

Oracle® Database

Database Client Installation Guide



19c for Linux
E96433-27
August 2026

ORACLE®

Contributors: Jean-Francois Verrier, Neha Avasthy, Kalambhatti Prashanth, Prasad Bagal, Subhranshu Banerjee, Tammy Bednar, John Arackal, Ilia Okomin, Sharad Raju, Gavin Bowe, Ranjit Noronha, Kalambhatti Prashanth, Bala Appulingam, Subrahmanyam Kodavaluru, Robert Chang, Darcy Christensen, Kiran Chamala, Jonathan Creighton, Benoit Dageville, Sudip Datta, Jim Erickson, Marcus Fallen, Joseph Francis, Richard Roddy, Allan Graves, Barbara Glover, Aneesh Khandelwal, Joel Kallman, Eugene Karichkin, Jai Krishnani, Sangeeth Kumar, Kevin Jernigan, Christopher Jones, Prasad Kuruvadi Nagaraj, Bryn Llewellyn, Saar Maoz, Sunil Surabhi, Gopal Mulagund, Sue Lee, Rich Long, Barb Lundhild, Rudregowda Mallegowda, Padmanabhan Manavazhi, Mughees Minhas, Krishna Mohan, Matthew McKerley, John McHugh, Gurudas Pai, Satish Panchumarthy, Rajesh Prasad, Rajendra Pingte, Sivaselvam Narayanasamy, Srinivas Poovala, Mohammed Shah Nawaz Quadri, Sangeeth Jose, Gurumurthy Ramamurthy, Hema Ramamurthy, Sunil Ravindrachar, Mark Richwine, Dipak Saggi, Trivikrama Samudrala, Pushkar Punit, David Schreiner, Ara Shakian, Mohit Singhal, Dharma Srinapalli, Akshay Shah, James Spiller, Roy Swonger, Binoy Sukumaran, Kamal Tbeileh, Ravi Thammaiah, Shekhar Vaggu, Peter Wahl, Terri Winters, 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, software documentation, data (as defined in the Federal Acquisition Regulation), 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," "commercial computer software documentation," or "limited rights data" 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®, Java, MySQL, and NetSuite 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.

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>.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

Preface

Audience	i
Documentation Accessibility	i
Diversity and Inclusion	ii
Set Up Java Access Bridge to Implement Java Accessibility	ii
Command Syntax	ii
Conventions	iii

1 Oracle Database Client Installation Checklist

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

2 Checking and Configuring Server Hardware for Oracle Database Client

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

3 Configuring Operating Systems for Oracle Database Client on Linux

About Oracle Linux with the Unbreakable Enterprise Kernel	1
Reviewing Operating System Security Common Practices	2
About Operating System Requirements	2
Using Oracle RPM Checker on IBM: Linux on System z	2
Operating System Requirements for x86-64 Linux Platforms	4
Supported Oracle Linux 9 Distributions for x86-64	5
Supported Oracle Linux 8 Distributions for x86-64	7
Supported Oracle Linux 7 Distributions for x86-64	9
Supported Red Hat Enterprise Linux 9 Distributions for x86-64	11
Supported Red Hat Enterprise Linux 8 Distributions for x86-64	12

Supported Red Hat Enterprise Linux 7 Distributions for x86-64	14
Supported SUSE Linux Enterprise Server 12 Distributions for x86-64	15
Supported SUSE Linux Enterprise Server 15 Distributions for x86-64	17
Installing Operating System Packages	19
Operating System Requirements for IBM: Linux on System z	19
Supported Red Hat Enterprise Linux 9 Distributions for IBM: Linux on System z	20
Supported Red Hat Enterprise Linux 8 Distributions for IBM: Linux on System z	21
Supported Red Hat Enterprise Linux 7 Distributions for IBM: Linux on System z	22
Supported SUSE Linux Enterprise Server 12 Distributions for IBM: Linux on System z	23
Supported SUSE Linux Enterprise Server 15 Distributions for IBM: Linux on System z	24
Operating System Requirements for Linux for Arm (aarch64)	27
Supported Oracle Linux 8 Distributions on Linux for Arm (aarch64)	27
Additional Drivers and Software Packages for Linux	29
Installing PAM for Login Authentication on Linux	29
Installing Oracle Messaging Gateway	29
Installation Requirements for ODBC and LDAP	30
About ODBC Drivers and Oracle Database	30
Installing ODBC Drivers for Linux x86-64	30
About LDAP and Oracle Plug-ins	30
Installing the LDAP Package	31
Installation Requirements for Programming Environments for Linux	31
Installation Requirements for Programming Environments for Linux x86-64	31
Installation Requirements for Programming Environments for Linux for Arm (aarch64)	32
Installation Requirements for Programming Environments for IBM: Linux on System z	33
Installation Requirements for Web Browsers	33
Checking Kernel and Package Requirements for Linux	33

4 Configuring Users, Groups and Environments for Oracle Database Client

Required Operating System Groups and Users	1
Determining If an Oracle Inventory and Oracle Inventory Group Exist	2
Creating the Oracle Inventory Group If an Oracle Inventory Does Not Exist	2
About Oracle Installation Owner Accounts	3
Identifying an Oracle Software Owner User Account	3
Creating Operating System Oracle Installation User Accounts	3
Creating an Oracle Software Owner User	4
Environment Requirements for Oracle Software Owners	4
Procedure for Configuring Oracle Software Owner Environments	5
Setting Remote Display and X11 Forwarding Configuration	7

5 Installing Oracle Database Client

About Image-Based Oracle Database Client Installation	1
Downloading Oracle Software	2
Downloading the Installation Archive Files from Oracle Website	2
Downloading the Software from Oracle Software Delivery Cloud Portal	3
About Character Set Selection During Installation	3
Running the Installer in a Different Language	4
Installing the Oracle Database Client Software	5
Running Setup Wizard to Install Oracle Database Client	5
Installing Oracle Database Client Using Image File	6
Using Oracle Net Configuration Assistant	7
Relinking Oracle Database Client Binaries After Installation	7

6 Installing Oracle Instant Client

About Oracle Instant Client	1
Hardware Requirements for Oracle Instant Client Installation on Linux for Arm (aarch64)	3
Operating System Requirements for Linux for Arm (aarch64)	3
Supported Oracle Linux 8 Distributions on Linux for Arm (aarch64)	3
Supported Oracle Linux 7 Distributions on Linux for Arm (aarch64)	4
Verifying Oracle Instant Client Code Signing	5
Verifying Signed Oracle Instant Client Zip Files	5
Verifying Signed Oracle Instant Client RPM Packages	6
Installing Oracle Instant Client Packages	8
Installing Oracle Instant Client Using Zip Files	8
Installing Oracle Instant Client Using RPMs	9
Installing Oracle Instant Client Using the Setup Wizard	12
Installing Oracle Instant Client Basic Light	13
About Oracle Instant Client Basic Light	13
Globalization Settings for Oracle Instant Client Basic Light	13
Installing Oracle Instant Client Basic Light Packages	14
Instant Client Libraries for OCI	14
Environment Variables for Oracle Instant Client	16
About the Oracle Instant Client SDK	17
Patching Oracle Instant Client Shared Libraries	18
Patching Oracle Instant Client	19
About Rebuilding Oracle Instant Client Packages and Libraries	19
Regenerating Data Shared Libraries	20

7 Oracle Database Client Postinstallation Tasks

Required Postinstallation Tasks	1
Downloading Release Update Patches	1
Recommended Postinstallation Tasks	2
Creating a Backup of the root.sh Script	2
Setting Language and Locale Preferences for Client Connections	3

8 Removing Oracle Database Software

About Oracle Deinstallation Options	1
Oracle Deinstallation (Deinstall)	3
Deinstallation Examples for Oracle Database Client	4

A Response Files

How Response Files Work	A-1
Reasons for Using Silent Mode or Response File Mode	A-2
Using Response Files	A-2
Preparing Response Files	A-2
Editing a Response File Template	A-3
Recording Response Files	A-4
Running Oracle Universal Installer Using a Response File	A-5

Index

List of Tables

1-1	<u>Server Hardware Checklist for Oracle Database Client Installations</u>	<u>1</u>
1-2	<u>Operating System General Checklist for Oracle Database Client on Linux</u>	<u>2</u>
1-3	<u>Server Configuration Checklist for Oracle Database Client</u>	<u>3</u>
1-4	<u>User Environment Configuration for Oracle Database</u>	<u>5</u>
1-5	<u>Storage Checklist for Oracle Database Client</u>	<u>6</u>
1-6	<u>Oracle Universal Installer Planning Checklist for Oracle Database Client Installation</u>	<u>6</u>
3-1	<u>x86-64 Oracle Linux 9 Minimum Operating System Requirements</u>	<u>5</u>
3-2	<u>x86-64 Oracle Linux 8 Minimum Operating System Requirements</u>	<u>7</u>
3-3	<u>x86-64 Oracle Linux 7 Minimum Operating System Requirements</u>	<u>9</u>
3-4	<u>x86-64 Red Hat Enterprise Linux 9 Minimum Operating System Requirements</u>	<u>11</u>
3-5	<u>x86-64 Red Hat Enterprise Linux 8 Minimum Operating System Requirements</u>	<u>13</u>
3-6	<u>x86-64 Red Hat Enterprise Linux 7 Minimum Operating System Requirements</u>	<u>14</u>
3-7	<u>x86-64 SUSE Linux Enterprise Server 12 Minimum Operating System Requirements</u>	<u>15</u>
3-8	<u>x86-64 SUSE Linux Enterprise Server 15 Minimum Operating System Requirements</u>	<u>17</u>
3-9	<u>Red Hat Enterprise Linux 9 Minimum Operating System Requirements</u>	<u>20</u>
3-10	<u>Red Hat Enterprise Linux 8 Minimum Operating System Requirements</u>	<u>21</u>
3-11	<u>Red Hat Enterprise Linux 7 Minimum Operating System Requirements</u>	<u>23</u>
3-12	<u>SUSE Linux Enterprise Server 12 Minimum Operating System Requirements</u>	<u>24</u>
3-13	<u>SUSE Linux Enterprise Server 15 Minimum Operating System Requirements</u>	<u>25</u>
3-14	<u>Linux Arm (aarch64) Oracle Linux 8 Minimum Operating System Requirements</u>	<u>27</u>
3-15	<u>Requirements for Programming Environments for Linux X86-64</u>	<u>31</u>
3-16	<u>Requirements for Programming Environments for Linux for Arm (aarch64)</u>	<u>32</u>
3-17	<u>Requirements for Programming Environments for IBM: Linux on System z</u>	<u>33</u>
6-1	<u>Oracle Instant Client Packages</u>	<u>2</u>
6-2	<u>Server Hardware Checklist for Oracle Instant Client Installation on Linux for Arm (aarch64)</u>	<u>3</u>
6-3	<u>Linux for Arm (aarch64) Oracle Linux 8 Minimum Operating System Requirements for Oracle Instant Client</u>	<u>4</u>
6-4	<u>Linux for Arm (aarch64) Oracle Linux 7 Minimum Operating System Requirements for Oracle Instant Client</u>	<u>5</u>
6-5	<u>Instant Client Basic Package Shared Libraries for OCI</u>	<u>15</u>
6-6	<u>Instant Client Basic Light Package Shared Libraries for OCI</u>	<u>15</u>
6-7	<u>Common Environment Variables for Oracle Instant Client</u>	<u>16</u>
6-8	<u>Commands to Regenerate the Zip Files and RPM Files</u>	<u>21</u>
A-1	<u>Response Files for Oracle Database Client</u>	<u>A-3</u>

Preface

This guide explains how to install and configure Oracle Database Client.

This guide also provides information about postinstallation tasks and how to remove the database client software.

Oracle Database Client enables development and deployment of applications that connect to Oracle Database, either on-premises or in the Cloud. The Oracle Database Client libraries provide the necessary network connectivity and advanced data features to make full use of Oracle Database. Tools, libraries, and SDKs included in Oracle Database Client provide quick and convenient data access.

- [Audience](#)
This guide is intended for anyone responsible for installing Oracle Database Client 19c.
- [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 Client 19c.

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 customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

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: <code>dd if=/dev/rdsd/c0t1d0s6 of=/dev/rst0 bs=10b \ count=10000</code>
braces { }	Braces indicate required items: <code>.DEFINE {macro1}</code>

Convention	Description
brackets []	Brackets indicate optional items: cvtcrt termname [outfile]
ellipses ...	Ellipses indicate an arbitrary number of similar items: CHKVAL fieldname value1 value2 ... valueN
<i>italic</i>	Italic type indicates a variable. Substitute a value for the variable: <i>library_name</i>
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.
<i>italic</i>	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 Client Installation Checklist

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

- [Server Hardware Checklist for Oracle Database Client Installation](#)
Use this checklist to check hardware requirements for Oracle Database Client installations.
- [Operating System Checklist for Oracle Database Client on Linux](#)
Use this checklist to check minimum operating system requirements for Oracle Database Client.
- [Server Configuration Checklist for Oracle Database Client](#)
Use this checklist to check minimum server configuration requirements for Oracle Database Client 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 Client](#)
Use this checklist to review storage minimum requirements and assist with configuration planning.
- [Installer Planning Checklist for Oracle Database Client](#)
Use this checklist to assist you to be prepared before starting Oracle Universal Installer.

Server Hardware Checklist for Oracle Database Client Installation

Use this checklist to check hardware requirements for Oracle Database Client installations.

Table 1-1 Server Hardware Checklist for Oracle Database Client Installations

Check	Task
Runlevel	3 or 5
Server Display Cards	At least 1024 x 768 display resolution, which Oracle Universal Installer requires.
Minimum network connectivity	Client is connected to a network.
Minimum RAM	At least 256 MB of RAM.
Linux for Arm (aarch64) processor	Oracle Database 19c Linux for Arm (aarch64) requires a CPU capable of Neoverse N1. Oracle recommends using a server with an Ampere Altra or Ampere Altra Max CPU.

Operating System Checklist for Oracle Database Client on Linux

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

Table 1-2 Operating System General Checklist for Oracle Database Client 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 9 with the Unbreakable Enterprise Kernel 7: 5.15.0-1.43.4.2.el9uek.x86_64 or later - Oracle Linux 9 with the Red Hat Compatible Kernel: 5.14.0-70.22.1.0.2.el9_0.x86_64 or later - Oracle Linux 8.8 with the Unbreakable Enterprise Kernel 7: 5.15.0-202.135.2.el8uek.x86_64 or later - Oracle Linux 8.1 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.0.7.el8uek.x86_64 or later - Oracle Linux 8 with the Red Hat Compatible kernel: 4.18.0-80.el8.x86_64 or later - Oracle Linux 7.4 with the Unbreakable Enterprise Kernel 4: 4.1.12-124.19.2.el7uek.x86_64 or later - Oracle Linux 7.4 with the Unbreakable Enterprise Kernel 5: 4.14.35-1818.1.6.el7uek.x86_64 or later - Oracle Linux 7.7 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.4.4.el7uek.x86_64 or later - Oracle Linux 7.5 with the Red Hat Compatible Kernel: 3.10.0-862.11.6.el7.x86_64 or later - Red Hat Enterprise Linux 9: 5.14.0-70.22.1.el9_0.x86_64 or later - Red Hat Enterprise Linux 8: 4.18.0-80.el8.x86_64 or later - Red Hat Enterprise Linux 7.5: 3.10.0-862.11.6.el7.x86_64 or later Note: Oracle Database 19c continues to be supported on Oracle Linux 7 and Red Hat Enterprise Linux 7, however, Oracle strongly recommends that you upgrade your operating system to Oracle Linux 8 or Red Hat Enterprise Linux 8 or later. - SUSE Linux Enterprise Server 12 SP3: 4.4.162-94.72-default or later - SUSE Linux Enterprise Server 15: 4.12.14-23-default or later Review the system requirements section for a list of minimum package requirements.
IBM: Linux on System z operating system requirements	The following IBM: Linux on System z kernels are supported: - Red Hat Enterprise Linux 9.2: 5.14.0-284.30.1.el9_2.s390x or later - Red Hat Enterprise Linux 8.3: 4.18.0-240.el8.s390x or later - Red Hat Enterprise Linux 7.4: 3.10.0-693.el7.s390x or later - SUSE Linux Enterprise Server 15: 5.3.18-57-default s390x or later - SUSE Linux Enterprise Server 12: 4.4.73-5-default s390x or later Review the system requirements section for a list of minimum package requirements.

Table 1-2 (Cont.) Operating System General Checklist for Oracle Database Client on Linux

Item	Task
Linux for Arm (aarch64) operating system requirements	Starting with Oracle Database 19c Release Update (19.19), the Linux for Arm (aarch64) operating system is supported with the following minimum supported version: Oracle Linux 8.6 with Unbreakable Enterprise Kernel 7: 5.15.0-6.80.3.1.el8uek.aarch64 or later Note: Oracle recommends that you update Oracle Linux 8 to the latest available release level.
Linux for Arm (aarch64) operating system requirements for Oracle Instant Client	The following Linux for Arm (aarch64) kernels are supported for Oracle Instant Client only: - Oracle Linux 8.6 with Unbreakable Enterprise Kernel 7: 5.15.0-6.80.3.1.el8uek.aarch64 or later - Oracle Linux 7.8 with the Unbreakable Enterprise Kernel 5: 4.14.35-1902.303.4.1.el7uek.aarch64 or later
Oracle Preinstallation RPM for Oracle Linux	If you use Oracle Linux, then Oracle recommends that you run an Oracle preinstallation RPM for your Linux release to configure your operating system for Oracle Database and Oracle Grid Infrastructure installations.
Oracle RPM Checker utility for IBM: Linux on System z	Oracle recommends that you use the Oracle RPM Checker utility to verify that you have the required Red Hat Enterprise Linux or SUSE packages installed on your IBM: Linux on System z operating system before you start the Oracle Database or Oracle Grid Infrastructure installation.

Server Configuration Checklist for Oracle Database Client

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

Table 1-3 Server Configuration Checklist for Oracle Database Client

Check	Task
Disk space allocated to the /tmp directory	At least 400 MB of space in the temporary disk space (/tmp) directory.
Swap space allocation relative to RAM	256 MB: 3 times the size of RAM Between 256 MB and 512 MB: 2 times the size of RAM Between 512 MB and 2 GB: 1.5 times the size of RAM Between 2 GB and 16 GB: Equal to the size of 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.

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

Check	Task
Oracle Inventory (oraInventory) and OINSTALL Group Requirements	- For new installs, if you have not configured an oraInventory directory, then the installer creates an Oracle inventory that is one directory level up from the Oracle base for the Oracle Grid Infrastructure install, and designates the installation owner's primary group as the Oracle Inventory group. - For upgrades, Oracle Universal Installer (OUI) detects an existing oraInventory directory from the <code>/etc/oraInst.loc</code> file, and uses the existing oraInventory. 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 <i>Oracle Database Installation Guide</i> for your platform.
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.
Determine root privilege delegation option for installation	During 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 such as Sudo. 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.
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.
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

- 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 <code>/etc/oraInst.loc</code> 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	The Preinstallation RPM automatically creates the <code>oracle</code> user and the <code>oraInventory</code> (<code>oinstall</code>) and <code>OSDBA</code> (<code>dba</code>) groups for you. Create additional operating system groups and users depending on your security requirements as described in this installation 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 <code>ORACLE_HOME</code>, <code>ORACLE_BASE</code>, <code>ORACLE_SID</code>, <code>TNS_ADMIN</code> 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 <code>oracle</code> or <code>grid</code> user by performing the following tasks: - Set the default file mode creation mask (<code>umask</code>) to <code>022</code> in the shell startup file. - Set the <code>DISPLAY</code> environment variable.

Storage Checklist for Oracle Database Client

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

Table 1-5 Storage Checklist for Oracle Database Client

Check	Task
Minimum local disk storage space for Oracle Database Client software	For Linux x86-64 and Linux for Arm (aarch64): At least 350 MB for an Instant Client installation. At least 2.4 GB for Administrator installation type. At least 2.0 GB for Runtime installation type. At least 2.4 GB for Custom installation type. At least 2.4 GB for image-based client installation. For IBM: Linux on System z: At least 274 MB for an Instant Client installation. At least 1.9 GB for Administrator installation type. At least 1.5 GB for Runtime installation type. At least 1.9 GB for Custom installation type.

Installer Planning Checklist for Oracle Database Client

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 Client 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 the Licensing Information	You are permitted to use only those components in the Oracle Database media pack for which you have purchased licenses. For more information about licenses, refer to the following URL: <i>Oracle Database Licensing Information</i>
Review Oracle Support Certification Matrix	New platforms and operating system software versions might 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.
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: <i>Oracle Clusterware Administration and Deployment Guide</i>

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

Check	Task
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.
Decide the client installation type	You can choose one of the following installation types when installing Oracle Database Client: • Instant Client: Enables you to install only the shared libraries required by Oracle Call Interface (OCI), Oracle C++ Call Interface (OCCI), Pro*C, or Java database connectivity (JDBC) OCI applications. This installation type requires much less disk space than the other Oracle Database Client installation types. For more information about Oracle Database Instant Client see the following URL: https://www.oracle.com/database/technologies/instant-client.html • Administrator: Enables applications to connect to an Oracle Database instance on the local system or on a remote system. It also provides tools that enable you to administer Oracle Database. • Runtime: Enables applications to connect to an Oracle Database instance on the local system or on a remote system. • Custom: Enables you to select individual components from the list of Administrator and Runtime components.
Obtain your My Oracle Support account information.	During installation, you require a My Oracle Support user name and password to configure security updates, download software updates, and other installation tasks. You can register for My Oracle Support at the following URL: https://support.oracle.com/
Decide if you need 32-bit client software	The 64-bit Oracle Database Client software does not contain any 32-bit client binaries. If you require 32-bit client binaries on 64-bit platforms, then install the 32-bit binaries from the respective 32-bit client software into a separate Oracle home. The 64-bit Oracle Database Client preinstallation requirements apply to 32-bit Oracle Database Client also. For more information, refer to My Oracle Support note 883702.1: https://support.oracle.com/rs?type=doc&id=883702.1
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

2

Checking and Configuring Server Hardware for Oracle Database Client

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

This section provides minimum server requirements to complete installation of Oracle Database Client. 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 clients to have full access to the original server 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.

① 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.

3

Configuring Operating Systems for Oracle Database Client on Linux

Complete operating system configuration requirements and checks for Linux operating systems before you start 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.
- [Reviewing Operating System Security Common Practices](#)
Secure operating systems are an important basis for general system security.
- [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.
- [Using Oracle RPM Checker on IBM: Linux on System z](#)
Use the Oracle RPM Checker utility to verify that you have the required Red Hat Enterprise Linux or SUSE packages installed on the operating system before you start the Oracle Database or Oracle Grid Infrastructure installation.
- [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.
- [Operating System Requirements for IBM: Linux on System z](#)
The Linux distributions and packages listed in this section are supported for this release on IBM: Linux on System z.
- [Operating System Requirements for Linux for Arm \(aarch64\)](#)
The Linux distribution and packages listed in this section are supported for this release on Linux for Arm (aarch64).
- [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.

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.

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 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.

Using Oracle RPM Checker on IBM: Linux on System z

Use the Oracle RPM Checker utility to verify that you have the required Red Hat Enterprise Linux or SUSE packages installed on the operating system before you start the Oracle Database or Oracle Grid Infrastructure installation.

1. Download the Oracle RPM Checker utility for your IBM: Linux on System z distribution from the link in [My Oracle Support Note 2553465.1](#).
2. Unzip the RPM, and install the RPM as root.

3. Run the utility as root to check your operating system packages. For example:

Download the Oracle RPM Checker utility for your IBM: Linux on System z distribution.
Then,

On Red Hat Enterprise Linux 9:

```
# rpm -ivh ora-val-rpm-RH9-DB-19c-19.0.1-1.s390x.rpm
```

On Red Hat Enterprise Linux 8:

```
# rpm -ivh ora-val-rpm-RH8-DB-19c.s390x.rpm
```

On Red Hat Enterprise Linux 7:

```
# rpm -ivh ora-val-rpm-RH7-DB-19c.s390x.rpm
```

On SUSE Linux Enterprise Server 12:

```
# rpm -ivh ora-val-rpm-S12-DB-19c.s390x.rpm
```

On SUSE Linux Enterprise Server 15:

```
# rpm -ivh ora-val-rpm-S15-DB-19c.s390x.rpm
```

On Red Hat Enterprise Linux, the utility checks and also installs all required RPMs. For example:

On Red Hat Enterprise Linux 9:

```
# yum install ora-val-rpm-RH9-DB-19c-19.0.1-1.s390x.rpm
```

On Red Hat Enterprise Linux 8:

```
# yum install ora-val-rpm-RH8-DB-19c.s390x.rpm
```

On Red Hat Enterprise Linux 7:

```
# yum install ora-val-rpm-RH7-DB-19c.s390x.rpm
```

To remove the Oracle RPM Checker utility:

On Red Hat Enterprise Linux 9:

```
# rpm -e ora-val-rpm-RH9-DB-19c-19.0.1-1.s390x
```

On Red Hat Enterprise Linux 8:

```
# rpm -e ora-val-rpm-RH8-DB-19c.s390x
```

On Red Hat Enterprise Linux 7:

```
# rpm -e ora-val-rpm-RH7-DB-19c.s390x
```

On SUSE Linux Enterprise Server 12:

```
# rpm -e ora-val-rpm-S12-DB-19c.s390x
```

On SUSE Linux Enterprise Server 15:

```
# rpm -e ora-val-rpm-S15-DB-19c.s390x
```

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/>

- [Supported Oracle Linux 9 Distributions for x86-64](#)
Use the following information to check supported Oracle Linux 9 distributions:
- [Supported Oracle Linux 8 Distributions for x86-64](#)
Use this 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 9 Distributions for x86-64](#)
Use the following information to check supported Red Hat Enterprise Linux 9 distributions:
- [Supported Red Hat Enterprise Linux 8 Distributions for x86-64](#)
Use this information to check supported Red Hat Enterprise Linux 8 distributions.
- [Supported Red Hat Enterprise Linux 7 Distributions for x86-64](#)
Use the following information to check supported Red Hat Enterprise Linux 7 distributions:
- [Supported SUSE Linux Enterprise Server 12 Distributions for x86-64](#)
Use the following information to check supported SUSE Linux Enterprise Server 12 distributions:
- [Supported SUSE Linux Enterprise Server 15 Distributions for x86-64](#)
Use this 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.

Supported Oracle Linux 9 Distributions for x86-64

Use the following information to check supported Oracle Linux 9 distributions:

Table 3-1 x86-64 Oracle Linux 9 Minimum Operating System Requirements

Item	Requirements
SSH Requirement	Ensure that OpenSSH is installed on your servers. OpenSSH is the required SSH software.
Oracle Linux 9	Starting with Oracle Database 19c Release 19.19, the following minimum versions are supported: • Oracle Linux 9 with the Unbreakable Enterprise Kernel 7: 5.15.0-1.43.4.2.el9uek.x86_64 or later • Oracle Linux 9 with the Red Hat Compatible Kernel: 5.14.0-70.22.1.0.2.el9_0.x86_64 or later Note: Oracle recommends that you update Oracle Linux to the latest available version and release level.

Table 3-1 (Cont.) x86-64 Oracle Linux 9 Minimum Operating System Requirements

Item	Requirements
Packages for Oracle Linux 9	Subscribe to the Oracle Linux 9 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, <code>oracle-database-preinstall-19c</code>. The Oracle Database Preinstallation RPM, <code>oracle-database-preinstall-19c</code>, 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, <code>oracle-database-preinstall-19c</code>, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. Install the latest released versions of the following packages: <code>bc</code> <code>binutils</code> <code>compat-openssl11</code> <code>elfutils-libelf</code> <code>fontconfig</code> <code>glibc</code> <code>glibc-devel</code> <code>ksh</code> <code>libaio</code> <code>libasan</code> <code>liblsan</code> <code>libX11</code> <code>libXau</code> <code>libXi</code> <code>libXrender</code> <code>libXtst</code> <code>libxcrypt-compat</code> <code>libgcc</code> <code>libibverbs</code> <code>libnsl</code> <code>librdmacm</code> <code>libstdc++</code> <code>libxcb</code> <code>libvirt-libs</code> <code>make</code> <code>policycoreutils</code> <code>policycoreutils-python-utils</code> <code>smartmontools</code> <code>sysstat</code> 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 3-1 (Cont.) x86-64 Oracle Linux 9 Minimum Operating System Requirements

Item	Requirements
Optional Packages for Oracle Linux 9	Based on your requirement, install the latest released versions of the following packages: - glibc-headers (for application development only) - ipmiutil (for Intelligent Platform Management Interface) - libns12 (for Oracle Database Client only) - libns12-devel (for Oracle Database Client only) - net-tools (for Oracle RAC and Oracle Clusterware) - nfs-utils (for Oracle ACFS)

Related Topics

- Installing Oracle Database Client 19c on Oracle Linux 9 and Red Hat Enterprise Linux 9
- Known Issues and Bugs

Supported Oracle Linux 8 Distributions for x86-64

Use this information to check supported Oracle Linux 8 distributions.

Table 3-2 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 8.1 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.0.7.el8uek.x86_64 or later • Oracle Linux 8 with the Red Hat Compatible Kernel: 4.18.0-80.el8.x86_64 or later • Starting with Oracle Database 19c Release 19.21 the following minimum version is supported: Oracle Linux 8.8 with the Unbreakable Enterprise Kernel 7: 5.15.0-202.135.2.el8uek.x86_64 or later • Starting with Oracle Grid Infrastructure 19c Release 19.23, the following minimum version is supported: Oracle Linux 8.8 with the Unbreakable Enterprise Kernel 7: 5.15.0-202.135.2.el8uek.x86_64 or later Note: Oracle recommends that you update Oracle Linux to the latest available version and release level.

Table 3-2 (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, <code>oracle-database-preinstall-19c</code>. The Oracle Database Preinstallation RPM, <code>oracle-database-preinstall-19c</code>, 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, <code>oracle-database-preinstall-19c</code>, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. <code>bc</code> <code>binutils</code> <code>compat-openssl10</code> <code>elfutils-libelf</code> <code>elfutils-libelf-devel</code> <code>fontconfig</code> <code>glibc</code> <code>glibc-devel</code> <code>ksh</code> <code>libaio</code> <code>libaio-devel</code> <code>libXrender</code> <code>libX11</code> <code>libXau</code> <code>libXi</code> <code>libXtst</code> <code>libgcc</code> <code>libnsl</code> <code>librdmacm</code> <code>libstdc++</code> <code>libstdc++-devel</code> <code>libxcb</code> <code>libibverbs</code> <code>libasan</code> <code>liblsan</code> <code>make</code> <code>policycoreutils</code> <code>policycoreutils-python-utils</code> <code>rng-tools</code> <code>smartmontools</code> <code>sysstat</code> 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 3-2 (Cont.) x86-64 Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
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)
Patches and Known Issues	- For a list of latest Oracle Database Release Updates (RU) and Release Update Revisions (RUR) patches for Oracle Linux Enterprise Linux 8 and Red Hat Enterprise Linux 8, visit My Oracle Support - For a list of known issues and open bugs for Oracle Linux 8 and Red Hat Enterprise Linux 8, read the Oracle Database Release Notes
KVM virtualization	Kernel-based virtual machine (KVM), also known as KVM virtualization, is certified on Oracle Database 19c 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 3-3 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: 4.1.12-124.19.2.el7uek.x86_64 or later - Oracle Linux 7.4 with the Unbreakable Enterprise Kernel 5: 4.14.35-1818.1.6.el7uek.x86_64 or later - Oracle Linux 7.7 with the Unbreakable Enterprise Kernel 6: 5.4.17-2011.4.4.el7uek.x86_64 or later - Oracle Linux 7.5 with the Red Hat Compatible Kernel: 3.10.0-862.11.6.el7.x86_64 or later Note: - Oracle recommends that you update Oracle Linux to the latest available version and release level. - Oracle Database 19c continues to be supported on Oracle Linux 7 and Red Hat Enterprise Linux 7, however, Oracle strongly recommends that you upgrade your operating system to Oracle Linux 8 or Red Hat Enterprise Linux 8 or later.

Table 3-3 (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, <code>oracle-database-preinstall-19c</code>. The Oracle Database Preinstallation RPM, <code>oracle-database-preinstall-19c</code>, 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, <code>oracle-database-preinstall-19c</code>, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. <code>bc</code> <code>binutils</code> <code>compat-libcap1</code> <code>compat-libstdc++-33</code> <code>elfutils-libelf</code> <code>elfutils-libelf-devel</code> <code>fontconfig-devel</code> <code>glibc</code> <code>glibc-devel</code> <code>ksh</code> <code>libaio</code> <code>libaio-devel</code> <code>libXrender</code> <code>libXrender-devel</code> <code>libX11</code> <code>libXau</code> <code>libXi</code> <code>libXtst</code> <code>libgcc</code> <code>libstdc++</code> <code>libstdc++-devel</code> <code>libxcb</code> <code>make</code> <code>policycoreutils</code> <code>policycoreutils-python</code> <code>smartmontools</code> <code>sysstat</code> 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 3-3 (Cont.) x86-64 Oracle Linux 7 Minimum Operating System Requirements

Item	Requirements
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) net-tools (for Oracle RAC and Oracle Clusterware) libvirt-libs (for KVM) nfs-utils (for Oracle ACFS) python (for Oracle ACFS Remote) python-configshell (for Oracle ACFS Remote) python-rtslib (for Oracle ACFS Remote) python-six (for Oracle ACFS Remote) targetcli (for Oracle ACFS Remote)
KVM virtualization	Kernel-based virtual machine (KVM), also known as KVM virtualization, is certified on Oracle Database 19c 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

Supported Red Hat Enterprise Linux 9 Distributions for x86-64

Use the following information to check supported Red Hat Enterprise Linux 9 distributions:

Table 3-4 x86-64 Red Hat Enterprise Linux 9 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 9	Starting with Oracle Database 19c Release 19.19, the following minimum versions are supported: Red Hat Enterprise Linux 9: 5.14.0-70.22.1.el9_0.x86_64 or later

Table 3-4 (Cont.) x86-64 Red Hat Enterprise Linux 9 Minimum Operating System Requirements

Item	Requirements
Packages for Red Hat Enterprise Linux 9	Install the latest released versions of the following packages: - bc - binutils - compat-openssl11 - elfutils-libelf - fontconfig - glibc - glibc-devel - ksh - libaio - libasan - liblsan - libX11 - libXau - libXi - libXrender - libXtst - libxcrypt-compat - libgcc - libibverbs - libnsl - librdmacm - libstdc++ - libxcb - libvirt-libs - 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 Red Hat Enterprise Linux 9	Based on your requirement, install the latest released versions of the following packages: - chkconfig (for Oracle Database and Oracle RAC only) - glibc-headers (for application development only) - 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 Red Hat Enterprise Linux 8 Distributions for x86-64

Use this information to check supported Red Hat Enterprise Linux 8 distributions.

Table 3-5 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 version: Red Hat Enterprise Linux 8: 4.18.0-80.el8.x86_64 or later
Packages for Red Hat Enterprise Linux 8	Install the latest released versions of the following packages: bc binutils elfutils-libelf elfutils-libelf-devel glibc glibc-devel ksh libaio libaio-devel libXrender libX11 libXau libXi libXtst libgcc libnsl librdmacm libstdc++ libstdc++-devel 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 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)
Patches and Known Issues	- For a list of latest Oracle Database Release Updates (RU) and Release Update Revisions (RUR) patches for Oracle Linux Enterprise Linux 8 and Red Hat Enterprise Linux 8, visit My Oracle Support - For a list of known issues and open bugs for Oracle Linux 8 and Red Hat Enterprise Linux 8, read the Oracle Database Release Notes

Supported Red Hat Enterprise Linux 7 Distributions for x86-64

Use the following information to check supported Red Hat Enterprise Linux 7 distributions:

Table 3-6 x86-64 Red Hat Enterprise Linux 7 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 7	Minimum supported version: Red Hat Enterprise Linux 7.5: 3.10.0-862.11.6.el7.x86_64 or later Note: Oracle Database 19c continues to be supported on Oracle Linux 7 and Red Hat Enterprise Linux 7, however, Oracle strongly recommends that you upgrade your operating system to Oracle Linux 8 or Red Hat Enterprise Linux 8 or later.
Packages for Red Hat Enterprise Linux 7	Install the latest released versions of the following packages: bc binutils compat-libcap1 compat-libstdc++-33 elfutils-libelf elfutils-libelf-devel fontconfig-devel glibc glibc-devel ksh libaio libaio-devel libX11 libXau libXi libXtst libXrender libXrender-devel libgcc libstdc++ libstdc++-devel libxcb 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 3-6 (Cont.) x86-64 Red Hat Enterprise Linux 7 Minimum Operating System Requirements

Item	Requirements
Optional Packages for Red Hat Enterprise Linux 7	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-utils (for Oracle ACFS) - python (for Oracle ACFS Remote) - python-configshell (for Oracle ACFS Remote) - python-rtplib (for Oracle ACFS Remote) - python-six (for Oracle ACFS Remote) - targetcli (for Oracle ACFS Remote)

Supported SUSE Linux Enterprise Server 12 Distributions for x86-64

Use the following information to check supported SUSE Linux Enterprise Server 12 distributions:

Table 3-7 x86-64 SUSE Linux Enterprise Server 12 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 12 SP3: 4.4.162-94.72-default or later

Table 3-7 (Cont.) x86-64 SUSE Linux Enterprise Server 12 Minimum Operating System Requirements

Item	Requirements
Packages for SUSE Linux Enterprise Server 12	Install the latest released versions of the following packages: - bc - binutils - glibc - glibc-devel - libX11 - libXau6 - libXtst6 - libcap-ng-utils - libcap-ng0 - libcap-progs - libcap1 - libcap2 - libelf-devel - libgcc_s1 - libjpeg-turbo - libjpeg62 - libjpeg62-turbo - libpcap1 - libpcre1 - libpcre16-0 - libpng16-16 - libstdc++6 - libtiff5 - libaio-devel - libaio1 - libXrender1 - make - mksh - pixz - rdma-core - rdma-core-devel - smartmontools - sysstat - xorg-x11-libs - xz 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 12	Based on your requirement, install the latest released versions of the following packages: - net-tools (for Oracle RAC and Oracle Clusterware) - nfs-kernel-server (for Oracle ACFS)

Supported SUSE Linux Enterprise Server 15 Distributions for x86-64

Use this information to check supported SUSE Linux Enterprise Server 15 distributions.

Table 3-8 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: 4.12.14-23-default or later

Table 3-8 (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 - libpcre16-0 - libpng16-16 - libstdc++6 - libtiff5 - libgfortran4 - mksh - make - pixz - rdma-core - rdma-core-devel - smartmontools - sysstat - systemd-sysvcompat - xorg-x11-libs - xz 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 3-8 (Cont.) x86-64 SUSE Linux Enterprise Server 15 Minimum Operating System Requirements

Item	Requirements
Optional Packages for SUSE Linux Enterprise Server 15	Based on your requirement, install the latest released versions of the following packages: net-tools (for Oracle RAC and Oracle Clusterware) nfs-kernel-server (for Oracle ACFS)
Patches and Known Issues	- For a list of latest Release Updates (RU) and Release Update Revisions (RUR) patches for SUSE Linux Enterprise Server 15, visit My Oracle Support - For a list of known issues and open bugs for SUSE Linux Enterprise Server 15, read the Oracle Database Release Notes

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
```

Operating System Requirements for IBM: Linux on System z

The Linux distributions and packages listed in this section are supported for this release on IBM: Linux on System z.

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

Note

32-bit packages in these requirements lists are needed only if you intend to use 32-bit client applications to access 64-bit servers.

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/>

- [Supported Red Hat Enterprise Linux 9 Distributions for IBM: Linux on System z](#)
Use the following information to check supported Red Hat Enterprise Linux 9 distributions:
- [Supported Red Hat Enterprise Linux 8 Distributions for IBM: Linux on System z](#)
Use the following information to check supported Red Hat Enterprise Linux 8 distributions:
- [Supported Red Hat Enterprise Linux 7 Distributions for IBM: Linux on System z](#)
Use the following information to check supported Red Hat Enterprise Linux 7 distributions:
- [Supported SUSE Linux Enterprise Server 12 Distributions for IBM: Linux on System z](#)
Use the following information to check supported SUSE Linux Enterprise Server 12 distributions:
- [Supported SUSE Linux Enterprise Server 15 Distributions for IBM: Linux on System z](#)
Use the following information to check supported SUSE Linux Enterprise Server 15 distributions:

Supported Red Hat Enterprise Linux 9 Distributions for IBM: Linux on System z

Use the following information to check supported Red Hat Enterprise Linux 9 distributions:

Table 3-9 Red Hat Enterprise Linux 9 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 9	Starting with Oracle Database 19c Release 19.23, the following minimum version is supported: • Red Hat Enterprise Linux 9.2: 5.14.0-284.30.1.el9_2.s390x or later

Table 3-9 (Cont.) Red Hat Enterprise Linux 9 Minimum Operating System Requirements

Item	Requirements
Packages for Red Hat Enterprise Linux 9	Install the latest released versions of the following packages: - glibc-devel - ksh - libaio - libasan - liblsan - libX11 - libXau - libXi - libXrender - libXtst - libxcrypt-compat - libgcc - libibverbs - libns - librdmacm - libstdc++ - libxcb - libvirt-libs - make - policycoreutils - policycoreutils-python-utils - smartmontools - sysstat
Optional Packages for Red Hat Enterprise Linux 9	Based on your requirement, install the latest released versions of the following packages: - chkconfig (for Oracle RAC and Oracle Clusterware only) - glibc-headers (for application development only) - ipmiutil (for Intelligent Platform Management Interface) - libns2 (for Oracle Database Client only) - libns2-devel (for Oracle Database Client only) - net-tools (for Oracle RAC and Oracle Clusterware only)

Supported Red Hat Enterprise Linux 8 Distributions for IBM: Linux on System z

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

Table 3-10 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.

Table 3-10 (Cont.) Red Hat Enterprise Linux 8 Minimum Operating System Requirements

Item	Requirements
Red Hat Enterprise Linux 8	Red Hat Enterprise Linux 8.3: 4.18.0-240.el8.s390x or later
Packages for Red Hat Enterprise Linux 8	The following packages (or later versions) must be installed: - bc-1.07.1-5.el8 (s390x) - binutils-2.30-79.el8 (s390x) - elfutils-libelf-0.180-1.el8 (s390x) - elfutils-libelf-devel-0.180-1.el8 (s390x) - fontconfig-2.13.1-3.el8 (s390x) - gcc-8.3.1-5.1.el8 (s390x) - gcc-c++-8.3.1-5.1.el8 (s390x) - glibc-2.28-127.el8_3.2 (s390x) - glibc-devel-2.28-127.el8_3.2 (s390x) - ksh-20120801-254.el8 (s390x) - libns12-1.2.0-2.20180605git4a062cf.el8 (s390x) - libX11-1.6.8-3.el8 (s390x) - libXau-1.0.9-3.el8 (s390x) - libXaw-1.0.13-10.el8 (s390x) - libXi-1.7.10-1.el8 (s390x) - libXrender-0.9.10-7.el8 (s390x) - libXtst-1.2.3-7.el8 (s390x) - libaio-0.3.112-1.el8 (s390x) - libaio-devel-0.3.112-1.el8 (s390x) - libattr-devel-2.4.48-3.el8 (s390x) - libgcc-8.3.1-5.1.el8 (s390x) - libgfortran-8.3.1-5.1.el8 (s390x) - libibverbs-29.0-3.el8 (s390x) - libns12-2.28-127.el8_3.2 (s390x) - librdmacm-29.0-3.el8 (s390x) - libstdc++-8.3.1-5.1.el8 (s390x) - libstdc++-devel-8.3.1-5.1.el8 (s390x) - libxcb-1.13.1-1.el8 (s390x) - make-4.2.1-10.el8 (s390x) - net-tools-2.0-0.52.20160912git.el8 (s390x) - pam-1.3.1-11.el8 (s390x) - pam-devel-1.3.1-11.el8 (s390x) - policycoreutils-2.9-9.el8 (s390x) - policycoreutils-python-utils-2.9-9.el8.noarch - smartmontools-7.1-1.el8 (s390x) - sysstat-11.7.3-5.el8 (s390x)

Supported Red Hat Enterprise Linux 7 Distributions for IBM: Linux on System z

Use the following information to check supported Red Hat Enterprise Linux 7 distributions:

Table 3-11 Red Hat Enterprise Linux 7 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 7	Red Hat Enterprise Linux 7.4: 3.10.0-693.el7.s390x or later
Packages for Red Hat Enterprise Linux 7	The following packages (or later versions) must be installed: binutils-2.25.1-31.base.el7 (s390x) compat-libcap1-1.10-7.el7 (s390x) gcc-4.8.5-16.el7 (s390x) gcc-c++-4.8.5-16.el7 (s390x) glibc-2.17-196.el7 (s390) glibc-2.17-196.el7 (s390x) glibc-devel-2.17-196.el7 (s390) glibc-devel-2.17-196.el7 (s390x) ksh-20120801-34.el7 (s390x) libX11-1.6.5-1.el7 (s390) libX11-1.6.5-1.el7 (s390x) libXaw-1.0.13-4.el7 (s390x) libXft-2.3.2-2.el7 (s390) libXi-1.7.9-1.el7 (s390) libXi-1.7.9-1.el7 (s390x) libXmu-1.1.2-2.el7 (s390) libXp-1.0.2-2.1.el7 (s390) libXtst-1.2.3-1.el7 (s390) libXtst-1.2.3-1.el7 (s390x) libaio-0.3.109-13.el7 (s390) libaio-0.3.109-13.el7 (s390x) libaio-devel-0.3.109-13.el7 (s390x) libattr-devel-2.4.46-12.el7 (s390) libattr-devel-2.4.46-12.el7 (s390x) libgcc-4.8.5-16.el7 (s390) libgcc-4.8.5-16.el7 (s390x) libgfortran-4.8.5-16.el7 (s390x) libstdc++-4.8.5-16.el7 (s390x) libstdc++-devel-4.8.5-16.el7 (s390x) make-3.82-23.el7 (s390x) pam-1.1.8-18.el7 (s390x) pam-devel-1.1.8-18.el7 (s390x) sysstat-10.1.5-12.el7 (s390x)

Supported SUSE Linux Enterprise Server 12 Distributions for IBM: Linux on System z

Use the following information to check supported SUSE Linux Enterprise Server 12 distributions:

Table 3-12 SUSE Linux Enterprise Server 12 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 12	SUSE Linux Enterprise Server 12: 4.4.73-5-default s390x or later
Packages for SUSE Linux Enterprise Server 12	The following packages (or later versions) must be installed: - binutils-2.26.1-9.12.1 (s390x) - gcc-4.8-6.189 (s390x) - gcc-c++-4.8-6.189 (s390x) - glibc-2.22-61.3 (s390x) - glibc-2.22-61.3 (s390) - glibc-devel-2.22-61.3 (s390x) - glibc-devel-2.22-61.3 (s390) - libaio-devel-0.3.109-17.15 (s390x) - libaio1-0.3.109-17.15 (s390x) - libaio1-0.3.109-17.15 (s390) - libX11-6-1.6.2-11.1 (s390x) - libXau6-1.0.8-4.58 (s390x) - libXau6-1.0.8-4.58 (s390x) - libXaw7-1.0.12-4.1 (s390x) - libXext6-1.3.2-3.61 (s390x) - libXext6-1.3.2-3.61 (s390) - libXft2-2.3.1-9.32 (s390x) - libXft2-2.3.1-9.32 (s390) - libXi6-1.7.4-17.1 (s390x) - libXi6-1.7.4-17.1 (s390) - libXmu6-1.1.2-3.60 (s390x) - libXp6-1.0.2-3.58 (s390x) - libXp6-1.0.2-3.58 (s390) - libXtst6-1.2.2-7.1 (s390x) - libXtst6-1.2.2-7.1 (s390) - libcap2-2.22-13.1 (s390x) - libstdc++48-devel-4.8.5-30.1 (s390) - libstdc++48-devel-4.8.5-30.1 (s390x) - libstdc++6-6.2.1+r239768-2.4 (s390) - libstdc++6-6.2.1+r239768-2.4 (s390x) - libxcb1-1.10-3.1 (s390x) - libxcb1-1.10-3.1 (s390) - libXmu6-1.1.2-3.60 (s390x) - mksh-50-2.13 (s390x) - make-4.0-4.1 (s390x)

Supported SUSE Linux Enterprise Server 15 Distributions for IBM: Linux on System z

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

Table 3-13 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 15	SUSE Linux Enterprise Server 15: 5.3.18-57-default s390x or later

Table 3-13 (Cont.) SUSE Linux Enterprise Server 15 Minimum Operating System Requirements

Item	Requirements
Packages for SUSE Linux Enterprise Server 15	The following packages (or later versions) must be installed: bc-1.07.1-11.37.s390x.rpm binutils-2.35.1-7.18.1.s390x gcc-7-3.3.22.s390x gcc-c++-7-3.3.22.s390x glibc-2.31-7.30.s390x glibc-devel-2.31-7.30.s390x libaio-devel-0.3.109-1.25.s390x libaio1-0.3.109-1.25.s390x libX11-6-1.6.5-3.15.1.s390x libXau6-1.0.8-1.26.s390x libXaw7-1.0.13-3.3.8.s390x libXext6-1.3.3-1.30.s390x libXft2-2.3.2-1.33.s390x libXi-devel-1.7.9-3.2.1.s390x libXi6-1.7.9-3.2.1.s390x libXmu6-1.1.2-1.30.s390x libXp6-1.0.3-1.24.s390x libXrender-devel-0.9.10-1.30.s390x libXrender1-0.9.10-1.30.s390x libXtst6-1.2.3-1.24.s390x libcap-ng-utils-0.7.9-4.37.s390x libcap-ng0-0.7.9-4.37.s390x libcap-progs-2.26-4.6.1.s390x libcap1-1.97-1.15.s390x libcap2-2.26-4.6.1.s390x libelf1-0.168-4.5.3.s390x libgcc_s1-10.2.1+git583-1.3.4.s390x libjpeg8-8.1.2-5.15.7.s390x libpcap1-1.9.1-1.33.s390x libpcre1-8.41-4.20.s390x libstdc++6-10.2.1+git583-1.3.4.s390x libtiff5-4.0.9-5.30.28.s390x libgfortran4-7.5.0+r278197-4.25.1.s390x libxcb1-1.13-3.5.1.s390x libXmu6-1.1.2-1.30.s390x mksh-56c-1.10.s390x make-4.2.1-7.3.2.s390x net-tools-2.0+git20170221.479bb4a-3.11.s390x net-tools-deprecated-2.0+git20170221.479bb4a-3.11.s390x.rpm pixz-1.0.6-1.13.s390x rdma-core-31.0-2.14.s390x rdma-core-devel-31.0-2.14.s390x smartmontools-7.0-6.1.s390x sysstat-12.0.2-3.27.1.s390x xorg-x11-libs-7.6.1-1.16.noarch xz-5.2.3-4.3.1.s390x

Table 3-13 (Cont.) SUSE Linux Enterprise Server 15 Minimum Operating System Requirements

Item	Requirements
	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.

Operating System Requirements for Linux for Arm (aarch64)

The Linux distribution and packages listed in this section are supported for this release on Linux for Arm (aarch64).

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/>

- [Supported Oracle Linux 8 Distributions on Linux for Arm \(aarch64\)](#)
Use the following information to check supported Oracle Linux 8 distributions:

Supported Oracle Linux 8 Distributions on Linux for Arm (aarch64)

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

Table 3-14 Linux Arm (aarch64) Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
Oracle Linux 8	Starting with Oracle Database 19c Release 19.19, the Linux for Arm (aarch64) operating system is supported with the following minimum supported version: Oracle Linux 8.6 with Unbreakable Enterprise Kernel 7: 5.15.0-6.80.3.1.el8uek.aarch64 or later Note: • Oracle recommends that you update Oracle Linux 8 to the latest available release level. • The Oracle Database 19c Release 19.19 Linux for Arm (aarch64) includes all the content from the Oracle Database 19c RU (19.19).

Table 3-14 (Cont.) Linux Arm (aarch64) Oracle Linux 8 Minimum Operating System Requirements

Item	Requirements
Packages for Oracle Linux 8	Install the latest released versions of the following packages: 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, <code>oracle-database-preinstall-19c</code>. The Oracle Database Preinstallation RPM, <code>oracle-database-preinstall-19c</code>, 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, <code>oracle-database-preinstall-19c</code>, then you do not have to install these packages, as the Oracle Database Preinstallation RPM automatically installs them. <code>bc</code> <code>binutils</code> <code>elfutils-libelf</code> <code>elfutils-libelf-devel</code> <code>fontconfig-devel</code> <code>gcc</code> <code>gcc-c++</code> <code>glibc</code> <code>glibc-devel</code> <code>ksh</code> <code>libaio</code> <code>libaio-devel</code> <code>libgcc</code> <code>libgfortran</code> <code>libibverbs</code> <code>libns</code> <code>libns2</code> <code>librdmacm</code> <code>libstdc++</code> <code>libstdc++-devel</code> <code>libxcb</code> <code>libX11</code> <code>libXau</code> <code>libXi</code> <code>libXrender</code> <code>libXtst</code> <code>make</code> <code>policycoreutils</code> <code>policycoreutils-python-utils</code> <code>smartmontools</code> <code>sysstat</code>

Related Topics

- [Installation Requirements for Programming Environments for Linux for Arm \(aarch64\)](#)
Ensure that your system meets the requirements for the programming environment you want to configure:

- Issues Affecting Linux for Arm (aarch64) for Oracle Database 19c

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.
- [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.

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.
- Oracle Messaging Gateway is not supported on Linux for Arm (aarch64).

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 install Oracle's ODBC driver, then you must also 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 install Oracle's ODBC driver, then you must also 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 for Arm \(aarch64\)](#)
Ensure that your system meets the requirements for the programming environment you want to configure:
- [Installation Requirements for Programming Environments for IBM: Linux on System z](#)
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 3-15 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.
Oracle Call Interface (OCI) Oracle C++ Call Interface Pro*C/C++	Latest released version of the Intel C/C++ Compiler or the GNU C and C++ compilers. Oracle C++ Call Interface (OCCI) applications can be built with the latest released version of the g++ or Intel C++ Compiler used with the standard template libraries of the gcc compilers listed in this table.

Table 3-15 (Cont.) Requirements for Programming Environments for Linux X86-64

Programming Environments	Support Requirements
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 OCCL.
Pro*COBOL	• Micro Focus Visual COBOL for Eclipse 2.3 - Update 2 • Micro Focus Visual COBOL v6.0
Python	cx_oracle users should install Python 3.9 or later. Python 3.9 is the minimum supported version on Oracle Linux 9. For installing Python on Oracle Linux 9, see Installing Python .

Installation Requirements for Programming Environments for Linux for Arm (aarch64)

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

Table 3-16 Requirements for Programming Environments for Linux for Arm (aarch64)

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.
Oracle Call Interface (OCI) Oracle C++ Call Interface Pro*C/C++	The GNU C and C++ compilers listed in this table. Oracle C++ Call Interface (OCCL) applications can be built with g++ used with the standard template libraries of the gcc compilers listed in this table.
gcc compiler packages	Install the latest released versions of the gcc packages listed here. gcc gcc-c++
Oracle XML Developer's Kit (XDK)	Oracle XML Developer's Kit is supported with the same compilers as OCCL.

Installation Requirements for Programming Environments for IBM: Linux on System z

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

Table 3-17 Requirements for Programming Environments for IBM: Linux on System z

Programming Environments	Support Requirements
Java Database Connectivity (JDBC) / Oracle Call Interface (OCI)	JDK 8 (Java SE Development Kit) with the JNDI extension with Oracle Java Database Connectivity.
Pro*COBOL	Micro Focus Visual COBOL v6.0 Micro Focus Server Express 5.1 Micro Focus Visual COBOL Development Hub 2.3 - Update 2

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 7 system:

```
4.14.35-1902.0.18.el7uek.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.

4

Configuring Users, Groups and Environments for Oracle Database Client

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.
- [Creating Operating System Oracle Installation User Accounts](#)
Before starting installation, create Oracle software owner user accounts, and configure their environments.
- [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 from your system administrator the name of an existing Oracle installation owner. Confirm that the existing owner is a member of the Oracle Inventory group.

```
oinstallobinstall
```

```
$ grep "oinstall" /etc/group
oinstall:x:54321: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(oper),54322(dba)
```

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

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) 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:
- [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:

Creating an Oracle Software Owner User

If the Oracle software owner user (oracle) 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 group dba.

```
# /usr/sbin/useradd -u 54321 -g oinstall -G dba oracle
```

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.

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
```

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.

5

Installing Oracle Database Client

The Oracle Database Client software is available as an image file for download and installation. Oracle Database Client installation binaries continue to be available in the traditional format as non-image zip files.

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 the installer to install the software. However, you can also run silent mode installations, without using the GUI.

Note

You cannot use the installer from an earlier Oracle release to install components from this release.

- [About Image-Based Oracle Database Client Installation](#)
Starting with Oracle Database 19c, installation and configuration of Oracle Database Client software is simplified with image-based installation.
- [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.
- [Running the Installer in a Different Language](#)
Describes how to run the installer in other languages.
- [Installing the Oracle Database Client Software](#)
These topics explain how to run the Setup Wizard to perform most database client installations.
- [Relinking Oracle Database Client Binaries After Installation](#)
After an Oracle Database Client installation, if required, you can modify the binaries using the `relink as_installed` option.

About Image-Based Oracle Database Client Installation

Starting with Oracle Database 19c, installation and configuration of Oracle Database Client software is simplified with image-based installation.

To install Oracle Database Client, 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.

You must extract the image software (`client_home.zip`) into the directory where you want your Oracle Database Client home to be located, and then run the Setup Wizard to start the Oracle Database Client installation and configuration. Oracle recommends that the Oracle home directory path you create is in compliance with the Oracle Optimal Flexible Architecture recommendations.

Using image-based installation, you can install Oracle Database Client 32-bit and 64-bit configurations of the Administrator installation type.

As with Oracle Database and Oracle Grid Infrastructure image file installations, Oracle Database Client image installations simplify Oracle Database Client installations and ensure best practice deployments. Oracle Database Client installation binaries continue to be available in the traditional format as non-image zip files.

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 Oracle Website](#)
Download installation archive files from the Oracle website.
- [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 Oracle Website

Download installation archive files from the Oracle website.

1. Use any browser to access the software download page on the Oracle website:
<http://www.oracle.com/technetwork/indexes/downloads/index.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, OraDB19c) 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.

Starting with Oracle Database 12c Release 2 (12.2), 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, database operations on character data may be slightly slower when compared to single-byte database character sets, such as WE8ISO8859P1 or WE8MSWIN1252. Storage space requirements for text in most languages that use characters outside of the ASCII repertoire are higher in AL32UTF8 compared to legacy character sets supporting the language. English data may require more space only if stored in CLOB (character large object) columns. Storage for non-character data types, such as NUMBER or DATE, does not depend on a character set. The universality and flexibility of Unicode usually outweighs these additional costs.

Consider legacy character sets only when the database needs to support a single group of languages and the use of a legacy character set is critical for fulfilling compatibility, storage, or performance requirements. The database character set to be selected in this case is the character set of most clients connecting to this database.

The database character set of a multitenant container database (CDB) determines which databases can be plugged in later. Ensure that the character set you choose for the CDB is compatible with the database character sets of the databases to be plugged into this CDB. If you use Unicode AL32UTF8 as your CDB character set, then you can plug in a pluggable database (PDB) in any database character set supported by Oracle Database (with the exception of EBCDIC-based character sets).

① See Also

Oracle Database Globalization Support Guide for more information about choosing a database character set for a multitenant container database (CDB)

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 Client Software

These topics explain how to run the Setup Wizard to perform most database client installations.

Starting with Oracle Database 19c, the Oracle Database client software is also available as an image file for download and installation. Oracle Database client installation binaries continue to be available in the traditional format as non-image zip files.

- [Running Setup Wizard to Install Oracle Database Client](#)
Use the `runInstaller` command to start the Oracle Database Client installation.
- [Installing Oracle Database Client Using Image File](#)
Extract the Oracle Database Client image files and use the `runInstaller` command to start the Oracle Database Client installation.
- [Using Oracle Net Configuration Assistant](#)
Run Oracle Net Configuration Assistant in standalone mode after the Oracle Database Client installation is complete to configure the listener, naming methods, net service names, and directory server usage.

Related Topics

- Installing Oracle Database Client 19c on Oracle Linux 9 and Red Hat Enterprise Linux 9

Running Setup Wizard to Install Oracle Database Client

Use the `runInstaller` command to start the Oracle Database Client installation.

Have all the information you need to provide regarding users groups, and storage paths before you start the installation.

During 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 such as `sudo`.

1. Log in as the Oracle installation owner user account (`oracle`).
2. From where you have downloaded the installation binaries, run the `runInstaller` command to start the Oracle setup wizard.

For example:

```
$ cd /home/oracle_sw/  
$ ./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.
- Oracle home or Oracle base cannot be symlinks, nor can any of their parent directories, all the way up to the root directory.

3. Select your installation type.

Installation screens vary depending on the installation option you select. Respond to the configuration prompts as needed.

Note

At any time during installation, if you have a question about what you are being asked to do, click **Help**.

Installing Oracle Database Client Using Image File

Extract the Oracle Database Client image files and use the `runInstaller` command to start the Oracle Database Client installation.

Starting with 19c, the Oracle Database Client software is also available as an image file for download and installation.

Have all the information you need to provide regarding storage paths before you start the installation. Oracle recommends that you have your My Oracle Support credentials available during installation. During installation, you are asked to run configuration scripts as the root user. You must run these scripts manually as root when prompted.

1. Log in as the Oracle installation owner user account (`oracle`).
2. Download the Oracle Database Client installation image files (`client_home.zip`) to a directory of your choice. For example, you can download the image files to the `/tmp` directory.
3. Create the 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/19.0.0/client_1
$ chgrp oinstall /u01/app/oracle/product/19.0.0/client_1
$ cd /u01/app/oracle/product/19.0.0/client_1
$ unzip -q /tmp/client_home.zip
```

Note

- Oracle recommends 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.

4. From the Oracle home directory, run the `runInstaller` command to start the Oracle Database Client Setup Wizard.

```
$ cd /u01/app/oracle/product/19.0.0/client_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.

5. The setup wizard starts an Administrator type installation of Oracle Database Client. Installation screens vary depending on the installation option you select. Respond to the configuration prompts as needed.

Note

At any time during installation, if you have a question about what you are being asked to do, click **Help**.

Using Oracle Net Configuration Assistant

Run Oracle Net Configuration Assistant in standalone mode after the Oracle Database Client installation is complete to configure the listener, naming methods, net service names, and directory server usage.

Oracle recommends that you have information ready about the host name of the computer where the Oracle database is installed.

To start Oracle Net Configuration Assistant in standalone mode:

1. Run `netca` from the `$ORACLE_HOME/bin` directory.
2. Respond to the configuration prompts and screens as needed. The screens vary depending on the options you select. At any time during the configuration, if you have a question about what you are being asked to do, click **Help**.

Related Topics

- *Oracle Database Net Services Administrator's Guide*

Relinking Oracle Database Client Binaries After Installation

After an Oracle Database Client installation, if required, you can modify the binaries using the `relink as_installed` option.

For example, you might want to relink the Oracle Database Client 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.

1. Log in as the Oracle Database Client owner user (`oracle`).

2. Set the ORACLE_HOME environment variable.

```
$ ORACLE_HOME=/u01/app/oracle/product/19.0.0/client_1
```

3. Go to the \$ORACLE_HOME/bin directory:

```
$ cd $ORACLE_HOME/bin
```

4. Run the relink script with the as_installed option to relink the binaries.

```
$ ./relink as_installed
```

The relinking is complete and the log files are generated under the \$ORACLE_HOME/install directory.

6

Installing Oracle Instant Client

Starting with Oracle Database 19c Release Update (19.10), Oracle Instant Client is available on Linux x86-64 and Linux for Arm (aarch64) platforms.

You can install Oracle Instant Client by downloading either the zip files or RPMs from the Oracle Instant Client download page on the Oracle website.

- [About Oracle Instant Client](#)
Learn how Oracle Instant Client enables applications to connect to a local or remote Oracle Database for development and production deployment.
- [Hardware Requirements for Oracle Instant Client Installation on Linux for Arm \(aarch64\)](#)
Use this checklist to check hardware requirements for an Oracle Instant Client installation on Linux for Arm (aarch64)
- [Operating System Requirements for Linux for Arm \(aarch64\)](#)
The Linux distribution and packages listed in this section are supported for this release on Linux for Arm (aarch64).
- [Verifying Oracle Instant Client Code Signing](#)
Starting with Oracle Database Client 19c, you can verify the code sign for Oracle Instant Client zips and RPMs to ensure the integrity of the packages before you deploy them in your environments.
- [Installing Oracle Instant Client Packages](#)
Learn the different methods to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms.
- [Installing Oracle Instant Client Basic Light](#)
Learn how to install Oracle Instant Client Basic Light on Linux x86-64 and Linux for Arm (aarch64) platforms.
- [Instant Client Libraries for OCI](#)
Learn about the Oracle Database client-side files required to deploy Oracle Call Interface (OCI) applications.
- [Environment Variables for Oracle Instant Client](#)
Some Oracle environment variables that can influence Oracle Instant Client installations.
- [About the Oracle Instant Client SDK](#)
Learn about the software development kit (SDK) development tools for Oracle Instant Client.
- [Patching Oracle Instant Client Shared Libraries](#)
Learn how to patch Oracle Instant Client shared libraries on top of a complete Oracle Database Client installation.

About Oracle Instant Client

Learn how Oracle Instant Client enables applications to connect to a local or remote Oracle Database for development and production deployment.

The Oracle Instant Client libraries provide the necessary network connectivity, as well as Oracle Database client-side files to create and run Oracle Call Interface (OCI), Oracle C++ Call Interface (OCCI), ODBC, and JDBC OCI applications to make full use of Oracle Database.

Oracle Instant Client is commonly used by the Oracle APIs of popular languages and environments including Python, Node.js, Go, Ruby, and PHP. Tools included in Oracle Instant Client, such as SQL*Plus, SQL*Loader, and Oracle Data Pump provide quick and convenient data access. Oracle Instant Client simplifies the deployment of applications by eliminating the need for an Oracle home on the client machines.

Oracle's standard client-server network interoperability allows connections between different versions of Oracle Instant Client and Oracle Database. For certified configurations, refer to [My Oracle Support note 207303.1](#). Tools such as Data Pump may have other restrictions.

Benefits of Oracle Instant Client

- Easy installation by extracting and installing zip files or RPM packages.
- Storage space requirement of applications running in Instant Client-mode is significantly reduced compared to the same application running in a full client-side installation.
- No loss of functionality or performance for applications deployed using Oracle Instant Client.
- Easy for independent software vendors to package applications.

Oracle Instant Client Packages

Various Oracle Instant Client packages are available that help you run applications.

Table 6-1 Oracle Instant Client Packages

Package Name	Description
Basic	All files required to run Oracle Call Interface (OCI), OCCI, and JDBC-OCI applications for Oracle Database
Basic Light	Smaller version of the Basic package, with only English error messages and Unicode, ASCII, and Western European character set support.
SDK	Additional header files and an example <code>makefile</code> for developing OCI and OCCI applications.
SQL*Plus	Additional package providing the SQL*Plus command-line tool for executing SQL and PL/SQL statements and scripts.
Tools	Additional tools including Data Pump, SQL*Loader and Workload Replay Client.
ODBC	Additional libraries providing ODBC.
Precompilers	Additional tools and libraries providing the Pro*C and Pro*COBOL precompilers.
JDBC-OCI Supplement	Additional libraries to support Internationalization.

Oracle Instant Client Utilities

Oracle Instant Client and Oracle Instant Client Basic Light contain the following utilities:

- **genezi:** You can retrieve information about your Oracle Instant Client installation using the `genezi` utility. This utility displays information such as the time zone files that are part of the Oracle Instant Client data shared library. The timezone information shows either the embedded timezone files, or those specified by the `ORA_TZFILE` environment variable, if it is set.

Run the `genezi -v` command to determine the client library information.

- **adrci**: Automatic Diagnostic Repository Command Interpreter (`adrci`) is a command-line tool that you can use to manage Oracle Database diagnostic data. `adrci` enables you to investigate problems, view health check reports, and package first-failure diagnostic data, all within a command-line environment.
- **uidrvci**: Do not use the `uidrvci` file as it is used to access diagnostic data.

Hardware Requirements for Oracle Instant Client Installation on Linux for Arm (aarch64)

Use this checklist to check hardware requirements for an Oracle Instant Client installation on Linux for Arm (aarch64)

Table 6-2 Server Hardware Checklist for Oracle Instant Client Installation on Linux for Arm (aarch64)

Check	Task
Minimum RAM	At least 256 MB of RAM.
Minimum Disk Space	At least 350 MB for an Oracle Instant Client installation.
Minimum network connectivity	Client is connected to a network.

Operating System Requirements for Linux for Arm (aarch64)

The Linux distribution and packages listed in this section are supported for this release on Linux for Arm (aarch64).

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/>

- [Supported Oracle Linux 8 Distributions on Linux for Arm \(aarch64\)](#)
Check supported Oracle Linux 8 kernel and package requirements for Oracle Instant Client on Linux for Arm (aarch64).
- [Supported Oracle Linux 7 Distributions on Linux for Arm \(aarch64\)](#)
Check supported Oracle Linux 7 kernel and package requirements for Oracle Instant Client on Linux for Arm (aarch64).

Supported Oracle Linux 8 Distributions on Linux for Arm (aarch64)

Check supported Oracle Linux 8 kernel and package requirements for Oracle Instant Client on Linux for Arm (aarch64).

Table 6-3 Linux for Arm (aarch64) Oracle Linux 8 Minimum Operating System Requirements for Oracle Instant Client

Item	Requirements
Oracle Linux 8	The Oracle Instant Client 19c Release 19.27, Linux for Arm (aarch64) operating system is supported with the following minimum supported version: Oracle Linux 8.6 with Unbreakable Enterprise Kernel 7: 5.15.0-6.80.3.1.el8uek.aarch64 or later Note: Oracle recommends that you update Oracle Linux 8 to the latest available release level.
Packages for Oracle Linux 8	Install the latest released versions of the following packages: bc binutils elfutils-libelf elfutils-libelf-devel fontconfig-devel gcc gcc-c++ glibc glibc-devel ksh libaio libaio-devel libgcc libgfortran libibverbs libns1 libns12 librdmacm libstdc++ libstdc++-devel libxcb libX11 libXau libXi libXrender libXtst make policycoreutils policycoreutils-python-utils smartmontools sysstat
JDK	JDK 8 (Java SE Development Kit).

Supported Oracle Linux 7 Distributions on Linux for Arm (aarch64)

Check supported Oracle Linux 7 kernel and package requirements for Oracle Instant Client on Linux for Arm (aarch64).

Table 6-4 Linux for Arm (aarch64) Oracle Linux 7 Minimum Operating System Requirements for Oracle Instant Client

Item	Requirements
Oracle Linux 7	The Oracle Instant Client 19c Release 19.10, Linux for Arm (aarch64) operating system is supported with the following minimum supported version: Oracle Linux 7.8 with the Unbreakable Enterprise Kernel 5: 4.14.35-1902.303.4.1.el7uek.aarch64 or later Note: Oracle recommends that you update Oracle Linux to the latest available version and release level.
Packages for Oracle Linux 7	Install the latest released versions of the following packages: devtoolset-8-binutils devtoolset-8-gcc pcre glibc libaio
JDK	JDK 8 (Java SE Development Kit).

Verifying Oracle Instant Client Code Signing

Starting with Oracle Database Client 19c, you can verify the code sign for Oracle Instant Client zips and RPMs to ensure the integrity of the packages before you deploy them in your environments.

- [Verifying Signed Oracle Instant Client Zip Files](#)
Use the Java utility `jar signer` to verify the integrity of your Oracle Instant Client zip files.
- [Verifying Signed Oracle Instant Client RPM Packages](#)
Use the `rpm` command to perform operations such as listing and verifying the signature fingerprint for Oracle Instant Client RPM packages for Oracle Linux 7 and Oracle Linux 8.

Verifying Signed Oracle Instant Client Zip Files

Use the Java utility `jar signer` to verify the integrity of your Oracle Instant Client zip files.

To verify the integrity of the Oracle Instant Client installation zip files before you extract the installation files:

1. Go to the directory where you have downloaded the zip file.
2. Run the `jar signer` command with the `-verify` option to quickly check your installation zip file:

```
jar signer -verify instant_client_zip_file
```

For example, to check the Oracle Instant Client Basic package zip file for **Linux x86-64**:

```
jar signer -verify instantclient-basic-linux.x64-19.27.0.0.0dbru.zip
jar verified.
```

3. If you want detailed certificate information, then use the `-verify` option along with the `-verbose:summary` and `-certs` options. For example:

```
jarsigner -verify -verbose:summary -certs instantclient-basic-  
linux.x64-19.27.0.0.0dbru.zip
```

For more jarsigner options, type `jarsigner -h` or review the [jarsigner reference documentation](#).

Verifying Signed Oracle Instant Client RPM Packages

Use the `rpm` command to perform operations such as listing and verifying the signature fingerprint for Oracle Instant Client RPM packages for Oracle Linux 7 and Oracle Linux 8.

Ensure that the RPM signature fingerprint matches the expected key fingerprint and also verify the RPM package has a signature. Before you start this process, perform the following tasks:

1. List the installed keys on your system:

```
rpm -qa gpg-pubkey
```

The output of this command is similar to the following:

```
gpg-pubkey-ad986da3-5cabf60d  
gpg-pubkey-16c083cd-49af3996
```

2. Import the GPG keys (optional):

```
rpm --import GPG-KEY
```

Note

Depending on your verification purpose you can install any GPG keys in the RPM database on your system to verify the signed packages.

Scenario 1: To verify an RPM package that has a signature with the key not installed on the system by default

1. Run the `rpm -qip` command to check the Oracle Instant Client Basic RPM package fingerprint. For example, for **Linux x86-64**:

```
rpm -qip oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm | grep  
Signature
```

The output of this command is similar to the following:

```
warning: oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm: Header  
V3 RSA/SHA256 Signature, key ID 8d8b756f: NOKEY  
Signature: RSA/SHA256, Mon 09 Jan 2023 07:38:06 PM GMT, Key ID  
bc4d06a08d8b756f
```

2. Run the `rpm --checksig --verbose` command to verify the package and print headers statuses and fingerprint details for the Oracle Instant Client Basic RPM package. For example, for **Linux x86-64**:

```
rpm --checksig --verbose oracle-instantclient19.27-  
basic-19.27.0.0.0-1.x86_64.rpm
```

The output of this command is similar to the following:

```
oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm:  
Header V3 RSA/SHA256 Signature, key ID 8d8b756f: NOKEY
```

This output verifies the Oracle Instant Client Basic RPM package signature by printing the key ID and returning NOKEY verification status as the GPG key with this ID is not present in the RPM key management system.

Scenario 2: To verify RPM package with both signature and key installed

1. Run the `rpm -qip` command to check the Oracle Instant Client Basic RPM package fingerprint. For example, for **Linux x86-64**:

```
rpm -qip oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm | grep  
Signature
```

The output of this command is similar to the following:

```
Signature: RSA/SHA256, Fri 06 Jan 2023 11:11:04 PM GMT, Key ID  
82562ea9ad986da3
```

2. Run the `rpm --checksig --verbose` command to verify the package and print headers statuses and fingerprint details for Oracle Instant Client Basic RPM package. For example, for **Linux x86-64**:

```
rpm --checksig --verbose oracle-instantclient19.27-  
basic-19.27.0.0.0-1.x86_64.rpm
```

The output of this command is similar to the following:

```
oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm:  
Header V3 RSA/SHA256 Signature, key ID ad986da3: OK  
  Header SHA256 digest: OK  
  Header SHA1 digest: OK  
  Payload SHA256 digest: OK  
  V3 RSA/SHA256 Signature, key ID ad986da3: OK  
  MD5 digest: OK
```

This output verifies the Oracle Instant Client Basic RPM package signature with the key and displays the fingerprint.

Related Topics

- [Oracle Linux Keys](#)
- [yum.oracle.com FAQs](#)

Installing Oracle Instant Client Packages

Learn the different methods to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms.

- [Installing Oracle Instant Client Using Zip Files](#)
Learn how to download and use zip files to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms.
- [Installing Oracle Instant Client Using RPMs](#)
Learn how to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms by downloading the RPMs using the `dnf install` or the `yum install` command.
- [Installing Oracle Instant Client Using the Setup Wizard](#)
Use the `runInstaller` command to start the Oracle Instant Client installation.

Installing Oracle Instant Client Using Zip Files

Learn how to download and use zip files to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms.

1. Go to the Oracle Instant Client Downloads page:
<https://www.oracle.com/database/technologies/instant-client/downloads.html>
2. Download the desired Oracle Instant Client zip file. Select the correct platform, architecture, and packages of your choice. For example, if your application is 64-bit, then ensure that you select 64-bit Instant Client and download the Basic Package zip file.
3. Unzip each zip file into a single directory such as `/opt/oracle/instantclient_19_27` that is accessible to your application.

For example, for **Linux x86-64**:

```
$ cd /opt/oracle
$ unzip instantclient-basic-linux.x64-19.27.0.0.0dbru.zip
```

For example, for Oracle Linux 8 on **Linux for Arm (aarch64)**:

```
$ cd /opt/oracle
$ unzip instantclient-basic-linux.arm64-19.27.0.0.0dbru.zip
```

The various packages unzip into `/opt/oracle/instantclient_19_27`.

4. Install the operating system `libaio` package. This is available as `libaio1` on some Linux distributions. For example, on Oracle Linux run:

```
$ sudo yum install libaio
```

5. If Oracle Instant Client is the only Oracle software installed on your system, then update the runtime link path, for example:

```
$ sudo sh -c "echo /opt/oracle/instantclient_19_27 > \
/etc/ld.so.conf.d/oracle-instantclient.conf"
$ sudo ldconfig
```

Alternatively, set the `LD_LIBRARY_PATH` environment variable prior to running applications. For example:

```
$ export LD_LIBRARY_PATH=/opt/oracle/instantclient_19_27:$LD_LIBRARY_PATH
```

Optionally, add the variable to configuration files such as `~/.bash_profile` and to application configuration files such as `/etc/sysconfig/httpd`.

6. If you intend to colocate optional Oracle configuration files such as `tnsnames.ora`, `sqlnet.ora`, `ldap.ora`, or `oraaccess.xml` with Oracle Instant Client, then move these files to the `network/admin` subdirectory.

This is the default Oracle configuration directory for applications linked with this Oracle Instant Client.

Alternatively, you can move the Oracle configuration files to another, accessible directory. Then set the environment variable `TNS_ADMIN` to that directory name.

7. To use binaries such as `sqlplus` from the SQL*Plus package, unzip the package to the same directory as the Basic package and then update your `PATH` environment variable. For example:

```
$ export PATH=/opt/oracle/instantclient_19_27:$PATH
```

8. Start your application.

Installing Oracle Instant Client Using RPMs

Learn how to install Oracle Instant Client on Linux x86-64 and Linux for Arm (aarch64) platforms by downloading the RPMs using the `dnf install` or the `yum install` command.

Note

Starting with 19c, by default, you can install only one version of the Oracle Instant Client RPM packages at a time. If you need multiple versions, use the Oracle Instant Client zip files.

Installing Oracle Instant Client From the yum Server

If you are using Oracle Linux and have configured your Oracle Linux yum server, then you can install Oracle Instant Client using the `yum install` command for Oracle Linux 7 and `dnf install` for Oracle Linux 8 and Oracle Linux 9.

1. Install the Oracle Linux release package to configure repository definitions for Oracle Instant Client:

For Oracle Linux 9:

```
$ sudo dnf install oracle-release-el9
```

For Oracle Linux 8:

```
$ sudo dnf install oracle-release-el8
```

For Oracle Linux 7:

```
$ sudo yum install oracle-release-el7
```

2. Go to the Oracle Instant Client repositories to view available Oracle Instant Client packages:

For Oracle Linux 9 on **Linux x86-64**:

https://yum.oracle.com/repo/OracleLinux/OL9/oracle/instantclient/x86_64/index.html

For Oracle Linux 8 on **Linux x86-64**:

https://yum.oracle.com/repo/OracleLinux/OL8/oracle/instantclient/x86_64/index.html

For Oracle Linux 7 on **Linux x86-64**:

https://yum.oracle.com/repo/OracleLinux/OL7/oracle/instantclient/x86_64/index.html

For Oracle Linux 8 on **Linux for Arm (aarch64)**:

<https://yum.oracle.com/repo/OracleLinux/OL8/oracle/instantclient/aarch64/index.html>

For Oracle Linux 7 on **Linux for Arm (aarch64)**:

<https://yum.oracle.com/repo/OracleLinux/OL7/oracle/instantclient/aarch64/index.html>

All installations require a Basic or Basic Light package. All other packages are optional.

3. Install Oracle Instant Client using yum for Oracle Linux 7 and dnf for Oracle Linux 8 and Oracle Linux 9. For example:

For Oracle Linux 8 and Oracle Linux 9 on **Linux x86-64** and **Linux for Arm (aarch64)**:

```
$ sudo dnf install oracle-instantclient19.27-basic
```

For Oracle Linux 7 on **Linux x86-64**:

```
$ sudo yum install oracle-instantclient19.27-basic
```

For Oracle Linux 7 on **Linux for Arm (aarch64)**:

```
$ sudo yum install oracle-instantclient19.10-basic
```

4. To install other Oracle Instant Client packages like SQL*Plus, run:

For Oracle Linux 8 and Oracle Linux 9 on **Linux x86-64** and **Linux for Arm (aarch64)**:

```
$ sudo dnf install oracle-instantclient19.27-sqlplus
```

For Oracle Linux 7 on **Linux x86-64**:

```
$ sudo yum install oracle-instantclient19.27-sqlplus
```

For Oracle Linux 7 on **Linux for Arm (aarch64)**:

```
$ sudo yum install oracle-instantclient19.10-sqlplus
```


Manually Downloading and Installing Oracle Instant Client RPM Packages

An alternative to installing Oracle Instant Client is to manually download and install the Oracle Instant Client packages.

1. Go to the Oracle Instant Client Downloads page:
<https://www.oracle.com/database/technologies/instant-client/downloads.html>
2. Download the desired Oracle Instant Client RPM package. Select the correct platform, architecture, and packages of your choice. For example, if your application is 64-bit, then ensure that you select 64-bit Instant Client and download the Basic Package RPM file.
3. Install the packages using yum for Oracle Linux 7 and dnf for Oracle Linux 8 and Oracle Linux 9. For example:

For Oracle Linux 8 and Oracle Linux 9 on **Linux x86-64**:

```
$ sudo dnf install oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm
```

For Oracle Linux 7 on **Linux x86-64**:

```
$ sudo yum install oracle-instantclient19.27-basic-19.27.0.0.0-1.x86_64.rpm
```

For Oracle Linux 8 on **Linux for Arm (aarch64)**:

```
$ sudo dnf install oracle-instantclient19.27-  
basic-19.27.0.0.0-1.aarch64.rpm
```

For Oracle Linux 7 on **Linux for Arm (aarch64)**:

```
$ sudo yum install oracle-instantclient19.10-  
basic-19.10.0.0.0-1.aarch64.rpm
```

Additionally, you may have to perform the following tasks before you start your application:

- If you intend to colocate optional Oracle configuration files such as `tnsnames.ora`, `sqlnet.ora`, `ldap.ora`, or `oraaccess.xml` with Oracle Instant Client, then move these files to the `/usr/lib/oracle/19.27/client64/lib/network/admin` or `/usr/lib/oracle/19.10/client64/lib/network/admin` subdirectory.

This is the default Oracle configuration directory for applications linked with this Oracle Instant Client.

Alternatively, you can move the Oracle configuration files to another, accessible directory. Then set the environment variable `TNS_ADMIN` to that directory name.

- To use binaries from the tools package, use yum to install the package and then update your `PATH` environment variable. For example:

For Oracle Linux 8 and Oracle Linux 9:

```
$ export PATH=/usr/lib/oracle/19.27/client64/bin:$PATH
```

For Oracle Linux 7:

```
$ export PATH=/usr/lib/oracle/19.10/client64/bin:$PATH
```

Installing Oracle Instant Client Using the Setup Wizard

Use the `runInstaller` command to start the Oracle Instant Client installation.

1. Log in as the Oracle installation owner user account (`oracle`).
2. Use any browser to access the Oracle Database Software Downloads page:
<https://www.oracle.com/database/technologies/oracle-database-software-downloads.html>
3. Download the Oracle Database Client installation zip file to a directory of your choice. For example, you can download the zip file to the `/tmp` directory.
4. Extract the client zip file and use the `runInstaller` command to start the installation.

```
$ ./runInstaller
```

5. In the Select Installation Type screen, select **Instant Client**. Click **Next**.
6. In the Specify Installation Location screen, enter an OFA-compliant Oracle home directory path like `/u01/app/oracle/product/19.0.0/client_1`. Click **Next**.
7. Note the information in the Summary screen and click **Next**.
8. The Finish screen displays the status of the Oracle Instant Client installation. Click **Close** to complete the installation and exit the installer.
9. The Oracle Instant Client shared libraries like `libociei.so`, `libclntsh.so.19.1`, `libclntshcore.so.19.1`, and `libnnz19.so` are available in the Oracle client home directory `/u01/app/oracle/product/19.0.0/client_1`
10. Set the operating system shared library path `LD_LIBRARY_PATH` environment variable to the Oracle client home directory.

```
$ export LD_LIBRARY_PATH=/u01/app/oracle/product/19.0.0/
client_1:$LD_LIBRARY_PATH
```

11. If you intend to colocate optional Oracle configuration files such as `tnsnames.ora`, `sqlnet.ora`, `ldap.ora`, or `oraaccess.xml` with Oracle Instant Client, then move these files to the `network/admin` subdirectory.

This is the default Oracle configuration directory for applications linked with this Oracle Instant Client.

Alternatively, you can move the Oracle configuration files to another, accessible directory. Then set the environment variable `TNS_ADMIN` to that directory name.

12. To use binaries such as `sqlplus` from the SQL*Plus package, unzip the package to the same directory as the Basic package and then update your `PATH` environment variable. For example:

```
$ export PATH=/u01/app/oracle/product/19.0.0/client_1:$PATH
```

13. Start your application.

To remove this Oracle Instant Client installation use the `runInstaller` command along with the `-deinstall -home` options:

```
$ ./runInstaller -deinstall -home /u01/app/oracle/product/19.0.0/client_1
```

Installing Oracle Instant Client Basic Light

Learn how to install Oracle Instant Client Basic Light on Linux x86-64 and Linux for Arm (aarch64) platforms.

- [About Oracle Instant Client Basic Light](#)
The size of the Instant Client Basic Light library is smaller than Oracle Instant Client. This is because Instant Client Basic Light contains only English language error message files and only a few supported character set definitions out of around 250.
- [Globalization Settings for Oracle Instant Client Basic Light](#)
Learn about the character sets supported by Oracle Instant Client Basic Light.
- [Installing Oracle Instant Client Basic Light Packages](#)
Learn how to install Oracle Instant Client Basic Light by downloading zip files or RPM files, or by using the package repository.

About Oracle Instant Client Basic Light

The size of the Instant Client Basic Light library is smaller than Oracle Instant Client. This is because Instant Client Basic Light contains only English language error message files and only a few supported character set definitions out of around 250.

The Instant Client Basic Light version is geared toward applications that use either Unicode, ASCII, and Western European character set support. There is no restriction on the `LANGUAGE` and the `TERRITORY` fields of the `NLS_LANG` setting, hence, Instant Client Basic Light operates with any language and territory settings. Only English error messages are provided with Instant Client Basic Light, hence, error messages generated by the client, such as Net connection errors, are always reported in English, even if you set `NLS_LANG` to a language other than `AMERICAN`. Error messages generated by the database server, such as SQL statement syntax errors, are generated in the selected language provided that you install the appropriate translated message files in the Oracle home of the database instance.

Globalization Settings for Oracle Instant Client Basic Light

Learn about the character sets supported by Oracle Instant Client Basic Light.

Oracle Instant Client Basic Light supports the following client character sets:

Single-byte

- US7ASCII
- WE8DEC
- WE8MSWIN1252
- WE8ISO8859P1

Unicode

- UTF8

- AL16UTF16
- AL32UTF8

Oracle Instant Client Basic Light can connect to databases having one of these database character sets:

- US7ASCII
- WE8DEC
- WE8MSWIN1252
- WE8ISO8859P1
- WE8EBCDIC37C
- WE8EBCDIC1047
- UTF8
- AL32UTF8

Oracle Instant Client Basic Light returns an error if any other character set is used as the client or database character set. Oracle Instant Client Basic Light can also operate with the OCI Environment handles created in the OCI_UTF16 mode.

Installing Oracle Instant Client Basic Light Packages

Learn how to install Oracle Instant Client Basic Light by downloading zip files or RPM files, or by using the package repository.

Installing Oracle Instant Client Basic Light Using Zip Files

To download and use zip files to install Oracle Instant Client Basic Light, follow the instructions in [Installing Oracle Instant Client Using Zip Files](#). However, unzip the `basiclite` zip file instead of the `basic` zip file.

Installing Oracle Instant Client Basic Light Using RPMs

To remove earlier installations and install Oracle Instant Client Basic Light using RPMs, follow the instructions in [Installing Oracle Instant Client Using RPMs](#). However, use the `basiclite` packages instead of `basic`.

Note

To install other packages like SQL*Plus, install the Oracle Instant Client Basic package.

Instant Client Libraries for OCI

Learn about the Oracle Database client-side files required to deploy Oracle Call Interface (OCI) applications.

Table 6-5 Instant Client Basic Package Shared Libraries for OCI

File Name	Description
libclntsh.so.19.1 libclntshcore .so.19.1	Client code library
libociei.so	OCI Instant Client Data Shared Library
libnnz19.so	Security Library

Table 6-6 Instant Client Basic Light Package Shared Libraries for OCI

File Name	Description
libclntsh.so.19.1 libclntshcore .so.19.1	Client code library
libociicus.so	OCI Instant Client Data Shared Library
libnnz19.so	Security Library

Guidelines for Basic and Basic Light Packages

- The libraries must reside in the same directory in order to use Oracle Instant Client.
- Other libraries and utilities are available when you install Oracle Instant Client, but are not needed for OCI runtime use. For example, you can delete files such as `liboramysql19.so` and `ojdbc8.jar`.
- In general, all OCI functionality is available to applications run using Oracle Instant Client, except that the Instant Client is for client-side operations only. Therefore, server-side external procedures cannot use Oracle Instant Client libraries.
- OCI applications, by default, look for the OCI Data Shared Library, `libociei.so` in the runtime library search path, for example, in `$LD_LIBRARY_PATH` to determine if the application should operate in the Instant Client mode. If the OCI Data Shared Library is not found, then OCI tries to load the Oracle Instant Client Basic Light Data Shared Library `libociicus.so`.

Guidelines for Running OCI Applications

- If you have multiple directories containing Oracle Client libraries, then only one such directory should be in the operating system library search path.
- If Oracle Instant Client is the only Oracle software installed on your system, then update the runtime link path using `ldconfig`. For example:

For Oracle Linux 8 and Oracle Linux 9:

```
$ sudo sh -c "echo /opt/oracle/instantclient_19_27 > \
/etc/ld.so.conf.d/oracle-instantclient.conf"
$ sudo ldconfig
```

For Oracle Linux 7:

```
$ sudo sh -c "echo /opt/oracle/instantclient_19_10 > \
/etc/ld.so.conf.d/oracle-instantclient.conf"
$ sudo ldconfig
```

Alternatively, to run OCI applications using Oracle Instant Client, set the operating system library search path, for example LD_LIBRARY_PATH, to the directory with the Oracle Instant Client libraries.

- If an Oracle Database installation or full Oracle Database Client installation exists on the same system, then you should not have \$ORACLE_HOME/lib and the Oracle Instant Client directory on the operating system library search path simultaneously regardless of the order in which they appear in the operating system library search path. Either the \$ORACLE_HOME/lib directory (for non-Instant Client operations) or Oracle Instant Client directory (for Oracle Instant Client operations) should be in the operating system library search path variable, but not both.
- If you intend to colocate optional Oracle configuration files such as tnsnames.ora, sqlnet.ora, ldap.ora, or oraaccess.xml with Oracle Instant Client, then move these files to the network/admin subdirectory.

Environment Variables for Oracle Instant Client

Some Oracle environment variables that can influence Oracle Instant Client installations.

Table 6-7 Common Environment Variables for Oracle Instant Client

Environment Variable	Description	For More Information
LD_LIBRARY_PATH	Used on Linux and some UNIX platforms. Set this to include the Oracle Instant Client libraries. For example, /opt/oracle/instantclient_19_27. Not needed if the libraries are located by an alternative method, such as from running ldconfig. On other platforms you will need to set the OS specific equivalent, such as LIBPATH or SHLIB_PATH. Note: You should not set this environment variable if you have installed the Instant Client RPMs, or if the libraries are located by an alternative method, such as by running ldconfig.	Oracle Database Administrator's Reference
NLS_LANG	If necessary, set the NLS_LANG environment variable. The NLS_LANG environment variable sets the language and territory used by the client application and the database server. It also sets the client's character set, which is the character set for data entered or displayed by a client program. If not set, a default value will be chosen by Oracle Instant Client.	Oracle Database Globalization Support Guide
TNS_ADMIN	The location of the optional Oracle Net configuration files and Oracle Instant Client configuration files, including tnsnames.ora, sqlnet.ora, and oraaccess.xml, if they are not in the default location. If TNS_ADMIN is not set, then the instantclient_19_27/network/admin subdirectory should contain your Oracle Net Services configuration files.	Oracle Database Net Services Administrator's Guide

Table 6-7 (Cont.) Common Environment Variables for Oracle Instant Client

Environment Variable	Description	For More Information
ORA_TZFILE	Oracle Instant Client can use the ORA_TZFILE environment variable to read the time zone file from the file system when this environment variable is set. If ORA_TZFILE is not set, then Oracle Instant Client uses the larger, default, <code>timezlg_n.dat</code> file from the shared libraries. If you want Oracle Instant Client to use the smaller <code>timezone_n.dat</code> file, then set the ORA_TZFILE environment variable to the name of the file without any absolute or relative path names. For example: <pre>\$ export ORA_TZFILE=timezone_n.dat</pre> Where, n is the time zone data file version number. Run the <code>genezi -v</code> command to determine the client library timezone information. To use an external time zone file, create an <code>oracore/zoneinfo</code> subdirectory under the Oracle Instant Client directory, and move the time zone file into this <code>oracore/zoneinfo</code> subdirectory. Now, set ORA_TZFILE to the time zone file name, without any absolute or relative path names. You can alternatively place the external time zone file in any directory, and then set the ORA_TZFILE environment variable to the absolute path of the file. For example: <pre>\$ export ORA_TZFILE=/opt/oracle/config/timezone_n.dat</pre>	<i>Oracle Database Globalization Support Guide</i>
ORA_SDTZ	Specifies the default session time zone.	<i>Oracle Database Globalization Support Guide</i>

Note

Do not set the ORACLE_HOME environment variable for Oracle Instant Client installations.

About the Oracle Instant Client SDK

Learn about the software development kit (SDK) development tools for Oracle Instant Client.

The SDK is a set of development tools that you can use to create applications for Oracle Instant Client. The SDK is available for download as an RPM or a zip file for Oracle Linux x86-64 and Arm platforms. Here are some examples of these RPM and zip files:

For Linux x86-64

- RPM (Oracle Linux 9): `oracle-instantclient19.27-devel-19.27.0.0.0-1.el9.x86_64.rpm`
- RPM (Oracle Linux 8): `oracle-instantclient19.27-devel-19.27.0.0.0-1.el8.x86_64.rpm`
- RPM (Oracle Linux 7): `oracle-instantclient19.27-devel-19.27.0.0.0-1.x86_64.rpm`
- Zip file: `instantclient-sdk-linux.x64-19.27.0.0.0dbru.zip`

For Linux for Arm (aarch64)

- RPM: `oracle-instantclient19.27-devel-19.27.0.0.0-1.el8.aarch64.rpm`
- Zip file: `instantclient-sdk-linux.arm64-19.27.0.0.0dbru.zip`

The SDK contains the following:

- Both C and C++ header files and a makefile for developing OCI and OCCI applications while in an Oracle Instant Client environment. Developed applications can be deployed in any client environment.
- Both C and C++ demonstration programs.
- The Object Type Translator (OTT) utility and its classes to generate application header files.
- On Linux, the `demo.mk` makefile is included to build demos. For example, the `instantclient_19_27` directory must be in the runtime library search path, and set `LD_LIBRARY_PATH` before linking the application.

Patching Oracle Instant Client Shared Libraries

Learn how to patch Oracle Instant Client shared libraries on top of a complete Oracle Database Client installation.

You must rebuild your Oracle Instant Client packages and libraries as part of the patching process.

Note

Patching Oracle Instant Client is not supported on Linux for Arm (aarch64).

- [Patching Oracle Instant Client](#)
Perform the following steps to patch Oracle Instant Client.
- [About Rebuilding Oracle Instant Client Packages and Libraries](#)
Separate makefile targets are used to create the data shared libraries, zip files, and RPM files either individually or all together.
- [Regenerating Data Shared Libraries](#)
Learn how to regenerate the Oracle Instant Client data shared libraries `libociei.so` and `libociicus.so`.

- [Regenerating Zip Files and RPM Files](#)
Learn how to regenerate the zip files and RPM files for the various Oracle Instant Client packages.

Patching Oracle Instant Client

Perform the following steps to patch Oracle Instant Client.

1. Complete a full Oracle Database Client Administration installation based on Oracle home for patching.
2. Apply the patch to your full Oracle Database Client Administration installation using the OPatch utility.
3. Regenerate the data shared libraries, zip files, and RPM files.
See [Regenerating Data Shared Libraries](#) and [Regenerating Zip Files and RPM Files](#).
4. Copy the data shared libraries, zip files, and RPM files to the target system and unzip them to the Oracle Instant Client directory.

The OPatch utility stores the patching information of the ORACLE_HOME installation in `libcintsh.so`. You can retrieve this patching information using the `genezi -v` command.

If the Oracle Instant Client deployment system does not have the `genezi` utility, copy it from the `$ORACLE_HOME/bin` directory.

Related Topics

- [Patching Using OPatch](#)

About Rebuilding Oracle Instant Client Packages and Libraries

Separate `makefile` targets are used to create the data shared libraries, zip files, and RPM files either individually or all together.

Regenerating the data shared libraries requires both a compiler and a linker, which may not be available on all installations.

The regenerated Oracle Instant Client binaries contain only the Oracle Instant Client files installed in the Oracle Client Administrator Home from which you regenerate the libraries and files. Hence, error messages, character set encodings, and time zone files that are available in the regeneration environment are the only ones that are packaged in the data shared libraries. Error messages, character set encodings, and time zone files depend on which national languages were selected for the installation of the Oracle Client Administrator Home.

Note

Regeneration of data shared libraries, zip files, and RPM files is not available on Microsoft Windows.

Regenerating Data Shared Libraries

Learn how to regenerate the Oracle Instant Client data shared libraries `libociei.so` and `libociicus.so`.

You can regenerate the Oracle Call Interface (OCI) Oracle Instant Client data shared libraries `libociei.so` and `libociicus.so` using the following commands in an Administrator install of a full Oracle Database Client:

1. Set `ORACLE_HOME` to the software install directory. For example:

```
$ export ORACLE_HOME=/home/user/product/19.0.0/client_1/
```

2. Set `LD_LIBRARY_PATH` to the library directory. For example:

```
$ export LD_LIBRARY_PATH=$ORACLE_HOME/lib
```

3. Create the following directory:

```
$ mkdir -p $ORACLE_HOME/rdbms/install/instantclient/light
```

4. Under Oracle home directory, go to the `/rdbms/lib` directory.

```
$ cd $ORACLE_HOME/rdbms/lib
```

5. To regenerate `libociei.so`:

```
$ make -f ins_rdbms.mk igenlibociei
```

The generated file will be in `$ORACLE_HOME/instantclient/libociei.so`

6. To regenerate `libociicus.so`:

```
$ make -f ins_rdbms.mk igenlibociicus
```

The generated file will be in `$ORACLE_HOME/instantclient/light/libociicus.so`

7. To regenerate both `libociei.so` and `libociicus.so`:

```
$ make -f ins_rdbms.mk igenliboci
```

Regenerating Zip Files and RPM Files

Learn how to regenerate the zip files and RPM files for the various Oracle Instant Client packages.

Before you regenerate the zip files and RPM files, install the `rpm-build` package.

For Oracle Linux 7:

```
$ sudo yum install rpm-build
```

For Oracle Linux 8 and later:

```
$ sudo dnf install rpm-build
```

Table 6-8 Commands to Regenerate the Zip Files and RPM Files

Package Name	Commands
All Packages	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_all_zip</pre>
Basic	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_basic_zip</pre>
Basic Light	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_basiclite_zip</pre>
JDBC	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_jdbc_zip</pre>
ODBC	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_odbc_zip</pre>
SQL*Plus	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_sqlplus_zip</pre>
Tools	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_tools_zip</pre>
SDK	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_sdk_zip</pre>
Precompilers	<pre>\$ cd \$ORACLE_HOME/rdbms/lib \$ make -f ins_rdbms.mk ic_precomp_zip</pre>

The new zip files and RPM files are generated under the \$ORACLE_HOME/rdbms/install/instantclient directory.

Oracle Database Client Postinstallation Tasks

Complete configuration tasks after you install Oracle Database.

You are required to complete some configuration tasks after Oracle Database Client 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.

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 Monthly Recommended Patches (MRPs) patches for your Oracle software after you complete your installation.

Downloading Release Update Patches

Download and install Release Updates (RU) and Monthly Recommended Patches (MRPs) patches for your Oracle software after you complete your installation.

Oracle provides quarterly updates in the form of Release Updates (RU) and Monthly Recommended Patches (MRPs). 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. Log in to the My Oracle Support website:
<https://support.oracle.com>
2. In the **My Oracle Support Support for Oracle Hardware, Software, and Managed Cloud services** section, click **Sign-in or create your Oracle account**.

Note

If you are not a My Oracle Support registered user, then register.

3. From the search bar, select **Patches and Product Certification Matrix**. The **Patch Search** window is displayed.
4. To search for patches, in the **Patch Search** window, search by **Patch Name or Number** or **Bug Number** if you have it. Otherwise, search using the **Product**, **Release**, **Platform**, and **Language** criteria.
5. Click **Apply**. The patch search results are displayed in the main window.
6. Click the patch number in the **Patch Name** column to open the Patch details in a new window.
7. Click **Download** to download the patch. Next, uncompress the Oracle patch updates that you downloaded from My Oracle Support.
8. Click **ReadMe**.

The README page is displayed. It contains information about the patch that you just downloaded, and how to apply the patch to your installation.

Note

The use of OPatch and OPatchAuto for out-of-place patching continues to be deprecated.

For patching in Oracle AI Database 26ai, Oracle recommends that you use out-of-place patching using Gold images to apply quarterly release updates (RUs). This deprecation does not affect in-place patching. If you want to perform in-place patching, then OPatch and OPatchAuto continue to be available for this purpose. The Oracle AI Database 26ai Free release is not supported for patching with RUs.

Related Topics

- [My Oracle Support note 888.1](#)
- [Patch Delivery Methods for Oracle AI Database](#)

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.

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.

① See Also

Oracle Database Globalization Support Guide for more information about configuring user locale preferences

8

Removing Oracle Database Software

These topics describe how to remove Oracle software and configuration files.

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 standalone 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 Client](#)
Use these examples to help you understand how to run the `deinstall` command.

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)
- 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 19c Oracle home to remove Oracle software from an existing 11.2.0.4 Oracle home.
- Starting with Oracle Database 12c Release 1 (12.1.0.2), the `roothas.sh` script replaces the `roothas.pl` script in the Oracle Grid Infrastructure home for Oracle Restart, and the `rootcrs.sh` script replaces the `rootcrs.pl` script in the Grid home for Oracle Grid Infrastructure for a cluster.

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] [-help]
```

Parameters

Parameter	Description
-silent	Use this flag to run <code>deinstall</code> 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 <code>-silent</code> 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 the <code>deinstall</code> command with the <code>-checkonly</code> flag. <code>deinstall</code> then discovers information from the Oracle home to deinstall and deconfigure. It generates the response file that you can then use with the <code>-silent</code> option. You can also modify the template file <code>deinstall.rsp.tmpl</code>, located in the <code>\$ORACLE_HOME/deinstall/response</code> directory.
-checkonly	Use this flag to check the status of the Oracle software home configuration. Running <code>deinstall</code> with the <code>-checkonly</code> flag does not remove the Oracle configuration. The <code>-checkonly</code> flag generates a response file that you can use with the <code>deinstall</code> command and <code>-silent</code> option.

Parameter	Description
<code>-paramfile</code> <i>complete path of input response file</i>	Use this flag to run <code>deinstall</code> 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 <code>\$ORACLE_HOME/deinstall/response</code> .
<code>-params</code> [<i>name1=value name2=value name3=value . . .</i>]	Use this flag with a response file to override one or more values to change in a response file you have created.
<code>-o</code> <i>complete path of directory for saving response files</i>	Use this flag to provide a path other than the default location where the response file (<code>deinstall.rsp.tpl</code>) is saved. The default location of the response file is <code>\$ORACLE_HOME/deinstall/response</code> .
<code>-tmpdir</code> <i>complete path of temporary directory to use</i>	Use this flag to specify a non-default location where <code>deinstall</code> writes the temporary files for the deinstallation.
<code>-logdir</code> <i>complete path of log directory to use</i>	Use this flag to specify a non-default location where <code>deinstall</code> writes the log files for the deinstallation.
<code>-local</code>	Use this flag on a multinode environment to deinstall Oracle software in a cluster. When you run <code>deinstall</code> with this flag, it deconfigures and deinstalls the Oracle software on the local node (the node where <code>deinstall</code> is run). On remote nodes, it deconfigures Oracle software, but does not deinstall the Oracle software.
<code>-help</code>	Use this option to obtain additional information about the command option flags.

Deinstallation Examples for Oracle Database Client

Use these examples to help you understand how to run the `deinstall` command.

You can run `deinstall` from the `$ORACLE_HOME/deinstall` directory. The deinstallation starts without prompting you for the Oracle home path.

```
$ ./deinstall
```

If you have a response file, then use the optional flag `-paramfile` to provide a path to the response file.

You can generate a deinstallation response file by running the `deinstall` command with the `-checkonly` flag. Alternatively, you can use the response file template located at `$ORACLE_HOME/deinstall/response/deinstall.rsp.tpl`.

In the following example, the `deinstall` command is in the path `/u01/app/oracle/product/19.0.0/client_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/19.0.0/client_1/deinstall
$ ./deinstall -paramfile /home/usr/oracle/my_db_paramfile.tmpl
```

A

Response Files

Review the following topics to install and configure Oracle Database, Oracle Grid Infrastructure, and other Oracle products using response files.

- [How Response Files Work](#)
Response files can assist you with installing an Oracle product multiple times on multiple computers.
- [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](#)
Review this information to use response files.
- [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.

How Response Files Work

Response files can assist you with installing an Oracle product multiple times on multiple computers.

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/19.0.0/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

Review this information to use response files.

Use the following general steps 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](#)
For Oracle Database Client, response files are located in the \$ORACLE_HOME/response directory.
- [Recording Response Files](#)
You can use Oracle Universal Installer (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

For Oracle Database Client, response files are located in the \$ORACLE_HOME/response directory.

All response file templates contain comment entries, sample formats, examples, and other useful instructions. Please read these instructions as they help you specify values for the variables listed in the response files and customize your installation.

The following table lists the response files provided with this software:

Table A-1 Response Files for Oracle Database Client

Response File	Description
client_install.rsp	Silent installation of Oracle Database Client
netca.rsp	Silent configuration of Oracle Net using Oracle Net Configuration Assistant.

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:

```
$ cp /directory_path/inventory/response/response_file.rsp local_directory
```

In this example, *directory_path* is the path of the directory where you have copied the installation binaries.

2. Open the response file in a text editor:

```
$ vi /local_dir/response_file.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/response_file.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 Client 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 Oracle Universal Installer (OUI) in interactive mode to record response files, which you can then edit and use to complete silent mode or response file mode 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.

To record a response file:

1. Complete preinstallation tasks for an Oracle Database Client 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 will specify when you run the installer.
3. On each installation screen, specify the required information.
4. When OUI 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 executable, `runInstaller`, provides several options. For help information on the full set of these options, run the `runInstaller` command with the `-help` option. For example:

```
$ directory_path/runInstaller -help
```

The help information appears in a window after some time.

To run the installer using a response file:

1. Complete the preinstallation tasks as 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:

```
$ /directory_path/runInstaller [-silent] [-noconfig] \  
-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:

- `directory_path` is the path of the directory where you have copied the installation binaries.
- `-silent` runs the installer in silent mode.
- `-noconfig` suppresses running the configuration assistants during installation, and a software-only installation is performed instead.

- *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_path/root.sh
```

Index

Numerics

32-bit client software, [6](#)

B

Bash shell

default user startup file, [5](#)

Bourne shell

default user startup file, [5](#)

C

C shell

default user startup file, [5](#)

CDBs

character sets, [3](#)

central inventory

See Oracle inventory directory

character sets, [3](#)

checklists

and installation planning, [1](#)

command syntax conventions, [ii](#)

commands

df -h, [2](#)

free, [2](#)

grep MemTotal, [2](#)

grep SwapTotal, [2](#)

root.sh, [2](#)

umask, [4](#)

uname —m, [2](#)

useradd, [4](#)

cron jobs, [6](#)

D

dbca.rsp file, [A-3](#)

default file mode creation mask

setting, [4](#)

deinstall, [1](#)

See also removing Oracle software

deinstall command, [1](#)

deinstallation, [1](#)

examples, [4](#)

df command, [5](#)

display variable, [5](#)

E

editing shell startup file, [5](#)

enterprise.rsp file, [A-3](#)

errors

X11 forwarding, [7](#)

F

file mode creation mask

setting, [4](#)

files

bash_profile, [5](#)

dbca.rsp, [A-3](#)

enterprise.rsp, [A-3](#)

login, [5](#)

profile, [5](#)

response files, [A-2](#)

filesets, [2](#)

G

globalization, [6](#)

localization for client connections, [3](#)

NLS_LANG

and client connections, [3](#)

groups

creating an Oracle Inventory Group, [2](#)

OINSTALL group, [3](#)

H

hardware requirements, [1, 3](#)

display, [1, 3](#)

I

image

install, [1](#)

installation, [6](#)

response files, [A-2](#)

preparing, [A-2, A-4](#)

templates, [A-2](#)

silent mode, [A-5](#)

installation planning, [1](#)

installer
supported languages, [4](#)

J

JDK requirements, [2](#)

K

Korn shell
default user startup file, [5](#)

L

licensing, [6](#)

M

mask
setting default file mode creation mask, [4](#)
mixed binaries, [2](#)
mode
setting default file mode creation mask, [4](#)
multitenant container database
character sets, [3](#)

N

netca.rsp file, [A-3](#)
noninteractive mode
See response file mode

O

oinstall group
creating, [2](#)
OINSTALL group, [5](#)
See also Oracle Inventory directory
operating system
different on cluster members, [2](#)
requirements, [2](#)
operating system privileges groups, [5](#)
operating system requirements, [2](#)
ORAchk
and Upgrade Readiness Assessment, [6](#)
Oracle Database Client
image-based installation, [6](#)
installation, [1](#)
Oracle Database Client, installation, [5, 6](#)
Oracle Database Client, relinking, [7](#)
Oracle Database Configuration Assistant
response file, [A-3](#)
Oracle home
ASCII path restriction for, [3](#)

Oracle Instant Client
about, [1](#)
about SDK, [17](#)
environment variables, [16](#)
installation, [1, 8, 12](#)
Instant Client Basic Light, [13](#)
Instant client light, [13](#)
patching, [18, 19](#)
regenerating libraries, [19, 20](#)
RPM installation, [9](#)
shared libraries, [14](#)
ZIP files installation, [8](#)
Oracle Instant Client Basic Light
Client Light, [13](#)
ZIP files installation, [14](#)
Oracle Inventory, [5](#)
identifying existing, [2](#)
Oracle Net Configuration Assistant
response file, [A-3](#)
Oracle Net Configuration Assistant, installation, [7](#)
Oracle Software Owner user
creating, [3, 4](#)
Oracle Software Owner users, [5](#)
Oracle Universal Installer
response files
list of, [A-3](#)
oracle user, [5](#)
creating, [3](#)
OSDBA, [5](#)
OTN website
downloading installation software from, [2](#)

P

postinstallation
recommended tasks
root.sh script, backing up, [2](#)

R

removing Oracle software, [1](#)
examples, [4](#)
response file installation
preparing, [A-2](#)
response files
templates, [A-2](#)
silent mode, [A-5](#)
response file mode, [A-1](#)
about, [A-1](#)
reasons for using, [A-2](#)
See also response files, silent mode
response files, [A-1](#)
about, [A-1](#)
creating with template, [A-3](#)
dbca.rsp, [A-3](#)
enterprise.rsp, [A-3](#)

response files (*continued*)
 general procedure, [A-2](#)
 netca.rsp, [A-3](#)
 passing values at command line, [A-1](#)
 specifying with Oracle Universal Installer, [A-5](#)
 See also silent mode.
root user
 logging in as, [1](#)
root.sh script
 backing up, [2](#)
rootcrs.sh, [1](#)
roothas.sh, [1](#)

S

silent mode
 about, [A-1](#)
 reasons for using, [A-2](#)
silent mode installation, [A-5](#)
software requirements, [2](#)
ssh
 and X11 Forwarding, [7](#)
supported languages
 installer, [4](#)
swap space
 allocation, [3](#)
system requirements, [1](#)

T

troubleshooting
 cron jobs and installation, [6](#)
 disk space errors, [3](#)
 environment path errors, [3](#)
 installation owner environment variables and
 installation errors, [8](#)
 unset environment variables, [3](#)

U

umask command, [4](#)
uninstall
 See removing Oracle software
UNIX commands
 xhost, [1](#)
UNIX workstation
 installing from, [1](#)
unset installation owners environment variables, [8](#)
upgrading
 and ORAchk Upgrade Readiness
 Assessment, [6](#)
useradd command, [4](#)
users
 creating the oracle user, [3](#)

X

X Window System
 enabling remote hosts, [1](#)
X11 forwarding errors, [7](#)
xhost command, [1](#)