell:

```
% setenv ORACLE_SID +ASM
% setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/grid
```

2. Use ASMCMD to list the disk groups for the Oracle ASM instance:

   ```
   $ORACLE_HOME/bin/asmcmd lsdg
   ```

   If the Oracle ASM instance is running, then ASMCMD connects by default as the SYS user with SYSASM privileges, and is available.

3. If the Oracle ASM instance is not running, then start the Oracle ASM instance using the following command:

   ```
   $ORACLE_HOME/bin/srvctl start asm
   ```

Related Topics

- *Oracle Automatic Storage Management Administrator's Guide*

Modifying Oracle Grid Infrastructure for a Standalone Server Binaries After Installation

After installation, if you need to modify the Oracle Grid Infrastructure for a standalone server configuration, then you must unlock the Grid home and relink the binaries.

You must first stop the Oracle Restart stack to modify the software installed in your Grid home. For example, to apply a one-off patch or modify any of the dynamic link libraries (DLLs) used by Oracle Restart or Oracle ASM, you must stop the stack manually, modify the software, and then restart the stack.

However, if you run the OPatch utility with the auto option, `opatchauto`, then OPatch stops and starts the software stack for you. If you run OPatch with the apply option, `opatchauto apply`, then you must also stop and restart the stack manually .

You must relink the Oracle Restart and Oracle ASM binaries every time you apply an operating system patch or after an operating system upgrade.

> **Caution:**
>
> Before relinking executables, you must shut down all executables that run in the Oracle home directory that you are relinking. In addition, shut down applications linked with Oracle shared libraries.

Modifying Oracle Grid Infrastructure for a Standalone Server Binaries

1. Log in as the Oracle Grid Infrastructure for a standalone server software owner user and change the directory to the path *Grid_home*/bin, where *Grid_home* is the path to the Oracle Grid Infrastructure for a standalone server home:

   ```
   $ cd Grid_home/bin
   ```

2. Shut down the Oracle Restart stack using the following command:

   ```
   $ crsctl stop has -f
   ```

3. Log in as `root` and unlock the grid home:

   ```
   # cd Grid_home/crs/install
   # roothas.sh -unlock
   ```

4. Apply the patches using `opatchauto apply`.
5. Relink the binaries.

   ```
   $ export ORACLE_HOME=Grid_home
   $ Grid_home/bin/relink
   ```

6. Lock the grid home:

   ```
   # cd Grid_home/crs/install
   # roothas.sh -lock
   ```

7. Enter the following command to restart the Oracle Restart stack:

   ```
   $ crsctl start has
   ```

Relinking the Oracle Restart and Oracle ASM Binaries

1. Log in as `root` and unlock the grid home:

   ```
   # cd Grid_home/crs/install
   # roothas.sh -unlock
   ```

2. Log in as the `grid` user and relink the binaries:

   ```
   $ export ORACLE_HOME=Grid_home
   $ Grid_home/bin/relink
   ```

3. Log in as `root` again and perform the following steps:

```
# cd Grid_home/rdbms/install/
# ./rootadd_rdbms.sh
# cd Grid_home/crs/install
# roothas.sh -lock
```

Related Topics

- *Oracle OPatch User's Guide for Windows and UNIX*

Configuring Oracle ASM Disk Groups Manually using Oracle ASMCA

After installing Oracle Grid Infrastructure for a standalone server, you can also use Oracle ASMCA to create and configure disk groups, Oracle ADVM, and Oracle ACFS.

During Oracle Grid Infrastructure for a standalone server installation, Oracle Automatic Storage Management Configuration Assistant (Oracle ASMCA) utility creates a new Oracle Automatic Storage Management instance if there is no Oracle ASM instance currently configured on the computer. After installation, you can create and configure additional disk groups, and you can configure Oracle ADVM and Oracle ACFS.

To create disk groups or manually configure Oracle ASM disks, start Oracle ASMCA, where *Grid_home* is the path to the Oracle Grid Infrastructure home:

```
$ cd Grid_home/bin
$ ./asmca
```

Related Topics

- *Oracle Automatic Storage Management Administrator's Guide*

Enabling Oracle ACFS on Oracle Restart Configurations

You must enable root access to use Oracle ACFS.

To enable root access, log in as `root`, navigate to the path *Grid_home*`/crs/install`, and run the script `roothas.sh –lockacfs`.
Where, *Grid_home* is the Oracle Grid Infrastructure home directory path.

For example:

```
# cd /u01/app/oracle/product/21.0.0/grid/crs/install
# roothas.sh -lockacfs
```

Applying Patches During an Oracle Grid Infrastructure Installation or Upgrade

You can download and apply Release Updates (RUs) and one-off patches during an Oracle Grid Infrastructure installation or upgrade.

1. Download the patches you want to apply from My Oracle Support:

   https://support.oracle.com

2. Select the **Patches and Updates** tab to locate the patch.

   Oracle recommends that you select **Recommended Patch Advisor**, and enter the product group, release, and platform for your software.

3. Move the patches to an accessible directory like `/tmp`.

4. Change to the Oracle Grid Infrastructure home directory:

   ```
   $ cd /u01/app/oracle/product/21.0.0/grid
   ```

5. Apply Release Updates (RUs) and any one-off patches during the installation or upgrade process:

   ```
   $ ./gridSetup.sh -applyRU patch_directory_location -applyOneOffs
   comma_seperated_list_of_patch_directory_locations
   ```

   > **Note:**
   >
   > You can apply RUs and one-off patches separately or together in the same command.

6. Complete the remaining steps in the Oracle Grid Infrastructure configuration wizard to complete the installation or upgrade.

Patching and Switching Oracle Grid Infrastructure Homes

Use `switchGridHome` command to switch from the current Oracle Grid Infrastructure home to the patched Oracle Grid Infrastructure home.

You can use the `switchGridHome` command to switch between different RUs of the same release, for example from Oracle Grid Infrastructure 21.3 to Oracle Grid Infrastructure 21.4. You can also use the `switchGridHome` command to apply one-off patches.

1. Download the Release Updates (RUs) you want to apply from My Oracle Support.

2. As the `grid` user, download the Oracle Grid Infrastructure image files and extract the files into a new Oracle Grid Infrastructure home directory.

   ```
   $ mkdir -p /u01/app/oracle/product/21.4.0/grid
   $ chown grid:oinstall /u01/app/oracle/product/21.4.0/grid
   $ cd /u01/app/oracle/product/21.4.0/grid
   $ unzip -q download_location/grid.zip
   ```

   > **Note:**
   >
   > The new Oracle Grid Infrastructure home path must be different from the current Oracle Grid Infrastructure home path.

3. Start the Oracle Grid Infrastructure installer with the `-switchGridHome` flag to switch to the patched Oracle Grid Infrastructure home after the installation and the optional `-applyRU` flag to apply Release Updates (RUs) during the installation.

   ```
   $ /u01/app/oracle/product/21.4.0/grid/gridSetup.sh -switchGridHome [-
   applyRU patch_directory_location]
   [-applyOneOffs comma_seperated_list_of_patch_directory_locations]
   ```

4. Follow the steps in the configuration wizard to complete the Oracle Grid Infrastructure installation.

   All Oracle Grid Infrastructure services start running from the new Grid home after the installation is complete.

11
Installing Oracle Database

Oracle Database and Oracle Grid Infrastructure installation software is available as image-based zip files and can be installed using several options.

You can download Oracle Database software from the Oracle website or the Oracle Software Delivery Cloud portal. In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install the software. However, you can also run silent mode installations, without using the GUI. You can also use Oracle Fleet Patching and Provisioning for Oracle Database and Oracle Grid Infrastructure (clusterware) deployments.

Oracle Database software may be available on installation media on-demand.

> **Note:**
>
> To install Oracle Database files on Oracle Automatic Storage Management (Oracle ASM), you must first complete an Oracle Grid Infrastructure for a standalone server installation. Oracle Grid Infrastructure for a standalone server includes Oracle Restart and Oracle ASM.
>
> To upgrade an existing Oracle ASM installation, upgrade Oracle ASM by running an Oracle Grid Infrastructure upgrade. If you do not have Oracle ASM installed and you want to use Oracle ASM as your storage option, then you must complete an Oracle Grid Infrastructure for a standalone server installation before you start your Oracle Database installation.
>
> You cannot use Oracle Universal Installer from an earlier Oracle release to install components from this release.

- About Image-Based Oracle Database Installation
  Understand image-based installation to simplify installation and configuration of Oracle Database software.
- About Deploying Oracle Database Using Oracle Fleet Patching and Provisioning
  You can use Oracle Fleet Patching and Provisioning (Oracle FPP) to provision Oracle Database software.
- Downloading Oracle Software
  Select the method you want to use to download the software.
- About Character Set Selection During Installation
  Before you create the database, decide the character set that you want to use.
- About Automatic Memory Management Installation Options
  Decide if you want to configure Automatic Memory Management during installation.
- Running the Installer in a Different Language
  Describes how to run the installer in other languages.

- Installing the Oracle Database Software
  These topics explain how to run Oracle Universal Installer to perform most database installations.
- Installing Oracle Database Using RPM Packages
  Learn how to leverage and take advantage of an RPM framework to easily deploy Oracle Database.
- Installing Standard Edition High Availability
  Learn how to Install high availability on Oracle Database Standard Edition 2.
- About Creating Databases Using Oracle DBCA
  Create additional databases using Oracle Database Configuration Assistant (Oracle DBCA).

About Image-Based Oracle Database Installation

Understand image-based installation to simplify installation and configuration of Oracle Database software.

To install Oracle Database, create the new Oracle home, extract the image file into the newly-created Oracle home, and run the setup wizard to register the Oracle Database product.

Using image-based installation, you can install and upgrade Oracle Database for single-instance and cluster configurations.

This installation feature streamlines the installation process and supports automation of large-scale custom deployments. You can also use this installation method for deployment of customized images, after you patch the base-release software with the necessary Release Updates (Updates) or Release Update Revisions (Revisions).

> **Note:**
>
> You must extract the image software (`db_home.zip`) into the directory where you want your Oracle Database home to be located, and then run the Oracle Database Setup Wizard to start the Oracle Database installation and configuration. Oracle recommends that the Oracle home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations.

About Deploying Oracle Database Using Oracle Fleet Patching and Provisioning

You can use Oracle Fleet Patching and Provisioning (Oracle FPP) to provision Oracle Database software.

Starting with Oracle Database 19c, Rapid Home Provisioning is renamed to Oracle Fleet Patching and Provisioning (Oracle FPP).

With Oracle Fleet Patching and Provisioning, you create, store, and manage templates of Oracle homes as images (called gold images) of Oracle software, such as databases, middleware, and applications. You can make a working copy of any gold

image and then you can provision that working copy to any node in the data center or cloud computing environment.

You can use Oracle Fleet Patching and Provisioning to provision, patch, and upgrade single-instance databases running on Oracle Restart, on clusters, or on single, standalone nodes. These may be on nodes with or without Oracle Grid Infrastructure installed.

> **Note:**
>
> Oracle Fleet Patching and Provisioning is not supported on Microsoft Windows and HP-UX Itanium systems.

Related Topics

- *Oracle Fleet Patching and Provisioning Administrator's Guide*

Downloading Oracle Software

Select the method you want to use to download the software.

You can download Oracle Database software from the Oracle website or the Oracle Software Delivery Cloud portal and extract them to the Oracle home. Ensure that you review and understand the terms of the license.

- Downloading the Installation Archive Files from the Oracle Database Website
  Download the installation archive files from the Oracle Database downloads page.
- Downloading the Software from Oracle Software Delivery Cloud Portal
  You can download the software from Oracle Software Delivery Cloud.

Downloading the Installation Archive Files from the Oracle Database Website

Download the installation archive files from the Oracle Database downloads page.

1. Use any browser to access the Oracle Database software downloads page:

   https://www.oracle.com/database/technologies/oracle-database-software-downloads.html
2. Go to the download page for the product to install.
3. On the download page, identify the required disk space by adding the file sizes for each required file.

   The file sizes are listed next to the file names.
4. Select a file system with enough free space to store and expand the archive files.

   In most cases, the available disk space must be at least twice the size of all of the archive files.
5. On the file system, create a parent directory for each product (for example, `OraDB21c`) to hold the installation directories.
6. Download all of the installation archive files to the directory you created for the product.

> **Note:**
>
> For Oracle Database Client installations, there are two installation archive files available for download. The first file is the client installation binary and the second file is a client gold image file. Download the appropriate zip file based on the type of installation you want to perform.

7. Verify that the files you downloaded are the same size as the corresponding files on the Oracle website. Also verify the checksums are the same as noted on the Oracle website using a command similar to the following, where *filename* is the name of the file you downloaded:

   ```
   cksum filename.zip
   ```

8. Extract the files in each directory that you just created.

Downloading the Software from Oracle Software Delivery Cloud Portal

You can download the software from Oracle Software Delivery Cloud.

1. Use a browser to access the Oracle Software Delivery Cloud portal:

   https://edelivery.oracle.com/

2. Click **Sign In** and enter your Oracle account username and password.
3. Type **Oracle Database** in the search bar. Click the **Add to Cart** button corresponding to the Oracle Database version that you want to download
4. In the Checkout page, click **Checkout** and deselect any products that you do not want to download.
5. Select the operating system platform on which you want to install the software from the **Platform/Languages** column.
6. Click **Continue**.
7. Review the license agreement.
8. Select the **I reviewed and accept the Oracle License Agreement** checkbox. Click **Continue**.
9. Click **Download** to start downloading the software.
10. After you download the files, click **View Digest** to verify that the checksum matches the value listed on the download page.

About Character Set Selection During Installation

Before you create the database, decide the character set that you want to use.

After a database is created, changing its character set is usually very expensive in terms of time and resources. Such operations may require converting all character data by exporting the whole database and importing it back. Therefore, it is important that you carefully select the database character set at installation time.

Oracle Database uses character sets for the following:

- Data stored in SQL character data types (`CHAR`, `VARCHAR2`, `CLOB`, and `LONG`).
- Identifiers such as table names, column names, and PL/SQL variables.
- Stored SQL and PL/SQL source code, including text literals embedded in this code.

The default database character set of a database created from the General Purpose/Transaction Processing or the Data Warehousing template is Unicode `AL32UTF8`.

Unicode is the universal character set that supports most of the currently spoken languages of the world. It also supports many historical scripts (alphabets). Unicode is the native encoding of many technologies, including Java, XML, XHTML, ECMAScript, and LDAP. Unicode is ideally suited for databases supporting the Internet and the global economy.

Because `AL32UTF8` is a multibyte character set, it requires slightly more CPU time for text processing compared to single-byte character sets. Also, storage space requirements are higher for text in most languages compared to corresponding legacy character sets. However, the universality and flexibility of Unicode that enables easy addition of data in new languages to applications running in an `AL32UTF8` database generally outweighs these additional costs.

The database character set of an Oracle Database, that is, of its `CDB$ROOT` container, determines which pluggable databases (PDBs) can be plugged into it. If you use Unicode `AL32UTF8` as your database character set, then you can plug in a PDB in any database character set supported by Oracle Database (with the exception of `EBCDIC`-based character sets). If you use any character set other than `AL32UTF8` when creating the container database, you will be able to plug in PDBs in the same character set only. Therefore, you should generally use the default option for the database character set when installing a new database.

If you need to deploy PDBs in a given legacy character set to fulfill a specific compatibility, storage, or performance requirement, create a temporary container database in this legacy character set with one empty PDB. This PDB will have the same legacy database character set. Then, unplug this PDB and plug it into the target `AL32UTF8` container database. Drop the temporary container database. You can use such a plugged-in PDB as a template to clone further PDBs in the same legacy character set as needed. You can use the same method to add further legacy character set template PDBs to the same `AL32UTF8` container database, as required.

> **See Also:**
>
> *Oracle Database Globalization Support Guide* for more information about choosing a database character set for an Oracle Database.

About Automatic Memory Management Installation Options

Decide if you want to configure Automatic Memory Management during installation.

During a **Typical** installation, you create your database with Oracle Database Configuration Assistant (DBCA), and automatic memory management is enabled. If you choose **Advanced** installation, then you can either specify memory allocation manually, or enable automatic memory management.

If the total physical memory of your database instance is greater than 4 GB, then you cannot select the Oracle Automatic Memory Management option during database installation and

creation. Instead, use automatic shared memory management. Automatic shared memory management automatically distributes the available memory among the various components as required, allowing the system to maximize the use of all available SGA memory.

With automatic memory management, the Oracle Database instances automatically manage and tune memory for you. With automatic memory management, you choose a memory target, and the instance automatically distributes memory between the system global area (SGA) and the instance program global area (instance PGA). As memory requirements change, the instance dynamically redistributes memory between the SGA and instance PGA.

You can enable automatic memory management either during, or after the database installation. Enabling automatic memory management after installation involves a shutdown and restart of the database.

> **Note:**
>
> By default, automatic memory management is disabled when you perform typical installation on a node that has more than 4 GB of RAM.

Related Topics

- *Oracle Database Administrator's Guide*

Running the Installer in a Different Language

Describes how to run the installer in other languages.

Your operating system locale determines the language in which the database installer runs. You can run the installer in one of these languages:

- Brazilian Portuguese (pt_BR)
- French (fr)
- German (de)
- Italian (it)
- Japanese (ja)
- Korean (ko)
- Simplified Chinese (zh_CN)
- Spanish (es)
- Traditional Chinese (zh_TW)

To run the database installer in a supported language, change the locale in which your operating system session is running before you start the installer.

If the selected language is not one of the supported languages, then the installer runs in English.

Installing the Oracle Database Software

These topics explain how to run Oracle Universal Installer to perform most database installations.

> **Note:**
>
> - If you plan to use Oracle Restart or Oracle ASM, then you must install Oracle Grid Infrastructure for a standalone server before you install and create the database. Otherwise, you must manually register the database with Oracle Restart.
> - You can install Oracle Database by using the silent or response file installation method, without the GUI. This method is useful to perform multiple installations of Oracle Database.

- Setup Wizard Installation Options for Creating Images
- Applying Patches During an Oracle Database Installation or Upgrade
  Starting with Oracle Database 18c, you can download and apply Release Updates (RUs) during an Oracle Database installation or upgrade.
- Running Oracle Database Setup Wizard to Install Oracle Database
  Extract the database image files and use the `runInstaller` command to start the installation.

Setup Wizard Installation Options for Creating Images

Gold image-creation options to use with your Oracle Database or Oracle Grid Infrastructure installation setup wizards.

Options

In image-based installations, you can start your Oracle Database installation or Oracle Grid Infrastructure installations by running the setup wizards `runInstaller` and `gridSetup.sh` respectively. Both these wizards come with the following image-creation options.

Table 11-1 Image-Creation Options for Setup Wizard

Option	Description
`-createGoldImage`	Creates a gold image from the current Oracle home.
`-destinationLocation`	Specify the complete path, or location, where the gold image will be created.
`-exclFiles`	Specify the complete paths to the files to be excluded from the newly created gold image.
`–help`	Displays help for all the available options.

For example:

```
./runInstaller -createGoldImage -destinationLocation /tmp/my_db_images -
exclFiles /u01/app/oracle/product/21.0.0/dbhome_1/relnotes
```

```
./gridSetup.sh -createGoldImage -destinationLocation /tmp/
my_grid_images -exclFiles /u01/app/oracle/product/21.0.0/dbhome_1/
relnotes
```

Where:

`/tmp/my_db_images` is a temporary file location where the image zip file is created.

`/tmp/my_grid_images` is a temporary file location where the image zip file is created.

`/u01/app/oracle/product/21.0.0/dbhome_1/relnotes` is the file to be excluded in the newly created gold image.

Applying Patches During an Oracle Database Installation or Upgrade

Starting with Oracle Database 18c, you can download and apply Release Updates (RUs) during an Oracle Database installation or upgrade.

1. Download the patches you want to apply from My Oracle Support:

   https://support.oracle.com

2. Select the **Patches and Updates** tab to locate the patch.

   Oracle recommends that you select **Recommended Patch Advisor**, and enter the product group, release, and platform for your software.

3. Move the patches to an accessible directory like `/tmp`.

4. Change to the Oracle Database home directory:

   ```
   $ cd $ORACLE_HOME
   ```

5. Apply Release Updates (RUs) during the installation or upgrade process:

   ```
   $ ./runInstaller -applyRU patch_directory_location
   ```

6. Complete the remaining steps in the Oracle Database configuration wizard to complete the installation or upgrade.

Running Oracle Database Setup Wizard to Install Oracle Database

Extract the database image files and use the `runInstaller` command to start the installation.

Have all the information you need to provide regarding users groups, and storage paths before you start installation. Oracle recommends that you have your My Oracle Support credentials available during installation. You should also be prepared to run root scripts or provide information to automate root scripts.

1. Log in as the Oracle installation owner user account (`oracle`) that you want to own the software binaries.
2. If this is the first time you are installing Oracle software, then create the Oracle base and the Oracle inventory directories as per the Oracle Optimal Flexible Architecture (OFA) recommendations. Specify the correct owner, group, and permissions for these directories.

```
# mkdir -p /u01/app/oracle
# mkdir -p /u01/app/oraInventory
# chown -R oracle:oinstall /u01/app/oracle
# chown -R oracle:oinstall /u01/app/oraInventory
# chmod -R 775 /u01/app
```

3. Download the Oracle Database installation image files (`db_home.zip`) to a directory of your choice. For example, you can download the image files to the `/tmp` directory.
4. Create an OFA-compliant Oracle home directory and extract the image files that you have downloaded in to this Oracle home directory. For example:

```
$ mkdir -p /u01/app/oracle/product/21.0.0/dbhome_1
$ cd /u01/app/oracle/product/21.0.0/dbhome_1
$ unzip -q /tmp/db_home.zip
```

> **Note:**
>
> - Ensure that the Oracle home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations. Also, unzip the installation image files only in this Oracle home directory that you created.
> - Oracle home or Oracle base cannot be `symlinks`, nor can any of their parent directories, all the way to up to the `root` directory.

5. From the Oracle home directory, run the `runInstaller` command to start the Oracle Database Setup Wizard.

```
$ cd /u01/app/oracle/product/21.0.0/dbhome_1
$ ./runInstaller
```

> **Note:**
>
> - Run the `runInstaller` command from the Oracle home directory only. Do not use the `runInstaller` command that resides at `$ORACLE_HOME/oui/bin/`, or any other location, to install Oracle Database, Oracle Database Client, or Oracle Grid Infrastructure.
> - Use the `runInstaller` command with the `-applyRU` option to install Release Updates (RUs) during an Oracle Database installation or upgrade.

6. In the Select Configuration Option screen, select **Create and configure a single instance database.**

7. Select your installation type.

   Starting with Oracle Database 21c, installation of non-CDB Oracle Database architecture is no longer supported.
   The non-CDB architecture was deprecated in Oracle Database 12c. It is desupported in Oracle Database 21c. Oracle Universal Installer can no longer be used to create non-CDB Oracle Database instances.

   Installation screens vary depending on the installation option you select. Respond to the configuration prompts as needed.

8. Provide information to automate root scripts, or run scripts as root when prompted by the setup wizard.

   If you configure automation for running root scripts, and a root script fails, then you can fix the problem manually, and click **Retry** to run the root script again.

> **Note:**
>
> Click **Help** if you have any questions about the information you are asked to submit during installation.

Related Topics

- Downloading Oracle Software
  Select the method you want to use to download the software.

Installing Oracle Database Using RPM Packages

Learn how to leverage and take advantage of an RPM framework to easily deploy Oracle Database.

- About RPM-Based Oracle Database Installation
  Starting with Oracle Database 18c, you can install a single-instance Oracle Database or an Oracle Database Instant Client software using RPM packages.
- Restrictions and Guidelines for RPM-Based Installations
  Review these guidelines before you start your RPM-based Oracle Database installation.
- RPM Packages Naming Convention
  The RPM packages naming convention includes information such as the version, release, and architecture.
- Running RPM Packages to Install Oracle Database
  Perform the following steps to install and configure Oracle Database using RPM packages.

About RPM-Based Oracle Database Installation

Starting with Oracle Database 18c, you can install a single-instance Oracle Database or an Oracle Database Instant Client software using RPM packages.

An RPM-based installation performs preinstallation checks, extracts the database software, reassigns ownership of the extracted software to the preconfigured user and groups, maintains the Oracle inventory, and executes all root operations required to

configure the Oracle Database software for a single-instance Oracle Database creation and configuration.

The RPM–based installation process detects when the minimum requirements for an installation are not met and prompts you to finish these minimum preinstallation requirements.

An RPM-based installation performs a software-only Oracle Database installation and creates an Oracle home. After the Oracle home is created, you can then use Oracle Database Configuration Assistant (Oracle DBCA) to create an Oracle Database.

The RPM-based installation process provides you with the option to create a database with the default settings using the `/etc/init.d/oracledb_ORCLCDB-21c` service configuration script.

Restrictions and Guidelines for RPM-Based Installations

Review these guidelines before you start your RPM-based Oracle Database installation.

- Patching Oracle Database software using RPMs is not supported. Please use the OPatch utility and follow the regular patching process to apply Oracle Database patches.
- An RPM-based Oracle Database installation is not available for Standard Edition 2.
- RPM-based database upgrades using `rpm -Uvh` is not supported. For Oracle Database upgrades, follow the regular upgrade process.
- An RPM-based installation supports the installation of multiple Oracle Database software versions into different Oracle homes on the same machine.

RPM Packages Naming Convention

The RPM packages naming convention includes information such as the version, release, and architecture.

The naming convention for RPM packages is *name-version-release.architecture.operating system*`.rpm`

The following table describes the variables used in the RPMs `oracle-database-ee-21c-1.0-1.ol7.x86_64.rpm` and `oracle-database-ee-21c-1.0-1.ol8.x86_64.rpm`

Table 11-2 RPM Packages Naming Convention Example

Variable	Example	Description
name	`oracle-database-ee-21c`	The name of the Oracle Database edition and release.
version	`1.0`	The version number of the RPM package.
release	`1`	The release number of the RPM package.
operating system	`ol7` `ol8`	The Oracle Linux operating system version of the RPM package.

Table 11-2 (Cont.) RPM Packages Naming Convention Example

Variable	Example	Description
architecture	x86_64	The architecture of the RPM package.

Related Topics

- RPM Packaging Guide
- Format of the RPM File

Running RPM Packages to Install Oracle Database

Perform the following steps to install and configure Oracle Database using RPM packages.

Installing Oracle Database RPM using yum

1. Log in as `root`.
2. If you are using Oracle Linux and have subscribed to Unbreakable Linux Network (ULN), you can install Oracle Database via a single `yum` command.

   ```
   # yum -y install oracle-database-ee-21c
   ```

   This command downloads and installs the Oracle Database Preinstallation RPM and the Oracle Database RPM packages.

   The installation of Oracle Database software is now complete, continue to the "Creating and Configuring an Oracle Database" section.

Installing Oracle Database RPM Manually

1. Log in as `root`.
2. Download and install the Oracle Database Preinstallation RPM:
   a. On Oracle Linux 7

      ```
      # yum -y install oracle-database-preinstall-21c
      ```

      On Oracle Linux 8

      ```
      # dnf -y install oracle-database-preinstall-21c
      ```

   b. On Red Hat Enterprise Linux 8

      ```
      # curl -o oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
      https://yum.oracle.com/repo/OracleLinux/OL8/latest/x86_64/
      getPackage/oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
      # yum -y localinstall oracle-database-
      preinstall-21c-1.0-1.el8.x86_64.rpm
      ```

After successful installation, you can delete the downloaded RPM file:

```
# rm oracle-database-preinstall-21c-1.0-1.el8.x86_64.rpm
```

3. Access the software download page for Oracle Database RPM-based installation from Oracle Technology Network:

   http://www.oracle.com/technetwork/indexes/downloads/index.html

4. Depending on your operating system, download the `.rpm` file required for performing an RPM-based installation to a directory of your choice. For Oracle Linux 7, download `oracle-database-ee-21c-1.0-1.ol7.x86_64.rpm` and for Oracle Linux 8 download `oracle-database-ee-21c-1.0-1.ol8.x86_64.rpm`.

   For example, for Oracle Linux 7, download the `oracle-database-ee-21c-1.0-1.ol7.x86_64.rpm` file to the `/tmp` directory.

5. Install the database software using the `yum localinstall` command. For example, for Oracle Linux 7:

   ```
   # cd /tmp
   # yum -y localinstall oracle-database-ee-21c-1.0-1.ol7.x86_64.rpm
   ```

   Where, `oracle-database-ee-21c-1.0-1.ol7.x86_64.rpm` is the fully qualified name of the Oracle Linux 7 Oracle Database RPM.

The installation of Oracle Database software is now complete.

Creating and Configuring an Oracle Database

To create a sample database with the default settings, perform the following steps:

1. Log in as `root`.
2. To configure a sample Oracle Database instance, run the following service configuration script:

   ```
   # /etc/init.d/oracledb_ORCLCDB-21c configure
   ```

   > **Note:**
   >
   > You can modify the configuration parameters by editing the `/etc/sysconfig/oracledb_ORCLCDB-21c.conf` file.

   This script creates a container database (`ORCLCDB`) with one pluggable database (`ORCLPDB1`) and configures the listener at the default port (1521).

   Review the status information that is displayed on your screen.

Related Topics

- Automatically Configuring Oracle Linux with Oracle Database Preinstallation RPM
  Use Oracle Database Preinstallation RPM to simplify operating system configuration in preparation for Oracle software installations.
- Removing an RPM-based Oracle Database Installation
  Use this procedure to remove and deinstall a database installed using the `yum` command.

Installing Standard Edition High Availability

Learn how to Install high availability on Oracle Database Standard Edition 2.

- About Standard Edition High Availability
  In this release, you can install Oracle Database Standard Edition 2 in high availability mode.
- Requirements for Installing Standard Edition High Availability
  Review these requirements before you install and deploy the Standard Edition High Availability feature.
- Deploying Standard Edition High Availability
  Learn the process and options to deploy high availability on Oracle Database Standard Edition 2.

About Standard Edition High Availability

In this release, you can install Oracle Database Standard Edition 2 in high availability mode.

Standard Edition High Availability provides cluster-based failover for single-instance Standard Edition Oracle Databases using Oracle Clusterware.

Oracle Standard Edition High Availability benefits from the cluster capabilities and storage solutions that are already part of Oracle Grid Infrastructure, such as Oracle Clusterware, Oracle Automatic Storage Management (Oracle ASM) and Oracle ASM Cluster File System (Oracle ACFS).

Using integrated, shared, and concurrently mounted storage, such as Oracle ASM and Oracle ACFS for database files as well as for unstructured data, enables Oracle Grid Infrastructure to restart an Oracle Database on a failover node much faster than any cluster solution that relies on failing over and remounting volumes and file systems.

Standard Edition High Availability is supported on Linux x86-64.

> **Note:**
>
> This section is specific to Standard Edition High Availability, which provides cluster-based database failover for Standard Edition Oracle Databases 21c and later. For more information about high availability options for Oracle Database, see *Oracle Clusterware Administration and Deployment Guide*.

Requirements for Installing Standard Edition High Availability

Review these requirements before you install and deploy the Standard Edition High Availability feature.

- You must configure Standard Edition High Availability using at least two nodes of a cluster running Oracle Grid Infrastructure 21c or later for Standalone Cluster.
- You must configure Standard Edition High Availability using Oracle Database home of version 21c or later.

> **Note:**
>
> When updating your Oracle Database home using 21c or later, ensure that you apply the Oracle Clusterware (OCW) RU of the same version to the Oracle Database home.

- Ensure that all cluster nodes on which you plan to install Oracle Database have the same operating system configuration, database users, database groups, and resource limits.
- You must store the Oracle Database binaries only on local storage or Oracle Automatic Storage Management Cluster File System (Oracle ACFS).
- You must store the Oracle Database data files only on Oracle ASM or Oracle ACFS.
- If you are using Oracle ACFS for storing data files, then you must register Oracle ACFS as a clusterware resource. The Oracle Database software owner user (`oracle`) must be the mount owner of the Oracle ACFS volume. When using an Oracle home on Oracle ACFS, Oracle recommends that you have Oracle base on the local file system.
- If you are using a local file system, then you must have the same single-instance Standard Edition 2 installation and updates on each node. You must also use the same Oracle base and Oracle home directory structure on each node.
- You must perform the same Oracle home operations on all the nodes.
- You must use an SPFILE for the database instance initialization parameters and a database password file stored for the database instance initialization parameters in Oracle ASM or Oracle ACFS. This approach ensures that the parameters are consistent across all nodes and the password file is available after a failover or a relocation.
- You must register the Standard Edition 2 single-instance database with SCAN listeners as remote listener and node listeners as local listener.

Deploying Standard Edition High Availability

Learn the process and options to deploy high availability on Oracle Database Standard Edition 2.

After installing Oracle Clusterware, as described in *Oracle Grid Infrastructure Installation and Upgrade Guide* for your platform, install single instance Standard Edition 2 Oracle Database software on the cluster nodes on which you want to configure Standard Edition High Availability.

- Installing Standard Edition High Availability Database Software on Local File System
  You can install Oracle Database software binaries on a local file system to enable the Oracle Database Standard Edition high availability feature.
- Installing Standard Edition High Availability Database Software on Oracle ACFS
  You can install Oracle Database software binaries on an Oracle ASM Cluster File System (Oracle ACFS) volume to enable the Oracle Database Standard Edition High Availability feature.

Related Topics

- *Oracle Grid Infrastructure Installation and Upgrade Guide for Linux*

Installing Standard Edition High Availability Database Software on Local File System

You can install Oracle Database software binaries on a local file system to enable the Oracle Database Standard Edition high availability feature.

Ensure that all the cluster nodes, on which you plan to configure Standard Edition High Availability, have the same operating system configuration, database users, database groups, resource limits, and SSH equivalence for the Oracle Database software owner user (`oracle`).

Before you start the installation, have all the information you need about users, groups, and storage paths. You should also be prepared to run root scripts or provide information to automate root scripts.

1. As the `root` user, log into the first cluster node on which you want to configure Standard Edition High Availability and create the Oracle base directory on the local file system. Follow the Oracle Optimal Flexible Architecture (OFA) recommendations and specify the correct owner, group, and permissions for this directory.

   ```
   # mkdir -p /u01/app/oracle
   # chown oracle:oinstall /u01/app/oracle
   ```

2. Log in to the first cluster node as the Oracle Database software owner user (`oracle`).
3. Download the Oracle Database 21c release installation image file (`db_home.zip`) from Oracle Software Delivery Cloud website to a directory of your choice.

   https://edelivery.oracle.com/

4. Create an OFA-compliant Oracle home directory on the local file system and extract the image files that you have downloaded in to this Oracle home directory. For example:

   ```
   $ mkdir -p /u01/app/oracle/product/21.0.0/dbhome_1
   $ cd /u01/app/oracle/product/21.0.0/dbhome_1
   $ unzip -q /tmp/db_home.zip
   ```

5. From the Oracle home directory, run the `runInstaller` command to start the Oracle Database Setup Wizard.

   ```
   $ ./runInstaller
   ```

   > **Note:**
   >
   > Run the `runInstaller` command from the Oracle home directory only. Do not use the `runInstaller` command that resides at `$ORACLE_HOME/oui/bin/`, or any other location, to install Oracle Database.

6. In the Select Configuration Option screen, select **Set Up Software Only**.

7. In the Select Database Installation Option screen, select **Single instance database installation**.
8. In the Select Database Edition screen, select **Standard Edition 2**.
9. Respond to the configuration prompts as needed.
10. Provide information to automate root scripts, or run scripts as root when prompted by the setup wizard.

    If you configure automation for running root scripts, and a root script fails, then you can fix the problem manually, and click **Retry** to run the root script again.

    > **Note:**
    >
    > Click **Help** if you have any questions about the information you are asked to submit during installation.

11. As the `root` user, create the Oracle base directory on all of the other cluster nodes on which you want to configure Standard Edition High Availability.

    ```
    # mkdir -p /u01/app/oracle
    # chown oracle:oinstall /u01/app/oracle
    ```

12. As the `oracle` user, run the `addnode.sh` script from the first node to perform the following operations on the other nodes on which you want to configure Standard Edition High Availability:
    - Copy the Oracle home directory from the first node to the other nodes.
    - Setup Oracle base and Oracle inventory directories on the other nodes.

    ```
    $ $ORACLE_HOME/addnode/addnode.sh -silent
    CLUSTER_NEW_NODES=comma_separated_list_of_other_nodes
    ```

13. As the `root` user, run the `root.sh` script on all the other cluster nodes on which you are configuring Standard Edition High Availability.

    ```
    # /u01/app/oracle/product/21.0.0/dbhome_1/root.sh
    ```

After the Oracle Database software installation is complete, use Oracle Database Configuration Assistant (Oracle DBCA), in either interactive or silent mode, to create a Standard Edition database on the first cluster node on which you installed the Oracle Database software.

For more information about the requirements for creating a database, and the procedure for enabling and configuring Standard Edition High Availability for Oracle Databases, refer to *Oracle Multitenant Administrator's Guide*.

Related Topics

- *Oracle Grid Infrastructure Installation and Upgrade Guide for Linux*
- *Creating a Database with Oracle DBCA*

Installing Standard Edition High Availability Database Software on Oracle ACFS

You can install Oracle Database software binaries on an Oracle ASM Cluster File System (Oracle ACFS) volume to enable the Oracle Database Standard Edition High Availability feature.

Ensure that all the cluster nodes, on which you plan to configure Standard Edition High Availability, have the same operating system configuration, database users, database groups, resource limits, and SSH equivalence for the Oracle Database software owner user (`oracle`) between the nodes.

Before you start the installation, have all the information you need about users groups, and storage paths. You should also be prepared to run root scripts or provide information to automate root scripts.

1. As the `root` user, register Oracle ACFS as an Oracle Clusterware resource specifying the Oracle Database software owner user (`oracle`) as the mount owner.

   ```
   # Grid_home/bin/srvctl add filesystem -volume acfs_volume_name -
   diskgroup diskgroup_name -path mount_point -fstype ACFS -autostart
   ALWAYS -user oracle
   ```

   > **Note:**
   >
   > Mount Oracle ACFS to a mount point where you plan to create Oracle base and Oracle home directories for this installation. For example, `/u01/app/oracle`.

2. As the `oracle` user, mount the Oracle ACFS file system on all of the cluster nodes on which you want to configure Standard Edition High Availability.

   ```
   $ Grid_home/bin/srvctl start filesystem -volume acfs_volume_name -
   diskgroup diskgroup_name
   ```

3. As the `root` user, create the Oracle base directory in an Oracle ACFS volume on the first cluster node on which you want to configure Standard Edition High Availability. Follow the Oracle Optimal Flexible Architecture (OFA) recommendations and specify the correct owner, group, and permissions for this directory.

   ```
   # mkdir -p /u01/app/oracle
   # chown oracle:oinstall /u01/app/oracle
   ```

   > **Note:**
   >
   > If you have configured the Oracle ACFS file system with the `oracle` user as the mount owner and the `/u01/app/oracle` directory as the mount point, then this step is not required.

4. Log in to the first cluster node as the Oracle Database software owner user (`oracle`).
5. Download the Oracle Database 21c installation image file (`db_home.zip`) from Oracle Software Delivery Cloud website to a directory of your choice.

   https://edelivery.oracle.com/
6. Create an OFA-compliant Oracle home directory in an Oracle ACFS volume and extract the image files that you have downloaded in to this Oracle home directory. For example:

   ```
   $ mkdir -p /u01/app/oracle/product/21.0.0/dbhome_1
   $ cd /u01/app/oracle/product/21.0.0/dbhome_1
   $ unzip -q /tmp/db_home.zip
   ```
7. From the Oracle home directory, run the `runInstaller` command to start the Oracle Database Setup Wizard.

   ```
   $ ./runInstaller
   ```

   > **Note:**
   >
   > Run the `runInstaller` command from the Oracle home directory only. Do not use the `runInstaller` command that resides at `$ORACLE_HOME/oui/bin/`, or any other location, to install Oracle Database.
8. In the Select Configuration Option screen, select **Set Up Software Only**.
9. In the Select Database Installation Option screen, select **Single instance database installation**.
10. In the Select Database Edition screen, select **Standard Edition 2**.
11. Respond to the configuration prompts as needed.
12. Provide information to automate root scripts, or run scripts as root when prompted by the setup wizard.

    If you configure automation for running root scripts, and a root script fails, then you can fix the problem manually, and click **Retry** to run the root script again.

    > **Note:**
    >
    > Click **Help** if you have any questions about the information you are asked to submit during installation.
13. Attach the Oracle home on the first node to the other cluster nodes on which you want to configure Standard Edition High Availability.

    ```
    $ $ORACLE_HOME/addnode/addnode.sh -silent
    CLUSTER_NEW_NODES=comma_separated_list_of_other_nodes
    ```
14. As the `root` user, run the `root.sh` script on all the other cluster nodes on which you are configuring Standard Edition High Availability.

    ```
    # /u01/app/oracle/product/21.0.0/dbhome_1/root.sh
    ```

After the Oracle Database software installation is complete, use Oracle Database Configuration Assistant (Oracle DBCA), in either interactive or silent mode, to create a Standard Edition database on the first cluster node on which you installed the Oracle Database software.

For more information about the requirements for creating a database, and the procedure for enabling and configuring Standard Edition High Availability for Oracle Databases, refer to *Oracle Multitenant Administrator's Guide*.

Related Topics

- *Oracle Grid Infrastructure Installation and Upgrade Guide for Linux*
- *Configuring Oracle Automatic Storage Management Cluster File System*
- *Creating a Database with Oracle DBCA*

About Creating Databases Using Oracle DBCA

Create additional databases using Oracle Database Configuration Assistant (Oracle DBCA).

Oracle Universal Installer (OUI) runs Oracle DBCA and creates a starter database for you as part of the single-instance Oracle Database installation. If you want to create additional databases, then you can launch Oracle DBCA as a standalone tool anytime after the Oracle Database installation.

You can run Oracle DBCA in interactive mode or silent mode.

To run Oracle DBCA in interactive mode:

```
$ cd $ORACLE_HOME/bin
$ ./dbca
```

Related Topics

- Creating a Database with Oracle DBCA

12
Configuring Oracle Homes

Understand how read-only Oracle homes work and how you can configure read-only Oracle homes.

- Evolution of Oracle Homes
  Learn about read-only Oracle home concepts like Oracle base home, Oracle base config, and orabasetab.
- Copying demo Directories to Oracle Base Home
  In a read-only mode ORACLE_HOME, you must copy the `demo` directories listed in this topic from ORACLE_HOME to ORACLE_BASE_HOME.
- File Path and Directory Changes in Read-Only Oracle Homes
  Learn from these examples the differences between hierarchical file mappings in a read-only Oracle home compared to a read/write Oracle home.

Evolution of Oracle Homes

Learn about read-only Oracle home concepts like Oracle base home, Oracle base config, and orabasetab.

- About Read-Only Oracle Homes
  Starting with Oracle Database 21c, an Oracle Database installation configures all Oracle Database homes in read-only mode by default.
- About Oracle Base Home
  In a read-only ORACLE_HOME the user-specific files, instance-specific files, and log files reside in a location known as the ORACLE_BASE_HOME.
- About Oracle Base Config
  In a read-only ORACLE_HOME the configuration files reside in a location known as ORACLE_BASE_CONFIG.
- About orabasetab
  The `orabasetab` file is used to define fundamental directories based on $ORACLE_HOME, ORACLE_BASE, ORACLE_BASE_HOME and ORACLE_BASE_CONFIG.

About Read-Only Oracle Homes

Starting with Oracle Database 21c, an Oracle Database installation configures all Oracle Database homes in read-only mode by default.

Before Oracle Database 21c, the default `ORACLE_HOME` layout combined `ORACLE_HOME`, `ORACLE_BASE_HOME` and `ORACLE_BASE_CONFIG` into a single location. Starting with Oracle Database 21c, the only available configuration is a read-only `ORACLE_HOME` where `ORACLE_BASE_HOME` and `ORACLE_BASE_CONFIG` are located separately from `ORACLE_HOME`.

In a read-only Oracle home, all the configuration data and log files reside outside of the read-only Oracle home. This feature allows you to use the read-only Oracle home as a software image that can be distributed across multiple servers.

Apart from the traditional `ORACLE_BASE` and `ORACLE_HOME` directories, the following directories contain files that used to be in `ORACLE_HOME`:

- `ORACLE_BASE_HOME`
- `ORACLE_BASE_CONFIG`

Benefits of a Read-Only Oracle Home

- Enables seamless patching and updating of Oracle Database binaries without extended downtime.
- Simplifies patching and mass rollout, as only one image needs to be updated to distribute a patch to many servers.
- Simplifies provisioning by implementing separation of installation and configuration.

> **Note:**
>
> This feature does not affect how database administrators monitor, diagnose, and tune their system performance.

About Oracle Base Home

In a read-only ORACLE_HOME the user-specific files, instance-specific files, and log files reside in a location known as the ORACLE_BASE_HOME.

Starting with Oracle Database 21c, the only available configuration is a read-only ORACLE_HOME where ORACLE_BASE_HOME and ORACLE_BASE_CONFIG are located separately from ORACLE_HOME.

In a read-only ORACLE_HOME, the ORACLE_BASE_HOME directory is not co-located with ORACLE_HOME but is located at `ORACLE_BASE/homes/HOME_NAME`. Where, HOME_NAME is the internal name for ORACLE_HOME.

For example, the networking directories `network/admin`, `network/trace`, and `network/log` are located in the ORACLE_BASE_HOME directory. In a read-only ORACLE_HOME the networking directories are located in `ORACLE_BASE/homes/HOME_NAME`.

To print the ORACLE_BASE_HOME path, run the `orabasehome` command from the `$ORACLE_HOME/bin` directory:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabasehome
```

For example:

```
$ ./orabasehome
/u01/app/oracle/homes/OraDB21Home1
```

Where, `/u01/app/oracle` is ORACLE_BASE and `OraDB21Home1` is HOME_NAME

About Oracle Base Config

In a read-only ORACLE_HOME the configuration files reside in a location known as ORACLE_BASE_CONFIG.

Starting with Oracle Database 21c, the only available configuration is a read-only ORACLE_HOME where ORACLE_BASE_HOME and ORACLE_BASE_CONFIG are located separately from ORACLE_HOME.

In a read-only ORACLE_HOME, the ORACLE_BASE_CONFIG path is the same as ORACLE_BASE.

`ORACLE_BASE_CONFIG/dbs` contains the configuration files for ORACLE_HOME. Each file in the `dbs` directory contains $ORACLE_SID so that the directory can be shared by many different ORACLE_SIDs.

To print the ORACLE_BASE_CONFIG path, run the `orabaseconfig` command from the `$ORACLE_HOME/bin` directory:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabaseconfig
```

For example:

```
$ ./orabaseconfig
/u01/app/oracle
```

Where, `/u01/app/oracle` is ORACLE_BASE.

About orabasetab

The `orabasetab` file is used to define fundamental directories based on $ORACLE_HOME, ORACLE_BASE, ORACLE_BASE_HOME and ORACLE_BASE_CONFIG.

Starting with Oracle Database 21c, the only available configuration is a read-only ORACLE_HOME where ORACLE_BASE_HOME and ORACLE_BASE_CONFIG are located separately from ORACLE_HOME.

The `orabasetab` file resides in `ORACLE_HOME/install/orabasetab`. To determine if an ORACLE_HOME is read-only, check for the presence of the `orabasetab` file. The `orabasetab` file also defines the ORACLE_BASE and the HOME_NAME of the Oracle home. HOME_NAME is the internal name for ORACLE_HOME.

The last line in the `orabasetab` file, which starts with $ORACLE_HOME, defines the directories for $ORACLE_HOME. The last line consists of four fields, each separate by a colon delimiter`(:)`.

1. The first field matches the current $ORACLE_HOME.
2. The second field defines the ORACLE_BASE for the current ORACLE_HOME.
3. The third field defines the HOME_NAME which is used in constructing the ORACLE_BASE_HOME path in a read-only ORACLE_HOME.
4. The fourth field displays `Y` in a read-only ORACLE_HOME.

In a read-only ORACLE_HOME, the ORACLE_BASE_HOME path is `ORACLE_BASE/homes/HOME_NAME` and ORACLE_BASE_CONFIG is the same as ORACLE_BASE.

Viewing an orabasetab File

1. Log in as the Oracle installation owner user account (`oracle`).
2. Go to the `$ORACLE_HOME/install` directory.

   ```
   $ cd /u01/app/oracle/product/21.0.0/dbhome_1/install
   ```

3. View the contents of the `orabasetab` file.

   ```
   $ cat orabasetab
   /u01/app/oracle/product/21.0.0/dbhome_1:/u01/app/
   oracle:OraDB21Home1:Y:
   ```

   In this example, a `Y` in the fourth field at the end of the line indicates you have a read-only Oracle home.

Copying demo Directories to Oracle Base Home

In a read-only mode ORACLE_HOME, you must copy the `demo` directories listed in this topic from ORACLE_HOME to ORACLE_BASE_HOME.

Oracle Database contains various `demo` directories that include a variety of examples and product demonstrations that you can use to learn about the products, options, and features of Oracle Database. In a read-only mode ORACLE_HOME, you cannot use these `demo` directories in ORACLE_HOME because writes are performed to these `demo` directories when they are used.

Many of the demo directories are not available by default. You must install Oracle Database Examples to view and use the examples and product demonstrations.

Copy the respective `demo` directory to the corresponding location in ORACLE_BASE_HOME. Now, you can use this copy of the `demo` directory.

You must copy the following `demo` directories from ORACLE_HOME to ORACLE_BASE_HOME:

- `jdbc/demo`
- `odbc/demo`
- `ord/http/demo`
- `precomp/demo`
- `rdbms/demo`
- `sqlj/demo`

- `sqlplus/demo`
- `xdk/demo`

You must also create symbolic links for the `odbc/demo`, `precomp/demo`, `rdbms/demo`, and `xdk/demo` demo directories. See the "Creating Symbolic Links" section in this topic.

Copying demo Directories

For example, to copy the `rdbms/demo` directory from ORACLE_HOME to ORACLE_BASE_HOME, perform the following:

1. Login as the Oracle software owner user (`oracle`).
2. Check if the `rdbms/demo` directory is copied to ORACLE_BASE_HOME.

   ```
   $ ls -l -d $(orabasehome)/rdbms/demo
   ```

3. If the `rdbms/demo` directory has not been copied, then create it and copy it.

   ```
   $ mkdir -p $(orabasehome)/rdbms
   $ cp -r $ORACLE_HOME/rdbms/demo $(orabasehome)/rdbms/demo
   ```

Similarly, copy all the `demo` directories listed earlier from ORACLE_HOME to ORACLE_BASE_HOME.

Creating Symbolic Links

You must create symbolic links for the `odbc/demo`, `precomp/demo`, `rdbms/demo`, and `xdk/demo` demo directories.

For `rdbms/demo`, replace `$ORACLE_HOME/rdbms/demo` with a symbolic link to the copy.

1. Ensure that the symbolic link does not already exist.

   ```
   $ ls -l -d $ORACLE_HOME/rdbms/demo
   ```

2. If `$ORACLE_HOME/rdbms/demo` is still the original `demo` directory, rename it and replace it with the symbolic link.

   ```
   $ cd $ORACLE_HOME/rdbms
   $ mv demo demo.installed
   $ ln -s $(orabasehome)/rdbms/demo $ORACLE_HOME/rdbms/demo
   ```

For `odbc/demo`, replace `$ORACLE_HOME/odbc/demo` with a symbolic link to the copy.

1. Ensure that the symbolic link does not already exist.

   ```
   $ ls -l -d $ORACLE_HOME/odbc/demo
   ```

2. If `$ORACLE_HOME/odbc/demo` is still the original `demo` directory, rename it and replace it with the symbolic link.

   ```
   $ cd $ORACLE_HOME/odbc
   $ mv demo demo.installed
   $ ln -s $(orabasehome)/odbc/demo $ORACLE_HOME/odbc/demo
   ```

For `precomp/demo`, replace `$ORACLE_HOME/precomp/demo` with a symbolic link to the copy.

1. Ensure that the symbolic link does not already exist.

   ```
   $ ls -l -d $ORACLE_HOME/precomp/demo
   ```

2. If `$ORACLE_HOME/precomp/demo` is still the original `demo` directory, rename it and replace it with the symbolic link.

   ```
   $ cd $ORACLE_HOME/precomp
   $ mv demo demo.installed
   $ ln -s $(orabasehome)/precomp/demo $ORACLE_HOME/precomp/demo
   ```

The `xdk/demo` directory requires a symbolic link at `$ORACLE_HOME/xdk/include` pointing to `$(orabasehome)/xdk/include` after you copy the `xdk/demo` directory.

1. Ensure that the symbolic link does not already exist:

   ```
   $ ls -l -d $ORACLE_HOME/xdk/include
   ```

2. If the symbolic link does not exist, then, run the following command:

   ```
   $ ln -s $ORACLE_HOME/xdk/include $(orabasehome)/xdk/include
   ```

> **Note:**
>
> In the `plsql/demo` directory, `ncmpdemo.sql` is unusable in read-only mode.

Copying the init.ora File

Copy the `init.ora` file from ORACLE_HOME to ORACLE_BASE_HOME.

1. Login as the Oracle software owner user (`oracle`).
2. Check if the `init.ora` file exists in ORACLE_BASE_HOME.

   ```
   $ ls $(orabasehome)/init.ora
   ```

   If an `init.ora` file exists in ORACLE_BASE_HOME, then update this `init.ora` file to be in-sync with the `$ORACLE_HOME/init.ora` file.
3. If the `init.ora` file does not exist in ORACLE_BASE_HOME, then copy it from ORACLE_HOME.

   ```
   $ cp $ORACLE_HOME/init.ora $(orabasehome)/init.ora
   ```

Related Topics

- *Oracle Database Examples Installation Guide*

File Path and Directory Changes in Read-Only Oracle Homes

Learn from these examples the differences between hierarchical file mappings in a read-only Oracle home compared to a read/write Oracle home.

Prior to Oracle Database 21c, the default ORACLE_HOME layout combined ORACLE_HOME, ORACLE_BASE_HOME and ORACLE_BASE_CONFIG into a single location. Starting with Oracle Database 21c, the only available configuration is a read-only ORACLE_HOME where ORACLE_BASE_HOME and ORACLE_BASE_CONFIG are located separately from ORACLE_HOME.

This example shows an Optimal Flexible Architecture-compliant Oracle Database installation, for the user `oracle`, with the ORACLE_HOME, ORACLE_BASE, ORACLE_BASE_HOME, and ORACLE_BASE_CONFIG logical locations. The database files are mounted on `/u01` and the HOME_NAME is `OraDB21Home1`.

This example also shows the changes in the Oracle Database software defined paths of configuration files, log files, and other directories in a read-only Oracle home when compared to a read/write Oracle home.

Table 12-1 Read-Only Oracle Home and Read/Write Oracle Home File Path Examples

Directory	Read-Only Oracle Home File Path (21c onwards)	Read/Write Oracle Home File Path (before 21c)
ORACLE_HOME	`/u01/app/oracle/product/21.0.0/dbhome_1`	`/u01/app/oracle/product/19.0.0/dbhome_1`
ORACLE_BASE	`/u01/app/oracle/`	`/u01/app/oracle/`
ORACLE_BASE_HOME	ORACLE_BASE/homes/HOME_NAME (or) `/u01/app/oracle/homes/OraDB21Home1`	ORACLE_HOME (or) `/u01/app/oracle/product/19.0.0/dbhome_1`
ORACLE_BASE_CONFIG	ORACLE_BASE (or) `/u01/app/oracle/`	ORACLE_HOME (or) `/u01/app/oracle/product/19.0.0/dbhome_1`
`network`	ORACLE_BASE_HOME/network/admin (or) `/u01/app/oracle/homes/OraDB21Home1/network/admin`	ORACLE_HOME/network/admin (or) `/u01/app/oracle/product/19.0.0/dbhome_1/network/admin`

Table 12-1 (Cont.) Read-Only Oracle Home and Read/Write Oracle Home File Path Examples

Directory	Read-Only Oracle Home File Path (21c onwards)	Read/Write Oracle Home File Path (before 21c)
`dbs`	ORACLE_BASE_CONFIG/dbs (or) `/u01/app/oracle/dbs`	ORACLE_HOME/dbs (or) `/u01/app/oracle/product/19.0.0/dbhome_1/dbs`
`hs`	ORACLE_BASE_HOME/hs/admin (or) `/u01/app/oracle/homes/OraDB21Home1/hs/admin`	ORACLE_HOME/hs/admin (or) `/u01/app/oracle/product/19.0.0/dbhome_1/hs/admin`
`mgw`	ORACLE_BASE_HOME/mgw/admin (or) `/u01/app/oracle/homes/OraDB21Home1/mgw/admin`	ORACLE_HOME/mgw/admin (or) `/u01/app/oracle/product/19.0.0/dbhome_1/mgw/admin`
`drdaas`	ORACLE_BASE_HOME/drdaas/admin (or) `/u01/app/oracle/homes/OraDB21Home1/drdaas/admin`	ORACLE_HOME/drdaas/admin (or) `/u01/app/oracle/product/19.0.0/dbhome_1/drdaas/admin`

- To print the ORACLE_BASE_HOME path, run:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabasehome
```

- To print the ORACLE_BASE_CONFIG path, run:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabaseconfig
```

13
Oracle Database Postinstallation Tasks

Complete configuration tasks after you install Oracle Database.

You are required to complete some configuration tasks after Oracle Database is installed. In addition, Oracle recommends that you complete additional tasks immediately after installation. You must also complete product-specific configuration tasks before you use those products.

> **Note:**
>
> This chapter describes basic configuration only. Refer to product-specific administration and tuning guides for more detailed configuration and tuning information.

- Required Postinstallation Tasks
  Download and apply required patches for your software release after completing your initial installation.
- Recommended Postinstallation Tasks
  Oracle recommends that you complete these tasks after installation.
- About Changes in Default SGA Permissions for Oracle Database
  Starting with Oracle Database 12c Release 2 (12.2.0.1), by default, permissions to read and write to the System Global Area (SGA) are limited to the Oracle software installation owner.
- Checking Installed Oracle Database Contents and Directory Location
  Use these steps to check the contents and directory location of an Oracle Database installation:
- Enabling and Disabling Oracle Database Options After Installation
  The `chopt` tool changes your database options after installation.
- Starting Oracle Enterprise Manager Database Express
  To start Oracle Enterprise Manager Database Express, use the EM Express URL provided by Oracle Database Configuration Assistant (Oracle DBCA) during the database installation and creation.
- Creating a Fast Recovery Area
  During an Oracle Restart installation, you can create only one disk group. During an Oracle Clusterware installation, you can create multiple disk groups. If you plan to add an Oracle Database for a standalone server or an Oracle RAC database, then you should create the fast recovery area for database files.
- Cloning Oracle Database
  Cloning an Oracle home involves creating a copy of the Oracle home and then configuring it for a new environment.

Required Postinstallation Tasks

Download and apply required patches for your software release after completing your initial installation.

- Downloading Release Update Patches
  Download and install Release Updates (RU) and Release Update Revisions (RUR) patches for your Oracle software after you complete installation.
- Unlocking and Resetting Oracle Database User Passwords
  Passwords for all Oracle system administration accounts except SYS, SYSTEM, and DBSMP are revoked after installation. Before you use a locked account, you must unlock it and reset its password.

Downloading Release Update Patches

Download and install Release Updates (RU) and Release Update Revisions (RUR) patches for your Oracle software after you complete installation.

Starting with Oracle Database 18c, Oracle provides quarterly updates in the form of Release Updates (RU) and Release Update Revisions (RUR). Oracle no longer releases patch sets. For more information, see My Oracle Support Note 2285040.1.

Check the My Oracle Support website for required updates for your installation.

1. Use a web browser to view the My Oracle Support website:

   https://support.oracle.com

2. Log in to My Oracle Support website.

   > **Note:**
   >
   > If you are not a My Oracle Support registered user, then click **Register for My Oracle Support** and register.

3. On the main My Oracle Support page, click **Patches & Updates**.
4. In the **Patch Search** region, select **Product or Family (Advanced)**.
5. On the **Product or Family (Advanced)** display, provide information about the product, release, and platform for which you want to obtain patches, and click **Search**.

   The Patch Search pane opens, displaying the results of your search.

6. Select the patch number and click **ReadMe**.

   The README page is displayed. It contains information about the patch and how to apply the patches to your installation.

7. Uncompress the Oracle patch updates that you downloaded from My Oracle Support.

Related Topics

- My Oracle Support note 2285040.1

Unlocking and Resetting Oracle Database User Passwords

Passwords for all Oracle system administration accounts except SYS, SYSTEM, and DBSMP are revoked after installation. Before you use a locked account, you must unlock it and reset its password.

If you created a preconfigured database during the installation, but you did not unlock accounts required to use the database, then you must unlock and reset those accounts using these procedures.

- Requirements for Database Passwords
  To secure your database, use passwords that satisfy the Oracle recommended password requirements, even the passwords for predefined user accounts.
- Oracle Database System Privileges Accounts and Passwords
  Review these system privileges accounts after installation in preparation for unlocking accounts and changing passwords.
- Guidelines for Changing System Privileges Account Passwords
  Follow these rules for changing Oracle Database system privileges account passwords.
- Locking and Unlocking User Accounts
- Using SQL*Plus to Unlock Accounts and Reset Passwords
  Use this SQL*Plus procedure to unlock and reset user account passwords.

Requirements for Database Passwords

To secure your database, use passwords that satisfy the Oracle recommended password requirements, even the passwords for predefined user accounts.

Oracle Database provides a set of predefined user accounts. Create passwords in a secure fashion. If you have default passwords, change these passwords to secure passwords.

You can manage the security for Oracle Database users in various ways:

- Enforce restrictions on the way that passwords are created
- Create user profiles
- Use user resource limits to further secure user accounts

Related Topics

- *Oracle Database Security Guide*

Oracle Database System Privileges Accounts and Passwords

Review these system privileges accounts after installation in preparation for unlocking accounts and changing passwords.

All databases created by the Database Configuration Assistant (DBCA) include the `SYS`, `SYSTEM`, and `DBSNMP` database accounts. In addition, Oracle Database provides several other administrative accounts. Before using these accounts, you must unlock them and reset their passwords.

Starting with Oracle Database 21c, all sample schemas are distributed on GitHub:

https://github.com/oracle/db-sample-schemas

> **Note:**
>
> This list contains some of the important system privileges user accounts, but it is not complete. Use Oracle Enterprise Manager Database Express to view the complete list of database accounts.

Table 13-1 Partial List of Oracle Database System Privileges Accounts Locked After Installation

User Name	Description	For More Information
ANONYMOUS	Enables HTTP access to Oracle XML DB.	*Oracle XML DB Developer's Guide*
APEX_050100	The account that owns the Oracle Application Express schema and metadata.	*Oracle Application Express App Builder User's Guide*
APEX_PUBLIC_USER	The minimally privileged account used for Oracle Application Express configuration with Oracle Application Express Listener or Oracle HTTP Server and `mod_plsql`.	*Oracle Application Express App Builder User's Guide*
APPQOSSYS	Used for storing and managing all data and metadata required by Oracle Quality of Service Management.	None
AUDSYS	The account where the unified audit data trail resides.	*Oracle Database Security Guide*
CTXSYS	The Oracle Text account.	*Oracle Text Application Developer's Guide*
DBSFWUSER	The account used to run the DBMS_SFW_ACL_ADMIN package.	*Oracle Database PL/SQL Packages and Types Reference*
DBSNMP	The account used by the Management Agent component of Oracle Enterprise Manager to monitor and manage the database.	*Oracle Enterprise Manager Cloud Control Administrator's Guide*
DIP	The account used by the Directory Integration Platform (DIP) to synchronize the changes in Oracle Internet Directory with the applications in the database.	None
DVSYS	There are two roles associated with this account. The Database Vault owner role manages the Database Vault roles and configurations. The Database Vault Account Manager is used to manage database user accounts. **Note**: Part of Oracle Database Vault user interface text is stored in database tables in the DVSYS schema. By default, only the English language is loaded into these tables. You can use the `DVSYS.DBMS_MACADM.ADD_NLS_DATA` procedure to add other languages to Oracle Database Vault.	*Oracle Database Vault Administrator's Guide*

Table 13-1 (Cont.) Partial List of Oracle Database System Privileges Accounts Locked After Installation

User Name	Description	For More Information
DVF	The account owned by Database Vault that contains public functions to retrieve the Database Vault Factor values.	*Oracle Database Vault Administrator's Guide*
FLOWS_FILES	The account owns the Oracle Application Express uploaded files.	*Oracle Application Express App Builder User's Guide*
GGSYS	The internal account used by Oracle GoldenGate. It should not be unlocked or used for a database login.	None
GSMADMIN_INTERNAL	The internal account that owns the Global Data Services schema. It should not be unlocked or used for a database login.	*Oracle Database Global Data Services Concepts and Administration Guide*
GSMCATUSER	The account used by Global Service Manager to connect to the Global Data Services catalog.	*Oracle Database Global Data Services Concepts and Administration Guide*
GSMUSER	The account used by Global Service Manager to connect to the database.	*Oracle Database Global Data Services Concepts and Administration Guide*
LBACSYS	The Oracle Label Security administrator account. Starting with Oracle Database 18c, the LBACSYS user account is created as a schema-only account.	*Oracle Label Security Administrator's Guide*
MDDATA	The schema used by Oracle Spatial and Graph for storing geocoder and router data.	*Oracle Spatial and Graph Developer's Guide*
MDSYS	The Oracle Spatial and Graph administrator account.	*Oracle Spatial and Graph Developer's Guide*
OUTLN	The account that supports plan stability. Plan stability enables you to maintain the same execution plans for the same SQL statements. OUTLN acts as a role to centrally manage metadata associated with stored outlines.	None
ORACLE_OCM	This account contains the instrumentation for configuration collection used by the Oracle Configuration Manager.	None
REMOTE_SCHEDULER_AGENT	The account to disable remote jobs on a database. This account is created during the remote scheduler agent configuration. You can disable the capability of a database to run remote jobs by dropping this user.	*Oracle Database Administrator's Guide*
SYS	The account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSTEM	Another account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSBACKUP	The account used to perform backup and recovery tasks.	*Oracle Database Administrator's Guide*

Table 13-1 (Cont.) Partial List of Oracle Database System Privileges Accounts Locked After Installation

User Name	Description	For More Information
SYSKM	The account used to perform encryption key management.	*Oracle Database Administrator's Guide*
SYSDG	The account used to administer and monitor Oracle Data Guard.	*Oracle Database Administrator's Guide*
SYSRAC	The account used to administer Oracle Real Application Clusters (RAC).	*Oracle Database Administrator's Guide*
SYS$UMF	The account used to administer Remote Management Framework, including the remote Automatic Workload Repository (AWR).	*Oracle Database Performance Tuning Guide*
WMSYS	The account used to store the metadata information for Oracle Workspace Manager.	*Oracle Database Workspace Manager Developer's Guide*
XDB	The account used for storing Oracle XML DB data and metadata.	*Oracle XML DB Developer's Guide*
XS$NULL	The internal account that represents the absence of a database schema user in a session, and indicates an application user session is in use. XS$NULL cannot be authenticated to a database, nor can it own any database schema objects, or possess any database privileges.	*Oracle Database Real Application Security Administrator's and Developer's Guide*

Except for the accounts provided with the Oracle Sample Schemas, most of these database accounts are locked by default and created without passwords as schema only. This prevents malicious users from logging into these accounts using the default password set during catalog creation. To find the status of an account, query the `AUTHENTICATION_TYPE` column of the `DBA_USERS` data dictionary view. If `AUTHENTICATION_TYPE` is schema only, then the status is `NONE`.

Many of these accounts are automatically created when you run standard scripts such as the various `cat*.sql` scripts. To find user accounts that are created and maintained by Oracle, query the `USERNAME` and `ORACLE_MAINTAINED` columns of the `ALL_USERS` data dictionary view. If the output for `ORACLE_MAINTAINED` is `Y`, then you must not modify the user account except by running the script that was used to create it.

Related Topics

- *Oracle Database Security Guide*
- *Oracle Database Sample Schemas*

Guidelines for Changing System Privileges Account Passwords

Follow these rules for changing Oracle Database system privileges account passwords.

Before you use a locked account, you must unlock it and reset its password. Passwords for all Oracle system administration accounts except SYS, SYSTEM, and

DBSNMP are revoked after installation. If you created a starter database during the installation, Oracle Database Configuration Assistant displays a screen with your database information and the Password Management button. Use the Password Management button to unlock only the user names you use.

For more information about how to create a secure password, see:

Oracle Database Security Guide

Locking and Unlocking User Accounts

To temporarily deny access to the database for a particular user account, you can lock the user account. If the user then attempts to connect, then the database displays an error message and does not allow the connection. You can unlock the user account when you want to permit database access again for that user. You can use Oracle Enterprise Manager Database Express (EM Express) to lock and unlock user accounts.

To lock or unlock a user account:

1. In EM Express, go to the Users page, as described in the “Viewing User Accounts" topic.
2. Click the desired user account.
3. From the **Actions** menu, select **Alter Account**.

   The Alter Account page appears.
4. Do one of the following:
   - To lock the account, enable the **Account Locked** option, and then click **OK**.
   - To unlock the account, disable the **Account Locked** option, and then click **OK**.

Using SQL*Plus to Unlock Accounts and Reset Passwords

Use this SQL*Plus procedure to unlock and reset user account passwords.

1. Log in as the Oracle Database software owner user.
2. Set the ORACLE_HOME and ORACLE_SID environment variables.

   Bourne, Bash or Korn shell:

   ```
   $ ORACLE_SID=orcl
   $ export ORACLE_SID
   $ ORACLE_HOME=/u01/app/oracle/product/21.0.0/dbhome_1
   $ export ORACLE_HOME
   ```

   C shell:

   ```
   % setenv ORACLE_SID orcl
   % setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
   ```
3. Start SQL*Plus and log in as the SYS user, connecting as SYSDBA:

   ```
   $ $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSDBA
   Enter password: sys_password
   ```

4. To unlock an account:

```
ALTER USER account ACCOUNT UNLOCK;
```

5. To reset the password:

```
ALTER USER user_name IDENTIFIED BY new_password;
```

> **Note:**
>
> If you unlock an account but do not reset the password, then the password remains expired. The first time someone connects as that user, they must change the user's password.

Related Topics

- *Oracle Database Administrator's Guide*

Recommended Postinstallation Tasks

Oracle recommends that you complete these tasks after installation.

- Creating a Backup of the root.sh Script
  Oracle recommends that you back up the `root.sh` script after you complete an installation.
- Setting Language and Locale Preferences for Client Connections
  Configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set.
- Recompiling All Invalid Objects
  Identify and recompile invalid objects on the CDB and PDBs using the `catcon` utility to run `utlrp.sql` after you install, patch, or upgrade a database.
- About Installing Oracle Autonomous Health Framework
  Oracle Autonomous Health Framework is pre-installed with Oracle Database.

Creating a Backup of the root.sh Script

Oracle recommends that you back up the `root.sh` script after you complete an installation.

If you install other products in the same Oracle home directory subsequent to this installation, then Oracle Universal Installer updates the contents of the existing `root.sh` script during the installation. If you require information contained in the original `root.sh` script, then you can recover it from the backed up `root.sh` file.

Setting Language and Locale Preferences for Client Connections

Configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set.

You must configure client applications connecting to an Oracle Database according to your locale preferences and your I/O device character set. If your applications do not

have their own specific methods to configure locale preferences, then the method you use to configure an Oracle database client connection depends on the access API you use to connect to the database. Check your application documentation, before you configure locale preferences for your applications.

For applications that connect to Oracle Databases using Oracle Call Interface (OCI) use NLS_LANG and other client settings with names that start with `NLS_` to set the locale conventions and client character set for Oracle Database sessions. It is important that you set the character set part of the NLS_LANG value properly. The character set you set must correspond to the character set used by your I/O devices, which in case of Microsoft Windows is either the ANSI Code Page (for GUI applications), such as WE8MSWIN1252, or the OEM Code Page (for Console mode applications), such as US8PC437. By doing this, the OCI API is notified about the character set of data that it receives from the application. OCI can then convert this data correctly to and from the database character set.

NLS_LANG and the other NLS settings can be specified either as environment variables or as Windows Registry settings. Environment variable values take precedence over Registry values.

Oracle Universal Installer sets a default value for the NLS_LANG setting in Registry when it creates a new Oracle home on Microsoft Windows. The NLS_LANG value is based on the language of the Windows user interface, which is the language of Windows menu items and dialog box labels.

Caution:

Failure to set the client character set correctly can cause data loss.

Java applications that connect to Oracle Databases by using Oracle JDBC do not use NLS_LANG. Instead, Oracle JDBC maps the default locale of the Java VM in which the application runs to the Oracle Database language and territory settings. Oracle JDBC then configures the connected database session using these settings. Because Java works internally in Unicode, the client character set is always set to Unicode. Unless an application explicitly changes it, the default locale of the Java VM is set based on the locale of the user operating system on which the Java VM runs. Check your Java VM documentation for information about configuring the Java VM default locale.

Note:

In 3-tier architecture deployments, application servers that are database clients can have settings in their configuration files that specify the NLS_LANG value or the Java VM locale. Check the documentation accompanying these servers.

See Also:

Oracle Database Globalization Support Guide for more information about configuring user locale preferences

Recompiling All Invalid Objects

Identify and recompile invalid objects on the CDB and PDBs using the `catcon` utility to run `utlrp.sql` after you install, patch, or upgrade a database.

> **Note:**
>
> If you upgraded using the AutoUpgrade utility, then AutoUpgrade automatically takes care of this task during the upgrade. You do not need to perform this task.

Oracle recommends that you use the `catcon.pl` utility to run `utlrp.sql` on all containers in your container database (CDB). The `utlrp.sql` script recompiles all invalid objects. Run the script immediately after installation, to ensure that users do not encounter invalid objects.

1. Change directory to *`Oracle_home`*`/rdbms/admin`. For example

   ```
   $ cd $ORACLE_HOME/rdbms/admin
   ```

2. Use the `catcon.pl` script in the Oracle home to run `utlrp.sql`. For example:

   ```
   $ORACLE_HOME/perl/bin/perl catcon.pl --n 1 --e --b utlrp --d
   '''.''' utlrp.sql
   ```

   Note the following conditions of this use case:

   - `--n` parameter: is set to 1, so the script runs each PDB recompilation in sequence.
   - `--e` parameter: turns echo on.
   - `--b` parameter: Sets the log file base name. It is set to `utlrp`.

Expect a time delay for the serial recompilation of PDBs to complete. Depending on the number of PDBs that you are upgrading, the recompilation can extend significantly beyond the time required for the upgrade scripts to complete.

The `utlrp.sql` script automatically recompiles invalid objects in either serial or parallel recompilation, based on both the number of invalid objects, and on the number of CPUs available. CPUs are calculated using the number of CPUs (`cpu_count`) multiplied by the number of threads for each CPU (`parallel_threads_per_cpu`). On Oracle Real Application Clusters (Oracle RAC), this number is added across all Oracle RAC nodes.

For more information about catcon utility syntax and options, refer to *Oracle Multitenant Administrator's Guide*.

Related Topics

- Syntax and Parameters for catcon.pl

About Installing Oracle Autonomous Health Framework

Oracle Autonomous Health Framework is pre-installed with Oracle Database.

Oracle Autonomous Health Framework performs proactive heath checks and collects diagnostics data for the Oracle software stack. Oracle Autonomous Health Framework updates are available on My Oracle Support and also shipped with Oracle Database Release Updates.

Oracle recommends that you update to the latest version of Oracle Autonomous Health Framework by either applying the latest Database Release Updates or by downloading and installing it from My Oracle Support Note 2550798.1:

https://support.oracle.com/epmos/faces/DocContentDisplay?id=2550798.1&parent=DOCUMENTATION&sourceId=USERGUIDE

Oracle Autonomous Health Framework includes the functionality from Oracle ORAchk, Oracle EXAchk, and Oracle Trace File Analyzer (TFA). Oracle Autonomous Health Framework extends health check coverage to the entire Oracle software stack, based on critical and reoccurring problems. Oracle Autonomous Health Framework proactively scans for known problems with Oracle products and deployments, including the following:

- Single-instance Oracle Database
- Oracle Grid Infrastructure
- Oracle Real Application Clusters
- Maximum Availability Architecture (MAA) Validation
- Upgrade Readiness Validations
- Oracle GoldenGate
- Oracle Exadata
- Oracle SuperCluster
- Oracle Zero Data Loss Recovery Appliance

Related Topics

- *Oracle Autonomous Health Framework Checks and Diagnostics User's Guide*

About Changes in Default SGA Permissions for Oracle Database

Starting with Oracle Database 12c Release 2 (12.2.0.1), by default, permissions to read and write to the System Global Area (SGA) are limited to the Oracle software installation owner.

In previous releases, both the Oracle installation owner account and members of the OSDBA group had access to shared memory. The change in Oracle Database 12c Release 2 (12.2) and later releases to restrict access by default to the Oracle installation owner account provides greater security than previous configurations. However, this change may prevent DBAs who do not have access to the Oracle installation owner account from administering the database.

The Oracle Database initialization parameter `ALLOW_GROUP_ACCESS_TO_SGA` determines if the Oracle Database installation owner account (`oracle` in Oracle documentation examples)

is the only user that can read and write to the database System Global Area (SGA), or if members of the OSDBA group can read the SGA. In Oracle Database 12c Release 2 (12.2) and later releases, the default value for this parameter is FALSE, so that only the Oracle Database installation owner has read and write permissions to the SGA. Group access to the SGA is removed by default. This change affects all Linux and UNIX platforms.

If members of the OSDBA group require read access to the SGA, then you can change the initialization parameter `ALLOW_GROUP_ACCESS_TO_SGA` setting from FALSE to TRUE. Oracle strongly recommends that you accept the default permissions that limit access to the SGA to the `oracle` user account.

Related Topics

- *Oracle Database Reference*

Checking Installed Oracle Database Contents and Directory Location

Use these steps to check the contents and directory location of an Oracle Database installation:

1. Go to `$ORACLE_HOME/oui/bin`.
2. Start Oracle Universal Installer.

   ```
   $ ./runInstaller
   ```

3. Click **Installed Products** to display the Inventory dialog box on the Welcome screen.
4. Select an Oracle Database product from the list to check the installed contents.
5. Click **Details** to find additional information about an installed product.
6. Click **Close** to close the Inventory dialog box.
7. Click **Cancel** to close Oracle Universal Installer, and then click **Yes** to confirm.

Enabling and Disabling Oracle Database Options After Installation

The `chopt` tool changes your database options after installation.

When you install Oracle Database, some options are enabled and others are disabled. To enable or disable a particular database feature for an Oracle home, shut down the database and use the `chopt` tool.

- Chopt Tool
  Use the `chopt` tool after installation to add or remove Oracle Database options.

Chopt Tool

Use the `chopt` tool after installation to add or remove Oracle Database options.

Purpose

The `chopt` tool is a command-line utility that enables and disables database options.

Prerequisites

You must complete installation before you can use the `chopt` tool.

File Path

The tool is located in the *`ORACLE_HOME`*`/bin` directory

Syntax

`chopt` [`enable` | `disable`] *db_option*

Options

Command Option	Description
`oaa`	Oracle Advanced Analytics
`olap`	Oracle OLAP
`partitioning`	Oracle Partitioning
`rat`	Oracle Real Application Testing

> **Note:**
>
> When you enable or disable OLAP, you must run the `SYS.XOQ_VALIDATE` and `SYS.APS_VALIDATE` procedures to update the database registry. When you disable OLAP, its status in the database registry should be OPTION OFF and when you enable OLAP, its status in the database registry should be VALID.

Examples

To use the `chopt` tool to modify your Oracle Database, you must shut down the database before you run the `chopt` tool, and then start up the database after you add or remove database options.

Example 13-1 Enabling Oracle Data Mining Using Chopt

The following example shows how to use the `chopt` tool to enable the Oracle Data Mining option in an Oracle Database called Sales:

```
cd $ORACLE_HOME/bin
srvctl stop database -d Sales
chopt enable oaa
srvctl start database -d Sales
```

Starting Oracle Enterprise Manager Database Express

To start Oracle Enterprise Manager Database Express, use the EM Express URL provided by Oracle Database Configuration Assistant (Oracle DBCA) during the database installation and creation.

Oracle Enterprise Manager Database Express (EM Express) is deprecated, and will be removed in a future Oracle Database release.

EM Express is a web-based database management tool that is built inside the Oracle Database. It supports key performance management and basic database administration functions. Many of EM Express's capabilities are also available in Oracle SQL Developer, which is included in all Oracle Database editions. Oracle recommends that you replace your use of EM Express with Oracle SQL Developer.

Creating a Fast Recovery Area

During an Oracle Restart installation, you can create only one disk group. During an Oracle Clusterware installation, you can create multiple disk groups. If you plan to add an Oracle Database for a standalone server or an Oracle RAC database, then you should create the fast recovery area for database files.

- About the Fast Recovery Area and the Fast Recovery Area Disk Group
  The fast recovery area is a unified storage location for all Oracle Database files related to recovery. Enabling rapid backups for recent data can reduce requests to system administrators to retrieve backup tapes for recovery operations.
- Creating the Fast Recovery Area Disk Group
  Procedure to create the fast recovery area disk group.

About the Fast Recovery Area and the Fast Recovery Area Disk Group

The fast recovery area is a unified storage location for all Oracle Database files related to recovery. Enabling rapid backups for recent data can reduce requests to system administrators to retrieve backup tapes for recovery operations.

Database administrators can define the `DB_RECOVERY_FILE_DEST` parameter to the path for the fast recovery area to enable on disk backups and rapid recovery of data. When you enable fast recovery in the `init.ora` file, Oracle Database writes all RMAN backups, archive logs, control file automatic backups, and database copies to the fast recovery area. RMAN automatically manages files in the fast recovery area by deleting obsolete backups and archiving files no longer required for recovery.

Oracle recommends that you create a fast recovery area disk group. Oracle Clusterware files and Oracle Database files can be placed on the same disk group, and you can also place fast recovery files in the same disk group. However, Oracle recommends that you create a separate fast recovery disk group to reduce storage device contention.

The fast recovery area is enabled by setting the `DB_RECOVERY_FILE_DEST` parameter. The size of the fast recovery area is set with `DB_RECOVERY_FILE_DEST_SIZE`. As a general rule, the larger the fast recovery area, the more useful it becomes. For ease of use, Oracle recommends that you

create a fast recovery area disk group on storage devices that can contain at least three days of recovery information. Ideally, the fast recovery area is large enough to hold a copy of all of your data files and control files, the online redo logs, and the archived redo log files needed to recover your database using the data file backups kept under your retention policy.

Multiple databases can use the same fast recovery area. For example, assume you have created a fast recovery area disk group on disks with 150 GB of storage, shared by 3 different databases. You can set the size of the fast recovery for each database depending on the importance of each database. For example, if database1 is your least important database, database2 is of greater importance, and database3 is of greatest importance, then you can set different `DB_RECOVERY_FILE_DEST_SIZE` settings for each database to meet your retention target for each database: 30 GB for database1, 50 GB for database2, and 70 GB for database3.

Creating the Fast Recovery Area Disk Group

Procedure to create the fast recovery area disk group.

1. Go to the Oracle Grid Infrastructure home `bin` directory, and start Oracle ASM Configuration Assistant (ASMCA).

   For example:

   ```
   $ cd /u01/app/oracle/product/21.0.0/grid/bin
   $ ./asmca
   ```

   ASMCA opens the home window.
2. Click **Disk Groups** in the left panel to open the Disk Groups tab.
3. Click **Create** to create a new disk group.

   The Create Disk Group window opens.
4. Provide configuration information for the fast recovery area as prompted:

   In the Disk Group Name field, enter a descriptive name for the fast recovery area group. For example: `FRA`.

   In the Redundancy section, select the level of redundancy you want to use. For example: `Normal`

   In the Select Member Disks field, select eligible disks you want to add to the fast recovery area, and click **OK**. If you do not see the disks, click the **Change Disk Discovery Path** button and provide a path and pattern match for the disk. For example, `/dev/sd*` for local Oracle ASM disks.
5. When the Fast Recovery Area disk group creation is complete, click **Exit** and click **Yes** to confirm closing the ASMCA application.

Cloning Oracle Database

Cloning an Oracle home involves creating a copy of the Oracle home and then configuring it for a new environment.

If you are performing multiple Oracle Database installations, then you may want to use cloning to create each Oracle home, because copying files from an existing Oracle Database installation takes less time than creating a new version of them. This method is also useful if

the Oracle home that you are cloning has had patches applied to it. When you clone the Oracle home, the new Oracle home has the patch updates.

- Cloning an Oracle Database Home Using an Image File
  Learn how to clone an Oracle Database home using the `-createGoldImage` option.
- Cloning an Oracle Database Home Using clone.pl
  Learn how to clone an Oracle home using the `clone.pl` script.

Cloning an Oracle Database Home Using an Image File

Learn how to clone an Oracle Database home using the `-createGoldImage` option.

Create an image file from the source Oracle home using the `-createGoldImage` option, unzip the image file, and perform a software-only Oracle Database installation to register the Oracle Database product.

1. Log in as the Oracle installation owner user account (`oracle`).
2. Go to the source Oracle home directory.

   ```
   $ cd /u01/app/oracle/product/21.0.0/dbhome_1
   ```

3. Use the `runInstaller` command with the `-createGoldImage` option to create a gold image from the source Oracle home.

   For example, to create an image file of `dbhome_1` and save it in the temporary destination location `my_db_images`:

   ```
   ./runInstaller -createGoldImage -destinationLocation /tmp/
   my_db_images
   ```

4. Create a new OFA-compliant Oracle home directory and extract the image file that you have created in to this destination Oracle home directory. For example:

   ```
   $ mkdir -p /u01/app/oracle/product/21.0.0/dbhome_2
   $ cd /u01/app/oracle/product/21.0.0/dbhome_2
   $ unzip -q /tmp/my_db_images/db_home.zip
   ```

   > **Note:**
   >
   > Oracle home or Oracle base cannot be symlinks, nor can any of their parent directories, all the way to up to the `root` directory.

5. From this new destination Oracle home directory, run the `runInstaller` command to start the Oracle Database Setup Wizard and register Oracle Database.

   ```
   $ cd /u01/app/oracle/product/21.0.0/dbhome_2
   $ ./runInstaller
   ```

6. In the Select Configuration Option screen, select **Set Up Software Only** to perform a software-only Oracle Database installation.

7. Select your installation type and respond to the configuration prompts as needed.
8. Provide information to automate root scripts, or run scripts as `root` when prompted by the setup wizard.

> **Note:**
>
> Click **Help** if you have any questions about the information you are asked to submit during installation.

9. Oracle Database is now successfully registered. Run Oracle Database Configuration Assistant (Oracle DBCA) to create a database and configure the listener for the newly-cloned Oracle home.

```
$ cd /u01/app/oracle/product/21.0.0/dbhome_2/bin
$ ./dbca
```

Cloning an Oracle Database Home Using clone.pl

Learn how to clone an Oracle home using the `clone.pl` script.

> **Note:**
>
> - This procedure using the `clone.pl` script is deprecated starting Oracle Database 19c and can be removed in a future release. Hence, Oracle recommends that you use the software-only installation option, available in the database installer, instead of `clone.pl` to clone your database.
> - During cloning, Oracle Universal Installer (OUI) prompts you to run scripts that require `root` privileges.

1. Verify that the installation of Oracle Database that you want clone is successful.

   You can do this by reviewing the `installActionsdate_time.log` file for the installation session, which is typically located in the `/u01/app/oracle/oraInventory/logs` directory.

   If you install patches, then check their status using the following:

   ```
   $ cd $ORACLE_HOME/OPatch
   ```

   Include `$ORACLE_HOME/OPatch` in `$PATH`

   ```
   $ opatch lsinventory
   ```

2. Stop all processes related to the Oracle home.
3. Create a ZIP or TAR file with the Oracle home (but not the Oracle base) directory.

For example, if the source Oracle installation is in the path `/u01/app/oracle/product/21.0.0/dbhome_1`, then you zip the `dbhome_1` directory by using the following command:

```
# zip -r dbhome_1.zip /u01/app/oracle/product/21.0.0/dbhome_1
```

You can also use the TAR command. For example:

```
# tar -cvf dbhome_1.tar /u01/app/oracle/product/21.0.0/dbhome_1
```

Do not include the `admin`, `fast_recovery_area`, and `oradata` directories that are under the Oracle base directory. These directories are created in the target installation later, when you create a new database there.

4. Copy the ZIP or TAR file to the root directory of the target computer. If you use File Transfer Protocol (FTP), then transfer the ZIP or TAR file in binary mode only.
5. Extract the ZIP or TAR file content using the following command:

```
# unzip -d / dbhome_1.zip
# tar -xvf dbhome_1.tar
```

6. If necessary, change the ownership of the Oracle Database home to the Oracle Database installation owner user (`oracle`) belonging to the Oracle Inventory group (`oinstall`).

```
# chown -R oracle:oinstall /u01/app/oracle/product/21.0.0/dbhome_1
```

7. On the target computer, change the directory to the unzipped Oracle base home directory, and remove all the `.ora` (`*.ora`) files present in the unzipped `ORACLE_BASE_HOME/network/admin` directory.

   To print the ORACLE_BASE_HOME path, run:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/21.0.0/dbhome_1
$ cd $ORACLE_HOME/bin
$ ./orabasehome
```

8. Delete unnecessary files from the unzipped Oracle base home directory.

   The unzipped Oracle base home directory contains files that are relevant only to the source Oracle home. The following example shows how to remove these unnecessary files from the unzipped Oracle base home directory:

   Remove the `.ora` files from the `ORACLE_BASE_HOME/network/admin` directory, and remove the old database entries from the `ORACLE_BASE/dbs` directory.

```
# cd $ORACLE_HOME/bin
# ./orabasehome
/u01/app/oracle/homes/OraDB21Home1
# cd /u01/app/oracle/homes/OraDB21Home1
# rm -rf network/admin/*.ora
# cd /u01/app/oracle
# rm dbs/old_database_entries
```

9. From the `$ORACLE_HOME/clone/bin` directory, run the `clone.pl` file for the unzipped Oracle home.

   Use the following syntax (you can also include one or more of the extended Oracle Database groups in the syntax):

   ```
   $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/clone/bin/clone.pl
   ORACLE_BASE="target_oracle_base" ORACLE_HOME="target_oracle_home"
   OSDBA_GROUP=OSDBA_privileged_group
   OSOPER_GROUP=OSOPER_privileged_group
   OSBACKUPDBA_GROUP=OSBACKUPDBA_privileged_group
   OSDGDBA_GROUP=OSDGDBA_privileged_group
   OSKMDBA_GROUP=OSKMDBA_privileged_group
   OSRACDBA_GROUP=OSRACDBA_privileged_group -defaultHomeName
   ```

   For example:

   ```
   $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/clone/bin/clone.pl
   ORACLE_BASE="/u01/app/oracle" ORACLE_HOME="/u01/app/oracle/product/21.0.0/
   dbhome_1"
   OSDBA_GROUP=dba OSOPER_GROUP=oper OSBACKUPDBA_GROUP=backupdba
   OSDGDBA_GROUP=dgdba OSKMDBA_GROUP=kmdba OSRACDBA_GROUP=racdba -
   defaultHomeName
   ```

   > **Note:**
   >
   > - In this command, if you do not provide the parameters for the operating system groups, then `clone.pl` uses the operating system group values from the source home.
   > - Run the `$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/clone/bin/clone.pl -help` command for more information about the command option flags.

   OUI starts, and then records the cloning actions in the `cloneActionstimestamp.log` file. This log file is typically located in `/u01/app/oracle/oraInventory/logs` directory.

10. Use the following commands to run Net Configuration Assistant to configure the connection information for the new database:

    ```
    $ cd $ORACLE_HOME/bin
    $ ./netca
    ```

11. Use the following commands to run Database Configuration Assistant to create a new Oracle Database for the newly-cloned oracle home:

    ```
    $ cd $ORACLE_HOME/bin
    $ ./dbca
    ```

Related Topics

- *Oracle Clusterware Administration and Deployment Guide*
- *Oracle Database Administrator's Guide*

14
Removing Oracle Database Software

Use the `deinstall` command that is included in Oracle homes to remove Oracle software. Oracle does not support the removal of individual products or components.

> **Caution:**
>
> If you have a single-instance database on a node in a cluster, and if you have multiple databases with the same global database name (GDN), then you cannot use the `deinstall` command to remove one database only.

- About Oracle Deinstallation Options
  You can stop and remove Oracle Database software and components in an Oracle Database home with the `deinstall` command.
- Oracle Deinstallation (Deinstall)
  You can run the `deinstall` command from an Oracle home directory after installation.
- Deinstallation Examples for Oracle Database
  Use these examples to help you understand how to run the `deinstall` command.
- Downgrading Oracle Restart
  Use this procedure to deconfigure and downgrade Oracle Restart, or to troubleshoot Oracle Restart installation errors.
- Removing an RPM-based Oracle Database Installation
  Use this procedure to remove and deinstall a database installed using the `yum` command.
- Deinstalling Previous Release Grid Home
  Use this procedure to deinstall the previous release Grid home.

About Oracle Deinstallation Options

You can stop and remove Oracle Database software and components in an Oracle Database home with the `deinstall` command.

You can remove the following software using `deinstall` :

- Oracle Database
- Oracle Grid Infrastructure, which includes Oracle Clusterware and Oracle Automatic Storage Management (Oracle ASM)
- Oracle Real Application Clusters (Oracle RAC)
- Grid Infrastructure Management Repository (GIMR)
- Oracle Database Client

The `deinstall` command is available in Oracle home directories after installation. It is located in the `$ORACLE_HOME/deinstall` directory.

`deinstall` creates a response file by using information in the Oracle home and using the information you provide. You can use a response file that you generated previously by running the `deinstall` command using the `-checkonly` option. You can also edit the response file template.

If you run `deinstall` to remove an Oracle Grid Infrastructure installation, then the deinstaller prompts you to run the `deinstall` command as the `root` user. For Oracle Grid Infrastructure for a cluster, the script is `rootcrs.sh`, and for Oracle Grid Infrastructure for a standalone server (Oracle Restart), the script is `roothas.sh`.

> **Note:**
>
> You must run the `deinstall` command from the same release to remove Oracle software. Do not run the `deinstall` command from a later release to remove Oracle software from an earlier release. For example, do not run the `deinstall` command from the 21c Oracle home to remove Oracle software from an existing 19c Oracle home.

If the software in the Oracle home is not running (for example, after an unsuccessful installation), then `deinstall` cannot determine the configuration, and you must provide all the configuration details either interactively or in a response file.

In addition, before you run `deinstall` for Oracle Grid Infrastructure installations:

- Dismount Oracle Automatic Storage Management Cluster File System (Oracle ACFS) and disable Oracle Automatic Storage Management Dynamic Volume Manager (Oracle ADVM).
- If Grid Naming Service (GNS) is in use, then notify your DNS administrator to delete the subdomain entry from the DNS.

Files Deleted by deinstall

When you run `deinstall`, if the central inventory (`oraInventory`) contains no other registered homes besides the home that you are deconfiguring and removing, then `deinstall` removes the following files and directory contents in the Oracle base directory of the Oracle Database installation owner:

- `admin`
- `cfgtoollogs`
- `checkpoints`
- `diag`
- `oradata`
- `fast_recovery_area`

Oracle strongly recommends that you configure your installations using an Optimal Flexible Architecture (OFA) configuration, and that you reserve Oracle base and Oracle home paths for exclusive use of Oracle software. If you have any user data in these locations in the Oracle base that is owned by the user account that owns the Oracle software, then `deinstall` deletes this data.

> **Caution:**
>
> `deinstall` deletes Oracle Database configuration files, user data, and fast recovery area (FRA) files even if they are located outside of the Oracle base directory path.

Oracle Deinstallation (Deinstall)

You can run the `deinstall` command from an Oracle home directory after installation.

Purpose

`deinstall` stops Oracle software, and removes Oracle software and configuration files on the operating system for a specific Oracle home.

Syntax

The `deinstall` command uses the following syntax:

```
(./deinstall [-silent] [-checkonly] [-paramfile complete path of input
response file]
[-params name1=value name2=value . . .]
[-o complete path of directory for saving files]
[-tmpdir complete path of temporary directory to use]
[-logdir complete path of log directory to use] [-local] [-
skipLocalHomeDeletion] [-skipRemoteHomeDeletion] [-help]
```

Parameters

Parameter	Description
`-silent`	Use this flag to run `deinstall` in noninteractive mode. This option requires one of the following: • A working system that it can access to determine the installation and configuration information. The `-silent` flag does not work with failed installations. • A response file that contains the configuration values for the Oracle home that is being deinstalled or deconfigured. You can generate a response file to use or modify by running `deinstall` with the `-checkonly` flag. `deinstall` then discovers information from the Oracle home to deinstall and deconfigure. It generates the response file that you can then use with the `-silent` option. You can also modify the template file `deinstall.rsp.tmpl`, located in the `$ORACLE_HOME/deinstall/response` directory.

Parameter	Description
`-checkonly`	Use this flag to check the status of the Oracle software home configuration. Running `deinstall` with the `-checkonly` flag does not remove the Oracle configuration. The `-checkonly` flag generates a response file that you can then use with the `deinstall` command and `-silent` option.
`-paramfile` *complete path of input response file*	Use this flag to run `deinstall` with a response file in a location other than the default. When you use this flag, provide the complete path where the response file is located. The default location of the response file is `$ORACLE_HOME/deinstall/response`.
`-params` [*name1=value name2=value name3=value . . .*]	Use this flag with a response file to override one or more values to change in a response file you have created.
`-o` *complete path of directory for saving response files*	Use this flag to provide a path other than the default location where the response file (`deinstall.rsp.tmpl`) is saved. The default location of the response file is `$ORACLE_HOME/deinstall/response` .
`-tmpdir` *complete path of temporary directory to use*	Use this flag to specify a non-default location where `deinstall` writes the temporary files for the deinstallation.
`-logdir` *complete path of log directory to use*	Use this flag to specify a non-default location where `deinstall` writes the log files for the deinstallation.
`-local`	Use this flag on a multinode environment to deinstall Oracle software in a cluster. When you run `deinstall` with this flag, it deconfigures and deinstalls the Oracle software on the local node (the node where `deinstall` is run). On remote nodes, it deconfigures Oracle software, but does not deinstall the Oracle software.
`-skipLocalHomeDeletion`	Use this flag in Oracle Grid Infrastructure installations on a multinode environment to deconfigure a local Grid home without deleting the Grid home.
`-skipRemoteHomeDeletion`	Use this flag in Oracle Grid Infrastructure installations on a multinode environment to deconfigure a remote Grid home without deleting the Grid home.
`-help`	Use this option to obtain additional information about the command option flags.

Deinstallation Examples for Oracle Database

Use these examples to help you understand how to run the `deinstall` command.

Run `deinstall` from the `$ORACLE_HOME/deinstall` directory. The deinstallation starts without prompting you for the Oracle home path.

```
$ ./deinstall
```

You can generate a deinstallation response file by running `deinstall` with the `-checkonly` flag. Alternatively, you can use the response file template located at `$ORACLE_HOME/deinstall/response/deinstall.rsp.tmpl`. If you have a response file, then use the optional flag `-paramfile` to provide a path to the response file.

In the following example, the `deinstall` command is in the path`/u01/app/oracle/product/21.0.0/dbhome_1/deinstall`. It uses a response file called `my_db_paramfile.tmpl` in the software owner location `/home/usr/oracle`:

```
$ cd /u01/app/oracle/product/21.0.0/dbhome_1/deinstall
$ ./deinstall -paramfile /home/usr/oracle/my_db_paramfile.tmpl
```

To remove the Oracle Grid Infrastructure home, use the `deinstall` command in the Oracle Grid Infrastructure home.

In this example, the Oracle Grid Infrastructure home is `/u01/app/oracle/product/21.0.0/grid`

```
$ cd  /u01/app/oracle/product/21.0.0/grid/deinstall
$ ./deinstall -paramfile /home/usr/oracle/my_grid_paramfile.tmpl
```

Downgrading Oracle Restart

Use this procedure to deconfigure and downgrade Oracle Restart, or to troubleshoot Oracle Restart installation errors.

1. As the `oracle` user, create a backup of the SPFILE to a PFILE from 21c Oracle home to 19c Oracle home.

   ```
   $ CREATE PFILE='/u01/app/oracle/product/19.0.0/dbhome_1/dbs/SIDinit.ora'
   FROM SPFILE
   ```

   You can use this PFILE during the database downgrade process.

2. List all Oracle Databases on the server with their version, unique name of the database, and Oracle home information.

   ```
   $ srvctl config database -home
   ```

3. Downgrade Oracle Database. Refer to *Oracle Database Upgrade Guide* for more information about required pre-downgrade tasks, downgrade tasks, post-downgrade tasks, and compatibility information.

> **Note:**
>
> Downgrade Oracle Database only if the Oracle Database version is higher than the Oracle Restart version to which you are downgrading Oracle Restart.

4. As the `oracle` user, downgrade the Oracle Restart resources corresponding to the Oracle Database, only if you have downgraded your Oracle Database.

```
$ srvctl downgrade database -d db_unique_name -
oraclehome $ORACLE_HOME -t to_version
```

5. Inspect the Oracle Restart configuration of each database, service, and listener.

```
$ srvctl config database -db db_unique_name
$ srvctl config service -db db_unique_name
$ srvctl config listener -listener listener_name
```

   Make a note of the configuration information and use this information when adding the components back to Oracle Restart.

6. Stop all databases and listeners that are running before you downgrade Oracle Restart.

```
$ srvctl stop database -db db_unique_name
$ srvctl stop listener [-listener listener_name]
```

7. Deinstall Oracle Restart while retaining the existing Oracle Automatic Storage Management (Oracle ASM) disk group.

```
$ cd /u01/app/oracle/product/21.0.0/dbhome_1/deinstall
$ ./deinstall
If you want to retain the existing diskgroups and associated ASM
filter driver labels
or if any of the information detected is incorrect, you can modify
by entering 'y'.
Do you want to modify above information  (y|n) [n]: y
Specify the ASM Diagnostic Destination
[ASM_diagnostic_destination]:
Specify the diskstring [AFD_diskstring]:
Specify the diskgroups that are managed by this ASM instance
[+DATA]:
De-configuring ASM will drop the diskgroups and clear the AFD
labels at cleanup time.
Do you want deconfig tool to drop the diskgroups and  all AFD
labels (y|n) [y]: n
```

8. As the `grid` user, download the Oracle Grid Infrastructure 19c image files and extract the files to the Grid home.

```
$ mkdir -p /u01/app/grid/product/19.0.0/grid
$ chown grid:oinstall /u01/app/grid/product/19.0.0/grid
```

```
$ cd /u01/app/grid/product/19.0.0/grid
$ unzip -q download_location/grid_home.zip
```

download_location/grid_home.zip is the path of the downloaded Oracle Grid Infrastructure image file.

> **Note:**
>
> You must extract the image software into the directory where you want your new Grid home to be located.

9. Start the Oracle Grid Infrastructure installation wizard.

```
$ /u01/app/grid/product/19.0.0/grid/gridSetup.sh
```

10. Select the Configuration Option as **Set Up Software Only** and select the installation options as prompted.
11. As the `grid` user, update the Oracle central inventory (`oraInventory`).

```
$ /u01/app/oracle/product/19.0.0/grid/oui/bin/runInstaller -
updateNodeList ORACLE_HOME=downgraded_Grid_home -defaultHomeName CRS=TRUE
```

12. As the `root` user, run `roothas.sh` to configure Oracle Restart without Oracle ASM.

```
# /u01/app/oracle/product/19.0.0/grid/crs/install/roothas.sh
```

13. As the `grid` user, start Oracle ASM Configuration Assistant (Oracle ASMCA).

```
$ /u01/app/oracle/product/19.0.0/bin/asmca
```

14. Create a new temporary Oracle ASM disk group.

> **Note:**
>
> Configure Oracle ASM to use the new disk group and provide `/.../.*` as the disk string so that Oracle ASM can discover the new disk.

15. Mount the existing Oracle ASM disk group and click **Exit** to close Oracle ASMCA.
16. Move the password file from the temporary disk group to the existing disk group.

```
$ /u01/app/oracle/product/19.0.0/bin/asmcmd
ASMCMD> pwmove --asm new_diskgroup/orapwasm existing_diskgroup/orapwasm -f
```

17. Remove the SPFILE from the existing Oracle ASM disk group.

```
ASMCMD> rm existing_diskgroup/ASM/ASMPARAMETERFILE/REGISTRY.253.1075263285
```

18. Copy the SPFILE from the new Oracle ASM disk group to the existing disk group.

```
ASMCMD> spcopy -u +new_diskgroup/asm/ASMPARAMETERFILE/
registry.253.722601213 +existing_diskgroup/spfile
ASMCMD> exit
```

19. Restart the high availability stack to verify that Oracle ASM starts with SPFILE from the existing disk group.
20. Start Oracle ASMCA and delete the new temporary disk group.
21. As the `oracle` user, add the components back to Oracle Restart with the same attributes that you noted in earlier steps.
    a. Add Oracle Database to the Oracle Restart configuration.

    ```
    $ srvctl add database -db db_unique_name -oraclehome Oracle_home
    -spfile spfile_name
    ```

    For the `-spfile` parameter, provide the SPFILE that you noted in Step 5.

    b. Add the listener to the Oracle Restart configuration.

    ```
    $ srvctl add listener -listener listener_name -oraclehome
    Oracle_home
    ```

    For the `-oraclehome` parameter, provide the Oracle home from where the listener was running before the downgrade.

    c. Add each service to the database using the `srvctl add service` command.

    ```
    $ srvctl add service -db db_unique_name -service
    service_name_list
    ```

22. Start the components that you have added back to Oracle Restart.
    a. Start the Oracle database.

    ```
    $ srvctl start database -db db_unique_name
    ```

    b. Start the listener.

    ```
    $ srvctl start listener -listener listener_name
    ```

    c. Start each service using the `srvctl start service` command.
23. Connect to Oracle Database and start the pluggable database (PDB).

```
$ sqlplus / as sysdba
SQL> ALTER PLUGGABLE DATABASE pdb_name OPEN;
SQL> exit
```

24. Verify that the Oracle Restart resources are running after the downgrade.

```
$ /u01/app/oracle/product/19.0.0/bin/crsctl stat res -t
```

Related Topics

- *Oracle Database Upgrade Guide*

Removing an RPM-based Oracle Database Installation

Use this procedure to remove and deinstall a database installed using the `yum` command.

1. Log in as `root`.
2. Run the following command to remove the RPM-based database installation:

```
# yum -y remove oracle-database-ee-21c
```

The `yum` command detects the default configured components in the Oracle home such as the database (ORCLCDB) and the listener (LISTENER) and removes these components for you.
Apart from the default database (ORCLCDB) and the listener (LISTENER), if the `yum` command detects other configured components in the Oracle home such as additional databases or listeners, then it stops the deinstallation process. A message is displayed that instructs you to remove these configured components manually. To remove these configured components:

1. Login as the Oracle Database installation owner user (`oracle`).
2. To remove any database associated with your installation, use Oracle Database Configuration Assistant (Oracle DBCA).

```
$ cd $ORACLE_HOME/bin
$ ./dbca
```

3. To remove any listeners associated with your installation, run Oracle Net Configuration Assistant (Oracle NETCA).

```
$ cd $ORACLE_HOME/bin
$ ./netca
```

4. Log in as `root` and run the `yum` command again to remove the rpm-based database installation.

```
# yum -y remove oracle-database-ee-21c
```

Related Topics

- Deleting a Database Using Oracle DBCA
- Using Oracle Net Configuration Assistant

Deinstalling Previous Release Grid Home

Use this procedure to deinstall the previous release Grid home.

For upgrades from previous releases, if you want to deinstall the previous release Grid home, then perform the following steps:

1. Log in as the `root` user.
2. Manually change the permissions of the previous release Grid home.

   ```
   # chmod -R 775 /u01/app/oracle/product/19.0.0/grid
   # chown -R oracle:oinstall /u01/app/oracle/product/19.0.0/grid
   # chown oracle /u01/app/oracle/product/19.0.0
   ```

   In this example:

   - `/u01/app/oracle/product/19.0.0/grid` is the previous release Oracle Grid Infrastructure for a standalone server home
   - `oracle` is the Oracle Grid Infrastructure installation owner user
   - `oinstall` is the name of the Oracle Inventory group (OINSTALL group)
   - `/u01/app/oracle/product/19.0.0` is the parent directory of the previous Grid home.
3. Log in as the Oracle Grid Infrastructure software owner user (`oracle`) and run the `deinstall` command.

A

Completing Preinstallation Tasks Manually

You can complete the preinstallation configuration tasks manually.

Oracle recommends that you use Oracle Universal Installer and Cluster Verification Utility fixup scripts to complete minimal configuration settings. If you cannot use fixup scripts, then complete minimum system settings manually.

- Configuring Kernel Parameters for Linux
  These topics explain how to configure kernel parameters manually for Linux if you cannot complete them using the fixup scripts.
- Configuring Default Thread Limits Value for SUSE Linux
  If you are on SUSE Linux Enterprise Server 12 SP2 or later, or SUSE Linux Enterprise Server 15 or later, then set the `DefaultTasksMax` parameter value to `65535`.
- Configuring Storage Device Path Persistence Using Oracle ASMLIB
  To use Oracle ASMLIB to configure Oracle ASM devices, complete the following tasks:
- Configuring Storage Device Path Persistence Manually
  You can maintain storage file path persistence by creating a rules file.

Configuring Kernel Parameters for Linux

These topics explain how to configure kernel parameters manually for Linux if you cannot complete them using the fixup scripts.

- Minimum Parameter Settings for Installation
  Use this table to set parameters manually if you cannot use the fixup scripts.
- Changing Kernel Parameter Values
  Use these instructions to display and change the kernel parameter values if they are different from the minimum recommended value.
- Configuring Additional Kernel Settings for SUSE Linux
  Additional kernel parameter settings configuration is required on SUSE Linux Enterprise Server systems.
- Setting UDP and TCP Kernel Parameters Manually
  If you do not use a Fixup script or CVU to set ephemeral ports, then set TCP/IP ephemeral port range parameters to provide enough ephemeral ports for the anticipated server workload.

Minimum Parameter Settings for Installation

Use this table to set parameters manually if you cannot use the fixup scripts.

Note:

- Unless otherwise specified, the kernel parameter and shell limit values shown in the following table are minimum values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. See the operating system documentation for more information about tuning kernel parameters.
- If the current value for any parameter is greater than the value listed in this table, then the Fixup scripts do not change the value of that parameter.

Table A-1 Minimum Operating System Resource Parameter Settings

Parameter	Value	File

`semmsl`
`semmns`
`semopm`
`semmni`
	250
	32000
	100
	128
		`/proc/sys/kernel/sem`

`shmall`	Greater than or equal to the value of `shmmax`, in pages.	`/proc/sys/kernel/shmall`
`shmmax`	Half the size of physical memory in bytes See My Oracle Support Note 567506.1 for additional information about configuring shmmax.	`/proc/sys/kernel/shmmax`
`shmmni`	4096	`/proc/sys/kernel/shmmni`
`panic_on_oops`	1	`/proc/sys/kernel/panic_on_oops`
`file-max`	6815744	`/proc/sys/fs/file-max`
`aio-max-nr`	1048576 Note: This value limits concurrent outstanding requests and should be set to avoid I/O subsystem failures.	`/proc/sys/fs/aio-max-nr`
`ip_local_port_range`	Minimum: 9000 Maximum: 65500	`/proc/sys/net/ipv4/ip_local_port_range`
`rmem_default`	262144	`/proc/sys/net/core/rmem_default`
`rmem_max`	4194304	`/proc/sys/net/core/rmem_max`
`wmem_default`	262144	`/proc/sys/net/core/wmem_default`
`wmem_max`	1048576	`/proc/sys/net/core/wmem_max`

Related Topics

- Setting UDP and TCP Kernel Parameters Manually
  If you do not use a Fixup script or CVU to set ephemeral ports, then set TCP/IP ephemeral port range parameters to provide enough ephemeral ports for the anticipated server workload.

Changing Kernel Parameter Values

Use these instructions to display and change the kernel parameter values if they are different from the minimum recommended value.

Use the following commands to display the current values of the kernel parameters:

Table A-2 Commands to Display Kernel Parameter Values

Parameter	Command
`semmsl`, `semmns`, `semopm`, and `semmni`	`# /sbin/sysctl -a | grep sem` This command displays the value of the semaphore parameters in the order listed.
`shmall`, `shmmax`, and `shmmni`	`# /sbin/sysctl -a | grep shm` This command displays the details of the shared memory segment sizes.
`file-max`	`# /sbin/sysctl -a | grep file-max` This command displays the maximum number of file handles.
`ip_local_port_range`	`# /sbin/sysctl -a | grep ip_local_port_range` This command displays a range of port numbers.
`rmem_default`	`# /sbin/sysctl -a | grep rmem_default`
`rmem_max`	`# /sbin/sysctl -a | grep rmem_max`
`wmem_default`	`# /sbin/sysctl -a | grep wmem_default`
`wmem_max`	`# /sbin/sysctl -a | grep wmem_max`
`aio-max-nr`	`# /sbin/sysctl -a | grep aio-max-nr`

If you used the Oracle Database Preinstallation RPM to complete you preinstallation configuration tasks, then the Oracle Database Preinstallation RPM sets these kernel parameters for you. However, if you did not use the Oracle Database Preinstallation RPM or the kernel parameters are different from the minimum recommended value, then to change these kernel parameter values:

1. Using any text editor, create or edit the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file, and add or edit lines similar to:

```
fs.aio-max-nr = 1048576
fs.file-max = 6815744
kernel.shmall = 2097152
kernel.shmmax = 4294967295
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
```

```
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
```

By specifying the values in the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file, the values persist on system restarts.

2. To change the current values of the kernel parameters:

```
# /sbin/sysctl --system
```

Review the output. If the values are incorrect, edit the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file, then enter this command again.

3. Confirm that the values are set correctly:

```
# /sbin/sysctl -a
```

4. Restart the computer, or run `sysctl --system` to make the changes in the `/etc/sysctl.d/97-oracle-database-sysctl.conf` file available in the active kernel memory.

Guidelines for Setting Kernel Parameter Values

- If you used the Oracle Database Preinstallation RPM, then your kernel parameter settings reside in the `/etc/sysctl.d/99-oracle-database-server-21c-preinstall-sysctl.conf` file.
- Include lines only for the kernel parameter values to change. For the semaphore parameters (`kernel.sem`), you must specify all four values. If any of the current values are larger than the minimum value, then specify the larger value.
- The `/etc/sysctl.conf` file has been deprecated.
- Avoid setting kernel parameter values in multiple files under `/etc/sysctl.d/`. The file with a lexically later name under `/etc/sysctl.d/` takes precedence, followed by `/etc/sysctl.conf`. Oracle recommends that you use the Oracle Database Preinstallation RPM which, among other preinstallation tasks, also sets the kernel parameter values for your database installation.

> **See Also:**
>
> `sysctl.conf(5)` and `sysctl.d(5)` man pages for more information

Configuring Additional Kernel Settings for SUSE Linux

Additional kernel parameter settings configuration is required on SUSE Linux Enterprise Server systems.

Perform these steps on SUSE Linux Enterprise Server systems only:

1. Enter the following command to enable the system to read the `/etc/sysctl.conf` file when it restarts:

```
# /sbin/chkconfig boot.sysctl on
```

2. Enter the GID of the oinstall group as the value for the parameter `/proc/sys/vm/hugetlb_shm_group`.

   For example, where the oinstall group GID is `501`:

   ```
   # echo 501 > /proc/sys/vm/hugetlb_shm_group
   ```

   Use `vi` to add the following text to `/etc/sysctl.conf`, and enable the `boot.sysctl` script to run on system restart:

   ```
   vm.hugetlb_shm_group=501
   ```

   > **Note:**
   >
   > Only one group can be defined as the `vm.hugetlb_shm_group`.

Setting UDP and TCP Kernel Parameters Manually

If you do not use a Fixup script or CVU to set ephemeral ports, then set TCP/IP ephemeral port range parameters to provide enough ephemeral ports for the anticipated server workload.

Ensure that the lower range is set to at least 9000 or higher, to avoid Well Known ports, and to avoid ports in the Registered Ports range commonly used by Oracle and other server ports. Set the port range high enough to avoid reserved ports for any applications you may intend to use. If the lower value of the range you have is greater than 9000, and the range is large enough for your anticipated workload, then you can ignore Oracle Universal Installer warnings regarding the ephemeral port range.

For example, with IPv4, use the following command to check your current range for ephemeral ports:

```
$ cat /proc/sys/net/ipv4/ip_local_port_range
32768 61000
```

In the preceding example, the lowest port (32768) and the highest port (61000) are set to the default range.

If necessary, update the UDP and TCP ephemeral port range to a range high enough for anticipated system workloads, and to ensure that the ephemeral port range starts at 9000 and above. For example:

```
# echo 9000 65500 > /proc/sys/net/ipv4/ip_local_port_range
```

Oracle recommends that you make these settings permanent. For example, as `root`, use a text editor to open `/etc/sysctl.conf`, and add or change to the following: `net.ipv4.ip_local_port_range = 9000 65500`, and then restart the network:

```
# /etc/rc.d/init.d/network restart
```

Refer to your Linux distribution system administration documentation for information about automating ephemeral port range alteration on system restarts.

Configuring Default Thread Limits Value for SUSE Linux

If you are on SUSE Linux Enterprise Server 12 SP2 or later, or SUSE Linux Enterprise Server 15 or later, then set the `DefaultTasksMax` parameter value to `65535`.

Increase the default thread limits, `DefaultTasksMax` value from `512` to `65535` to avoid running into service failures.

1. View the contents of `/etc/systemd/system.conf` to know the current `DefaultTasksMax` value. Alternatively, run the following command:

   ```
   $ systemctl show --property DefaultTasksMax
   ```

2. If your `DefaultTasksMax` value is not `65535`, then uncomment the line in `/etc/systemd/system.conf` and set the value to `65535`.
3. To enable the new settings, reboot your system or run the following command:

   ```
   $ systemctl daemon-reload
   ```

Configuring Storage Device Path Persistence Using Oracle ASMLIB

To use Oracle ASMLIB to configure Oracle ASM devices, complete the following tasks:

> **Note:**
>
> To create a database during the installation using the Oracle ASM library driver, you must choose an installation method that runs ASMCA in interactive mode. You must also change the disk discovery string to `ORCL:*`.
>
> Oracle ASMLIB is not supported on IBM:Linux on System z.

- About Oracle ASM with Oracle ASMLIB
  Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade.
- Installing and Configuring Oracle ASMLIB Software
  Review this information to install and configure the Oracle Automatic Storage Management library driver software manually.
- Configuring Disk Devices to Use Oracle ASMLIB
  Configure disk devices to use in an Oracle Automatic Storage Management disk group.
- Administering Oracle ASMLIB and Disks
  Review this information to administer the Oracle Automatic Storage Management library driver and disks.

- Deinstalling Oracle ASMLIB On Oracle Database
  If Oracle ASM library driver (Oracle ASMLIB) is installed but you do not use it for device path persistence, then deinstall Oracle ASMLIB:

About Oracle ASM with Oracle ASMLIB

Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade.

The Oracle Automatic Storage Management library driver simplifies the configuration and management of block disk devices by eliminating the need to rebind block disk devices used with Oracle Automatic Storage Management (Oracle ASM) each time the system is restarted.

With Oracle ASMLIB, you define the range of disks you want to have made available as Oracle ASM disks. Oracle ASMLIB maintains permissions and disk labels that are persistent on the storage device, so that the label is available even after an operating system upgrade.

> **Note:**
>
> If you configure disks using Oracle ASMLIB, then you must change the disk discovery string to `ORCL:*`. If the diskstring is set to `ORCL:*`, or is left empty (""), then the installer discovers these disks.

Installing and Configuring Oracle ASMLIB Software

Review this information to install and configure the Oracle Automatic Storage Management library driver software manually.

Oracle ASMLIB is included with the Oracle Linux packages, and with SUSE Linux Enterprise Server. If you are a member of the Unbreakable Linux Network, then you can install the Oracle ASMLIB RPMs by subscribing to the Oracle Linux channel, and using `yum` to retrieve the most current package for your system and kernel. For additional information, see the following URL:

http://www.oracle.com/technetwork/topics/linux/asmlib/index-101839.html

To install and configure the Oracle Automatic Storage Management library driver software manually, perform the following steps:

1. Enter the following command to determine the kernel version and architecture of the system:

   ```
   # uname -rm
   ```

2. Depending on your operating system version, download the required Oracle Automatic Storage Management library driver packages and driver:

   `http://www.oracle.com/technetwork/server-storage/linux/asmlib/index-101839.html`

> **See Also:**
>
> My Oracle Support note 1089399.1 for information about Oracle ASMLIB support with Red Hat distributions:
>
> https://support.oracle.com/rs?type=doc&id=1089399.1

3. Switch to the `root` user:

```
$ su -
```

4. Depending on your operating system, you must install the `oracleasmlib` and `oracleasm-support` packages for all Oracle ASMLIB installations.

   Method 1: After you have downloaded the Oracle Automatic Storage Management library driver packages as described in Step 2, run the following `yum` commands to install the `oracleasmlib` and `oracleasm-support` packages

```
# yum install -y oracleasm
# yum install -y oracleasm-support
# yum install oracleasmlib
```

   Method 2: Alternatively, install the following packages in sequence, where *`version`* is the version of the Oracle Automatic Storage Management library driver, *`arch`* is the system architecture, and *`kernel`* is the version of the kernel that you are using:

```
oracleasm-support-version.arch.rpm
oracleasm-kernel-version.arch.rpm
oracleasmlib-version.arch.rpm
```

   Enter a command similar to the following to install the packages:

```
# rpm -ivh oracleasm-support-version.arch.rpm \
           oracleasm-kernel-version.arch.rpm \
           oracleasmlib-version.arch.rpm
```

   For example, if you are using the Red Hat Enterprise Linux 5 AS kernel on an AMD64 system, then enter a command similar to the following:

```
# rpm -ivh oracleasm-support-2.1.3-1.el5.x86_64.rpm \
           oracleasm-2.6.18-194.26.1.el5xen-2.0.5-1.el5.x86_64.rpm \
           oracleasmlib-2.0.4-1.el5.x86_64.rpm
```

5. Enter the following command to run the `oracleasm` initialization script with the `configure` option:

```
# /usr/sbin/oracleasm configure -i
```

> **Note:**
>
> The oracleasm command in `/usr/sbin` is the command you should use. The `/etc/init.d` path is not deprecated, but the `oracleasm` binary in that path is now used typically for internal commands.

6. Enter the following information in response to the prompts that the script displays:

Prompt	Suggested Response
Default user to own the driver interface:	**Standard groups and users configuration**: Specify the Oracle software owner user (for example, `oracle`) **Job role separation groups and users configuration:** Specify the Grid Infrastructure software owner (for example, `grid`)
Default group to own the driver interface:	**Standard groups and users configuration**: Specify the OSDBA group for the database (for example, `dba`). **Job role separation groups and users configuration:** Specify the OSASM group for storage administration (for example, `asmadmin`).
Start Oracle ASM Library driver on boot (y/n):	Enter `y` to start the Oracle Automatic Storage Management library driver when the system starts.
Scan for Oracle ASM disks on boot (y/n)	Enter `y` to scan for Oracle ASM disks when the system starts.

The script completes the following tasks:

- Creates the `/etc/sysconfig/oracleasm` configuration file
- Creates the `/dev/oracleasm` mount point
- Mounts the ASMLIB driver file system

> **Note:**
>
> The Oracle ASMLIB file system is not a regular file system. It is used only by the Oracle ASM library to communicate with the Oracle ASMLIB.

7. Enter the following command to load the `oracleasm` kernel module:

```
# /usr/sbin/oracleasm init
```

Configuring Disk Devices to Use Oracle ASMLIB

Configure disk devices to use in an Oracle Automatic Storage Management disk group.

To configure the disk devices to use in an Oracle Automatic Storage Management disk group, perform the following steps:

1. If you intend to use IDE, SCSI, or RAID devices in the Oracle Automatic Storage Management disk group, then perform the following steps:
   a. Install or configure the disk devices that you intend to use for the disk group and restart the system.
   b. Enter the following command to identify the device name for the disks to use:

```
# /sbin/fdisk -l
```

Depending on the type of disk, the device name can vary.

Table A-3 Device Name Formats Based on Disk Type

Disk Type	Device Name Format	Description
IDE disk	`/dev/hd`*xn*	In this example, *x* is a letter that identifies the IDE disk, and *n* is the partition number. For example, `/dev/hda` is the first disk on the first IDE bus.
SCSI disk	`/dev/sd`*xn*	In this example, *x* is a letter that identifies the IDE disk, and *n* is the partition number. For example, `/dev/hda` is the first disk on the first IDE bus.
RAID disk	`/dev/rd/c`*x*`d`*y*`p`*z* `/dev/ida/c`*x*`d`*y*`p`*z*	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, `/dev/ida/c0d1` is the second logical drive on the first controller.

To include devices in a disk group, you can specify either whole-drive device names or partition device names.

> **Note:**
>
> Oracle recommends that you create a single whole-disk partition on each disk to use.

c. Use either `fdisk` or `parted` to create a single whole-disk partition on the disk devices.

2. Enter a command similar to the following to mark a disk as an Oracle Automatic Storage Management disk:

```
# /usr/sbin/oracleasm createdisk DISK1 /dev/sdb1
```

In this example, `DISK1` is a name assigned to the disk.

> **Note:**
>
> - The disk names you specify can contain uppercase letters, numbers, and the underscore character. They must start with an uppercase letter.
> - To create a database during the installation using the Oracle Automatic Storage Management library driver, you must change the disk discovery string to `ORCL:*`.
> - If you are using a multi-pathing disk driver with Oracle ASM, then make sure that you specify the correct logical device name for the disk.

3. To make the disk available on the other nodes in the cluster, enter the following command as `root` on each node:

```
# /usr/sbin/oracleasm scandisks
```

This command identifies shared disks attached to the node that are marked as Oracle ASM disks.

Administering Oracle ASMLIB and Disks

Review this information to administer the Oracle Automatic Storage Management library driver and disks.

To administer the Oracle Automatic Storage Management library driver and disks, use the `/usr/sbin/oracleasm` initialization script with different options, as described in the following table:

Table A-4 Disk Management Tasks Using ORACLEASM

Task	Command Example	Description
Configure or reconfigure ASMLIB	`oracleasm configure -i`	Use the configure option to reconfigure the Oracle Automatic Storage Management library driver, if necessary. To see command options, enter `oracleasm configure` without the `-i` flag.
Change system restart load options for ASMLIB	`oracleasm configure -e`	Options are `-e` to enable and `-d` to disable. Use the `-e` and `-d` options to change the actions of the Oracle Automatic Storage Management library driver when the system starts. The `-e` option causes the Oracle Automatic Storage Management library driver to load when the system starts.
Load or unload ASMLIB without restarting the system	`oracleasm init`	Options are `init` to load and `exit` to unload. Use the `init` and `exit` options to load or unload the Oracle Automatic Storage Management library driver without restarting the system.

Table A-4 (Cont.) Disk Management Tasks Using ORACLEASM

Task	Command Example	Description
Mark a disk for use with ASMLIB	`oracleasm createdisk VOL1 /dev/sda1`	Use the `createdisk` option to mark a disk device for use with the Oracle Automatic Storage Management library driver and give it a name, where *`labelname`* is the name you want to use to mark the device, and *`devicepath`* is the path to the device: `oracleasm createdisk` *`labelname devicepath`*
Unmark a named disk device	`oracleasm deletedisk VOL1`	Use the `deletedisk` option to unmark a named disk device, where *`diskname`* is the name of the disk: `oracleasm deletedisk` *`diskname`* **Caution:** Do not use this command to unmark disks that are being used by an Oracle Automatic Storage Management disk group. You must delete the disk from the Oracle Automatic Storage Management disk group before you unmark it.
Determine if ASMLIB is using a disk device	`oracleasm querydisk`	Use the `querydisk` option to determine if a disk device or disk name is being used by the Oracle Automatic Storage Management library driver, where *`diskname_devicename`* is the name of the disk or device that you want to query: `oracleasm querydisk` *`diskname_devicename`*
List Oracle ASMLIB disks	`oracleasm listdisks`	Use the `listdisks` option to list the disk names of marked Oracle ASM library driver disks.
Identify disks marked as ASMLIB disks	`oracleasm scandisks`	Use the `scandisks` option to enable cluster nodes to identify which shared disks have been marked as ASMLIB disks on another node.
Rename ASMLIB disks	`oracleasm renamedisk VOL1 VOL2`	Use the `renamedisk` option to change the label of an Oracle ASM library driver disk or device by using the following syntax, where *`manager`* specifies the manager device, *`label_device`* specifies the disk you intend to rename, as specified either by OracleASM label name or by the device path, and *`new_label`* specifies the new label you want to use for the disk: `oracleasm renamedisk [-l` *`manager`*`] [-v]` *`label_device new_label`* Use the `-v` flag to provide a verbose output for debugging. **Caution:** You must ensure that all Oracle Database and Oracle ASM instances have ceased using the disk before you relabel the disk. If you do not do this, then you may lose data.

Deinstalling Oracle ASMLIB On Oracle Database

If Oracle ASM library driver (Oracle ASMLIB) is installed but you do not use it for device path persistence, then deinstall Oracle ASMLIB:

1. Stop Oracle ASM and any running database instance:

   ```
   $ srvctl stop database
   ```

   ```
   $ srvctl stop asm
   ```

2. Log in as `root`.
3. Stop the Oracle Restart stack:

   ```
   # cd Grid_home/bin
   ```

   ```
   # crsctl stop has
   ```

   Where, *Grid_home* is the Oracle Grid Infrastructure home directory path.
4. Stop Oracle ASMLIB:

   ```
   # /usr/sbin/oracleasm configure -d
   ```

5. Remove the `oracleasm` library and tools RPMs:

   ```
   # rpm -e oracleasm-support
   ```

   ```
   # rpm -e oracleasmlib
   ```

6. Check if any `oracleasm` RPMs are remaining:

   ```
   # rpm -qa| grep oracleasm
   ```

7. If any `oracleasm` configuration files are remaining, remove them:

   ```
   # rpm -qa| grep oracleasm | xargs rpm -e
   ```

   ```
   Oracle ASMLIB and associated RPMs are now removed.
   ```

Start the Oracle Restart stack. Optionally, you can install and configure Oracle ASM Filter Driver (Oracle ASMFD) before starting the Oracle Restart stack.

Configuring Storage Device Path Persistence Manually

You can maintain storage file path persistence by creating a rules file.

- Configuring Device Persistence Manually for Oracle ASM
  Complete these tasks to create device path persistence manually for Oracle ASM.

Configuring Device Persistence Manually for Oracle ASM

Complete these tasks to create device path persistence manually for Oracle ASM.

1. On existing systems (single instance or clustered), to obtain current mount paths, run the command `scsi_id` (`/sbin/scsi_id`) on storage devices on a server to obtain their unique device identifiers. When you run the command `scsi_id` with the `-s` argument, the device path and name you provide should be relative to the `sysfs` directory `/sys` (for example, `/block/device`) when referring to `/sys/block/device`.

   For example, as the `root` user:

   ```
   # /sbin/scsi_id -g -s /block/sdb/sdb1
   360a98000686f6959684a453333524174

   # /sbin/scsi_id -g -s /block/sde/sde1
   360a98000686f6959684a453333524179
   ```

   Record the unique SCSI identifiers, so you can provide them when required.

   > **Note:**
   >
   > The command `scsi_id` should return the same device identifier value for a given device, regardless of which node the command is run from.

2. Configure SCSI devices as trusted devices, by editing the `/etc/scsi_id.config` file and adding `options=-g` to the file.

   ```
   # cat > /etc/scsi_id.config
   vendor="ATA",options=-p 0x80
   options=-g
   ```

3. Using a text editor, create a UDEV rules file for the Oracle ASM devices, setting permissions to 0660 for the installation owner and the operating system group you have designated the OSASM group, whose members are administrators of the Oracle Grid Infrastructure software. For example, on Oracle Linux, to create a role-based configuration `rules.d` file where the installation owner is `grid` and the OSASM group `asmadmin`, enter commands similar to the following:

   ```
   # vi /etc/udev/rules.d/99-oracle-asmdevices.rules

   KERNEL=="sdb1, OWNER="grid", GROUP="asmadmin", MODE="0660",
   BUS=="scsi", PROGRAM=="/sbin/scsi_id",
   RESULT=="14f70656e66696c00000000"
   KERNEL=="sdc1", OWNER="grid", GROUP="asmadmin", MODE="0660",
   BUS=="scsi", PROGRAM=="/sbin/scsi_id",
   RESULT=="14f70656e66696c00000001"
   KERNEL=="sdd1", OWNER="grid", GROUP="asmadmin", MODE="0660",
   ```

```
BUS=="scsi", PROGRAM=="/sbin/scsi_id", RESULT=="14f70656e66696c00000002"
```

4. On clustered systems, copy the `rules.d` file to all other nodes on the cluster. For example:

```
# scp 99-oracle-asmdevices.rules root@node2:/etc/udev/rules.d/99-oracle-
asmdevices.rules
```

5. Run the `partprobe` command using the syntax `/sbin/partprobe` *`devicename`*.

   For example:

```
# /sbin/partprobe /dev/sdc1
# /sbin/partprobe /dev/sdd1
# /sbin/partprobe /dev/sde1
# /sbin/partprobe /dev/sdf1
```

6. Run the command `udevtest` (`/sbin/udevtest`) to test the UDEV rules configuration you have created. The output should indicate that the devices are available and the rules are applied as expected. For example, for `/dev/ssd1`:

```
# udevtest /block/sdd/sdd1
main: looking at device '/block/sdd/sdd1' from subsystem 'block'
udev_rules_get_name: add symlink
'disk/by-id/scsi-360a98000686f6959684a453333524174-part1'
udev_rules_get_name: add symlink
'disk/by-path/ip-192.168.1.1:3260-iscsi-iqn.1992-08.com.netapp:sn.887085-
part1'
udev_node_mknod: preserve file '/dev/.tmp-8-17', because it has correct
dev_t
run_program: '/lib/udev/vol_id --export /dev/.tmp-8-17'
run_program: '/lib/udev/vol_id' returned with status 4
run_program: '/sbin/scsi_id'
run_program: '/sbin/scsi_id' (stdout) '360a98000686f6959684a453333524174'
run_program: '/sbin/scsi_id' returned with status 0
udev_rules_get_name: rule applied, 'sdd1' becomes 'data1'
udev_device_event: device '/block/sdd/sdd1' validate currently present
symlinks
udev_node_add: creating device node '/dev/data1', major = '8', minor =
'17',
mode = '0640', uid = '0', gid = '500'
udev_node_add: creating symlink
'/dev/disk/by-id/scsi-360a98000686f6959684a453333524174-part1' to '../../
data1'
udev_node_add: creating symlink
'/dev/disk/by-path/ip-192.168.1.1:3260-iscsi-
iqn.1992-08.com.netapp:sn.84187085
-part1' to '../../data1'
main: run: 'socket:/org/kernel/udev/monitor'
main: run: '/lib/udev/udev_run_devd'
main: run: 'socket:/org/freedesktop/hal/udev_event'
main: run: '/sbin/pam_console_apply /dev/data1
/dev/disk/by-id/scsi-360a98000686f6959684a453333524174-part1
/dev/disk/by-path/ip-192.168.1.1:3260-iscsi-
```

```
iqn.1992-08.com.netapp:sn.84187085-
part1'
```

In the example output, note that applying the rules renames OCR device `/dev/sdd1` to `/dev/data1`.

7. Load the rules and restart the UDEV service. For example:
   - Oracle Linux and Red Hat Enterprise Linux

     ```
     # udevadm control --reload-rules
     ```

   - SUSE Linux Enterprise Server

     ```
     # /etc/init.d boot.udev restart
     ```

Verify that the device permissions and ownerships are set correctly.

B

Installing and Configuring Oracle Database Using Response Files

Review the following topics to install and configure Oracle products using response files.

- Modes for Oracle DBCA Noninteractive (Silent) Configuration
  You run the Oracle DBCA installer from a command prompt using either silent mode or response file mode.
- Reasons for Using Silent Mode or Response File Mode
  Review this section for use cases for running the installer in silent mode or response file mode.
- Using Response Files
  Use response files to install and configure Oracle products using the installer in silent or response file mode.
- Preparing Response Files
  Review this information to prepare response files for use during silent mode or response file mode installations.
- Running Oracle Universal Installer Using a Response File
  After creating the response file, run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation.
- Running Configuration Assistants Using Response Files
  You can run configuration assistants in response file or silent mode to configure and start Oracle software after it is installed on the system. To run configuration assistants in response file or silent mode, you must copy and edit a response file template.
- Postinstallation Configuration Using Response File Created During Installation
  Use response files to configure Oracle software after installation. You can use the same response file created during installation to also complete postinstallation configuration.

Modes for Oracle DBCA Noninteractive (Silent) Configuration

You run the Oracle DBCA installer from a command prompt using either silent mode or response file mode.

When you start Oracle Universal Installer (OUI), you can use a response file to automate the installation and configuration of Oracle software, either fully or partially. OUI uses the values contained in the response file to provide answers to some or all installation prompts.

Typically, the installer runs in interactive mode, which means that it prompts you to provide information in graphical user interface (GUI) screens. When you use response files to provide this information, you run the installer from a command prompt using either of the following modes:

- Silent mode

  If you include responses for all of the prompts in the response file and specify the `-silent` option when starting the installer, then it runs in silent mode. During a silent

mode installation, the installer does not display any screens. Instead, it displays progress information in the terminal that you used to start it.

- Response file mode

  If you include responses for some or all of the prompts in the response file and omit the `-silent` option, then the installer runs in response file mode. During a response file mode installation, the installer displays all the screens, screens for which you specify information in the response file, and also screens for which you did not specify the required information in the response file.

You define the settings for a silent or response file installation by entering values for the variables listed in the response file. For example, to specify the Oracle home name, provide the Oracle home path for the `ORACLE_HOME` environment variable:

```
ORACLE_HOME=/u01/app/oracle/product/21.0.0/dbhome_1
```

Reasons for Using Silent Mode or Response File Mode

Review this section for use cases for running the installer in silent mode or response file mode.

Mode	Uses
Silent	Use silent mode for the following installations: • Complete an unattended installation, which you schedule using operating system utilities such as `at`. • Complete several similar installations on multiple systems without user interaction. • Install the software on a system that does not have X Window System software installed on it. The installer displays progress information on the terminal that you used to start it, but it does not display any of the installer screens.
Response file	Use response file mode to complete similar Oracle software installations on more than one system, providing default answers to some, but not all of the installer prompts.

Using Response Files

Use response files to install and configure Oracle products using the installer in silent or response file mode.

> **Note:**
>
> You must complete all required preinstallation tasks on a system before running the installer in silent or response file mode.

1. Prepare a response file.
2. Run the installer in silent or response file mode.
3. Run the root scripts as prompted by Oracle Universal Installer.

4. If you completed a software-only installation, then run Net Configuration Assistant and Oracle DBCA in silent or response file mode to create the database listener and an Oracle Database instance respectively.

Preparing Response Files

Review this information to prepare response files for use during silent mode or response file mode installations.

- Editing a Response File Template
  Oracle provides response file templates for each product and each configuration tool.
- Recording Response Files
  You can use OUI in interactive mode to record response files, which you can then edit and use to complete silent mode or response file mode installations.

Editing a Response File Template

Oracle provides response file templates for each product and each configuration tool.

About Response File Templates

For Oracle Database, the response file templates are located in the `$ORACLE_HOME/install/response` directory. For Oracle Grid Infrastructure, the response file templates are located in the *`Grid_home`*`/install/response` directory.

Where, *Grid_home* is the Oracle Grid Infrastructure home directory path.

> **Note:**
>
> If you copied the software to a hard disk, then the response files are located in the `$ORACLE_HOME/install/response` directory.

All response file templates contain comment entries, sample formats, examples, and other useful instructions. Read the response file instructions to understand how to specify values for the response file variables, so that you can customize your installation.

The following table lists the response files provided with this software:

Table B-1 Response Files for Oracle Database and Oracle Grid Infrastructure

Response File	Description
`db_install.rsp`	Silent installation of Oracle Database.
`dbca.rsp`	Silent creation and configuration of Oracle Database using Oracle DBCA.
`netca.rsp`	Silent configuration of Oracle Net using Oracle NETCA.
`gridsetup.rsp`	Silent configuration of Oracle Grid Infrastructure installations.

> **Caution:**
>
> When you modify a response file template and save a file for use, the response file may contain plain text passwords. Ownership of the response file should be given to the Oracle software installation owner only, and permissions on the response file should be changed to 600. Oracle strongly recommends that database administrators or other administrators delete or secure response files when they are not in use.

To copy and modify a response file:

1. Copy the response file from the response file directory to a directory on your system:

   For example, for Oracle Database:

   ```
   $ cp $ORACLE_HOME/install/response/db_install.rsp local_directory
   ```

2. Open the response file in a text editor:

   ```
   $ vi /local_directory/db_install.rsp
   ```

3. Follow the instructions in the file to edit it.

   > **Note:**
   >
   > The installer or configuration assistant fails if you do not correctly configure the response file. Also, ensure that your response file name has the `.rsp` suffix.

4. Secure the response file by changing the permissions on the file to 600:

   ```
   $ chmod 600 /local_dir/db_install.rsp
   ```

   Ensure that only the Oracle software owner user can view or modify response files or consider deleting them after the installation succeeds.

   > **Note:**
   >
   > A fully-specified response file for an Oracle Database installation contains the passwords for database administrative accounts and for a user who is a member of the OSDBA group (required for automated backups).

Recording Response Files

You can use OUI in interactive mode to record response files, which you can then edit and use to complete silent mode or response file mode installations.

Recording response files is useful for Advanced or software-only installations. You can save all the installation steps into a response file during installation by clicking **Save**

Response File on the **Summary** page. You can use the generated response file for a silent installation later. When you record the response file, you can either complete the installation, or you can exit from the installer on the Summary page, before OUI starts to set up the software to the system.

If you use record mode during a response file mode installation, then the installer records the variable values that were specified in the original source response file into the new response file.

> **Note:**
>
> OUI does not save passwords while recording the response file.

1. Complete preinstallation tasks as for a standard installation.

   When you run the installer to record a response file, it checks the system to verify that it meets the requirements to install the software. For this reason, Oracle recommends that you complete all of the required preinstallation tasks and record the response file while completing an installation.
2. Ensure that the Oracle software owner user (typically `oracle`) has permissions to create or write to the Oracle home path that you specify when you run the installer.
3. On each installation screen, specify the required information.
4. When the installer displays the **Summary** screen, perform the following steps:
   a. Click **Save Response File**. In the window, specify a file name and location for the new response file. Click **Save** to write the responses you entered to the response file.
   b. Click **Finish** to continue with the installation. Click **Cancel** if you do not want to continue with the installation. The installation stops, but the recorded response file is retained.

   > **Note:**
   >
   > Ensure that your response file name has the `.rsp` suffix.
5. Before you use the saved response file on another system, edit the file and make any required changes. Use the instructions in the file as a guide when editing it.

Running Oracle Universal Installer Using a Response File

After creating the response file, run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation.

Run Oracle Universal Installer at the command line, specifying the response file you created. The Oracle Universal Installer executables, `runInstaller` and `gridSetup.sh`, provide several options. For help information on the full set of these options, run the `gridSetup.sh` or `runInstaller` command with the `-help` option. For example:

- For Oracle Database:

```
$ $ORACLE_HOME/runInstaller -help
```

- For Oracle Grid Infrastructure:

```
$ /u01/app/oracle/product/21.0.0/grid/gridSetup.sh -help
```

The help information appears in a window after some time.

Perform the following steps to run the installer using a response file:

1. Complete the preinstallation tasks for a normal installation.
2. Log in as the software installation owner user.
3. If you are completing a response file mode installation, then set the operating system `DISPLAY` environment variable for the user running the installation.

> **Note:**
>
> You do not have to set the `DISPLAY` environment variable if you are completing a silent mode installation.

4. To start the installer in silent or response file mode, enter a command similar to the following:
   - For Oracle Database:

     ```
     $ $ORACLE_HOME/runInstaller [-silent] -responseFile
     responsefilename
     ```

   - For Oracle Grid Infrastructure:

     ```
     $ /u01/app/oracle/product/21.0.0/grid/gridSetup.sh [-silent] -
     responseFile responsefilename
     ```

> **Note:**
>
> Do not specify a relative path to the response file. If you specify a relative path, then the installer fails.

   In this example:

   - `-silent` runs the installer in silent mode.
   - *responsefilename* is the full path and file name of the installation response file that you configured.

5. If this is the first time you are installing Oracle software on your system, then Oracle Universal Installer prompts you to run the `orainstRoot.sh` script. Log in as the `root` user and run the `orainstRoot.sh` script:

```
$ su root
password:
# /u01/app/oraInventory/orainstRoot.sh
```

> **Note:**
>
> You do not have to manually create the `oraInst.loc` file. Running the `orainstRoot.sh` script is sufficient as it specifies the location of the Oracle Inventory directory.

6. When the installation completes, log in as the `root` user and run the `root.sh` script. For example:

```
$ su root
password:
# $ORACLE_HOME/root.sh
```

Running Configuration Assistants Using Response Files

You can run configuration assistants in response file or silent mode to configure and start Oracle software after it is installed on the system. To run configuration assistants in response file or silent mode, you must copy and edit a response file template.

> **Note:**
>
> If you copied the software to a hard disk, then the response file template is located in the `$ORACLE_HOME/install/response` directory.

- Running Net Configuration Assistant Using Response Files
  You can run NETCA in silent mode to configure and start an Oracle Net Listener on the system, and configure naming methods and Oracle Net service names.
- Running Oracle DBCA Using Response Files
  You can run Oracle DBCA in response file mode to configure and start an Oracle database on the system.

Running Net Configuration Assistant Using Response Files

You can run NETCA in silent mode to configure and start an Oracle Net Listener on the system, and configure naming methods and Oracle Net service names.

To run Oracle Net Configuration Assistant (NETCA) in silent mode, you must copy and edit a response file template. Oracle provides a response file template named `netca.rsp` in the `$ORACLE_HOME/assistants/netca` directory.

To run Net Configuration Assistant using a response file:

1. Copy the `netca.rsp` response file template from the response file directory to a directory on your system:

```
$ cp $ORACLE_HOME/assistants/netca/netca.rsp local_directory
```

2. Open the response file in a text editor:

   ```
   $ vi /local_directory/netca.rsp
   ```

3. Follow the instructions in the file to edit it.

   > **Note:**
   >
   > Net Configuration Assistant fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.
5. Enter a command similar to the following to run Net Configuration Assistant in silent mode:

   ```
   $ $ORACLE_HOME/bin/netca /silent /responsefile /local_directory/
   netca.rsp
   ```

   In this command:

   - The `/silent` option indicates to run Net Configuration Assistant in silent mode.
   - *local_directory* is the full path of the directory where you copied the `netca.rsp` response file template.

Running Oracle DBCA Using Response Files

You can run Oracle DBCA in response file mode to configure and start an Oracle database on the system.

To run Oracle Database Configuration Assistant (Oracle DBCA) in response file mode, you must copy and edit a response file template. Oracle provides a response file template named `dbca.rsp` in the `ORACLE_HOME/assistants/dbca` directory. To run Oracle DBCA in response file mode, you must use the `-responseFile` flag in combination with the `-silent` flag. You must also use a graphical display and set the `DISPLAY` environment variable.

To run Oracle DBCA in response file mode:

1. Copy the `dbca.rsp` response file template from the response file directory to a directory on your system:

   ```
   $ cp $ORACLE_HOME/assistants/dbca/dbca.rsp /local_directory
   ```

   In this example, *directory_path* is the path of the directory where you have copied the installation binaries.

   As an alternative to editing the response file template, you can also create a database by specifying all required information as command line options when you

run Oracle DBCA. For information about the list of options supported, enter the following command:

```
$ $ORACLE_HOME/bin/dbca -help
```

2. Open the response file in a text editor:

```
$ vi /local_dir/dbca.rsp
```

3. Follow the instructions in the file to edit the file.

> **Note:**
>
> Oracle DBCA fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.
5. To run Oracle DBCA in response file mode, set the `DISPLAY` environment variable.
6. Use the following command syntax to run Oracle DBCA in silent or response file mode using a response file:

```
$ORACLE_HOME/bin/dbca [-silent] -createDatabase -responseFile /local_dir/
dbca.rsp
```

In this example:

- `-silent` option indicates that Oracle DBCA runs in silent mode.
- *`local_dir`* is the full path of the directory where you copied the `dbca.rsp` response file template.

During configuration, Oracle DBCA displays a window that contains the status messages and a progress bar.

Postinstallation Configuration Using Response File Created During Installation

Use response files to configure Oracle software after installation. You can use the same response file created during installation to also complete postinstallation configuration.

- Using the Installation Response File for Postinstallation Configuration
  You can use the response file created during installation to also complete postinstallation configuration.
- Running Postinstallation Configuration Using Response File
  You can use a response file to complete postinstallation tasks on one or more servers simultaneously.

Using the Installation Response File for Postinstallation Configuration

You can use the response file created during installation to also complete postinstallation configuration.

Run the installer with the `-executeConfigTools` option to configure configuration assistants after installing Oracle Grid Infrastructure or Oracle Database. You can use the response file located at `$ORACLE_HOME/install/response/` *product_timestamp.rsp* to obtain the passwords required to run the configuration tools. You must update the response file with the required passwords before running the `-executeConfigTools` command.

Oracle strongly recommends that you maintain security with a password response file:

- Permissions on the response file should be set to 600.
- The owner of the response file should be the installation owner user, with the group set to the central inventory (`oraInventory`) group.

Example B-1 Response File Passwords for Oracle Grid Infrastructure (`grid` user)

```
grid.install.crs.config.ipmi.bmcPassword=password
grid.install.asm.SYSASMPassword=password
grid.install.asm.monitorPassword=password
grid.install.config.emAdminPassword=password
```

If you do not have a BMC card, or you do not want to enable IPMI, then leave the `ipmi.bmcPassword` input field blank.

If you do not want to enable Oracle Enterprise Manager for management, then leave the `emAdminPassword` password field blank.

Example B-2 Response File Passwords for Oracle Grid Infrastructure for a Standalone Server (`oracle` user)

```
oracle.install.asm.SYSASMPassword=password
oracle.install.asm.monitorPassword=password
oracle.install.config.emAdminPassword=password
```

If you do not want to enable Oracle Enterprise Manager for management, then leave the `emAdminPassword` password field blank.

Example B-3 Response File Passwords for Oracle Database (`oracle` user)

This example illustrates the passwords to specify for use with the database configuration assistants.

```
oracle.install.db.config.starterdb.password.SYS=password
oracle.install.db.config.starterdb.password.SYSTEM=password
oracle.install.db.config.starterdb.password.DBSNMP=password
oracle.install.db.config.starterdb.password.PDBADMIN=password
oracle.install.db.config.starterdb.emAdminPassword=password
oracle.install.db.config.asm.ASMSNMPPassword=password
```

You can also specify `oracle.install.db.config.starterdb.password.ALL=`*password* to use the same password for all database users.

The database configuration assistants require the SYS, SYSTEM, PDBADMIN, and DBSNMP passwords for use with Oracle DBCA. You must specify the following passwords, depending on your system configuration:

- If the database uses Oracle Automatic Storage Management (Oracle ASM) for storage, then you must specify a password for the `ASMSNMPPassword` variable. If you are not using Oracle ASM, then leave the value for this password variable blank.

Running Postinstallation Configuration Using Response File

You can use a response file to complete postinstallation tasks on one or more servers simultaneously.

Complete this procedure to run configuration assistants with the `executeConfigTools` command and a response file.

1. Edit the response file and specify the required passwords for your configuration. You can use the response file created during installation, located at `$ORACLE_HOME/install/response/`*`product_timestamp.rsp`*.
2. Change directory to the Oracle home containing the installation software. For example:

   For Oracle Grid Infrastructure:

   ```
   cd Grid_home
   ```

   Where, *Grid_home* is the path to the Oracle Grid Infrastructure home directory `/u01/app/oracle/product/21.0.0/grid`

   For Oracle Database:

   ```
   cd $ORACLE_HOME
   ```

3. Run the configuration script using the following syntax:

   For Oracle Grid Infrastructure:

   ```
   $ ./gridSetup.sh -executeConfigTools -responseFile Grid_home/install/
   response/product_timestamp.rsp
   ```

   For Oracle Database:

   ```
   $ ./runInstaller -executeConfigTools -responseFile $ORACLE_HOME/install/
   response/product_timestamp.rsp
   ```

   For Oracle Database, you can also run the response file located in the directory `$ORACLE_HOME/install/response/`:

   ```
   $ ./runInstaller -executeConfigTools -responseFile $ORACLE_HOME/install/
   response/db_install.rsp
   ```

   The postinstallation configuration tool runs the installer in the graphical user interface mode, displaying the progress of the postinstallation configuration. Specify the `[-silent]` option to run the postinstallation configuration in the silent mode.

For example, for Oracle Grid Infrastructure:

```
$ ./gridSetup.sh -executeConfigTools -responseFile /u01/app/oracle/
product/21.0.0/grid/install/response/grid_2019-01-09_01-03-36PM.rsp
[-silent]
```

For Oracle Database:

```
$ ./runInstaller -executeConfigTools -responseFile /u01/app/oracle/
product/21.0.0/dbhome_1/response/db_2019-01-09_01-03-36PM.rsp [-
silent]
```

C
Optimal Flexible Architecture

Oracle Optimal Flexible Architecture (OFA) rules are a set of configuration guidelines created to ensure well-organized Oracle installations, which simplifies administration, support and maintenance.

- About the Optimal Flexible Architecture Standard
  Oracle Optimal Flexible Architecture (OFA) rules help you to organize database software and configure databases to allow multiple databases, of different versions, owned by different users to coexist.
- About Multiple Oracle Homes Support
  Oracle Database supports multiple Oracle homes. You can install this release or earlier releases of the software more than once on the same system, in different Oracle home directories.
- About the Oracle Inventory Directory and Installation
  The directory that you designate as the Oracle Inventory directory (oraInventory) stores an inventory of all software installed on the system.
- Oracle Base Directory Naming Convention
  The Oracle Base directory is the database home directory for Oracle Database installation owners, and the log file location for Oracle Grid Infrastructure owners.
- Oracle Home Directory Naming Convention
  By default, Oracle Universal Installer configures Oracle home directories using these Oracle Optimal Flexible Architecture conventions.
- Optimal Flexible Architecture File Path Examples
  Review examples of hierarchical file mappings of an Optimal Flexible Architecture-compliant installation.

About the Optimal Flexible Architecture Standard

Oracle Optimal Flexible Architecture (OFA) rules help you to organize database software and configure databases to allow multiple databases, of different versions, owned by different users to coexist.

In earlier Oracle Database releases, the OFA rules provided optimal system performance by isolating fragmentation and minimizing contention. In current releases, OFA rules provide consistency in database management and support, and simplifies expanding or adding databases, or adding additional hardware.

By default, Oracle Universal Installer places Oracle Database components in directory locations and with permissions in compliance with OFA rules. Oracle recommends that you configure all Oracle components in accordance with OFA guidelines.

Oracle recommends that you accept the OFA default. Following OFA rules is especially of value if the database is large, or if you plan to have multiple databases.

> **Note:**
>
> OFA assists in identification of an ORACLE_BASE with its Automatic Diagnostic Repository (ADR) diagnostic data to properly collect incidents.

About Multiple Oracle Homes Support

Oracle Database supports multiple Oracle homes. You can install this release or earlier releases of the software more than once on the same system, in different Oracle home directories.

Careful selection of mount point names can make Oracle software easier to administer. Configuring multiple Oracle homes in compliance with Optimal Flexible Architecture (OFA) rules provides the following advantages:

- You can install this release, or earlier releases of the software, more than once on the same system, in different Oracle home directories. However, you cannot install products from one release of Oracle Database into an Oracle home directory of a different release.
- Multiple databases, of different versions, owned by different users can coexist concurrently.
- To install Oracle Database software in multiple Oracle homes, you must extract the image file in each Oracle home, and then run the setup wizard from the respective Oracle home.
- You must install a new Oracle Database release in a new Oracle home that is separate from earlier releases of Oracle Database.

  You cannot install multiple releases in one Oracle home. Oracle recommends that you create a separate Oracle Database Oracle home for each release, in accordance with the Optimal Flexible Architecture (OFA) guidelines.
- In production, the Oracle Database server software release is the release number in the format of major and RU release number. For example, with the release number 19.3.0.0.0, the major release is 19 and the RU release number is 3.
- Later Oracle Database releases can access earlier Oracle Database releases. However, this access is only for upgrades. For example, Oracle Database 19c can access an Oracle Database 18c if the 18c database is started up in upgrade mode.
- Oracle Database Client can be installed in the same Oracle Database home if both products are at the same release level. For example, you can install Oracle Database Client 19c into an existing Oracle Database 19c home but you cannot install Oracle Database Client 19c into an existing Oracle Database 18c home. If you apply a patch set or release update before installing the client, then you must apply the patch set or release update again.
- Structured organization of directories and files, and consistent naming for database files simplify database administration.
- Login home directories are not at risk when database administrators add, move, or delete Oracle home directories.
- You can test software upgrades in an Oracle home in a separate directory from the Oracle home where your production database is located.

- For information about release support timelines, refer to My Oracle Support Doc ID 742060.1

Related Topics

- My Oracle Support Note 742060.1

About the Oracle Inventory Directory and Installation

The directory that you designate as the Oracle Inventory directory (oraInventory) stores an inventory of all software installed on the system.

All Oracle software installation owners on a server are granted the OINSTALL privileges to read and write to this directory. If you have previous Oracle software installations on a server, then additional Oracle software installations detect this directory from the `/etc/oraInst.loc` file, and continue to use that Oracle Inventory. Ensure that the group designated as the OINSTALL group is available as a primary group for all planned Oracle software installation owners.

If you are installing Oracle software for the first time, then OUI creates an Oracle base and central inventory, and creates an Oracle inventory using information in the following priority:

- In the path indicated in the `ORACLE_BASE` environment variable set for the installation owner user account
- In an Optimal Flexible Architecture (OFA) path (u[01–99]`/app/`*owner* where *owner* is the name of the user account running the installation), and that user account has permissions to write to that path
- In the user home directory, in the path `/app/`*owner*, where *owner* is the name of the user account running the installation

For example:

If you are performing an Oracle Database installation, and you set `ORACLE_BASE` for user `oracle` to the path `/u01/app/oracle` before installation, and grant 755 permissions to `oracle` for that path, then Oracle Universal Installer creates the Oracle Inventory directory one level above the ORACLE_BASE in the path `ORACLE_BASE/../oraInventory`, so the Oracle Inventory path is `/u01/app/oraInventory`. Oracle Universal Installer installs the software in the `ORACLE_BASE` path. If you are performing an Oracle Grid Infrastructure for a Cluster installation, then the Grid installation path is changed to `root` ownership after installation, and the Grid home software location should be in a different path from the Grid user Oracle base.

If you create the OFA path `/u01`, and grant `oracle` 755 permissions to write to that path, then the Oracle Inventory directory is created in the path `/u01/app/oraInventory`, and Oracle Universal Installer creates the path `/u01/app/oracle`, and configures the `ORACLE_BASE` environment variable for the Oracle user to that path. If you are performing an Oracle Database installation, then the Oracle home is installed under the Oracle base. However, if you are installing Oracle Grid Infrastructure for a cluster, then be aware that ownership of the path for the Grid home is changed to `root` after installation and the Grid base and Grid home should be in different locations, such as `/u01/app/21.0.0/grid` for the Grid home path, and `/u01/app/grid` for the Grid base. For example:

`/u01/app/oraInventory`, owned by `grid:oinstall`
`/u01/app/oracle`, owned by `oracle:oinstall`
`/u01/app/oracle/product/21.0.0/dbhome_1/`, owned by `oracle:oinistall`

`/u01/app/grid`, owned by `grid:oinstall`
`/u01/app/21.0.0/grid`, owned by `root`

If you have neither set ORACLE_BASE, nor created an OFA-compliant path, then the Oracle Inventory directory is placed in the home directory of the user that is performing the installation, and the Oracle software is installed in the path `/app/`*owner*, where *owner* is the Oracle software installation owner. For example:

```
/home/oracle/oraInventory
/home/oracle/app/oracle/product/21.0.0/dbhome_1
```

Oracle Base Directory Naming Convention

The Oracle Base directory is the database home directory for Oracle Database installation owners, and the log file location for Oracle Grid Infrastructure owners.

Name Oracle base directories using the syntax */pm/h/u*, where *pm* is a string mount point name, *h* is selected from a small set of standard directory names, and *u* is the name of the owner of the directory.

You can use the same Oracle base directory for multiple installations. If different operating system users install Oracle software on the same system, then you must create a separate Oracle base directory for each installation owner. For ease of administration, Oracle recommends that you create a unique owner for each Oracle software installation owner, to separate log files.

Because all Oracle installation owners write to the central Oracle inventory file, and that file mountpoint is in the same mount point path as the initial Oracle installation, Oracle recommends that you use the same */pm/h* path for all Oracle installation owners.

Table C-1 Examples of OFA-Compliant Oracle Base Directory Names

Example	Description
`/u01/app/oracle`	Oracle Database Oracle base, where the Oracle Database software installation owner name is `oracle`. The Oracle Database binary home is located underneath the Oracle base path.
`/u01/app/grid`	Oracle Grid Infrastructure Oracle base, where the Oracle Grid Infrastructure software installation owner name is `grid`.

Caution:

The Oracle Grid Infrastructure Oracle base should not contain the Oracle Grid Infrastructure binaries for an Oracle Grid Infrastructure for a cluster installation. Permissions for the file path to the Oracle Grid Infrastructure binary home are changed to `root` during installation.

> **Note:**
>
> Oracle home or Oracle base cannot be `symlinks`, nor can any of their parent directories, all the way to up to the `root` directory.

Oracle Home Directory Naming Convention

By default, Oracle Universal Installer configures Oracle home directories using these Oracle Optimal Flexible Architecture conventions.

The directory pattern syntax for Oracle homes is /*pm*/*s*/*u*/product/*v*/*type*_[*n*]. The following table describes the variables used in this syntax:

Variable	Description
pm	A mount point name.
s	A standard directory name.
u	The name of the owner of the directory.
v	The version of the software.
type	The type of installation. For example: Database (`dbhome`), Client (`client`), or Oracle Grid Infrastructure (`grid`)
n	An optional counter, which enables you to install the same product more than once in the same Oracle base directory. For example: Database 1 and Database 2 (`dbhome_1`, `dbhome_2`)

For example, the following path is typical for the first installation of Oracle Database on this system:

```
/u01/app/oracle/product/21.0.0/dbhome_1
```

> **Note:**
>
> Oracle home or Oracle base cannot be `symlinks`, nor can any of their parent directories, all the way to up to the `root` directory.

Optimal Flexible Architecture File Path Examples

Review examples of hierarchical file mappings of an Optimal Flexible Architecture-compliant installation.

```
/u02/u03 /u04
```

Note:

- The Grid homes are examples of Grid homes used for an Oracle Grid Infrastructure for a standalone server deployment (Oracle Restart), or a Grid home used for an Oracle Grid Infrastructure for a cluster deployment (Oracle Clusterware). You can have either an Oracle Restart deployment, or an Oracle Clusterware deployment. You cannot have both options deployed at the same time.
- Oracle Automatic Storage Management (Oracle ASM) is included as part of an Oracle Grid Infrastructure installation. Oracle recommends that you use Oracle ASM to provide greater redundancy and throughput.

Table C-2 Optimal Flexible Architecture Hierarchical File Path Examples

Directory	Description
`/`	Root directory
`/u01/`	User data mount point 1
`/u01/app/`	Subtree for application software
`/u01/app/oraInventory`	Central OraInventory directory, which maintains information about Oracle installations on a server. Members of the group designated as the OINSTALL group have permissions to write to the central inventory. All Oracle software installation owners must have the OINSTALL group as their primary group, and be able to write to this group.
`/u01/app/oracle/`	Oracle base directory for user `oracle`. There can be many Oracle Database installations on a server, and many Oracle Database software installation owners. Oracle software homes that an Oracle installation owner owns should be located in the Oracle base directory for the Oracle software installation owner, unless that Oracle software is Oracle Grid Infrastructure deployed for a cluster.
`/u01/app/grid`	Oracle base directory for user `grid`. The Oracle home (Grid home) for Oracle Grid Infrastructure for a cluster installation is located outside of the Grid user. There can be only one Grid home on a server, and only one Grid software installation owner. The Grid home contains log files and other administrative files.
`/u01/app/oracle/admin/`	Subtree for database administration files

Table C-2 (Cont.) Optimal Flexible Architecture Hierarchical File Path Examples

Directory	Description
`/u01/app/oracle/admin/TAR`	Subtree for support log files
`/u01/app/oracle/admin/db_sales/`	Admin subtree for database named “sales”
`/u01/app/oracle/admin/db_dwh/`	Admin subtree for database named “dwh”
`/u01/app/oracle/fast_recovery_area/`	Subtree for recovery files
`/u01/app/oracle/fast_recovery_area/db_sales`	Recovery files for database named “sales”
`/u01/app/oracle/fast_recovery_area/db_dwh`	Recovery files for database named “dwh”

`/u02/app/oracle/oradata`
`/u03/app/oracle/oradata`
`/u04/app/oracle/oradata`
	Oracle data file directories

`/u01/app/oracle/product/`	Common path for Oracle software products other than Oracle Grid Infrastructure for a cluster
`/u01/app/oracle/product/21.0.0/dbhome_1`	Oracle home directory for Oracle Database 1, owned by Oracle Database installation owner account `oracle`

Table C-2 (Cont.) Optimal Flexible Architecture Hierarchical File Path Examples

Directory	Description
`/u01/app/oracle/product/21.0.0/dbhome_2`	Oracle home directory for Oracle Database 2, owned by Oracle Database installation owner account `oracle`
`/u01/app/oracle2/product/21.0.0/dbhome_2`	Oracle home directory for Oracle Database 2, owned by Oracle Database installation owner account `oracle2`
`/u01/app/oracle/product/21.0.0/grid`	Oracle home directory for Oracle Grid Infrastructure for a standalone server, owned by Oracle Database and Oracle Grid Infrastructure installation owner `oracle`.
`/u01/app/21.0.0/grid`	Oracle home directory for Oracle Grid Infrastructure for a cluster (Grid home), owned by user `grid` before installation, and owned by `root` after installation.
`/u01/app/oracle/product/21.0.0/client_1`	Oracle home directory for Oracle Database Client 1, owned by Oracle Database installation owner account `oracle`

D

Managing Oracle Database Port Numbers

Review default port numbers. If needed, use these steps to change assigned ports after installation.

- About Managing Ports
  During installation, Oracle Universal Installer assigns port numbers to components from a set of default port numbers.
- Oracle Database Component Port Numbers and Protocols
  This table lists the port numbers and protocols configured for Oracle Database components during a single-instance installation. By default, the first port in the range is assigned to the component, if it is available.

About Managing Ports

During installation, Oracle Universal Installer assigns port numbers to components from a set of default port numbers.

Many Oracle Database components and services use ports. As an administrator, it is important to know the port numbers used by these services, and to ensure that the same port number is not used by two services on your host. Enter the following command to identify the ports currently used on your computer:

```
$/bin/netstat -a
```

Most port numbers are assigned during installation. Every component and service has an allotted port range, which is the set of port numbers Oracle Database attempts to use when assigning a port. Oracle Database starts with the lowest number in the range and performs the following checks:

- Is the port used by another Oracle Database installation on the host?

  The installation may be up or down at the time. Oracle Database can still detect if the port is used.
- Is the port used by a process that is currently running?

  This can be any process on the host, even a non-Oracle Database process.
- Is the port listed in the `/etc/services` file?

If the answer to any of the preceding questions is yes, then Oracle Database moves to the next highest port in the allotted port range, and continues checking until it finds a free port.

Oracle Database Component Port Numbers and Protocols

This table lists the port numbers and protocols configured for Oracle Database components during a single-instance installation. By default, the first port in the range is assigned to the component, if it is available.

Table D-1 Protocols and Default Port Numbers for Oracle Database Components

Component	Description	Default Port Number	Port Range	Protocol
Oracle Net Services Listener	Enables Oracle client connections to the database over the Oracle Net Services protocol. You can configure it during installation. To reconfigure this port, use Net Configuration Assistant.	1521	Port number changes to the next available port. Modifiable manually to any available port.	TCP
Oracle Connection Manager	Listening port for Oracle client connections to Oracle Connection Manager. It is not configured during installation, but can be configured manually by editing the `cman.ora` parameter file. This file is located under the `/network/admin` directory.	1630	1630	TCP
Oracle XML DB	The Oracle XML DB HTTP port is used if web-based applications must access an Oracle database from an HTTP listener. You must configure this port manually.	0	Configured Manually	HTTP
Oracle XML DB	The Oracle XML DB FTP is used when applications must access an Oracle database from an FTP listener. You must configure this port manually.	0	Configured Manually	FTP

Related Topics

- Using HTTP(S) on a Standard Port
- Using FTP on the Standard Port
- *Oracle Real Application Clusters Installation Guide for Linux and UNIX*