Redbooks

ibm.com/redbooks

Oracle 19c on IBM LinuxONE

Sam Amsavelu

Armelle Cheve

Phillippe Deverge

Wayne Martin

David Simpson

LinuxONE

IBM

Redbooks

IBM Redbooks

Oracle 19c on IBM LinuxONE

November 2021

SG24-8487-00

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

First Edition (November 2021)

Contents

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

Db2®	Rational®	WebSphere®
FICON®	Redbooks®	z/Architecture®
IBM®	Redbooks (logo)  ®	z/OS®
IBM Spectrum®	System z®	z/VM®
IBM Z®	Think®	z15™

The following terms are trademarks of other companies:

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Red Hat, are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM® Redbooks® publication helps you install, tailor, and configure Oracle 19c on the IBM LinuxONE platform.

Authors

This book was produced by a team of specialists from around the world working at IBM Redbooks, Poughkeepsie Center.

Sam Amsavelu has worked at IBM® for the last 25 years, advising customers to build highly available, reliable, and secure infrastructure architectures for their databases. He is an expert in implementing virtualization, server consolidation, and cloud technologies to help customers choose the right platform and technologies for an optimized environment with the lowest total cost of ownership possible. He also is a SME in several operating systems (IBM z/OS®, IBM z/VM®, UNIX, and multiple Linux distributions), relational databases (IBM Db2®, Oracle, PostgreSQL, and SQLServer), NoSQL databases (MongoDB), and HADRsolutions (Oracle RAC, DataGuard, GoldenGate, replication, and sharding).

Armelle Cheve is an IT Economics Practitioner within the IBM IT Economics Team in EMEA. Armelle has three decades of experience in the IT industry, supporting the Mainframe since 1998. She is an Oracle solution on Linux on Z and LinuxONE SME. She has provided technical support on this solution since 2012 and more recently, Economic Assessments.

Philippe Deverge has 18 years of experience as an IT Architect in Montpellier Garage for Systems, working with ISVs and customers. In 2002, he joined IBM at the MOP System Center as a IBM Power Architect and performance expert. He has experience in designing, sizing, and benchmarking with ISVs solutions, such as SAP and Oracle for customers. He works in the Europe ATS Garage for Systems as a certified Infrastructure Architect for Oracle on IBM Z® and LinuxONE Systems.

Wayne Martin is the IBM Systems Solution Manager responsible for the technology relationship between IBM and Oracle Corporation for all IBM server brands. He is responsible for developing the mutual understanding between IBM and Oracle on technology innovations that generate benefits for mutual customers. Wayne has held various technical and management roles at IBM that focused on driving enhancements of ISV software that use IBM's mainframe, workstation, and scalable parallel products.

David Simpson is an Oracle Certified Database Administrator with IBM Linux on Z and LinuxONE. David has authored several IBM Redbooks® publications and has presented at numerous user conferences, including Oracle OpenWorld. David has assisted many IBM customers with implementing Oracle database solutions with IBM hardware and IBM Hybrid Cloud solutions.

Thanks to the following people for their contributions to this project:

William White, IBM Redbooks
Poughkeepsie Center

Susan Adamovich
IBM US

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, IBM Redbooks
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

  http://www.redbooks.ibm.com/rss.html

1

A shared commitment to client value with the IBM and Oracle alliance

Since 1986, Oracle and IBM have been providing clients with compelling joint solutions, combining Oracle's technology and application software with IBM's complementary hardware, software and services solutions. More than 80,000 joint clients benefit from the strength and stability of the Oracle and IBM alliance. Through this partnership, Oracle and IBM offer technology, applications, services and hardware solutions that are designed to mitigate risk, boost efficiency and lower total cost of ownership.

IBM is a Platinum level partner in the Oracle Partner Network, delivering the proven combination of industry insight, extensive real-world Oracle applications experience, deep technical skills and high-performance servers and storage to create a complete business solution with a defined return on investment. From application selection, purchase and implementation to upgrade and maintenance, we help organizations reduce the total cost of ownership and the complexity of managing their current and future applications environment while building a solid base for business growth.

This chapter includes the following topics:

1.1 Deploy resources flexibly and on-demand

Growth impacts your business. More clients, transactions, servers, mobile devices, networking, and databases can create significant challenges for any enterprise's infrastructure, resulting in unnecessary complexity, inefficiency, and cost. IBM and Oracle provide a premier enterprise infrastructure that is simple, secure, scalable, and cost-effective in supporting your transformation in the digital economy.

1.2 Solution overview

Enterprises require a trusted IT infrastructure that is dynamic, scalable, and flexible to support both mission-critical work and the development and deployment of new workloads. Coupled with innovative solutions from Oracle, the IBM LinuxONE III and the IBM z15™ share the IBM z/Architecture® and therefore execute identical versions of the Linux distributions, and Oracle product releases. While the balance of this book will refer to the LinuxONE III, the same solutions are supported on the z15 when it is configured to run Linux.

The LinuxOne III delivers advanced performance, security, resiliency, availability, and extreme virtualization to provide high levels of qualities of service. With up to 190 client configurable cores and 40 terabytes (TB) of available Redundant Array of Independent Memory (RAIM) real memory, SMT and now packaged in a 19-inch frame the LinuxONE III LT1 is ideally suited for consolidating large-scale distributed environments and for new in-memory and Java workloads. The LinuxOne III LT2, the newest member of LinuxONE family, is a single- frame system in a 19-inch industry standard rack allowing it to sit side-by-side with any other platform in a data center. Both LinuxONE III support SMT, up to 65 configurable cores and 16 terabytes (TB) of available Redundant Array of Independent Memory (RAIM) real memory per server. This new entry model is ideal for any growing business that seeks to use Z technologies' qualities of service, flexibility and performance.

1.3 Linux for the IBM Z family overview

Linux is available on various computing platforms, from set top boxes and hand held devices to the largest servers. After your applications are running on Linux, you are no longer tied to a specific hardware platform. You control the choice of hardware platform that supports your application. Workloads that are running on the IBM LinuxONE platform benefit from hardware that includes specialized processors, cryptographic cards with dedicated RISC processors, and a combination of hypervisors that allow unparalleled flexibility in Linux deployment. A major benefit of Linux is that it is open source. Hundreds of Linux distributions are available for almost every computing platform. The following enterprise distributions[1] of Linux are tested on IBM LinuxONE servers (Red Hat and SUSE are the certified and supported Linux distributions for deploying Oracle solutions on the IBM LinuxONE platform):

- Red Hat: Red Hat Enterprise Linux (RHEL)
- SUSE: SUSE Linux Enterprise Server
- Canonical: Ubuntu Server

These Linux distributions provide customers who are using Linux with various support options, including 24 x 7 support with one-hour response time worldwide for customers who are running production systems. Including the Linux operating system, all of the major Linux distributions offer several other open source products that they also fully support.

The IBM LinuxONE platform is a portfolio of hardware, software, and solutions for an enterprise-grade Linux environment. It is designed to run more transactions faster and with more security and reliability specifically for the open community. It fully embraces open source-based technology. The following are some key points to consider that tie to the value of deploying Oracle on IBM LinuxONE servers:

- Scale and grow Oracle Database 19c applications and data with confidence
- Simplify IT operations with advanced virtualization, multi-architecture workload support and operations management capabilities
- Minimize network security vulnerabilities, and reduce latency

1.4 How it works

IBM LinuxONE servers can help deliver advanced performance, security, resiliency, availability, and virtualization for a high quality of service. Additionally, they offer massive scalability for secure data serving, high-volume transaction processing, and large-scale consolidation.

A Linux distribution is a complete operating system and environment including compilers, file systems, and applications, such as Apache (Web server), SAMBA (file and print), Sendmail (mail server), Tomcat (Java application server), and MySQL (database)

The IBM LinuxONE platform has the world's fastest commercially available processor. Built for speed and scalability, Z servers support Simultaneous Multi-Threading (SMT) for Linux and Java workloads. These features help deliver outstanding transaction processing and data serving throughput performance. Leveraging these capabilities allows LinuxONE run average sustained CPU utilization of 80 to 90%, greatly exceeding that of other commodity based distributed systems and making them ideal for consolidating large-scale distributed environments and the introduction of new in-memory workloads.

LinuxONE can support up to thousands of virtual Linux servers on a single footprint. This means that the virtualization capabilities in a single Z server can result in a less complex Linux infrastructure with fewer components, less management, less space requirements, and optimized core workload efficiency.

For more information about the value of deploying Oracle Database Release 19c on IBM Z, see the *Oracle Database 19c and Oracle RAC 19c on IBM Z and IBM LinuxONE* and the solution data sheet *IBM LinuxONE servers running Oracle Database 19c on premises on Linux*.

1.5 Competitive advantage

Oracle and IBM experts work together to provide design, development, testing, technical support, and education to support your Oracle solutions that are running Linux on Z. These resources are provided to help ensure your solution performance and provide training to your IT staff.

For more information about how IBM LinuxONE servers provide the perfect platform for running Oracle and Open Source mission-critical Linux applications for fast deployment and cost savings, see the IBM website.

2

Oracle 19c database new features

Oracle 19c is the latest database version from Oracle with many new database features and enhancements. Oracle 19c is the last or terminal release of Oracle 12c available from Oracle. It was released for IBM LinuxONE in early 2019 along with other supported platforms.

Some of the new Oracle 19c database features include Oracle Autonomous Indexing, Data Guard DML redirect, Oracle SQL quarantine and achieving higher Oracle availability with Oracle Database release 19c.

This chapter discusses some of these new database features that are available in Oracle Release 19c and how they might be used with the IBM LinuxONE (Linux on System z®) platform.

This chapter includes the following topics:

2.1 Database stability

One of the most important new features of Oracle Database release 19c is stability, which is critical for LinuxONE customers that run mission critical workloads on this platform.

Oracle 19c is final release for Oracle 12c, which means it will have the longest service time of any database release. Currently, patching is available into 2026 for Oracle 19c.

Note: These support projections can be changed by Oracle at any time at their discretion. See Oracle support note for the latest information: Release Schedule of Current Database Releases [742060.1]

The availability a system that accept the latest security and bug fix patches is paramount for database stability and is why it is recommended for customers to start planning the upgrade to this release as soon as possible.

2.2 Autonomous automatic indexing

Another new feature that is available with Oracle 19c is automatic indexing as part of Oracle's 19c autonomous database capabilities:

- Oracle's 19c new automatic indexing capability
- Oracle on LinuxONE can dynamically allocate resources to improve performance
- Indexes can be created automatically (autonomous)

2.3 Data-guard DML redirect

Many LinuxONE customers use Oracle Data Guard for High Availability for local failover and their site disaster scenarios.

Oracle's new 19c Active Data Guard DML Redirection feature now allows for incidental Data Manipulation Language (DML) statements to be issued on standby databases. This means **`INSERT`**, **`UPDATE`**, **`DELETE`** and **`MERGE`** statements can be run directly on the standby database now, when the new 19c **`ADG_REDIRECT_DML`** database parameter is set to `true`.

The process includes the following steps:

1. A DML SQL statement executes against an open standby database.
2. The standby database redirects the statement to the primary database.
3. The DML statement is applied to the primary database.
4. That data change is streamed back to the standby database whereby the data is then visible to the standby database client when the transaction is applied to the standby database.

One business use of Data Guard DML Redirect is with auditing reporting. Audit compliance can be maintained by inserting a record into an audit table when running reporting operations on a standby database when a sensitive report is run by user on the standby database.

2.4 Real-time stats and stats-only queries

LinuxONE Oracle 19c databases benefit from more efficient SQL queries in real time by virtue of being a virtualized platform. Reducing CPU with more efficient SQL queries reduces CPU peaks on the system, which gains more efficiency across the system for other workloads by using fewer overall system resources.

For some workloads that insert and delete a lot of data, database statistics can go stale between running DBMS_STATS gathering jobs. Database tables with frequent changes in the number of rows can occur with batch processing jobs, for example.

With real-time statistics, tables with a high rate of change between statistics gathering can help the optimizer generate optimal plans for these frequently changing tables. With Oracle Real-time statistics, new statistics can be generated with a new SQL plan and run if a better SQL execution plan is available.

Real-time statistics runs are generated when Oracle performs a hard parse of a SQL statement. The optimizer now detects that rows were added to the table, and regathers statistics for the table to ensure the best plan is being used.

Real-time statistics are indicated by `STATS_ON_CONVENTIONAL_DML` statement in the Notes section of the `*_TAB_STATISTICS`, or the `*_TAB_COL_STATISTICS` database view tables (`*=ALL`, `DBA` or `USER`).

2.5 Partial JSON update support

LinuxONE provides great scale up capabilities for Oracle workloads, with the ability to dynamically to add CPU processors or memory to a Linux operating system dynamically. JavaScript Object Notation (JSON) is a lightweight data-interchange format that open-source technologies use for simplicity in JavaScript.

New in Oracle 19c is the **`JSON_MERGEPATCH`** function, which is used to modify parts of a JSON document in select and update operations. Previously with JSON, you retrieved the JSON document, processed the contents, and replaced the entire document with the modified document.

This new **`JSON_MERGEPATCH`** function can simplify the process significantly for some scenarios. You can update JSON document declaratively by using SQL function **`JSON_MERGEPATCH`**, as shown in the following example:

```
JSON_MERGEPATCH ( target_expr , patch_expr [ returning_clause ] [ PRETTY ] [ ASCII
]  [ TRUNCATE ] [ on_error_clause ] )
```

2.6 Schema-only Oracle accounts

LinuxONE features one of the highest security rating (EAL 4+) for commercially available systems running Oracle 19x for independence between Linux guest systems running virtualized between Logical Partitions (LPARs) and Linux guest operating systems running virtualized.

Oracle 19c increased the security of the database in 19c with schema-only user accounts. Unused and rarely accessed database user accounts with administrative privileges can now become schema (password less) only accounts to help make the database more secure by have less access points to the database.

Password-less accounts are typically used for applications that include many schemas, with one schema that is shared between the schemas for administration. Available are database procedures, functions and grants, and synonyms, such as the APPS schema in E-business suite applications. Schema-only user accounts help prevents administrators from having to manage the passwords on these accounts making the system more secure for your business.

2.7 Automated installation, configuration, and patching

LinuxONE runs Clusterware with the ability to run multiple technologies for the Oracle RAC interconnect network, such as Open System Adapter (OSA) network cards at 1, 10, or 25 Gbps, shared Memory Communications over RDMA adapter (RoCE) by using IP, or Hipersockets memory-to-memory networking.

Oracle 19c introduces several new features for cloud automation, such as root scripts automation, dry-run validation for Oracle Clusterware upgrades, and dbca duplication of an database in silent mode.

These new scripted automation features help make implementing a hybrid multi-cloud solution with IBM LinuxONE and Oracle 19c that much easier.

2.8 Support direct file placement of OCR and voting disks

Previously in Oracle 12.2, the placement of OCR and voting files directly on a shared file system was not supported. This issue was rescinded in 19c. Now LinuxONE Oracle 19c customers can place OCR and voting files directly on shared file systems, such as IBM Spectrum® Scale.

Having OCR and Voting files, which are required only for Oracle RAC systems, makes backing up these files easier to see and maintain by virtue of being on a file system.

2.9 Zero downtime Oracle Grid infrastructure patching

LinuxONE has one of the lowest percent downtime of any commercially available system, according to the latest (2019) ITIC Global Server Hardware, Server OS Reliability Surveys.

Increase database availability by patching Grid Infrastructure without interrupting database operations on the node being patched with zero downtime compliments well with IBM LinuxONE systems for mission critical or highly virtualized workloads.

Zero Downtime Oracle Grid Infrastructure Patching increases database availability by allowing DBAs to perform a rolling patch of Oracle Grid Infrastructure without interrupting database operations.

2.10 Oracle SQL quarantine

In a LinuxONE in a shared virtual environment, controlling run away ad-hoc user requests is beneficial to all users on the system.

Oracle Resource Manager limits resource consumption (CPU, IO) consuming SQL and is available in 11gR2 and 12c. Nothing prevented that same query from being repeatedly being executed again, which effected system performance.

New in Oracle 19c, SQL statements that are stopped by Oracle Database Resource Manager because of excessive use of CPU and I/O resources can be automatically quarantined. Execution plans that are associated with stopped SQL statements can be quarantined to prevent them from being run again.

This feature protects an Oracle Database from performance degradation by preventing execution of SQL statements that excessively use CPU and I/O resources.

You can use the DBMS_SQLQ package to create a quarantine configuration for an execution plan of a SQL statement, which must be quarantined, as shown in the following example:

```
DECLARE
quarantine_q    VARCHAR2(30);
BEGIN
quarantine_q:= DBMS_SQLQ.CREATE_QUARANTINE_BY_SQL_ID(SQL_ID => ' 9agxh488f3v88
');
END;
```

3

Setting up Linux to install Oracle Database

This chapter describes the preinstallation setup steps that are required for setting up Oracle 19c on an IBM LinuxONE system. We include information about how to obtain Oracle documentation, code, and My Oracle Support (MOS) Notes.

For product longevity and patching, IBM and Oracle recommend upgrading to Oracle 19c, which is the long-term release with the longest support end date. Oracle 19c is the terminal release of Oracle 12.2.

We also describe setting up and configuring Red Hat Enterprise Linux Servers and SUSE Linux Enterprise Servers.

This chapter includes the following topics:

3.1 Obtaining Oracle documentation, downloads, and My Oracle Support Notes

Changes in the recommended Oracle RPMs and kernel parameters can occur between release levels. We include the provided references throughout this chapter, with the corresponding MOS reference note for the most up to date guidance on a particular topic and verification.

3.1.1 Master Note of Linux OS requirements for database server

The Master Note is intended to provide an index and references to the most frequently used MOS articles concerning Linux OS requirements for an Oracle Database software installation.

For more information about the master note, see MOS Doc ID 851598.1, which is available by logging in to your Oracle Support account. The following key notes for Oracle 19c installation with Linux on System z are available:

- For more information about installing Red Hat 7 Oracle 19c, see Requirements for Installing Oracle 19c RDBMS on RH7 on IBM: Linux on System z (Doc ID 2628892.1).
- For more information about installing SUSE 12 Oracle 19c, see Requirements for Installing Oracle 19c on SUSE Linux Enterprise Server 12 on IBM: Linux on System z (Doc ID 2628889.1).

Oracle 19c is certified by Oracle for Red Hat 7.4 or greater and SUSE 12 Update 3 or greater. Oracle is certified to the base Major OS level. All service packs that are above the minimum required distribution level are supported by Oracle.

The latest available Red Hat 7.x or SUSE 12.x distribution service pack level is recommended for new installations to have the most up-to-date features and security functionality for running Oracle databases.

The Release Schedule of Current Database Releases (Doc ID 742060.1) is available by logging in to your Oracle Support account. Use the My Oracle Support Certification tab on the MOS site to determine the latest release status of Oracle version releases with Linux on System z.

3.1.2 Obtaining the Oracle 19c Software for IBM LinuxONE

Depending on your company's Oracle licensing agreement, the latest Oracle code can be downloaded from the Oracle Software Delivery cloud or from the Oracle Technology Network (OTN) developers download site.

If you have a commercial Oracle license, download your software from Oracle Software Delivery Cloud.

For trial and developer network downloads, the Oracle License agreement must be fully understood before agreeing and downloading the Oracle 19c database software from the OTN download site.

After you are signed into the Oracle Software Delivery cloud by using your Oracle user ID and password, enter `Oracle Database` in the search field for the Oracle Enterprise Edition that you want to install, as shown in Figure 3-1 on page 13.

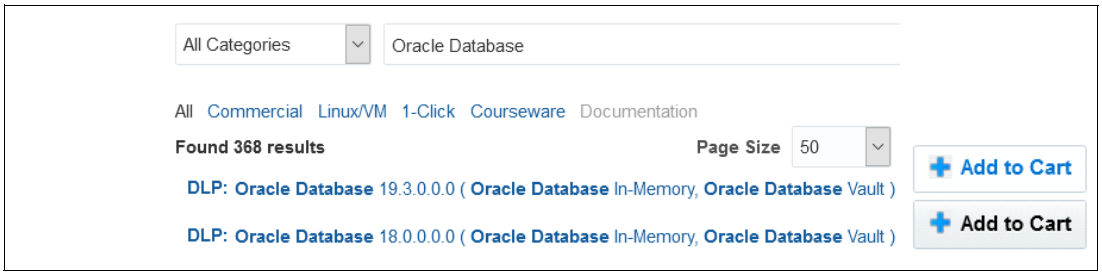

Figure 3-1 Oracle Software Delivery Cloud Download

Click the "**+**" sign of the **Add to Cart** button for the software version that you want to install, and then, click **View Cart** in the upper right of the window. An option to checkout then is available. Click **Checkout**.

In the software catalog for the selected software, select **IBM: Linux on System z**, as shown in Figure 3-2. Then, select the software components that you need for your installation under the Platforms / Languages column.

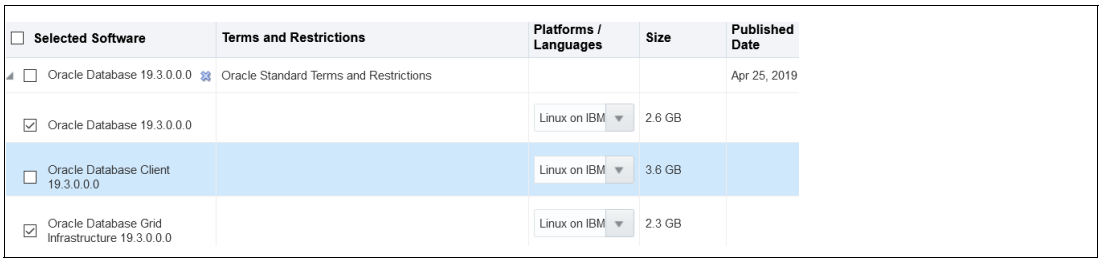

Figure 3-2 Oracle Software Delivery Cloud Select Software components

Most installations require the Oracle Database software to be download. If you use Oracle ASM, or are installing Oracle RAC or Oracle RACONE, you also need the Oracle Database Grid Infrastructure software.

For applications that must connect to an Oracle Database that is running on Linux on System z, if your application runs on Linux on System z by using sql*net network connectivity, the Oracle Database Client also must be downloaded.

After all of the software is selected, click **Download** to start downloading the software. If this download is the first time you are downloading software, you might be prompted to download and install the Oracle download installer first. After the installer is configured, the selected Oracle distribution software is downloaded to your workstation. You also can download by clicking the link for the software product individually.

In addition to the base software, the latest Patch Set or Release Update must be downloaded from Oracle Support.

Oracle 19c provides the latest Release Update (RU) for the Grid Infrastructure and Database from the Oracle support site. The latest RU for your release can be found in Oracle Note Master Note for Database Proactive Patch Program (Doc I 756671.1).

An Oracle 19c base installation is release 19.3 for Linux on System Z. It is recommended to be on the latest Grid Infrastructure and Database RU when the patches are made available every fiscal quarter.

The latest (as of September 2022) Oracle 19.16 Grid Infrastructure and Database security and RU patch is 34133642. Follow the readme file for the patch you plan to install. Make sure to download the latest OPatch installer utility patch 6880880 as well for the RU patch as described in the patch readme file.

3.2 Red Hat Enterprise Linux and SUSE Enterprise Linux Server setup

This section describes the common steps that are needed to implement an Oracle Database, including Oracle Grid Infrastructure on SUSE Enterprise Server 12 or Red Hat Enterprise 7 on LinuxONE.

3.2.1 Important information

In this section, the following preinstallation verification information is provided to give you a basic understanding before starting an Oracle Database installation on LinuxONE:

- Graphical user interface (GUI)

  A GUI is required to run the Oracle Installers interactively. We use the Linux `vncserver` on LinuxONE to start an X server and a VNC viewer client to establish a terminal emulator xterm session.

  To use GUIs, such as the Oracle Universal Installer (OUI), configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

  In our environment, we start a `vncserver` on LinuxONE, and use a VNC viewer client to establish the connection with the `vncserver`.

  You also can configure silent or scripted installations for cloud automation deployments.

- Linux minimal installation

  A minimal Linux installation lacks many RPMs that are required for database installation. Therefore, you must use an RPM package for Linux release to install the required packages.

- Automatic Storage Management

  To use Oracle's Automatic Storage Management (ASM), you must first install Oracle Grid Infrastructure for a stand-alone server before you install and create the database.

  Most customers use Oracle ASM to store database files or a file system for single instance installations. For Oracle Real Application Cluster (RAC) installations, Oracle ASM or IBM Spectrum scale can be used to store database files across Oracle RAC cluster nodes.

  To use Oracle ASM on LinuxONE, you must first install the Oracle Grid Infrastructure for a stand-alone server before you can install and create the database.

  Although the SUSE `orarun` RPM package can be used to configure the user or groups, `ulimits`, and kernel parms automatically, our experiences are based on manually configuring these settings for greater control.

3.2.2 General requirements and information for Oracle Database on LinuxONE

To obtain general LinuxONE information from a Linux session, run the following commands, ensuring that the system was started with a run level 3 or 5:

- Check information about the CPU that is defined in LinuxONE:

  ```
  cat /proc/cpuinfo
  ```

- Check information about the RAM that is defined in LinuxONE:

  ```
  cat /proc/meminfo | grep MemTotal
  cat /proc/meminfo | grep SwapTotal
  ```

- Check the amount of space that is available in the `/tmp` directory:

  ```
  df -h /tmp
  ```

- Check the amount of free disk space on the system (in GigaBytes GB).

  ```
  df -h
  ```

- Check the amount of free RAM and disk swap space on the system:

  ```
  free
  ```

- Check whether the system architecture can run the software:

  ```
  uname -m
  ```

  This command should return: `s390x`.

3.2.3 Server minimum requirements

The following minimum requirements for the server must be met:

- Disk Space Requirements for Oracle 19c on IBM LinuxONE

  Ensure that the system meets the disk space requirements for software files:

  - Enterprise Edition database: A total of 7.5 GB - 8.6 GB of disk space for the Oracle software, depending on the installation type. You might need extra space for patching logs. A minimum of 10 GB of disk space is recommended.
  - Grid Infrastructure: A total of 8.9 - 10GB is minimally required for installing Oracle ASM single instance (restart) or an Oracle RAC install. It is also recommended to provide more space for Oracle grid infrastructure log files. A minimum of 20 GB of disk space is recommended.

- Recommended memory

  A minimum of 4 GB of virtual memory is recommended in the Linux guest for Oracle 19c installations of Oracle Database, including the Grid Infrastructure requirements.

  For single instance file system-based installations, 2 GB minimum virtual memory is recommended.

  Note: Although 2 GB is the minimum amount of virtual memory that is required, the Oracle OPatch utility requires at least 3072 MB (3 GB) for Oracle patching.

- Swap

  Swap disk space is proportional to the system's physical memory (as listed in Table 3-1) and recommended by Oracle for general Linux swap.

Table 3-1 Swap size recommendation

RAM	Swap space
1 - 8 GB	Twice the size of RAM
8 - 32 GB	Equal to the size of RAM
More than 32 GB	32 GB

The Oracle guidelines for Linux swap can be reduced (if needed) when enough memory is available to run all the databases in the Linux guest. The Oracle Installer requires a minimum 500 MB of configured Linux swap to complete an installation and 1 GB of swap for database upgrades.

Customers that use IBM LinuxONE can use a layered virtual in-memory disk or VDisk for the Linux swap devices. Linux swap to a memory device (VDisk) is much quicker than using a physical disk storage device. z/VM does not allocate the memory that is used by a VDisk until the first swap page is "Used".

Figure 3-3 shows an example of a recommended VDisk configuration with a VDisk being used for the first and second Level Swap with higher priority. Then, a physical disk or DASD disk can be used as a lower priority swap in the case of unexpected memory usage. Linux use, the higher priority swap devices first. When the swap device is fully exhausted, the next priority swap device is used next.

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdo1                             partition       131000  0       10
/dev/dasdp1                             partition       524216  0       5
/dev/mapper/u603_swap3                  partition       6291448 0       1
```

Figure 3-3 Swap VDisk Configuration Priorities

The Linux **mkswap** and **swapon** commands can be used to configure swap manually, and the `/etc/fstab` file can be used to configure permanent swap devices, as shown in Figure 3-4. Swap usage is used with the highest priority (`pri=`) number. The default is `-1` for swap usage priority.

```
oratest:~ # cat /etc/fstab | grep swap
/dev/disk/by-path/ccw-0.0.0300-part1                    swap       swap pri=10          0 0
oratest:~ # swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasda1                             partition       215900  0       10
oratest:~ # lsdasd | grep 0.0.0300
0.0.0300   active      dasda      94:0    ECKD  4096   210MB     54000
```

Figure 3-4 Swap VDisk setup in fstab

- Temporary disk space required for installation

  A minimum of 1 GB of disk space in the `/tmp` directory on LinuxONE systems is required (and less than 2TB of disk space). If you do not have enough space in `/tmp` for an Oracle installation, you can delete any files that you can or you can set the `TMP` and `TMPDIR` environment variables to another directory with space when setting the `oracle` user's environment.

- Linux kernel parameters

  Verify that the required operating system kernel parameters are installed.

  An example of the required Oracle kernel parameters is shown in Table 3-2 for a system with an 8 GB Oracle System Global Area (SGA) database.

  If the current value of any parameter is higher than the value listed in Table 3-2, do not change the value of that parameter. These parameters must be configured in the Linux `/etc/sysctl.conf` file.

Table 3-2 Kernel parameters

Kernel parameter	Example value	File
`fs.file-max` - set to 512 x maximum number of user processes in Oracle per database, for the Linux Guest. 6815744 supports up to 13312 Oracle processes. Increase as needed.	`6815744`	`/proc/sys/fs/file-max`
`fs.aio-max-nr` - This value limits concurrent outstanding requests and should be set to avoid I/O subsystem failure. The formula is aio-max-nr =no of process per DB * no of databases * 4096	`1048576` or `3145728` for larger databases.	`/proc/sys/fs/aio-max-nr`
`kernl.shmmax` - Set this value to the size of at least the largest SGA in bytes. 8589934592 bytes will support a maximum shared memory segment of 8 GB. See MOS Note 567506.1 for more information.	`8589934592` (8 GB)	`/proc/sys/kernel/shmmax`
`kernel.shmall` - Set this parameter to a value that is equal to the total amount of shared memory in 4 K pages that the system can use at one time. The sum of all the SGA databases on the system in bytes / 4096. 20907152 supports 8 GB SGA. See MOS Note 301830.1 for more information.	`2097152`	`/proc/sys/kernel/shmall`
`kernel.sem` (**`semmsl`**, **`semmns`**, **`semopm`**, **`semmni`**) Minimum required semaphores required for Oracle.	`kernel.sem= 250 32000 100 128` `128`	`/proc/sys/kernel/sem`
`kernel.shmmni`	`4096`	`/proc/sys/kernel/shmmni`

Kernel parameter	Example value	File
kernel.panic_on_oops – setting to allow time for diagnostics in the event of system restart. 1 second on an Oracle node evic *n* seconds delay before a node eviction/reboot.		/proc/sys/kernel/panic_on_oops
net.ipv4.ip_local_port_range	9000 65500	/proc/sys/net/ipv4/ip_local_port_range
net.core.rmem_default	262144	/proc/sys/net/core/rmem_default
net.core.rmem_max	4194304	/proc/sys/net/core/rmem_max
net.core.wmem_default	262144	/proc/sys/net/core/wmem_default
net.core.wmem_max	1048576	/proc/sys/net/core/wmem_max
vm.swappiness - A value of 1 will prevent Linux swapping as much as possible.	1	/proc/sys/vm/swappiness

Optional performance-related kernel parameters are listed in Table 3-3.

Table 3-3 Optional performance related kernel parameters

Variable	Example value
vm.nr_hugepages Sum of the Oracle SGA's + 16 MB; for example, 8208 for 8 GB SGA memory for large pages.	8208 **Note:** Linux Guest memory should always be sized larger than this value or excessive Linux swap will occur.
vm.hugetlb_shm_group Set to oinstall group ID number from /etc/group file.	1001

Example 3-1 shows a sample sysctl.conf file. After kernel parameter values are updated, validate the kernel parameters by using the following **sysctl** command:

```
/sbin/sysctl -p
```

Output of the command is shown in Example 3-1.

Example 3-1 Sample Linux kernel parameters for Oracle

```
# Oracle specific parameters
fs.file-max = 6815744
fs.aio-max-nr = 1048576
#set shmmax to largest sga memory segment in bytes
kernel.shmmax = 8589934592
#set shmall to (sum of sga memory in bytes)/4096
kernel.shmall = 2097152
```

```
kernel.sem = 250 32000 100 128
kernel.shmmni = 4096
kernel.panic_on_oops = 1
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
vm.swappiness = 1
#set nr_hugepages to sum of all Oracle SGAs in MB if using large pages
#vm.nr_hugepages = 8208
vm.hugetlb_shm_group=1001
```

3.2.4 Creating the database installation owner user

The installation process requires at least an Oracle Database installation owner (`oracle`), an Oracle Inventory group (`oinstall`), and an Oracle administrative privileges group (*dba*).

Example 3-2 shows the commands that are used to create the `oracle` user and groups.

Example 3-2 Creating the Oracle user and groups

```
/usr/sbin/groupadd oinstall -g 1001
/usr/sbin/groupadd dba -g 1002
/usr/sbin/groupadd asmadmin -g 1003
/usr/sbin/groupadd asmoper -g 1004
/usr/sbin/useradd -m -g oinstall -G dba  oracle
echo "oracle:newPassw0rd" | /usr/sbin/chpasswd
```

For separation of duty, some sites set up a Linux `grid` user ID to manage the grid infrastructure ASM and Oracle Grid Infrastructure components. If this is the case, use the same useradd and change password steps that are shown in Example 3-7 on page 21 for the Linux grid user as well:

```
/usr/sbin/useradd -m -g oinstall -G dba grid
echo "grid:newPassw0rd" | /usr/sbin/chpasswd
```

3.2.5 User login security and limits configuration

To avoid operational problems with the Oracle Database using system resources, the Linux `ulimit` values must be adjusted to allow for the suitable operation of an Oracle Database.

The following settings must be verified before an installation is performed with the `/etc/security/limits.conf` file:

- If you are planning to use large pages and the Oracle SGA memory is increased, setting the `ulimit memlock` to `unlimited` is one less step that is needed to make that change:

  ```
  oracle soft memlock unlimited
  oracle hard memlock unlimited
  ```

- If you use another Oracle Linux user ID, such as `grid`, to manage the Grid Infrastructure components, define the same `ulimit` values for the `grid` Linux user and the `oracle` Linux user ID:

  ```
  grid soft memlock unlimited
  grid hard memlock unlimited
  ```

- If the Linux `oracle` user performs the installation, add the settings that are shown in Example 3-3 to your `/etc/security/limits.conf` file. If your current values are higher than these values, leave the higher value in place.

Example 3-3 Sample limits.conf file

```
oracle        soft    nproc   2047
oracle        hard    nproc   16384
oracle        soft    nofile  1024
oracle        hard    nofile  65536
oracle        soft    stack   10240
oracle        hard    stack   10240
#
grid          soft    nproc   2047
grid          hard    nproc   16384
grid          soft    nofile  1024
grid          hard    nofile  65536
grid          soft    stack   10240
grid          hard    stack   10240
# memlock for Huge Pages  support
oracle        soft    memlock  unlimited
oracle        hard    memlock  unlimited
grid          soft    memlock  unlimited
grid          hard    memlock  unlimited
```

Note: When the `limits.conf` file is changed, these changes take effect immediately. However, if the `oracle` users are logged in, these changes do not take effect until you log out these users and log them back in.

The `ulimit` take effect upon logging into a Linux guest by setting the `ulimit` values in each Linux users `.bash_profile` file or updating the system base `/etc/profile` for the system for all Linux users.

The values that are shown in Example 3-4 can be added to the `oracle` and `grid` (if used) user home directory `.bash_profile` file (see Example 3-4).

Example 3-4 Oracle user limits

```
ulimit -u 16384
ulimit -n 65536
ulimit -s 10240
```

3.2.6 Shared memory file system

For Oracle Database Installations that do not use Linux large pages but are using Automatic Memory Management (AMM) with the MEMORY_TARGET parameter, it is important to ensure that the `/dev/shm` mount area is of type `tmpfs` and is mounted with the following options:

- With RW (read/write) and execute permissions set on it
- With `noexec` or nosuid not set on it

Use the following command to check the shared memory file system:

```
cat /etc/fstab |grep tmpfs
```

The output for this command looks similar to following example:

```
tmpfs      /dev/shm     tmpfs      defaults      0 0
```

If needed, change the mount settings. As the root user, open the `/etc/fstab` file with a text editor and modify the `tmpfs` line. If you are planning to use AMM, set the `tmpfs` file system size to the sum of all the MEMORY_TARGET on the system, as shown in the following example:

```
tmpfs      /dev/shm     tmpfs      rw,exec,size=30G 0 0
```

LinuxONE can use large pages of 1 M with Oracle running under z/VM and 2GB when running under LPAR mode.

The Oracle **MEMORY_TARGET** parameter is not eligible for large pages; instead, **SGA_TARGET** must be used for the Oracle memory setting when Linux large pages are used.

3.2.7 Host name

Use the **ping** command to ensure that the computer host name is resolvable, as shown in Example 3-5.

Example 3-5 Example of the ping command and its output

```
ping l3oradb0.mop.fr.ibm.com
PING l3oradb0.mop.fr.ibm.com (10.3.58.144) 56(84) bytes of data.
64 bytes from l3oradb0.mop.fr.ibm.com (10.3.58.144): icmp_seq=1 ttl=64 time=0.290
ms
-- l3oradb0.mop.fr.ibm.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 999ms rtt min/avg/max/mdev
= 0.090/0.190/0.290/0.100 ms
```

If needed, add the Fully Qualified Domain Name to the `HOSTNAME` file by using the **hostname** command. Example 3-6 demonstrates showing the **hostname** command and its results.

Example 3-6 Show the host name

```
# hostname -f
Sample.yourbiz.com
```

Example 3-7 shows the command and results of displaying the contents of the `HOSTNAME` file.

Example 3-7 Show the contents of the HOSTNAME file

```
# cat /etc/hostname
l3oradb2.mop.fr.ibm.com
```

To update the host name, enter the fully qualified host name by using the **hostname** command, as shown in Example 3-8.

Example 3-8 Updating the host name

```
# hostname
# hostname l3oradb2.mop.fr.ibm.com
# hostname
l3oradb2.mop.fr.ibm.com
```

3.2.8 Storage options for Oracle Database

In this section, we describe the following disk storage configuration options that are available for database files on LinuxONE:

- Open Storage FCP/SCSI LUNs
- IBM Extended Count Key Data (ECKD) DASD with HyperPAV or PAV.
- File systems and Shared Storage (for example, Spectrum Scale GPFS)

To configure disk storage for LinuxONE, assign the LUNs (with multiple paths) to Linux to ensure high availability and better performance by spreading the I/O workload across more channel paths/host bus adapters (HBAs).

FCP/SCSI

If FCP/SCSI open system storage is used for your ASM disk devices, it is important to configure a multipathing and an UDEV rule that ensure device persistence and disk permissions for the Oracle ASM disks.

To configure multiple disk paths with FCP/SCSI open storage, it is necessary to set up `/etc/multipath.conf` file to enable multipathing for high availability to the disk paths and for performance reasons.

It is recommended to use the **`user_friendly_names`** parameter in the `multipath.conf` so that the disk permissions for the Oracle Database ASM LUNs can be set correctly. Example 3-9 shows a sample `multipath.conf` file.

Note: It is also recommended to consult your storage vendor for their recommendations for the `multipath.conf` settings for the Linux distribution and level that you use.

Example 3-9 Sample /etc/multipath.conf file for FCP/SCSI open system disk storage

```
defaults {
    dev_loss_tmo            inifinity
    fast_io_fail_tmo        5
    user_friendly_names     yes
}
blacklist {
#       devnode ".*"
}
multipaths {
       multipath {
               wwid 20020c240001221a8
               alias    ASMDISK00
          }
}
```

To restart the multipath service for any changes to take effect, run the following command to read in the new `multipath.conf` settings:

```
/sbin/multipath -v2
systemctl restart multipathd.service
```

The /etc/udev/rules.d/12-dm-permissions.rules file also must be configured to set the /dev/mapper/ user friendly disk names to the correct device permissions for the oracle or grid Linux user ID that owns the disk storage devices.

Example 3-10 shows that any disks that are configured with a **user_friendly_name** that begins with **"**ASM***"** are owned by the oracle Linux user (grid or oracle user).

Example 3-10 12-dm-permissions.rules file for FCP/SCSI open system disk storage

```
ENV{DM_NAME}=="ASM*",OWNER:="oracle",GROUP:="dba",MODE:="660"
```

You can get a sample /etc/udev/rules.d/12-dm-permissions.rules template file from /usr/share/doc/device-mapper-<your kernel level>/12-dm-permissions.rules for Red Hat systems, and /usr/share/doc/packages/device-mapper/12-dm-permissions.rules on SUSE Linux systems.

You can run the following Linux commands to enable any changes to the UDEV rule:

```
udevadm control –reload-rules
udevadm trigger
```

Then, verify the disk permissions are set correctly by using the **ls** command:

```
ls -lL ASM*
brw-rw---- 1 oracle dba 253, 14 Jun 12 08:30 ASMSAN00
brw-rw---- 1 oracle dba 253, 15 Jun 12 08:00 ASMSAN01
brw-rw---- 1 oracle dba 253, 16 Jun 12 08:00 ASMSAN02
brw-rw---- 1 oracle dba 253, 17 Jun 12 08:00 ASMSAN10
brw-rw---- 1 oracle dba 253, 18 Jun 12 08:00 ASMSAN11
```

ECKD/DASD

With ECKD/DASD storage, it is recommended to set up disk path aliases with HyperPAV, as described in Chapter 1, "A shared commitment to client value with the IBM and Oracle alliance" on page 1. z/VM manages the high availability of the disk paths and HyperPAV provides more I/O paths to help avoid any disk I/O pathing bottlenecks.

Consider the following points:

- To configure the correct disk permissions for the Oracle ASM disk user to write and read from the ASM devices, a UDEV rule is needed to assign the correct disk permissions.
- To configure the ECKD/DASD devices, it is helpful to use the lsdasd device name in the disk name to make it easier to assign the disk storage.
- If configuring Oracle RAC, the same disk devices must be set up as shareable on each of the nodes in the RAC cluster. Configure a new UDEV, such as /etc/udev/rules.d/99-udev-oracle.rules, as shown in Example 3-11.

Example 3-11 Reference 99-udev-oracle.rules file for ECKD/DASD storage devices for database files

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================
0.0.0200   active      dasda     94:0    ECKD  4096   7043MB    1803060
0.0.7406   active      dasdp     94:60   ECKD  4096   7043MB    1803060
0.0.7407   active      dasds     94:72   ECKD  4096   7043MB    1803060
# cat 99-udev-oracle.rules
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.7406", ENV{DEVTYPE}=="partition",
SYMLINK+="oracleasm/asm7406", GROUP="oinstall", OWNER="oracle", MODE="0660"
```

```
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.7407", ENV{DEVTYPE}=="partition",
SYMLINK+="oracleasm/asm7407", GROUP="oinstall", OWNER="oracle", MODE="0660"
# udevadm trigger
# ls -lL /dev/oracleasm
brw-rw---- 1 oracle oinstall 94,  61 Jun 12 19:18 asm7406
brw-rw---- 1 oracle oinstall 94,  73 Jun 12 17:15 asm7407
```

A restart is required for modifications of UDEV rules to take effect. For Red Hat 7 / SUSE Linux Enterprise Server 12, you can run the following UDEV commands:

```
udevadm control –reload-rules
udevadm trigger
```

This change can be done dynamically. Then, run the **ls** commands to confirm that the file permissions are set correctly (based on the devices configured in your UDEV rules), as shown in Example 3-11 on page 23.

3.2.9 Required software directories

Identify or create the directories that are described in this section that are needed by the Oracle software installer. The examples in this section use `/u01` for the mount point directory.

Oracle Base Directory

The following directory is used:

```
ORACLE_BASE = /u01/app/software_owner
```

In this installation example, we created a 20 GB Logical Volume to host Oracle software and data. Where `software owner` is the operating system user name of the software owner that is installing the Oracle software; for example, `oracle` or `grid`.

Oracle Inventory directory

This directory stores an inventory of all software that is installed on the system. It is required and shared by all Oracle software installations on a single system. If an old Oracle Inventory path exists, the OUI continues to use that Oracle Inventory.

The Oracle Inventory path is in the `/var/opt/oracle/oraInst.loc` file.

Example 3-12 shows the command that is used to check the contents and the existence of the `oraInst.loc` file with the results displaying the contents of the file.

Example 3-12 Examining the oraInst.loc

```
# cat /var/opt/oracle/oraInst.loc
 inventory_loc=/oraInventory inst_group=oinstall
```

Oracle Home directory

This directory is where the Oracle software product is installed. A typical `ORACLE_HOME` environment variable setting is shown in the following example:

```
ORACLE_HOME=$ORACLE_BASE/oracle_base/product/19c/dbhome_1
```

If ASM is not used, create a Linux file system by using the following Yast or Linux commands:

1. Create the initial mount point:

```
mkdir /u01
chown -R oracle:oinstall /u01
chmod -R 775 /u01
```

2. Create a simple file system:
   - ECKD DASD:

```
dasdfmt -p -y -b 4096 -f  /dev/dasdan
fdasd -a /dev/dasdan
pvcreate /dev/dasdan1
vgcreate orvg /dev/dasdan1
```

   - FCP/SCSI:

```
fdisk  /dev/mapper/mymp1
pvcreate /dev/mapper/mymp1p1
vgcreate orvg /dev/mapper/mymp1p1
```

3. For both disk types, create the logical volume and file system:

```
lvcreate --name orlv -l 100%FREE orvg
mkfs.xfs  /dev/orvg/orlv
```

4. Mount the logical volume on `ORACLE_BASE=/u01`, as shown in the following example:

```
mount -t xfs /dev/orvg/orlv  /u01
```

5. For resiliency after system reboot, insert the mount statement into the `/etc/fstab` file:

```
/dev/oravg/oralv                /u01                    xfs     defaults        1 1
```

3.2.10 Tiger VNC xterm Window for Interactive GUI installations (optional)

Install or configure the VNC viewer or xterm emulator if you want to perform GUI-based Oracle installations with `runInstaller`, `asmca`, `dbca`, or `netca` Oracle GUI installer programs.

Silent install options are available for these Oracle installers as well. However, first-time installations include GUI wizards that provide the system checks to verify your system configuration and then can be scripted with the silent installations later.

For Red Hat systems, the VNC Configurator summarizes the steps to install the `vnc` server and client RPMs, and then to configure your `vnc xstartup` file based on the xterm emulator of your choosing.

For more information about the Red Hat 7 installation, see this web page.

```
# echo "Install VNC server"
# yum install tigervnc-server
# echo "Install VNC client"
# yum install tigervnc
# For Minimal GUI twm
# yum install xterm xorg-x11-twm
```

For SUSE installations, the VNC viewer installer provides a good example of the use of the Remmina Desktop as a good option for Oracle installations.

3.3 Red Hat Enterprise Linux-specific steps

This section provides the preinstallation Linux steps that are required to set up of Oracle RDBMS 19c release on Red Hat 7 Update 4 (or higher 7.x version) on IBM: Linux on System Z (s390x).

These steps are in addition to the generic Linux setup steps that are described in 3.2, “Red Hat Enterprise Linux and SUSE Enterprise Linux Server setup” on page 14 and apply to Red Hat Enterprise Linux systems on IBM LinuxONE only.

3.3.1 Preinstallation checking

This section describes more preinstallation checking that you must perform after you ensure that a GUI is established.

Operating system requirements for IBM LinuxONE

RHEL 7 servers must be running Red Hat kernel 3.10.0-693.el7 (s390x) or higher to install Oracle RDBMS 19c on IBM LinuxONE.

Oracle RPM Checker with Red Hat Enterprise Linux Server

In this section, we describe how to use the Oracle RPM Checker utility to verify that the required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation.

Download the suitable Oracle RPM Checker utility from the MOS websites based on Oracle Database 19c Release and the Red Hat Enterprise Linux Server distribution level (for example, RHEL 7 / 19.3). *RPM Checker when Installing Oracle Database 19c on IBM Linux on System z (Doc ID 2553465.1* contains the RPM checker scripts.

After you download the RPM checker on LinuxONE, upload the RPM checker to your system and install it as `root`. Use the Red Hat **`yum`** command to install any required RPMs and their dependencies:

```
yum localnstall ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
```

Figure 3-5 and Figure 3-6 on page 28 shows the result of running the **`yum install`** command to install the Oracle RPM dependencies.

```
[root@techora stage]# yum localinstall ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to register.
Examining ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm: ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
Marking ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package ora-val-rpm-RH7-DB-19c.s390x 0:19.0.1-1 will be installed
--> Finished Dependency Resolution
local-rhn-server-debuginfo/7Server/s390x
local-rhn-server-optional/7Server/s390x
local-rhn-server-optional/7Server/s390x/primary_db
local-rhn-server-os/7Server/s390x
local-rhn-server-os/7Server/s390x/primary_db
local-rhn-server-supplementary/7Server/s390x

Dependencies Resolved

================================================================================================================================================================
 Package                                    Arch                         Version                          Repository
================================================================================================================================================================
Installing:
 ora-val-rpm-RH7-DB-19c                     s390x                        19.0.1-1                         /ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x

Transaction Summary
================================================================================================================================================================
Install  1 Package
```

Figure 3-5 The yum install command for an Oracle 19c Red Hat 7 installation (1 of 2)

```
Is this ok [y/d/N]: y
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Warning: RPMDB altered outside of yum.
 Installing : ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
1/1
********************************************************************************
*     Validation complete - please install any missing rpms                    *
*   The following output should display both (s390) - 31-bit and               *
*       (s390x) 64-bit rpms - Please provide the output to Oracle              *
*            Support If you are still encountering problems.                   *
********************************************************************************
Found          glibc (s390)
Found          glibc (s390x)
Found          libaio (s390)
Found          libaio (s390x)
Found          libaio-devel (s390)
Found          libaio-devel (s390x)
Found          libXi (s390)
Found          libXi (s390x)
Found          libXtst (s390)
Found          libXtst (s390x)
Found          glibc-devel (s390)
Found          glibc-devel (s390x)
Found          libgcc (s390)
Found          libgcc (s390x)
Found          libstdc++ (s390)
Found          libstdc++ (s390x)
Found          libstdc++-devel (s390)
Found          libstdc++-devel (s390x)
Found          libattr-devel (s390)
Found          libattr-devel (s390x)
Found          libX11 (s390)
Found          libX11 (s390x)
Found          libXaw (s390x)
Found          libXft (s390)
Found          libXmu (s390)
Found          libXp (s390)
 Verifying : ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
1/1
Installed:
  ora-val-rpm-RH7-DB-19c.s390x 0:19.0.1-1
Complete!
```

Figure 3-6 The yum install command for an Oracle 19c Red Hat 7 installation (2 of 2)

Running the command automatically pulls in the required RPMs if your YUM repository is set up. If you do not have your RPM repository set up, you can install the RPM dependency checker by using the following command (use the command corresponding to your distribution level):

```
rpm -ivh ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
```

Figure 3-7 shows checking RPMs if RPM checker is not yet installed.

```
# rpm -ivh ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:ora-val-rpm-RH7-DB-19c-19.0.1-1  ################################# [100%]
********************************************************************************
*     Validation complete - please install any missing rpms                    *
*   The following output should display both (s390) - 31-bit and               *
*       (s390x) 64-bit rpms - Please provide the output to Oracle              *
*            Support If you are still encountering problems.                   *
********************************************************************************
Found          glibc (s390)
Found          glibc (s390x)
Found          libaio (s390)
Found          libaio (s390x)
Found          libaio-devel (s390)
Found          libaio-devel (s390x)
Found          libXi (s390)
Found          libXi (s390x)
Found          libXtst (s390)
Found          libXtst (s390x)
Found          glibc-devel (s390)
Found          glibc-devel (s390x)
Found          libgcc (s390)
Found          libgcc (s390x)
Found          libstdc++ (s390)
Found          libstdc++ (s390x)
Found          libstdc++-devel (s390)
Found          libstdc++-devel (s390x)
Found          libattr-devel (s390)
Found          libattr-devel (s390x)
Found          libX11 (s390)
Found          libX11 (s390x)
Found          libXaw (s390x)
Found          libXft (s390)
Found          libXmu (s390)
Found          libXp (s390)
```

Figure 3-7 Red Hat RPM checker for an Oracle RDBMS 19c Installation

3.4 SUSE Linux Enterprise Server specific setup

This section provides the preinstallation Linux steps that are required to set up of Oracle RDBMS 19c release on SUSE Linux Enterprise Server 12 Update 3 (or higher 12.x version) on IBM: Linux on System Z (s390x).

These steps are in addition to the generic Linux setup steps that are described in 3.2, “Red Hat Enterprise Linux and SUSE Enterprise Linux Server setup” on page 14 and apply to SUSE Linux Enterprise systems on IBM LinuxONE only.

For more information about other installation options for the use of KVM for non-production deployments, see Chapter 8, “Optimizing Oracle Database for developing and testing on a SUSE Linux Enterprise Server KVM guest” in *Virtualization Cookbook for IBM Z Volume 5: KVM*, SG24-8463.

3.4.1 Important information

Although the SUSE **`orarun`** RPM package can be used to configure the user or groups, `ulimits`, and kernel parms automatically, our experiences are based on manually configuring these settings.

3.4.2 Preinstallation checking

This section describes other preinstallation checking that you must perform after ensuring that a GUI is established.

Operating system requirements for IBM LinuxONE

SUSE servers must be running SUSE Linux Enterprise Server 12 SP3 4.4.73-5.1-default (s390x) or higher to install Oracle RDBMS 19c on IBM LinuxONE.

Oracle RPM Checker with SUSE Linux Enterprise Server on LinuxONE

In this section, we describe how to use the Oracle RPM Checker utility to verify that the following required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation.

Download the suitable Oracle RPM Checker utility from the MOS websites based on Oracle Database 19c Release and the SUSE Linux Enterprise Server release level (for example, SUSE Linux Enterprise Server 12 / 19.3). *RPM Checker when Installing Oracle Database 19c on IBM Linux on System z (Doc ID 2553465.1)* contains the latest RPM checker scripts.

After you download the RPM checker on LinuxONE, upload the RPM checker to your system and install it as `root`. Use the SUSE **`zypper`** command to install any required RPMs and their dependencies:

```
zypper install ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm
```

Figure 3-8 on page 31 shows the result of running the **`zypper install`** command to install the Oracle required RPM dependencies.

```
# zypper install ./ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm
Loading repository data...
Reading installed packages...
Resolving package dependencies...
The following NEW package is going to be installed:
  ora-val-rpm-S12-DB-19c
The following package has no support information from it's vendor:
  ora-val-rpm-S12-DB-19c
1 new package to install.
Overall download size: 3.2 KiB. Already cached: 0 B. No additional space will
be used or freed after the operation.
Continue? [y/n/...? shows all options] (y): y
Retrieving package ora-val-rpm-S12-DB-19c-19.0.1-1.s390x
(1/1),   3.2 KiB (    0   B unpacked)
ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm:
    Package is not signed!
ora-val-rpm-S12-DB-19c-19.0.1-1.s390x (Plain RPM files cache): Signature
verification failed [File is unsigned]
Abort, retry, ignore? [a/r/i] (a): i
Checking for file conflicts:
.....................................................................[done]
(1/1) Installing: ora-val-rpm-S12-DB-19c-19.0.1-1.s390x
.....................................................................[done]
Additional rpm output:
*******************************************************************************
*  Validation complete - Your sles12 OS has required rpms for Oracle 19c.     *
* If you are using Grid Infrastructure, make sure you have gfortran compiler: *
*  at least v. 4.8.5-16                                                       *
*******************************************************************************
```

Figure 3-8 Zypper install command for required SUSE Linux Enterprise Server 12 Linux RPMs for Oracle 19c installation

Running the command automatically pulls in the required RPMs if your `zypper` repository is set up. If you do not have your `rpm` repository set up, you can install the RPM dependency checker by using the following command (use the command that corresponds to the RPM based on your distribution level):

```
rpm -ivh ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm
```

Figure 3-9 shows checking RPMs if RPM checker is not yet installed.

```
# rpm -ivh ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:ora-val-rpm-S12-DB-19c-19.0.1-1  ################################# [100%]
*******************************************************************************
*  Validation complete - Your sles12 OS has required rpms for Oracle 19c.     *
* If you are using Grid Infrastructure, make sure you have gfortran compiler: *
*  at least v. 4.8.5-16                                                       *
*******************************************************************************
```

Figure 3-9 SUSE RPM checker for an Oracle 19c SUSE Linux Enterprise Server 12 installation

3.5 Linux large pages and Oracle databases

It is recommended for performance and availability reasons to implement Linux large pages for Oracle databases running on IBM LinuxONE systems. Linux large pages are particularly beneficial for systems where the database's Oracle SGA is greater than 8 GB or if more than 50 database connections exist.

3.5.1 Linux large page setup

Complete the following steps to implement large pages:

1. Each database that is planned for Linux large pages cannot use AMM by setting the **MEMORY_MANAGEMENT** parameter. It is recommended to use Automatic Shared Memory Management (ASMM) by setting the **SGA_TARGET** and **PGA_AGGREGATE_TARGET** Oracle parameters.
2. The Oracle **use_large_pages** parameter can be set to `true`, `false`, or `only`. If you have a LinuxONE system with one large database require large pages, and other smaller databases that do not require large pages, you can set the larger SGA database to "`only`" and the smaller databases to "`false`". The default setting for `use_large_pages` is "`true`".
3. At the Linux level, it is recommended to set the `/etc/security/limits.conf` to `unlimited` to allow for changes to the Oracle SGA/Linux large page values dynamically, as shown in the following examples:

   ```
   soft memlock unlimited
   hard memlock unlimited
   ```

4. Set or update the following parameters in the `/etc/sysctl.conf` file:

   ```
   vm.nr_hugepages = ((sum of all large page SGA's)* 1024) + 16 = N
   vm.hugetlb_shm_group = <Linux group number from /etc/group>
   ```

5. Restart your Linux Image and restart Oracle for the changes to take effect.
6. Review your Oracle Database alert log to verify the database was started with large pages enabled.

3.5.2 Disable Transparent HugePages

It is recommended for performance and stability reasons to disable Transparent HugePages. Transparent HugePages are different than Linux large pages, which are still highly recommended to use.

You can check if your system has Transparent HugePages enabled by using the following command:

```
cat /sys/kernel/mm/transparent_hugepage/enabled
[always] madvise never
```

To disable Transparent HugePages, complete the following steps:

1. Add `vi /etc/zipl.conf add transparent_hugepage=never` to the end of the parameters line, as shown in Figure 3-10.

```
oratest:~ # cat /proc/cmdline
root=UUID=26180fe6-b0f8-4ca8-b60c-dc98a230bc37 hvc_iucv=8 TERM=dumb hvc_iucv=8
 TERM=dumb nobp=off transparent hugepage=never
```

Figure 3-10 Disable Transparent HugePages

2. Run **`zipl -VV`** to activate the Linux kernel parameter for the next system restart:

```
[root@zlnxl2 etc]# zipl -VV
Using config file '/etc/zipl.conf'
Target device information
  Device.....................: 5e:00
  Partition..................: 5e:01
  Device name................: dasda
  Device driver name.........: dasd
…
Preparing boot device: dasda (0200).
Syncing disks...
Done.
```

3. Restart your Linux guest by running the following command:

```
shutdown -r now
```

4. Verify that Transparent HugePages are disabled by running the following command:

```
cat /sys/kernel/mm/transparent_hugepage/enabled
always madvise [never]
```

3.6 Enable Linux random number generation (Entropy)

Foe Oracle Database versions 12.2 and later, it is important for systems to be configured to provide good random number generation to use the new security features of an Oracle Database installation.

Oracle version 12.2+ now uses `/dev/random` rather than `/dev/urandom`, which means without correctly configured entropy or random numbers replenishment, you might encounter intermittent access issues, such as `ORA-01017: invalid username/password` on your system.

3.6.1 Entropy replenishment configuration

The general rule is that `/proc/sys/kernel/random/entropy_avail` is always greater than 1000 random numbers available. To validate this number, the following command can be used:

```
cat /proc/sys/kernel/random/entropy_avail
348
```

This value is less than 1000 and needs replenishment.

The Pseudo Random Number Generator (PRNG) device can be used to provide statistically significant random numbers for running Oracle databases. IBM LinuxONE servers use the CP Assist for Cryptographic Function (CPACF) on each processor chip to replenish random numbers.

The first step is to verify if your system has the PRNG driver loaded. To verify whether the `/dev/prandom` file exists on your LinuxONE system, run the following command:

```
ls -la /dev/prandom
```

The latest Red Hat 7 and SUSE Linux Enterprise Server 12 distribution levels should have the driver `prandom` driver configured by default. If the `prandom` driver does not exist on your system, you can configure/load the driver by completing the following steps:

1. Load the `/dev/prandom` pseudo random module (PRNG) module at run time by using the following command:

   ```
   modprobe prng.
   ```

2. To load the `prng` module, on system restart you can configure in a local boot script (such as `rc.local`) to load the `prng` module after a system restart:

   ```
   cat /etc/rc.d/rc.local
   #!/bin/sh
   #
   # This script will be executed *after* all the other init scripts.
   # You can put your own initialization stuff in here if you don't
   # want to do the full Sys V style init stuff.
   touch /var/lock/subsys/local
   modprobe prng
   ```

3.6.2 Enabling the Red Hat rngd service for random number generation

Complete the following steps to install and enable the `rngd` service if the module is not installed:

1. Install the `rngd` service if it is not installed:

   ```
   yum install -y rng-tools
   ```

2. Create a copy of the template `rngd.service` file (if such a file does not exist):

   ```
   cp /usr/lib/systemd/system/rngd.service  /etc/systemd/system/rngd.service
   ```

3. Edit the `/etc/systemd/system/rngd.service` file and replace the line that starts with `ExecStart` with the random number method that is used:

   ```
   ExecStart=/sbin/rngd -f -r /dev/prandom
   ```

4. Reload the systemd configuration:

   ```
   # systemctl daemon-reload
   ```

5. Restart the rngd service:

   ```
   # systemctl restart rngd
   Then check the service status:
   # systemctl status rngd
   If you see the following error during install, make sure you have the rng-tools
   rpm installed.
   Failed to init entropy source 1: TPM RNG Device
   systemctl status rngd
   rngd.service - Hardware RNG Entropy Gatherer Daemon
   ```

```
   Loaded: loaded (/etc/systemd/system/rngd.service; enabled; vendor preset:
enabled)
   Active: active (running) since Fri 2020-04-03 08:36:26 CDT; 10s ago
 Main PID: 53599 (rngd)
   CGroup: /system.slice/rngd.service
           ??53599 /sbin/rngd -f -r /dev/prandom
Apr 03 08:36:26 zlnxl2.pbm.ihost.com systemd[1]: Started Hardware RNG Entropy
Gatherer Daemon.
Apr 03 08:36:26 zlnxl2.pbm.ihost.com systemd[1]: Starting Hardware RNG Entropy
Gatherer Daemon...
Apr 03 08:36:26 zlnxl2.pbm.ihost.com rngd[53599]: Initalizing available sources
Apr 03 08:36:26 zlnxl2.pbm.ihost.com rngd[53599]: Initalizing entropy source
Hardware RNG Device
Apr 03 08:36:27 zlnxl2.pbm.ihost.com rngd[53599]: Enabling JITTER rng support
Apr 03 08:36:27 zlnxl2.pbm.ihost.com rngd[53599]: Initalizing entropy source
JITTER Entropy generator
```

6. Verify that your system includes the required amount of entropy greater than 1000:

```
cat /proc/sys/kernel/random/entropy_avail
3099
```

3.6.3 Enabling the SUSE haveged service for random number generation

Complete the following steps to install and enable the `haveged` service if the module is not enabled on your system:

1. Verify or install the `haveged` service if it is not installed:

```
# rpm -qa | grep haveged
haveged-1.9.1-16.1.s390x
```

2. If `haveged` is not installed, you can use zypper to install the module and verify:

```
# cat /proc/sys/kernel/random/entropy_avail
2851
```

4

Setting up RHEL 8 Linux to install Oracle 19c Database

This chapter covers the pre-installation steps that are required for setting up Oracle 19c with the newly certified Red Hat Enterprise Linux Release 8 (RHEL 8) on an IBM LinuxONE system. We include information about how to obtain Oracle documentation, code, and My Oracle Support (MOS) Notes.

For product longevity and patching, IBM and Oracle recommend upgrading to Oracle 19c, which is the long-term release with the longest support end date. Oracle 19c is the terminal release of Oracle 12.2.

The minimum RHEL 8 Oracle Database release level is Oracle 19c (19.3) with the 19.11 RU installed.

This chapter includes the following topics:

4.1 Obtaining Oracle documentation, downloads, and My Oracle support notes

This section includes the following topics:

Changes in the recommended Oracle Rational Portfolio Managers and kernel parameters can occur between release levels. We provided references throughout this chapter with the corresponding MOS reference note for the most updated guidance on a specific topic and verification.

4.1.1 Master Note of Linux OS requirements for Database Server

The Master Note provides an index and references to the most frequently used MOS articles about Linux OS requirements for an Oracle Database Software Installation.

For more information about the master note, see MOS Doc ID 851598.1.

This support doc is available by logging in to your Oracle Support account. The key notes for Oracle 19c installation with Linux on System z are described next.

For more information about installing Red Hat 8 Oracle 19c, see *Requirements for Installing Oracle Database 19c on RHEL8 64-bit (IBM Linux on System z),* Doc ID 2781301.1 (log in required).

Oracle 19c is certified by Oracle for Red Hat release level 8.3 or greater. Oracle is certified to the base Major OS level. All service packs that are greater than the minimum required distribution level are supported by Oracle.

The latest available Red Hat service pack level (for example, 8.4) is recommended for new installations to have the most updated features and security features for running Oracle databases.

For more information about the release schedule of current database releases, see Doc ID 742060.1.

This doc is available by logging in to your Oracle Support account. Use the My Oracle Support Certification tab on the MOS site to determine the latest release status of Oracle version releases with Linux on System z.

4.1.2 Obtaining the Oracle 19c software for IBM LinuxONE

Depending on your company’s Oracle licensing agreement, the latest Oracle code can be downloaded from the Oracle Software Delivery cloud or from the Oracle Technology Network (OTN) developers download site.

If you have a commercial Oracle license, download your software from Oracle Software Delivery Cloud.

For trial and developer network downloads, the Oracle license agreement must be fully understood before agreeing and downloading the Oracle 19c database software from the OTN download site.

Complete the following steps:

1. Sign in to the Oracle Software Delivery cloud by using your Oracle user ID and password. Enter `Oracle Database` in the search field for the Oracle Enterprise Edition that you want to install.
2. Click the **“+”** sign of the **Add to Cart** button for the software version that you want to install, and then, click **View Cart** in the upper right of the window. The view cart window opens with an option to check out. Click **Checkout**.
3. Select **IBM: Linux on System z,** and then, select the software components that you need for your installation under the Platforms/Languages column.

   Most installations require the Oracle Database software to be download. If you use Oracle ASM, or are installing Oracle RAC or Oracle RACONE, you also need the Oracle Database Grid Infrastructure software for applications that must connect to an Oracle Database that is running on Linux on System z. If your application runs on Linux on System z by using `sql*net` network connectivity, the Oracle Database Client also must be downloaded.
4. After all of the software is selected, click **Download** to start downloading the software. If this download is the first time you are downloading software, you might be prompted to download and install the Oracle download installer first. After the installer is configured, the selected Oracle distribution software is downloaded to your workstation. You can also download by clicking the link for the software product individually.

In addition to the base software, the latest Patch Set or Release Update should be downloaded from the Oracle support site.

Oracle 19c provides the latest Release Update (RU) for the Grid Infrastructure and Database from the Oracle support site. The latest RU for your release can be found in Oracle Note: *Master Note for Database Proactive Patch Program*, Doc ID 756671.1.

An Oracle 19c base installation is for release 19.3 for Linux on System z. For Oracle RHEL 8 certification, RU 19.11 is the minimum RU level required.

As of October 2021, the RU patch Oracle 19.12 Grid Infrastructure and Database security and RU patch is 32895426 (log in required).

Follow the readme file for the patch that you plan to install. Make sure to download the latest OPatch installer utility patch 6880880 as well for the RU patch as described in the patch readme file.

4.2 Setting up RHEL 8.0 Oracle

This section describes the common steps that are needed to implement an Oracle Database, including Oracle Grid Infrastructure on Red Hat Enterprise 8 on LinuxONE.

4.2.1 Important information

The following pre-installation verification information is provided to give a basic understanding of the system requirements before starting an Oracle Database installation on LinuxONE:

- Graphical user interface (GUI)

  A GUI is required to run the Oracle Installers interactively. We use the Linux `vncserver` on LinuxONE to start an X server and a VNC viewer client to establish a terminal emulator xterm session.

  To use GUIs, such as the Oracle Universal Installer (OUI), configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

  In our environment, we start a `vncserver` on LinuxONE, and use a VNC viewer client to establish the connection with the `vncserver`.

  Note: You also can configure a silent or scripted installation for cloud automation deployments.

- Linux minimal installation

  A minimal Linux installation lacks many RPMs that are required for database installation. Therefore, you must use an IBM Rational® Portfolio Manager package for Linux release to install the required packages.

- Automatic Storage Management ASM

  To use Oracle's ASM, you must first install Oracle Grid Infrastructure for a stand-alone server before you install and create the database.

  Most customers use Oracle ASM to store database files or a file system for single instance installations. For Oracle Real Application Cluster (RAC) installations, Oracle ASM or IBM Spectrum scale can be used to store database files across Oracle RAC cluster nodes.

  To use Oracle ASM on LinuxONE, you must first install the Oracle Grid Infrastructure for a stand-alone server before you can install and create the database.

4.2.2 General requirements and information for Oracle Database on LinuxONE

To obtain general LinuxONE information from a Linux session, run the following commands, ensuring that the system was started with a run level 3 or 5:

- Check information about the CPU that is defined in LinuxONE:

  ```
  cat /proc/cpuinfo
  ```

- Check information about the RAM that is defined in LinuxONE:

  ```
  cat /proc/meminfo | grep MemTotal
  cat /proc/meminfo | grep SwapTotal
  ```

- Check the amount of space that is available in the `/tmp` directory:

  ```
  df -h /tmp
  ```

- Check the amount of free disk space on the system (in GB):

  ```
  df -h
  ```

- Check the amount of free RAM and disk swap space on the system:

  ```
  free
  ```

- Check whether the system architecture can run the software:

  ```
  uname -m
  ```

  This command should return: s390x

4.2.3 Server minimum requirements

The following minimum requirements for the server must be met:

- Disk Space Requirements for Oracle 19c on IBM LinuxONE

  Ensure that the system meets the disk space requirements for software files:

  - Enterprise Edition database: A total of 7.5 - 8.6 GB of disk space for the Oracle software, depending on the installation type. You might need some more space for patching logs. A minimum of 10 GB of disk space is recommended.
  - Grid Infrastructure: A minimum of 8.9 - 10 GB is required for installing Oracle ASM single instance (restart) or an Oracle RAC install. It is also recommended to provide more space for Oracle grid infrastructure log files. A minimum of 20 GB of disk space is recommended.

- Recommended memory

  A minimum of 4 GB of virtual memory is recommended in the Linux guest for Oracle 19c installations of Oracle Database, including the Grid Infrastructure requirements.

  For single instance file system-based installations, 2 GB minimum virtual memory is recommended.

  Note: Although 2 GB is the minimum amount of virtual memory that is required, the Oracle OPatch utility requires at least 3072 MB (3 GB) for Oracle patching.

- Swap

  Swap disk space is proportional to the system's physical memory (see Table 4-1) and recommended by Oracle for general Linux swap.

  Table 4-1 Swap size recommendation

RAM	Swap space
1 - 8 GB	Twice the size of RAM
8 - 32 GB	Equal to the size of RAM
More than 32 GB	32 GB

  The Oracle guidelines for Linux swap can be reduced when needed. If enough memory is available to run all the databases in the Linux guest, the system should not need to swap, which can affect system performance.

  The Oracle Installer requires a minimum of 500 MB of configured Linux swap to complete an installation and 1 GB of swap for database upgrades.

Customers that use IBM LinuxONE can use a layered virtual in-memory disk or VDisk for the Linux swap devices. Linux swap to a memory device (VDisk) is much quicker than using a physical disk storage device. z/VM does not allocate the memory that is used by a VDisk until the first swap page is used.

Then, a physical disk or DASD disk can be used as a lower priority swap if unexpected memory use occurs. Linux uses the higher priority swap devices first. When the swap device is fully exhausted, the next priority swap device is used. Figure 4-1 shows an example of a recommended VDisk configuration with a VDisk that is used for the first and second Level Swap with higher priority.

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdo1                             partition       131000  0       10
/dev/dasdp1                             partition       524216  0       5
/dev/mapper/u603_swap3                  partition       6291448 0       1
```

Figure 4-1 Swap VDisk configuration priorities

Swap usage is used with the highest priority (`pri=`) number. The default is `-1` for swap usage priority. The Linux **`mkswap`** and **`swapon`** commands can be used to manually configure swap. The `/etc/fstab` file can be used to configure permanent swap devices, as shown in the following example:

```
# cat /etc/fstab | grep swap
/dev/disk/by-path/ccw-0.0.0202-part1 swap          swap defaults   0 0
# lsdasd | grep 0202
0.0.0202  active    dasdc     94:8    ECKD         4096   42259MB   10818360
# swapon -s
Filename                                Type            Size    Used
Priority
/dev/dasdc1                             partition       43273340        0
-2
```

- Temporary disk space required for installation

  A minimum of 1 GB of disk space in the `/tmp` directory on LinuxONE systems is required (and less than 2 TB of disk space). If you do not have enough space in `/tmp` for an Oracle installation, you can delete any files that you can or set the TMP and TMPDIR environment variables to another directory with space when setting the Oracle user's environment.

- Linux kernel parameters

  Verify that the required operating system kernel parameters are installed.

  An example of the required Oracle kernel parameters is listed in Table 4-2 on page 43 for a system with an 8 GB Oracle System Global Area (SGA) database.

Table 4-2 Kernel parameters

Kernel parameter	Example value	File
`fs.file-max` Set to 512 x maximum number of user processes in Oracle per database for the Linux Guest. The 6815744 supports up to 13312 Oracle processes. Increase as needed.	6815744	`/proc/sys/fs/file-max`
`fs.aio-max-nr` This value limits concurrent outstanding requests and must be set to avoid I/O subsystem failure. The formula is `aio-max-nr =no of process per DB * no of databases * 4096`	1048576, or 3145728 for larger databases	`/proc/sys/fs/aio-max-nr`
`kernel.shmmax` Set this value to the size of at least the largest SGA in bytes. A total of 8589934592 bytes supports a maximum shared memory segment of 8 GB. For more information, see MOS note 567506.1.	8589934592 (8 GB)	`/proc/sys/kernel/shmmax`
kernel.shmall Set this parameter to a value that is equal to the total amount of shared memory in 4 K pages that the system can use at one time. The sum of all the SGA databases on the system in bytes is 4096. A total of 20907152 supports 8 GB SGA. For more information, see MOS note 301830.1.	2097152	`/proc/sys/kernel/shmall`
`kernel.sem` (`semmsl`, `semmns`, `semopm`, `semmni`) Minimum required semaphores that are required for Oracle	kernel.sem= 250 32000 100 128	`/proc/sys/kernel/sem`
`kernel.shmmni`	4096	`/proc/sys/kernel/shmmni`
`kernel.panic_on_oops` This setting allows time for diagnostics if a system restart occurs.	1	`/proc/sys/kernel/panic_on_oops`
`net.ipv4.ip_local_port_range`	9000 65500	`/proc/sys/net/ipv4/ip_local_port_range`
`net.core.rmem_default`	262144	`/proc/sys/net/core/rmem_default`
`net.core.rmem_max`	4194304	`/proc/sys/net/core/rmem_max`

Kernel parameter	Example value	File
net.core.wmem_default	262144	/proc/sys/net/core/wmem_default
net.core.wmem_max	1048576	/proc/sys/net/core/wmem_max
vm.swappiness A value of 1 prevents Linux swapping as much as possible.	1	/proc/sys/vm/swappiness

If the value of any parameter is higher than the value that is listed in Table 4-2 on page 43, do not change the value of that parameter. These parameters must be configured in the Linux /etc/sysctl.conf file.

Optional performance-related kernel parameters are listed in Table 4-3.

Table 4-3 Optional performance-related kernel parameters

Variable	Example value
vm.nr_hugepages Sum of the Oracle SGAs + 16 MB; for example,: 8208 for 8 GB SGA memory for large pages	8208 **Note:** Linux Guest memory always must be sized larger than this value or excessive Linux swap occurs.
vm.hugetlb_shm_group Set to oinstall group ID number from /etc/group file.	1001

Example 4-1 shows a sample sysctl.conf file with the required changes for an oracle RHEL 8 installation.

Example 4-1 sample sysctl.conf file

```
# cat /etc/sysctl.conf
# Oracle specific parameters
fs.file-max = 6815744
fs.aio-max-nr = 1048576
#set shmmax to largest sga memory segment in bytes
kernel.shmmax = 8589934592
#set shmall to (sum of sga memory in bytes)/4096
kernel.shmall = 2097152
kernel.sem = 250 32000 100 128
kernel.shmmni = 4096
kernel.panic_on_oops = 1
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
vm.swappiness = 1
#set nr_hugepages to sum of all Oracle SGAs in MB if using large pages
#vm.nr_hugepages = 8208
vm.hugetlb_shm_group=1001
```

After kernel parameter values are updated, validate them by using the following **sysctl** command to validate any kernel parameter changes:

```
/usr/sbin/sysctl -p
```

4.2.4 Creating the database installation owner user

The installation process requires at least an Oracle Database installation owner (`oracle`), an Oracle Inventory group (`oinstall`), and an Oracle administrative privileges group (`dba`).

The following example shows the commands that are used to create the `oracle` user and groups.

```
/usr/sbin/groupadd oinstall -g 1001
/usr/sbin/groupadd dba -g 1002
/usr/sbin/groupadd asmadmin -g 1003
/usr/sbin/groupadd asmoper -g 1004
/usr/sbin/useradd -m -g oinstall -G dba  oracle
echo "oracle:newPassw0rd" | /usr/sbin/chpasswd
```

For separation of duty, some sites create a Linux “grid” user ID to manage the grid infrastructure ASM and Oracle Grid Infrastructure components. If so, use the same **useradd** command and change password steps as shown in the following example for the Linux grid user as well:

```
/usr/sbin/useradd -m -g oinstall -G dba grid
echo "grid:newPassw0rd" | /usr/sbin/chpasswd
```

4.2.5 User login security and limits configuration

To avoid operational problems with the Oracle Database consuming system resources, the Linux `ulimit` values in the `/etc/security/limits.conf` file must be adjusted to allow for the proper operation of an Oracle Database.

An alternative is to use the `/etc/security/limits.d` directory for modifying the user `ulimit` values. The `ulimits` can be set only in one file on a system for any specific value.

If you are planning to use large pages, and the Oracle SGA memory is increased, setting the `ulimit memlock` to `unlimited`, is one less step that is needed to make that change:

```
oracle soft memlock unlimited
oracle hard memlock unlimited
```

If you use another Oracle Linux user ID, such as `grid`, to manage the Grid Infrastructure components, define the same `ulimit` values for the grid Linux user and the oracle Linux user ID:

```
grid soft memlock unlimited
grid hard memlock unlimited
```

If the Linux oracle user performs the installation, add the settings that are shown in Example 4-2 to your /etc/security/limits.conf file. If your current values are higher than these values, leave the higher value in place.

Example 4-2 /etc/security/limits.conf file

```
# cat /etc/security/limits.conf
oracle        soft    nproc   2047
oracle        hard    nproc   16384
oracle        soft    nofile  1024
oracle        hard    nofile  131072
oracle        soft    stack   10240
oracle        hard    stack   32768
grid          soft    nproc   2047
grid          hard    nproc   16384
grid          soft    nofile  1024
grid          hard    nofile  131072
grid          soft    stack   10240
grid          hard    stack   32768
#
# Use memlock for Huge Pages  support
oracle        soft    memlock  unlimited
oracle        hard    memlock  unlimited
grid          soft    memlock  unlimited
grid          hard    memlock  unlimited30G     0   30G   0% /dev/shm
```

Note: When the limits.conf file is changed, these changes take effect immediately. However, if the oracle users are logged in, these changes do not take effect until you log out these users and log them back in.

The ulimits take effect upon logging in to a Linux guest by setting the ulimit values in each Linux users' .bash_profile file or updating the system base /etc/profile for the system for all Linux users.

The following example shows the ulimit values that can be added to the oracle and grid (if used) user home directory .bash_profile file:

```
# cat /home/oracle/.bash_profile | grep ulimit
ulimit -n 131072
ulimit -s 32768
ulimit -u 32768
```

4.2.6 Shared memory file system

For Oracle Database Installations that are not using Linux large pages, it is recommended to use Automatic Memory Management (AMM) with the MEMORY_TARGET parameter. To use AMM, it is important to ensure that the /dev/shm mount area is of type tmpfs and is mounted with the following options:

- With RW (read/write) and run permissions set on it
- With noexec or nosuid not set on it

To use AMM, the tmpfs mount settings *must* be configured to the correct size. It is also recommended to back up your /etc/fstab file before making any changes.

As the root user, open the /etc/fstab file by using a text editor and modify the tmpfs line. If you are planning to use AMM, set the tmpfs file system size to the sum of all the MEMORY_TARGET on the system, as shown in the following example:

```
# cp /etc/fstab /etc/fstab.old
# cat /etc/fstab |grep tmpfs
tmpfs   /dev/shm        tmpfs   rw,exec,size=30G 0 0
# mount tmpfs
# df -h | grep shm
tmpfs                          30G     0   30G   0% /dev/shm
```

LinuxONE can use large pages of 1 M with Oracle running under z/VM and 2 GB when running under LPAR mode.

The Oracle MEMORY_TARGET parameter is not eligible for large pages; instead, use SGA_TARGET for the Oracle memory setting when Linux large pages are used.

4.2.7 Host name

Use the **ping** command to ensure that the computer host name is resolvable, as shown in the following example:

```
# ping  -c 2 oradb9.pbm.ihost.com
PING oradb9.pbm.ihost.com (129.40.181.112) 56(84) bytes of data.
64 bytes from oradb9.pbm.ihost.com (129.40.181.112):icmp_seq=1 ttl=64 time=0.022
ms
64 bytes from oradb9.pbm.ihost.com (129.40.181.112):icmp_seq=2 ttl=64 time=0.025
ms
--- oradb9.pbm.ihost.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1034ms
```

If needed, add the fully Qualified Domain Name to the HOSTNAME file by using the **hostname** command. The following example demonstrates showing the **hostname** command and its results:

```
# hostname -f
oradb9.pbm.ihost.com
```

The following example shows the command and results of displaying the contents of the HOSTNAME file:

```
# cat /etc/hostname
oradb9
```

4.2.8 Oracle Database storage options

In this section, we describe the following disk storage configuration options that are available for database files on LinuxONE:

- Open Storage FCP/SCSI LUNs
- IBM Extended Count Key Data (ECKD) DASD with HyperPAV or PAV

To configure disk storage for LinuxONE, assign the LUNs (with multiple paths) to Linux to ensure high availability and better performance by spreading the I/O workload across more channel paths and host bus adapters (HBAs).

FCP/SCSI

If FCP/SCSI open system storage is used for your ASM disk devices, it is important to configure a multipathing and UDEV rule that ensures device persistence and disk permissions for the Oracle ASM disks.

To configure multiple disk paths with FCP/SCSI open storage, the `/etc/multipath.conf` file must be set up to enable multipathing for high availability to the disk paths and for performance reasons.

Note: It is also recommended to consult your storage vendor for their recommendations for the `multipath.conf` settings for the Linux distribution and level that you use.

It is also recommended to use the `user_friendly_names` parameter in the `multipath.conf` so that the disk permissions for the Oracle Database ASM LUNs can be set correctly across various Linux guests for the same Oracle cluster. A `multipath.conf` file is shown in Example 4-3.

Example 4-3 Multipath configuration file

```
# cat /etc.multipath.conf
defaults {
    dev_loss_tmo            inifinity
    fast_io_fail_tmo        5
    user_friendly_names     yes
}
blacklist {
#       devnode ".*"
}
multipaths {
       multipath {
               wwid 20020c240001221a8
               alias    ASMDISK00
          }
}
```

To restart the multipath service so that any changes take effect, run the following command to read in the new `multipath.conf` settings:

```
# /sbin/multipath -F
# /sbin/multipath -v2
# systemctl restart multipathd.service
```

The `/etc/udev/rules.d/12-dm-permissions.rules` file also must be configured to set the `/dev/mapper/` disk names to the correct device permissions for the `oracle` or `grid` Linux user ID that owns the disk storage devices.

The following example shows that any disks that are configured with a name that begins with `ASM*` is owned by the oracle Linux user (`grid` or `oracle`):

```
# cat  /etc/udev/rules.d/12-dm-permissions.rules
ENV{DM_NAME}=="ASM*",OWNER:="oracle",GROUP:="dba",MODE:="660"
```

A sample `/etc/udev/rules.d/12-dm-permissions.rules` template file is provided at `/usr/share/doc/device-mapper-<your kernel level>/12-dm-permissions.rules` for Red Hat systems.

The following Linux commands can be used to enable any changes to a UDEV rule file:

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

Then, verify that the disk permissions are set correctly by using the **ls** command of the ASM disk devices:

```
# ls -lL ASM*
brw-rw---- 1 oracle dba 253, 14 Jun 12 08:30 ASMSAN00
brw-rw---- 1 oracle dba 253, 15 Jun 12 08:00 ASMSAN01
brw-rw---- 1 oracle dba 253, 16 Jun 12 08:00 ASMSAN02
brw-rw---- 1 oracle dba 253, 17 Jun 12 08:00 ASMSAN10
brw-rw---- 1 oracle dba 253, 18 Jun 12 08:00 ASMSAN11
```

ECKD/DASD

With ECKD/DASD storage, it is recommended to set up disk path aliases with HyperPAV. z/VM manages the high availability of the disk paths and HyperPAV provides more I/O paths to help avoid any disk I/O pathing bottlenecks.

To configure the ECKD/DASD devices, it is helpful to use the `lsdasd` device name in the disk name to make it easier to assign the disk storage, as shown in the following example:

```
# lsdasd
Bus-ID     Status      Name      Device  Type  BlkSz  Size      Blocks
==============================================================
0.0.0200   active      dasda     94:0    ECKD  4096   7043MB    1803060
0.0.7406   active      dasdp     94:60   ECKD  4096   7043MB    1803060
0.0.7407   active      dasds     94:72   ECKD  4096   7043MB    1803060
```

To configure the correct disk permissions for the Oracle ASM disk user to write and read from the ASM devices, a UDEV rule is needed to assign the correct disk permissions.

If configuring Oracle RAC, the same disk devices should be set up as shareable on each of the nodes in the RAC cluster, Configure a new UDEV, such as `/etc/udev/rules.d/99-udev-oracle.rules`, as show in the following example:

```
# cat 99-udev-oracle.rules
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.7406", SYMLINK+="oracleasm/asm7406",
GROUP="oinstall", OWNER="oracle", MODE="0660"
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.7407", SYMLINK+="oracleasm/asm7407",
GROUP="oinstall", OWNER="oracle", MODE="0660"
```

A restart is required for modifications of UDEV rules to take effect. You can run the following UDEV commands for any udev changes to take effect. This change can be done dynamically. Then, run the **ls** commands to confirm that the file permissions are set correctly (based on the devices that are configured in your UDEV rules):

```
# udevadm control -reload-rules
# udevadm trigger
# ls -lL /dev/oracleasm
brw-rw---- 1 oracle oinstall 94,  61 Jun 12 19:18 asm7406
brw-rw---- 1 oracle oinstall 94,  73 Jun 12 17:15 asm7407
```

4.2.9 Required software directories and pre-installation steps

Identify or create the directories that are described in this section that are needed by the Oracle software installer. The examples in this section use `/u01` for the mount point directory.

You can use any naming convention that is consistent across Oracle nodes, especially when configuring an Oracle Real Application Cluster. The Oracle software mount points must be consistent between all the nodes in the Oracle cluster.

Oracle base directory

The following directory is used:

```
ORACLE_BASE = /u01/app/software_owner
```

In this installation example, we created a 20 GB Logical Volume to host Oracle software and data, where `software owner` is the operating system username of the software owner that is installing the Oracle software; for example, `oracle` or `grid`.

Oracle inventory directory

This Oracle Inventory directory stores an inventory of all software that is installed on the system. It is required and shared by all Oracle software installations on a single system. If an existing or old Oracle Inventory paths exist, the OUI continues to use that Oracle Inventory.

The `/var/opt/oracle/oraInst.loc` file is the location of the Oracle inventory path.

The following command is used to check the contents and the existence of the `oraInst.loc` file:

```
cat /var/opt/oracle/oraInst.loc
inventory_loc=/oraInventory inst_group=oinstall
```

Oracle software file system

The Oracle Home directory is where the Oracle software product is installed. A typical `ORACLE_HOME` environment variable setting is described in this section.

If ASM is not used, create a Linux file system by using the Yast or Linux commands that are described next. Create physical volumes, a volume group, and add physical volumes to it and then, create a Logical Volume with stripping. Also, create an ext4 file system.

Use the following commands to create the initial mount point:

```
mkdir /u01
chown -R oracle:oinstall /u01
chmod -R 775 /u01
```

To create a simple file system, complete the following steps:

1. Issue the following commands for ECKD DASD:

   ```
   dasdfmt -p -y -b 4096 -f  /dev/dasdan
   fdasd -a /dev/dasdan
   pvcreate /dev/dasdan1
   vgcreate orvg /dev/dasdan1
   ```

2. Issue the following commands for FCP/SCSI:

   ```
   fdisk  /dev/mapper/mymp1
   pvcreate /dev/mapper/mymp1p1
   vgcreate orvg /dev/mapper/mymp1p1
   ```

3. Create the logical volume and file system for both disk types:

```
lvcreate --name orlv -l 100%FREE orvg
mkfs.xfs  /dev/orvg/orlv
```

4. Mount the logical volume on ORACLE_BASE=/u01:

```
mount -t xfs /dev/orvg/orlv  /u01
```

5. For resiliency after system reboot, insert the mount statement into the /etc/fstab file:

```
/dev/oravg/oralv                /u01                    xfs     defaults        1 1
```

Oracle OUI GUI Installer

If the GUI is used when running OUI, Database Configuration Assistant (DBCA), or any related tasks, run the following command as the oracle DB or grid user:

```
export CV_ASSUME_DISTID=RHEL8.0
```

Known installation symvers issue: Oracle installer (for example, runInstaller) might encounter errors because a file symvers that is no longer available on RHEL 8 systems under /boot. This potential issue is confirmed by Red Hat support note 5174441. The file in /usr/lib/modules no longer not includes the kernel version in the file name. This issue is also a known Oracle Bug: Bug 32501230 Remove Symvers Checks in CVU For RHEL8.

To work around this problem, install the latest Oracle RU that is greater than version 19.11 whereby the latest runInstaller does not encounter this error:

```
echo $ export CV_ASSUME_DISTID=RHEL8.0

/u01/db/runInstaller -applyRU /u01/software/<19.x patch area>
```

4.2.10 Tiger VNC xterm window for interactive GUI installations (optional)

Install or configure the VNC viewer or xterm emulator if you want to complete GUI-based Oracle installations by using runInstaller, asmca, dbca, or netca Oracle GUI installer programs.

Silent installation options also are available for these Oracle installers. However, the first time an installation is done by using the GUI wizards provides the system checks to verify your system configuration and then can be scripted with the silent installation later.

For Red Hat systems, the VNC Configurator provides a good summary of the steps that are completed to install the vnc server and client RPMs. Then, the vnc xstartup file is configured based on the xterm emulator that is used:

```
echo "Install VNC server"
yum install tigervnc-server

# echo "Install VNC client"
# yum install tigervnc

# For Minimal GUI twm
# yum install xterm xorg-x11-twm
```

For more information about the RHEL 8 installation, see this Red Hat web page (log in required).

4.2.11 Pre-installation checking

This section provides the pre-installation Linux steps that are required to set up Oracle RDBMS 19c release on Red Hat 8 Update 3 (or higher 8.x version) on IBM: Linux on System z (s390x). This section also describes more pre-installation checking that you must perform after you ensure that a GUI is established.

Operating system requirements for IBM LinuxONE

RHEL 8 servers must be running Red Hat kernel 4.18.0-240.el8.s390x (s390x) or higher to Install Oracle RDBMS 19c on IBM LinuxONE.

Oracle Rational Portfolio Manager Checker with Red Hat Enterprise Linux Server

In this section, we describe how to use the Oracle Rational Portfolio Manager Checker utility to verify that the required Linux packages are installed on the operating system before starting the Oracle Database or Oracle Grid Infrastructure installation.

Download the suitable Oracle Rational Portfolio Manager Checker utility from the MOS websites based on Oracle Database 19c Release and the Red Hat Enterprise Linux Server distribution level (for example, RHEL 8.3 or 19.11). The Rational Portfolio Manager Checker when Installing Oracle Database 19c on IBM Linux on System z (Doc ID 2553465.1) Oracle Support note contains the Rational Portfolio Manager checker scripts that are available for download (log in required).

After you download the suitable Rational Portfolio Manager checker, upload it to your system and install it as root. Use the Red Hat **`yum`** command to install any required RPMs and their dependencies.

The downloaded file from the support note is `RH8-DB-19c.s390x.rpm`.

To install, use the Red Hat yum utility to install the required Linux packages and their dependencies:

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

Running the **`yum`** command automatically pulls in the required RPMs if your yum repository is set up. If you do not have your Rational Portfolio Manager repository set up, you can install the Rational Portfolio Manager dependency checker by using the following command:

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

4.3 Linux large pages and Oracle Database

It is recommended for performance and availability reasons to implement Linux large pages for Oracle databases that are running on IBM LinuxONE systems. Linux large pages are especially beneficial for systems in which the database's Oracle SGA is greater than 8 GB or if more than 50 database connections are used.

4.3.1 Setting up Linux large pages

Complete the following steps to implement large pages:

1. Each database that is planned for Linux large pages cannot use AMM by setting the `MEMORY_MANAGEMENT` parameter. It is recommended to use Automatic Shared Memory Management (ASMM) by setting the `SGA_TARGET` and `PGA_AGGREGATE_TARGET` Oracle parameters.
2. The Oracle `use_large_pages` parameter can be set to `true`, `false`, or `only`. If you have a LinuxONE system with one large database that requires large pages, and other smaller database that does not require large pages, you can set the larger SGA database to `only` and the smaller database to `false`. The default setting for `use_large_pages` is `true`.
3. Set or update the following parameters in the `/etc/sysctl.conf` kernel parameters, with the `vm.nr_hugepages` value being the size of the Oracle SGA in 1 MB pages rounded up to the nearest granule size of the databases SGA sizes. This `nr_hugepage` value is the sum of all the Oracle Database SGAs on the Linux guest requiring large pages:

   ```
   vm.nr_hugepages = ((sum of all large page SGA's)* 1024)+16 (Granule)= N
   vm.hugetlb_shm_group = <Oracle user Linux group number from /etc/group>
   ```

   Note: Be careful checking that these kernel parameters are correct. A mis-configured `vm.nr_hugepage` kernel parameter setting can cause a Linux system to start incorrectly because of a lack of memory.

4. At the Linux level, it is recommended to set the `/etc/security/limits.conf` file's `memlock` `ulimit` parameters to `unlimited`. The Linux kernel parameter (`nr_hugepages`), Oracle SGA values, and the `ulimits` must be synchronized. By setting the `ulimit` to `unlimited`, one less change is needed if the Oracle SGA or Linux large page values must be increased, as shown in the following example:

   ```
   # cat /sys/kernel/mm/transparent_hugepage/enabled
   cat /etc/security/limits.conf | grep memlock | grep -v grep
   # Use memlock for Huge Pages  support
   oracle        soft    memlock  unlimited
   oracle        hard    memlock  unlimited
   grid          soft    memlock  unlimited
   grid          hard    memlock  unlimited
   ```

5. Restart your Linux Image and restart Oracle for the changes to take effect.
6. Review your Oracle Database alert log to verify that the database was started with large pages enabled.

4.3.2 Disabling transparent HugePages

It is recommended for performance and stability reasons to disable transparent HugePages. Transparent HugePages are different from Linux large pages, which are still highly recommended to use. You can check whether your system features transparent HugePages enabled by using the following command:

```
# cat /sys/kernel/mm/transparent_hugepage/enabled
[always] madvise never
```

To disable Transparent HugePages, it is recommended to update the LinuxONE grub file by using `transparent_hugepage=never` at the end of the parameters line. Then, run **`zipl -VV`** for the changes to take effect with the next restart of your system.

Complete the following steps:

1. Use the grubby utility to add `transparent_hugepage=never` to the end of the kernel parameters line:

   ```
   grubby --update-kernel=ALL --args="transparent_hugepage=never"
   ```

2. Run the **`zipl -VV`** command to activate the Linux kernel parameters for next system restart:

   ```
   # zipl -VV
   Using config file '/etc/zipl.conf'
   Using BLS config file
   '/boot/loader/entries/f50d8e9423f84071a0139c213bc16f91-4.1
   8.0-240.el8.s390x.conf'
   Using BLS config file
   '/boot/loader/entries/f50d8e9423f84071a0139c213bc16f91-0-r
   escue.conf'
   Target device information
     Device.......................: 5e:00
     Partition....................: 5e:01
     Device name..................: dasda
     Device driver name...........: dasd
     DASD device number...........: 0200
     Type.........................: disk partition
     Disk layout..................: ECKD/compatibledisk layout
     Geometry - heads.............: 15
     Geometry - sectors...........: 12
     Geometry - cylinders.........: 60102
     Geometry - start.............: 24
     File system block size.......: 4096
     Physical block size..........: 4096
     Device size in physical blocks..: 10818336
   Building bootmap in '/boot'
   Building menu 'zipl-automatic-menu'
   Adding #1: IPL section 'Red Hat Enterprise Linux (4.18.0-240.el8.s390x) 8.3
   (Ootpa)' (default)
     initial ramdisk...: /boot/initramfs-4.18.0-240.el8.s390x.img
     kernel image......: /boot/vmlinuz-4.18.0-240.el8.s390x
     kernel parmline...: 'root=/dev/disk/by-path/ccw-0.0.0200-part1
   rd.dasd=0.0.0200
   rd.znet=qeth,0.0.1410,0.0.1411,0.0.1412,layer2=1,portno=0,portname=DUMMY
   vmalloc=4096G transparent_hugepage=never'
   ```

3. Restart your Linux guest; for example, `shutdown -r now`.

4. Verify that transparent HugePages are disabled by using the following command:

   ```
   # cat /sys/kernel/mm/transparent_hugepage/enabled
   always madvise [never]
   ```

4.4 Enabling Linux random number generation

For Oracle Database versions 12.2 and later, it is important for systems to be configured to provide good random number generation to use the new security features of an Oracle Database installation.

Oracle version 12.2+ now uses `/dev/random` rather than `/dev/urandom`. This change means that without correctly configured entrop or random numbers replenishment, you might encounter intermittent access issues, such as: `ORA-01017: invalid username/password` on your system.

4.4.1 Entropy replenishment configuration

For this configuration, the general rule is that `/proc/sys/kernel/random/entropy_avail` always is greater than 1000 random numbers available. To validate this number, run the following command:

```
# cat /proc/sys/kernel/random/entropy_avail
348
```

This value (348) is less than 1000 and needs replenishment for Oracle to authenticate user connections correctly.

The Pseudo Random Number Generator (PRNG) device can be used to provide statistically significant random numbers for running Oracle databases. IBM LinuxONE servers use the CP Assist for Cryptographic Function (CPACF) on each processor chip to replenish random numbers.

The first step is to verify whether your system includes the PRNG driver loaded. To verify this information, the `/dev/prandom` file should exist on your LinuxONE system:

```
# ls -la /dev/prandom
```

The latest RHEL 8 distribution levels should have the driver `prandom` driver configured by default. If for some reason the `prandom` driver does not exist on your system, you can configure or load the driver as described next.

The `/dev/prandom` pseudo random module (PRNG) module can be loaded at runtime by using the following command:

```
# modprobe prng
```

To load the `prng` module, configure in a local boot script (such as `rc.local`) to load the `prng` module after a system restart; for example:

```
# cat /etc/rc.d/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.
touch /var/lock/subsys/local
modprobe prng
```

4.4.2 Enabling Red Hat rngd service for random number generation

Complete the following steps to install and enable the `rngd` service if the module is not installed:

1. Install the `rngd` service if it is not installed:

   ```
   # yum install -y rng-tools
   ```

2. Ensure the rngd service is enabled:

   ```
   # systemctl enable --now rngd
   Removed /etc/systemd/system/sysinit.target.wants/rngd.service.
   Created symlink /etc/systemd/system/sysinit.target.wants/rngd.service ?
   /etc/systemd/system/rngd.service.
   ```

3. Edit the `/etc/systemd/system/rngd.service` file and replace the line that starts with `ExecStart` with the random number methodology that is used (IBM recommends `prandom` or `hwrng`).

   Replace `ExecStart=/sbin/rngd -f --fill-watermark=0 --rng-device /dev/urandom` with `ExecStart=/sbin/rngd -f --fill-watermark=0 --rng-device /dev/prandom`.

4. Reload the systemd configuration:

   ```
   # systemctl daemon reload
   ```

5. Restart the `rngd` service:

   ```
   # systemctl restart rngd
   ```

   If you see the following error during installation, ensure that the `rng-tools` Rational Portfolio Manager is installed:

   ```
   Failed to init entropy source 1: TPM RNG Device
   ```

6. Verify that the `rngd` service is running correctly:

   ```
   # systemctl status rngd
   rngd.service - Hardware RNG Entropy Gatherer Daemon
      Loaded: loaded (/etc/systemd/system/rngd.service; enabled; vendor preset:
   enabled)
      Active: active (running) (thawing) since Thu 2021-10-28 15:35:22 EDT; 4min
   10s ago
    Main PID: 1373716 (rngd)
       Tasks: 5 (limit: 412334)
      Memory: 2.2M
      CGroup: /system.slice/rngd.service
              ??1373716 /sbin/rngd -f --fill-watermark=0 --rng-device /dev/urandom
   Oct 28 15:35:22 oradb9 systemd[1]: Started Hardware RNG Entropy Gatherer
   Daemon.
   Oct 28 15:35:22 oradb9 rngd[1373716]: Initializing available sources
   Oct 28 15:35:22 oradb9 rngd[1373716]: Initializing entropy source hwrng
   Oct 28 15:35:24 oradb9 rngd[1373716]: Initializing AES buffer
   Oct 28 15:35:24 oradb9 rngd[1373716]: Enabling JITTER rng support
   Oct 28 15:35:24 oradb9 rngd[1373716]: Initializing entropy source jitter
   Oct 28 15:38:15 oradb9 systemd[1]: rngd.service: Current command vanished from
   the unit file, execution of the command list won't be resumed.
   ```

7. Verify that your system features the required amount of entropy greater than 1000:

```
# cat /proc/sys/kernel/random/entropy_avail
3495

# ps -ef | grep rng | grep -v grep
root         152       2  0 Sep21 ?        00:25:50 [hwrng]
rngd     1376905       1  2 15:42 ?        00:00:18 /sbin/rngd -f
--fill-watermark=0 --rng-device /dev/prandom
```

5

Setting up Oracle 19c on SUSE Linux Enterprise Server 15 SP3 on IBM LinuxONE servers

This chapter covers the pre-installation setup steps that are required for setting up Oracle 19c on an IBM LinuxONE system with SUSE Linux Enterprise Server 15 Service pack 3 or later. We include information about how to obtain Oracle documentation, code, and My Oracle Support (MOS) Notes.

For product longevity and Oracle Technology Network (OTN) patching, IBM and Oracle recommend upgrading to Oracle 19c, which is the latest long-term release with the longest support end date at the time of writing. Oracle 19c is the terminal release of Oracle 12.2.

This chapter includes the following topics:

5.1 Obtaining Oracle Documentation, Downloads, and Support Notes

This section includes the following topics:

- Master Note of Linux OS requirements for Oracle Database Server
- Obtaining the Oracle 19c Software for IBM LinuxONE

Changes in the recommended Oracle RPMs and kernel parameters can occur between release levels. We have provided the references throughout this chapter, with the corresponding MOS reference note, for the most up-to-date guidance on a particular topic and verification.

5.1.1 Master Note of Linux OS requirements for Oracle Database Server

The Master Note is intended to provide an index and references to the most frequently used MOS articles concerning Linux OS requirements for an Oracle Database Software Installation.

For more information about the Master Note, see Primary Note of Linux OS Requirements for Database Server (Doc ID 851598.1), which is available by logging in to your Oracle Support account. To see the key notes for Oracle 19c Installation with Linux on System z for SUSE Linux Enterprise Server 15 Oracle 19c, see Requirements for Installing Oracle Database 19c 64-bit Linux on IBM System z on SUSE Linux Enterprise Server (Doc ID 2813961.1).

Oracle 19c is certified by Oracle for SUSE Linux Enterprise Server 15 Update 3 or later. Oracle is certified for the base Major OS level. All service packs above the minimum required distribution level are supported by Oracle.

The minimum required Release Update (RU) from Oracle is 19.12 with SUSE Linux Enterprise Server 15 SP3+ installations.

The latest available SUSE 15.x distribution service pack level is recommended for new installations to have the most up-to-date features and security functions for running Oracle databases.

The release schedule of current database releases is described in Release Schedule of Current Database Releases (Doc ID 742060.1), which is available by logging in to your Oracle Support account. Use the **My Oracle Support Certification** tab on the MOS site to determine the latest release status of Oracle version releases with Linux on System z.

5.1.2 Obtaining the Oracle 19c Software for IBM LinuxONE

Depending on your company's Oracle licensing agreement, the latest Oracle code can be downloaded from the Oracle Software Delivery cloud or from the Oracle Technology Network (OTN) developers download site.

If you have a commercial Oracle license, you should download your software from Oracle Software Delivery Cloud.

For trial and developer network downloads, the Oracle license agreement should be fully understood before agreeing and downloading the Oracle 19c database software from the OTN download site.

After you are logged in to the Oracle Software Delivery cloud by using your Oracle user ID and password, enter `Oracle Database` in the search field for the Oracle Enterprise Edition that you want to install, as shown in Figure 5-1.

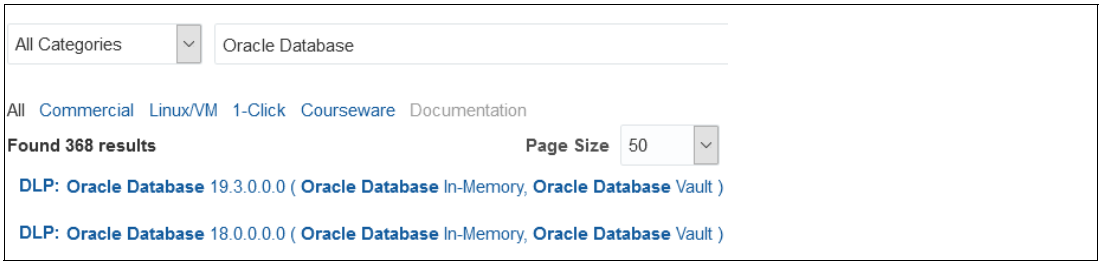

Figure 5-1 Oracle Software Delivery Cloud Download

Click the **+** sign of **Add to Cart** (see Figure 5-2) for the software version that you want to install, and then click **View Cart** in the upper right of the window. A window opens with an option to check out. Click **Checkout** link.

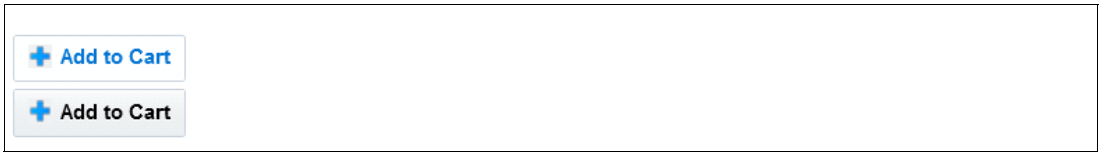

Figure 5-2 Add to Cart

In the software catalog for the selected software, Select **Linux on IBM System** z, as shown in Figure 5-3. Then, select the software components that you need for your installation under the Platforms / Languages column.

Figure 5-3 Oracle Software Delivery Cloud Select Software components

Most installations require that the Oracle Database software is downloaded. If you are using Oracle ASM, or installing Oracle RAC or Oracle RACONE, you also need the Oracle Database Grid Infrastructure software. For applications that must connect to an Oracle Database that is running on Linux on System z by using sql*net network connectivity, then the Oracle Database Client also should be downloaded.

After all the software is selected, click **Download** to start downloading the software. If this download is the first time that you are downloading software, you might be prompted to download and install the Oracle download installer first. After the installer is configured, the selected Oracle distribution software is downloaded to your workstation. You also can download the software by clicking the link for each software product one at a time.

In addition to the base software, the latest Patch Set or RU should be downloaded from MOS.

Oracle 19c provides the latest RU for both the Grid Infrastructure and Database from MOS. The latest RU for your release can be found in Master Note for Database Proactive Patch Program (Doc ID 756671.1).

An Oracle 19c base installation is release 19.3 for Linux on System z. It is recommended to be on the latest Grid Infrastructure and Database RU when the patches are made available every fiscal quarter. The minimum required RU is 19.13 for SUSE Linux Enterprise Server 15 SP3 systems running Oracle Database on IBM LInuxONE.

The latest (as of September 2022) Oracle 19.16 Grid Infrastructure and Database security and RU patch is 34133642. Follow the readme file for the patch that you plan to install. Make sure to download the latest OPatch installer utility patch 6880880 for the RU patch, as described in the patch readme file.

5.2 Setting up a SUSE Linux Enterprise Server 15 to run Oracle 19c

This section describes the common steps that are needed to implement an Oracle Database, including Oracle Grid Infrastructure, on a SUSE Linux Enterprise Server 15 Enterprise Linux Server SP3 or greater to run Oracle 19c on an IBM LinuxONE system.

The minimum Linux kernel version level that is certified by Oracle is SUSE Linux Enterprise Server 15: 5.3.18-57-default s390x or later.

5.2.1 Important information

In this section, the following pre-installation verification information is provided to give you a basic understanding before starting an Oracle Database installation on LinuxONE:

- Graphical user interface (GUI)

  A GUI is required to run the Oracle Installers interactively. We use the Linux vncserver on LinuxONE to start an X server and a VNC viewer client to establish a terminal emulator xterm session.

  To use GUIs, such as the Oracle Universal Installer (OUI), configuration assistants, and Oracle Enterprise Manager, set the display to a system with X Window System server packages.

  In our environment, we start a vncserver on LinuxONE, and use a VNC viewer client to establish the connection with the vncserver.

  You also can configure a silent or scripted installation for cloud automation deployments, which is described in Chapter 8, “Installing Oracle” on page 101.

- Linux minimal installation

  A minimal Linux installation lacks many RPMs that are required for database installation. Therefore, you must use an RPM package for Linux release to install the required packages, which are listed in 5.2.13, “Required Linux RPMs and pre-installation checking” on page 74.

- Oracle Automatic Storage Management

  To use Oracle Automatic Storage Management (ASM), you must first install Oracle Grid Infrastructure for a stand-alone server before you install and create the database.

  Most customers use either Oracle ASM to store database files or a file system for single-instance installations. For Oracle Real Application Cluster (RAC) installations, you can use Oracle ASM or IBM Spectrum Scale to store database files across Oracle RAC cluster nodes.

  To use Oracle ASM on LinuxONE, you must first install Oracle Grid Infrastructure for a stand-alone server before you can install and create the database.

  Although you can use the SUSE `orarun` RPM package to configure the user or groups, ulimits, and kernel parms automatically, it is a best practice to configure manually these settings for greater control.

5.2.2 General requirements and information for Oracle Database on LinuxONE

To obtain general LinuxONE information from a Linux session, run the following commands, ensuring that the system started with run level 3 or 5:

1. Check information about the CPU that is defined in LinuxONE:

   ```
   cat /proc/cpuinfo
   ```

2. Check information about the RAM that is defined in LinuxONE:

   ```
   cat /proc/meminfo | grep MemTotal
   cat /proc/meminfo | grep SwapTotal
   ```

3. Check the amount of space that is available in the `/tmp` directory:

   ```
   df -h /tmp
   ```

4. Check the amount of free disk space on the system (in gigabytes):

   ```
   df -h
   ```

5. Check the amount of free RAM and disk swap space on the system:

   ```
   free
   ```

6. Check whether the system architecture can run the software:

   ```
   uname -m
   ```

   This command should return `s390x`.

5.2.3 Server minimum requirements

The following minimum requirements for the server must be met:

- Disk space requirements for Oracle 19c on IBM LinuxONE

  Ensure that the system meets the disk space requirements for the software files:

  - Enterprise Edition database: 7.5 GB - 8.6 GB of disk space is required for the Oracle software, depending on the installation type. You might need some extra space for patching logs. 10 GB minimum of disk space is recommended.
  - Grid Infrastructure: 8.9 - 10 GB is required at a minimum to install an Oracle ASM single-instance (restart) or an Oracle RAC installation. It is a best practice to provide extra space for Oracle Grid Infrastructure log files. 20 GB of disk space minimum is recommended.

- Recommended memory

  A minimum of 4 GB of virtual memory is recommended in the Linux guest for Oracle 19c installations of Oracle Database, including the Grid Infrastructure requirements.

  For single-instance, file system-based installations, 2 GB minimum of virtual memory is recommended.

> **Note:** Although 2 GB is the minimum amount of virtual memory that is required, the Oracle OPatch utility requires at least 3072 MB (3 GB) for Oracle patching.

5.2.4 Linux swap recommendations

This section describes Linux swap recommendations for running an Oracle 19c database on a SUSE Linux Enterprise Server 15 Linux guest running on an IBM LinuxONE system. Swap disk space should be configured proportional to the system's physical memory, as shown in Table 5-1.

Table 5-1 Linux Swap size recommendations

RAM	Swap space
1 - 8 GB	Two times the size of the memory
8 - 32 GB	Equal to the size of the memory
More than 32 GB	32 GB

The Oracle guidelines for Linux swap can be reduced when needed if there is enough memory to run all the databases in the Linux guest. Linux swapping memory to disk is not recommended for an Oracle Database because it might make the database unresponsive. The Oracle Installer requires at bare minimum 500 MB of configured Linux swap to complete an installation, and 1 GB of swap for database upgrades.

Customers that use IBM LinuxONE can use a layered virtual in-memory disk (VDisk) for the Linux swap devices. Linux swap to a memory device (VDisk) is much quicker than using a physical disk storage device. IBM z/VM does not allocate the memory that is used by a VDisk until the first swap page is "Used".

Figure 5-4 illustrates an example of a recommended VDisk configuration with a VDisk being used for the first and second level swaps with higher priority. Then, a physical disk or DASD disk can be used as a lower priority swap in the case of unexpected memory usage. Linux uses the higher priority swap devices first. When the swap device is fully exhausted, the next priority swap device is used.

```
# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasdo1                             partition       131000  0       10
/dev/dasdp1                             partition       524216  0       5
/dev/mapper/u603_swap3                  partition       6291448 0       1
```

Figure 5-4 Swap VDisk configuration priorities

You can use the Linux **mkswap** and **swapon** commands to configure swapping manually, and you can use the `/etc/fstab` file to configure permanent swap devices, as shown in Figure 5-5. Swap usage is consumed with the highest priority (pri=) number. The default is -1 for swap usage priority.

```
oratest:~ # cat /etc/fstab | grep swap
/dev/disk/by-path/ccw-0.0.0300-part1                    swap       swap pri=10          0 0
oratest:~ # swapon -s
Filename                                Type            Size    Used    Priority
/dev/dasda1                             partition       215900  0       10
oratest:~ # lsdasd | grep 0.0.0300
0.0.0300   active      dasda      94:0    ECKD  4096   210MB     54000
```

Figure 5-5 Swap VDisk setup in fstab

Temporary disk space that is required for installation

A minimum of 1 GB of disk space in the `/tmp` directory on LinuxONE systems is required (and less than 2 TB of disk space). If you do not have enough space in `/tmp` for an Oracle installation, delete any files that you can, or set the `TMP` and `TMPDIR` environment variables to another directory with space when setting the Oracle user's environment.

5.2.5 Linux kernel parameter recommendations

The Linux kernel parameters that are described in this section are the minimum required Linux kernel parameters that are needed for an Oracle 19c installation.

Examples of the required Oracle kernel parameters for a system that uses an 8 GB Oracle System Global Area (SGA) database are shown in Table 5-2. Larger databases that require higher Oracle SGA values might require increased Linux kernel parameter settings based on the size of the Oracle SGA.

Table 5-2 Kernel parameters

Kernel parameter	Example value	File
fs.file-max: Set to 512 x maximum number of user processes in Oracle per database for the Linux guest. 6815744 supports up to 13312 Oracle processes. Increase as needed.	6815744	/proc/sys/fs/file-max
fs.aio-max-nr: This value limits concurrent outstanding requests and should be set to avoid I/O subsystem failure. The formula is aio-max-nr = number of processes per DB * number of databases * 4096.	1048576 3145728 for larger databases	/proc/sys/fs/aio-max-nr
kernel.shmmax: Set this value to the size of at least the largest SGA in bytes. 8589934592 bytes support a maximum shared memory segment of 8 GB. For more information, see Maximum SHMMAX Values for Linux x86 and x86-64 (Doc ID 567506.1).	8589934592 (8 GB)	/proc/sys/kernel/shmmax
kernel.shmall: Set this parameter to a value that is equal to the total amount of shared memory in 4 K pages that the system can use at one time. The sum of all the SGA databases on the system in bytes / 4096. 20907152 supports 8 GB SGA. For more information, see Upon startup of Linux database get ORA-27102: out of memory Linux-X86_64 Error: 28: No space left on device (Doc ID 301830.1).	2097152	/proc/sys/kernel/shmall
kernel.sem (semmsl, semmns, semopm, or semmni): The minimum required semaphores for Oracle.	kernel.sem= 250 32000 100 128	/proc/sys/kernel/sem
kernel.shmmni	4096	/proc/sys/kernel/shmmni
kernel.panic_on_oops: Set this value to allow time for diagnostics in the event of a system restart.	1 second on an Oracle node eviction. *n* seconds delay before a node eviction or restart.	/proc/sys/kernel/panic_on_oops
net.ipv4.ip_local_port_range	9000 65500	/proc/sys/net/ipv4/ip_local_port_range
net.core.rmem_default	262144	/proc/sys/net/core/rmem_default
net.core.rmem_max	4194304	/proc/sys/net/core/rmem_max
net.core.wmem_default	262144	/proc/sys/net/core/wmem_default

Kernel parameter	Example value	File
net.core.wmem_max	1048576	/proc/sys/net/core/wmem_max
vm.swappiness: A value of 1 prevents Linux swapping as much as possible.	1	/proc/sys/vm/swappiness

If the current value of any parameter is higher than the value that is listed in Table 5-2 on page 66, then we suggest to *not* change the value of that parameter to a lower value because in general a larger value means more system resources can be allocated if required. These parameters should be configured in the Linux `/etc/sysctl.conf` file.

Optional performance-related kernel parameters are shown in Table 5-3. They are used as a best practice for running Oracle 19c with SUSE Linux Enterprise Server 15 on an IBM LinuxONE system.

Table 5-3 Optional performance-related kernel parameters

Variable	Example value
vm.nr_hugepages: Sum of the Oracle SGAs + 16 MB, for example, 8208 for 8 GB SGA memory for large pages.	8208 The Linux guest memory always should be sized larger than this value or excessive Linux swap occurs.
vm.hugetlb_shm_group: Set to the `oinstall` group ID number from the `/etc/group` file for the `oracle` user.	1001

Example 5-1 provides a sample `sysctl.conf` file. After the kernel parameter values are updated, validate them by using the following sysctl command:

```
/sbin/sysctl -p
```

Example 5-1 Sample Linux kernel parameters for Oracle

```
# Oracle specific parameters
fs.file-max = 6815744
fs.aio-max-nr = 1048576
#set shmmax to largest sga memory segment in bytes
kernel.shmmax = 8589934592
#set shmall to (sum of sga memory in bytes)/4096
kernel.shmall = 2097152
kernel.sem = 250 32000 100 128
kernel.shmmni = 4096
kernel.panic_on_oops = 1
net.ipv4.ip_local_port_range = 9000 65500
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048576
vm.swappiness = 1
#set nr_hugepages to sum of all Oracle SGAs in MB if using large pages
#vm.nr_hugepages = 8208
vm.hugetlb_shm_group=1001
```

Finally, to ensure that `sysctl.conf` is read on system start, run the following command to enable the system to read the `/etc/sysctl.conf` file when it restarts:

```
# /sbin/chkconfig boot.sysctl on
```

5.2.6 Creating the database installation owner user

The installation process requires at least an Oracle Database installation owner (`oracle`), an Oracle Inventory group (`oinstall`), and an Oracle administrative privileges group (`dba`).

Example 5-2 shows the commands that are used to create the `oracle` user and groups.

Example 5-2 Creating the oracle user and groups

```
/usr/sbin/groupadd oinstall -g 1001
/usr/sbin/groupadd dba -g 1002
/usr/sbin/groupadd asmadmin -g 1003
/usr/sbin/groupadd asmoper -g 1004
/usr/sbin/useradd -m -g oinstall -G dba  oracle
echo "oracle:newPassw0rd" | /usr/sbin/chpasswd
```

For separation of duty, some sites set up a Linux “grid” user ID to manage the grid infrastructure ASM and Oracle Grid Infrastructure components. If so, use the same **useradd** and change password steps that are shown in Example 5-2 for the Linux grid user too.

```
/usr/sbin/useradd -m -g oinstall -G dba grid
echo "grid:newPassw0rd" | /usr/sbin/chpasswd
```

5.2.7 Linux ulimits for Oracle User Accounts

To avoid operational problems with the Oracle Database consuming system resources, the Linux ulimit values must be adjusted to allow for the proper operation of an Oracle Database. If you are configuring a separate user for an Oracle grid, the ulimit values must be set for this user too.

If you are planning to use large pages and the Oracle SGA memory is increased, then setting the ulimit memlock to `unlimited` is one fewer step that is needed to make that change. To do so, run the following commands:

```
oracle soft memlock unlimited
oracle hard memlock unlimited
```

Oracle recommends that the soft and hard limits should be at least 90% of the current memory when HugePages memory is enabled, and at least 3145728 KB (3 GB) when HugePages memory is disabled.

If you are using an extra Oracle Linux user ID such as `grid` to manage the Grid Infrastructure components, define the same ulimit values for the `grid` and `oracle` Linux user IDs by running the following commands:

```
grid soft memlock unlimited
grid hard memlock unlimited
```

If the Linux `oracle` user performs the installation, add the settings that are shown in Example 5-3 on page 69 to your `/etc/security/limits.conf` file. If your current values are higher than these values, leave the higher values in place.

Example 5-3 Sample limits.conf file

```
oracle        soft    nproc   2047
oracle        hard    nproc   16384
oracle        soft    nofile  1024
oracle        hard    nofile  65536
oracle        soft    stack   10240
oracle        hard    stack   10240
#
grid          soft    nproc   2047
grid          hard    nproc   16384
grid          soft    nofile  1024
grid          hard    nofile  65536
grid          soft    stack   10240
grid          hard    stack   10240
# memlock for Huge Pages  support
oracle        soft    memlock  unlimited
oracle        hard    memlock  unlimited
grid          soft    memlock  unlimited
grid          hard    memlock  unlimited
```

Note: When the `limits.conf` file changes, the changes take effect immediately. However, if the `oracle` users are logged in, these changes do not take effect until you log out these users and log them back in.

The ulimits take effect on logging in to a Linux guest either by setting the ulimit values in each Linux user's `.bash_profile` file or updating the system base `/etc/profile` for the system for all Linux users.

The values that are shown in Example 5-4 can be added to the `oracle` and `grid` (if used) user home directory `.bash_profile` files or `/etc/profile` so that the ulimits are set correctly for the `oracle` database user.

Example 5-4 Oracle user limits

```
ulimit -u 16384
ulimit -n 65536
ulimit -s 10240
```

5.2.8 Shared memory file system (/dev/shm)

For Oracle Database installations that are *not* using Linux large pages and using Automatic Memory Management (AMM) with the **`MEMORY_TARGET`** parameter, it is important to ensure that the `/dev/shm` mount area is type `tmpfs` and mounted with the following options:

- With read/write (RW) and execute permissions set on it.
- With noexec or nosuid not set on it.

To check the shared memory file system, run the following command:

```
cat /etc/fstab |grep tmpfs
```

The output for this command looks like the following example:

```
tmpfs     /dev/shm     tmpfs     defaults     0 0
```

If needed, change the mount settings. To do so, as the root user, open the `/etc/fstab` file in a text editor and modify the `tmpfs` line. If you are planning to use AMM, you should set the `tmpfs` file system size to the sum of all the **`MEMORY_TARGET`** values on the system, as shown in the following example:

```
tmpfs      /dev/shm     tmpfs      rw,exec,size=30G 0 0
```

LinuxONE can use large pages of 1 M in size with Oracle running under z/VM, and 2 GB when running under LPAR mode.

The Oracle **`MEMORY_TARGET`** parameter is not eligible for large pages; instead, **`SGA_TARGET`** should be used for the Oracle memory setting when Linux large pages are used.

5.2.9 Host file name and configuration

To ensure that the computer hostname is resolvable, run the **`ping`** command, as shown in Example 5-5, especially for Oracle RAC node members.

Example 5-5 Example of the ping command and its output

```
ping oradb13.pbm.ihost.com
oradb13:/ # ping -c 1 oradb13.pbm.ihost.com
PING oradb13.pbm.ihost.com (129.40.181.78) 56(84) bytes of data.
64 bytes from oradb13.pbm.ihost.com (129.40.181.78): icmp_seq=1 ttl=64 time=0.018
ms

--- oradb13.pbm.ihost.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.018/0.018/0.018/0.000 ms
```

If needed, add the Fully Qualified Domain Name to the `HOSTNAME` file by using the **`hostname`** command. Example 5-6 demonstrates the **`hostname`** command and its results.

Example 5-6 Showing the contents of the HOSTNAME file

```
# cat /etc/hostname
oradb13.pbm.ihost.com
```

5.2.10 Storage options for Oracle Database

In this section, we describe the disk storage configuration options that are available for database files on LinuxONE:

- Open Storage FCP/SCSI LUNs
- IBM Extended Count Key Data DASD with HyperPAV or PAV

To configure disk storage for LinuxONE, assign the LUNs (with multiple paths) to Linux to ensure high availability (HA) and better performance by spreading the I/O workload across more channel paths and host bus adapters (HBAs).

Open Storage FCP/SCSI LUNs

If you are using FCP/SCSI open system storage for your ASM disk devices, it is important to configure multipathing and an UDEV rule that ensures device persistence and disk permissions for the Oracle ASM disks.

To configure multiple disk paths with FCP/SCSI open storage, set up a `/etc/multipath.conf` file to enable multipathing for HA to the disk paths and for performance reasons.

As a best practice, use the **`user_friendly_names`** parameter in `multipath.conf` so that the disk permissions for the Oracle Database ASM LUNs can be set correctly.

Example 5-7 shows a sample `multipath.conf` file.

Note: Consult your storage vendor for their recommendations for the `multipath.conf` settings for the Linux distribution and level that you are using.

Example 5-7 Sample /etc/multipath.conf file for FCP/SCSI open system disk storage

```
defaults {
    dev_loss_tmo            infinity
    fast_io_fail_tmo        5
    user_friendly_names     yes
}
blacklist {
#       devnode ".*"
}
multipaths {
       multipath {
               wwid 20020c240001221a8
               alias    ASMDISK00
          }
}
```

To restart the multipath service so that any changes can take effect, run the following command to read in the new `multipath.conf` settings:

```
/sbin/multipath -v2
systemctl restart multipathd.service
```

The `/etc/udev/rules.d/12-dm-permissions.rules` file also must be configured to set the `/dev/mapper/` disk names to the correct device permissions for the `oracle` or `grid` Linux user ID that owns the disk storage devices.

Example 5-8 shows that any disks that are configured with a **`user_friendly_name`** value that begins with `"ASM*"` are owned by the `oracle` Linux user (either the `grid` or `oracle` user).

Example 5-8 Example 12-dm-permissions.rules file for FCP/SCSI open system disk storage

```
ENV{DM_NAME}=="ASM*",OWNER:="oracle",GROUP:="dba",MODE:="660"
```

You can get a sample `/etc/udev/rules.d/12-dm-permissions.rules` template file from `/usr/share/doc/packages/device-mapper/12-dm-permissions.rules` on SUSE Linux Enterprise Server 15 Linux systems.

To enable any changes to the UDEV rule, run the following Linux commands:

```
udevadm control -reload-rules
udevadm trigger
```

Then, verify that the disk permissions are set correctly by running the following **ls** command:

```
ls -lL /dev/mapper/ASM*
brw-rw---- 1 oracle dba 253, 29 Jun 24 06:16 /dev/mapper/ASMFLASH00
brw-rw---- 1 oracle dba 253, 30 Jun 24 06:16 /dev/mapper/ASMFLASH01
```

IBM Extended Count Key Data DASD with HyperPAV or PAV

With Extended Count Key Data (ECKD)/DASD storage, it is a best practice to set up disk path aliases with HyperPAV, as described in Chapter 1, “A shared commitment to client value with the IBM and Oracle alliance” on page 1. z/VM manages the HA of the disk paths, and HyperPAV provides extra I/O paths to help avoid any disk I/O pathing bottlenecks.

To configure the correct disk permissions for the Oracle ASM disk user to read/write from the ASM devices, use a UDEV rule to assign the correct disk permissions.

To configure the ECKD/DASD devices, use the "`lsdasd`" device name in the disk name to make it easier to assign the disk storage.

If you are configuring Oracle RAC, the same disk devices should be set up as sharable on each of the nodes in the RAC cluster, Configure a new UDEV rule for each Oracle ASM DASD volume in `/etc/udev/rules.d/99-udev-oracle.rules`, as shown in the following example:

```
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.b3c0", SYMLINK+="oracleasm/asmb3c0",
GROUP="asmadmin", OWNER="oracle", MODE="0660"
```

A restart of udev is required for modifications to the UDEV rules to take effect. For SUSE Linux Enterprise Server 15, you can run the following UDEV commands for any udev changes to take effect.

```
udevadm control --reload-rules
udevadm trigger
```

This change can be done dynamically. Then, run the **ls** commands to confirm that the file permissions are set correctly (based on the devices that are configured in your UDEV rules), as shown in Example 5-9.

Example 5-9 Referencing the 99-udev-oracle.rules file for ECKD/DASD storage devices for database files

```
# lsdasd
Bus-ID    Status    Name      Device  Type         BlkSz  Size      Blocks
==============================================================================
0.0.b3c0  active    dasdbr    94:276   ECKD         4096   42259MB   10818360
0.0.b3c1  active    dasdbu    94:288   ECKD         4096   42259MB   10818360
# cat 99-udev-oracle.rules
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.b3c0", SYMLINK+="oracleasm/asmb3c0",
GROUP="asmadmin", OWNER="oracle", MODE="0660"
ACTION=="add|change", ENV{ID_PATH}=="ccw-0.0.b3c1", SYMLINK+="oracleasm/asmb3c1",
GROUP="asmadmin", OWNER="oracle", MODE="0660"

# udevadm control --reload-rules && udevadm trigger

# ls -lL /dev/oracleasm
brw-rw---- 1 oracle oinstall 94,  61 Jun 12 19:18 asmb3c0
brw-rw---- 1 oracle oinstall 94,  73 Jun 12 17:15 asmb3c1
```

5.2.11 Required software directories

Identify or create the following directories, which are needed by the Oracle software installer. For the examples in this section, we use `/u01` for the mount point directory.

Oracle Base directory

The following directory is used:

```
ORACLE_BASE = /u01/app/software_owner
```

In this installation example, we created a 20 GB logical volume to host Oracle software and data, where `software_owner` is the operating system username of the software owner that is installing the Oracle software, for example, `oracle` or `grid`.

Oracle Inventory directory

This Oracle Inventory directory stores an inventory of all software that is installed on the system. It is required and shared by all Oracle software installations on a single system. If existing or earlier Oracle Inventory paths exist, the OUI continues to use that Oracle Inventory.

The Oracle Inventory path is in the `/var/opt/oracle/oraInst.loc`.file.

Example 5-10 shows the command that is used to check the contents and the existence of the `oraInst.loc` file. The results display the contents of the file.

Example 5-10 Examining oraInst.loc

```
# cat /var/opt/oracle/oraInst.loc
 inventory_loc=/oraInventory inst_group=oinstall
```

Oracle Home directory

The Oracle Home directory is where the Oracle software product is installed. A typical **`ORACLE_HOME`** environment variable setting is as follows:

```
ORACLE_HOME=$ORACLE_BASE/oracle_base/product/19c/dbhome_1
```

If ASM is not used, create a Linux file system by using the following YaST or Linux commands. Create physical volumes, a volume group, and add physical volumes to the volume group. Then, create a logical volume with striping, and an ext4 file system.

To create the initial mount point, run the following commands:

```
mkdir /u01
chown -R oracle:oinstall /u01
chmod -R 775 /u01
```

To create a simple file system, we completed the following steps:

1. Depending on the disk type that you use, run one set of the following commands:
   - For ECKD DASD, run the following commands:

     ```
     dasdfmt -p -y -b 4096 -f  /dev/dasdan
     fdasd -a /dev/dasdan
     pvcreate /dev/dasdan1
     vgcreate orvg /dev/dasdan1
     ```

   - For FCP/SCSI, run the following commands:

     ```
     disk  /dev/mapper/mymp1
     pvcreate /dev/mapper/mymp1p1
     vgcreate orvg /dev/mapper/mymp1p1
     ```

2. For both disk types, create the logical volume and file system by running the following commands:

   ```
   lvcreate --name orlv -l 100%FREE orvg
   mkfs.xfs  /dev/orvg/orlv
   ```

3. Mount the logical volume on `ORACLE_BASE=/u01` by running the following command:

   ```
   mount -t xfs /dev/orvg/orlv  /u01
   ```

4. For persistence after a system restart, insert the following **`mount`** statement into the `/etc/fstab` file:

   ```
   /dev/oravg/oralv                /u01                    xfs     defaults        1 1
   ```

5.2.12 Tiger VNC xterm window for interactive GUI installations (optional)

If you want to do GUI-based installations of Oracle software with either the runInstaller, asmca, dbca, or netca Oracle GUI installer programs, install or configure the VNC viewer or xterm emulator.

Silent installation options are available for these Oracle installers too, but the first time that you do an installation, the GUI wizards provide the system checks to verify your system configuration, which then can be scripted with a silent installation afterward.

For SUSE installs, the VNC viewer installer provides a good example of using the Remmina Desktop for Oracle installations.

5.2.13 Required Linux RPMs and pre-installation checking

This section describes other pre-installation checking that you must perform after ensuring that a GUI is established.

Operating system requirements for IBM LinuxONE

SUSE servers must be running SUSE Linux Enterprise Server 15 SP3 5.3.18-57-default (s390x) or later to Install Oracle RDBMS 19c on IBM LinuxONE.

SUSE Linux Enterprise Server 15: Required RPMs for an Oracle 19c installation

Here are the required Linux RPMs for a standard Oracle 19.12+ installation on a LinuxONE (s390x) system:

- 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

Note: Make sure that `compat-libpthread-nonshared-0-150300.3.6.1`, which is not part of the default installed RPMs, is installed before installing Oracle.

Oracle RPM Checker with SUSE Linux Enterprise Server on LinuxONE

In this section, we describe how to use the Oracle RPM Checker utility to verify that the required Linux packages are installed on the operating system before starting an Oracle Database or Oracle Grid Infrastructure installation.

Download the appropriate Oracle RPM Checker utility from the MOS websites based on the Oracle Database 19c Release and the SUSE Linux Enterprise Server release level (for example, SUSE Linux Enterprise Server 15 / 19.3). The MOS Note RPM Check when Installing Oracle Database 19c on Linux on System z (Doc ID 2553465.1) contains the latest RPM checker scripts.

After you download the RPM checker on LinuxONE, upload the RPM checker to your system and install it as root. To install any required RPMs and their dependencies, use the SUSE **zypper** command:

```
zypper install ora-val-rpm-S15-DB-19c-19.0.1-1.s390x.rpm
```

Example 5-11 shows the result of running the **zypper install** command to install the required Oracle RPM dependencies.

Example 5-11 The zypper install command for required SUSE Linux Enterprise Server 15 Linux RPMs for an Oracle 19c Installation

```
# zypper install ./ora-val-rpm-S15-DB-19c-19.0.1-1.s390x.rpm
Loading repository data...
Reading installed packages...
Resolving package dependencies...
The following NEW package is going to be installed:
  ora-val-rpm-S15-DB-19c

The following package has no support information from its vendor:
ora-val-rpm-S15-DB-19c

1 new package to install.

Overall download size: 3.2 KiB. Already cached: 0 B. No additional space will be
used or freed after the operation.
Continue? [y/n/v/...? shows all options] (y): y
Retrieving package ora-val-rpm-S15-DB-19c-19.0.1-1.s390x
(1/1),   3.2 KiB (    0   B unpacked)
ora-val-rpm-S15-DB-19c-19.0.1-1.s390x.rpm:
 Package is not signed!

ora-val-rpm-S15-DB-19c-19.0.1-1.s390x (Plain RPM files cache): Signature
verification failed [6-File is unsigned]

Abort, retry, ignore? [a/r/i] (a): i

Checking for file conflicts:
.....................................................................................
.....................................................................[done]
(1/1) Installing: ora-val-rpm-S15-DB-19c-19.0.1-1.s390x
.....................................................................................
.........................................[done]
Additional rpm output:
*************************************************************************
```

```
*Validation complete - Your sles15 OS has required rpms for Oracle 19c.
**********************************************************************
```

Running the command automatically imports the required RPMs if they are missing when your `zypper` repository is set up correctly to do so. If you do not have your RPM repository set up, you can install the RPM dependency checker by using the following command (use the command with th corresponding RPM based on your distribution level):

```
rpm -ivh ora-val-rpm-S15-DB-19c-19.0.1-1.s390x.rpm
```

Example 5-12 illustrates that the Oracle validation checker is installed correctly.

Example 5-12 SUSE RPM checker for an Oracle 19c SUSE Linux Enterprise Server 12 installation

```
# rpm -ivh ora-val-rpm-S12-DB-19c-19.0.1-1.s390x.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:ora-val-rpm-S12-DB-19c-19.0.1-1  ################################# [100%]
********************************************************************************
*  Validation complete - Your sles12 OS has required rpms for Oracle 19c.     *
* If you are using Grid Infrastructure, make sure you have gfortran compiler: *
*  at least v. 4.8.5-16                                                       *
********************************************************************************
```

5.3 Linux large pages and Oracle databases

For performance and availability reasons, implement Linux large pages for Oracle databases running on IBM LinuxONE systems. Linux large pages are beneficial for systems where the database's Oracle SGA is greater than 8 GB or if there are more than 50 database connections.

5.3.1 Linux large page setup

To implement large pages, complete the following steps:

1. If you plan to use Linux large pages for Oracle databases, you cannot use AMM by setting the **`MEMORY_MANAGEMENT`** parameter. Instead, use Automatic Shared Memory Management (ASMM) by setting the **`SGA_TARGET`** and **`PGA_AGGREGATE_TARGET`** Oracle parameters.
2. The Oracle **`use_large_pages`** parameter can be set to `true`, `false`, or `only`. If you have a LinuxONE system with one large database that requires large pages, and other smaller databases that do not require large pages, set the larger SGA database to "`only`" and the smaller databases to "`false`". The default setting for **`use_large_pages`** is "`true`".
3. At the Linux level, set `/etc/security/limits.conf` to `unlimited` to allow for changes to the Oracle SGA/Linux large page values dynamically, as shown in the following entries:
   - `soft memlock unlimited`
   - `hard memlock unlimited`
4. Set or update the following parameters in the `/etc/sysctl.conf` file:

   ```
   vm.nr_hugepages = ((sum of all large page SGAs)* 1024) + 16 = N
   vm.hugetlb_shm_group = <Linux group number of oinstall from /etc/group>
   ```

5. Restart your Linux Image and restart Oracle for the changes to take effect.
6. Review your Oracle Database alert log to verify that the database started with large pages enabled.

5.3.2 Disabling transparent HugePages

For performance and stability reasons, disable transparent HugePages. Transparent HugePages are different than Linux large pages.

You can check whether your system has transparent HugePages enabled by running the following command:

```
cat /sys/kernel/mm/transparent_hugepage/enabled
[always] madvise never
```

To disable transparent HugePages, complete the following steps:

1. In the `/etc/default/grub` file, add `transparent_hugepage=never` to the end of the parameters line:

   ```
   GRUB_CMDLINE_LINUX_DEFAULT="hvc_iucv=8 TERM=dumb mitigations=auto
   transparent_hugepage=never vmalloc=4096G"
   ```

2. To ensure that the Linux kernel parameter changes take effect, run the **`mkconfig`** command:

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

3. To activate the Linux kernel parameter for the next system restart, run **`zipl -VV`**:

   ```
   [root@zlnxl2 etc]# zipl -VV
   Using config file '/etc/zipl.conf'
   Target device information
     Device.......................: 5e:00
     Partition....................: 5e:01
     Device name..................: dasda
     Device driver name...........: dasd
   Preparing boot device: dasda (0200).
   Syncing disks...
   Done.
   ```

4. Restart your Linux guest for the parameter change to take effect by running the following command:

   ```
   shutdown -r now
   ```

5. Verify that transparent HugePages are disabled by running the following command:

   ```
   cat /sys/kernel/mm/transparent_hugepage/enabled
   always madvise [never]
   ```

5.4 Enabling Linux random number generation (entropy)

For Oracle Database 19c and later, it is important for systems to be configured to provide good random number generation to use the new security features of an Oracle Database installation.

Oracle 19c now uses `/dev/random` rather than `/dev/urandom`, which means without properly configured entropy/random numbers replenishment, you might encounter intermittent access issues, such as `ORA-01017: invalid username/password` on your system.

5.4.1 Entropy replenishment configuration

The general rule is that `/proc/sys/kernel/random/entropy_avail` always should be greater than 1000 available random numbers. To validate this number, run the following command:

```
cat /proc/sys/kernel/random/entropy_avail
348
```

If this value is less than 1000, then your Linux guest might require more setup steps so that random numbers are replenished properly, and to avoid any Oracle user logon-related errors with the database.

You can use the Pseudo Random Number Generator (PRNG) device to provide statistically significant random numbers for running Oracle databases. IBM LinuxONE servers use the CP Assist for Cryptographic Function (CPACF) on each processor chip to replenish random numbers.

5.4.2 Enabling the SUSE haveged service for random number generation

To install and enable the haveged service if the module is not enabled already on your system, complete the following steps:

1. Verify or install the haveged service if it is not installed already by running the following command:

   ```
   # rpm -qa | grep haveged
   haveged-1.9.2-6.1.s390x
   ```

2. If haveged is not installed, you can use **`zypper`** to install the RPM. Verify that random number replenishment is working properly with a value always over 1000 by running the following command:

   ```
   # cat /proc/sys/kernel/random/entropy_avail
   2851
   ```

5.5 Running the Oracle Database Setup Wizard to install Oracle Database

You should now be ready to follow the standard Oracle Installation procedure on your SUSE Linux Enterprise Server 15 SP3+ system. To do so, complete the following steps:

1. Log in as the Oracle installation owner user account (`oracle`) that you want to own the software binary files.
2. To install Oracle Database 19c on SUSE Linux Enterprise Server 15, download the Oracle Database 19.3 software binary files from OTN, as described in 5.1.2, “Obtaining the Oracle 19c Software for IBM LinuxONE” on page 60, and then apply the 19.12 or later RUs during the Oracle Database installation or upgrade process.
3. Create the Oracle home directory and then extract the database image files. For example:

   ```
   $ mkdir -p /u01/db
   $ chgrp oinstall /u01/db
   $ cd /u01/db
   $ unzip -q /tmp/LINUX.ZSERIES64_193000_db_home.zip
   ```

4. From the Oracle home directory, run the **`runInstaller`** command to start the Oracle Database Setup Wizard by using a vnc or similar xterm window.

   It is a best practice not to use any form of “su” to start the runInstaller to avoid potential display-related problems.

5. During the Oracle Database installation process, run the **`runInstaller`** command with the **`-applyRU`** option to install the 19.12 or later RU for SUSE Linux Enterprise Server 15:

   ```
   ./runInstaller -applyRU <DBRU patch unzip location> -applyOneOffs <oneoff patch unzip location>
   ```

   On our system, we ran from a vnc xterm window to apply the 19.15 RU to Oracle `HOME` as part of the installation:

   ```
   /u01/db/runInstaller -applyRU /u01/patches/33803476
   ```

After your database/grid software is installed, you can proceed with creating a database, as shown in Chapter 8, “Installing Oracle” on page 101.

6

Disk storage options for LinuxONE

This chapter describes the disk storage options that are available when configuring Oracle Database files on LinuxONE.

IBM recommends consulting the following Oracle documents when storage configurations are created:

- *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 on IBM: Linux on System z under Red Hat 6*, Doc ID 1377392.1
- *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 on IBM: Linux on System z under SUSE Linux Enterprise Server*, Doc ID 1350008.1, and the update note for SUSE Linux Enterprise Server 11 SP2+, Doc ID 1584934.1.

This chapter includes the following topics:

6.1 Overview

In this section, we discuss Open Storage FCP/SCSI LUNs and IBM Extended Count Key Data (ECKDTM) DASD with HyperPAV.

ECKD DASD uses less CPU, but does not get the same transactional throughput as FCP/SCSI implementations.

Your choice depends on your environment, including your High Availability and Disaster Recovery (HA/DR) requirements.

It is recommended when configuring disk storage for LinuxONE to assign the LUNs with multiple paths to ensure high availability and better performance by spreading the I/O workload across more channel paths/host bus adapters (HBAs).

It also is recommended to prepare the Oracle Database disk Layout, multiple paths, number of disks, disk size, type of FS, and so on, before installing the Linux Guest with Red Hat or SUSE. If done so, the disks set up, configuration, definition, and persistence is managed automatically through the Linux installation tools.

If it is not the case and you are installing an Oracle Database on a guest, you must configure the disk device persistence by using UDEV rules.

To ensures device persistence, configure a UDEV rule by using **`chzdev`** and **`lszdev`** commands, which configures the correct udev configuration files for the devices:

- FCP/SCSI: `/etc/udev/rules.d/12-dm-permissions.rules`
- ECKD/DASD: `/etc/udev/rules.d/99-udev-oracle.rules`

A restart is required for modifications of UDEV rules to take effect.
You can run the **`udevadm trigger`** command for any udev changes to take effect. This change can be done dynamically.

Also, check or set the udev file permissions for Oracle. Then, run the **`ls`** commands to confirm that the file permissions are set correctly (based on the devices that are configured in your UDEV rules), as shown in the following example:

```
   # ls -la /dev/dasde1
brw-rw---- 1 oracle oinstall 94, 5 Sep 25 08:59 /dev/dasde1
```

6.1.1 File systems and shared storage

For a stand-alone database on LinuxONE, database files and recovery files also can use a supported Linux file system for database files.

When a database is used with GRID or RAC, ASM provides shared storage and is certified on LinuxONE. A standard file system (for example, ext4 or XFS) cannot be used for shared database files that are used with Oracle RAC, RACONE, and ASM systems.

Spectrum Scale 5.0 (GPFS) can be used for storing database and non-database shared files, including Oracle RAC database data files.

As described in *Oracle Supported and Recommended File Systems on Linux, Doc ID 236826.1*, the recommended file systems xfs or ex14 when not using Oracle ASM or Spectrum scale (GPFS) for database files for SUSE 12 SP3+ systems and Red Hat 7.4+ systems.

6.2 Open Storage FCP/SCSI disk configuration

To configure disk storage for Linux on z System, LinuxONE, assign the LUNs (with multiple paths) to your Linux system guest.

First, verify that the LUNs are available to the Linux system. Run the **lsluns** or **lsscsi** commands to verify SAN connectivity.

Example 6-1 shows sample LUN listings.

Example 6-1 LUN listings

```
[root@orastgu1 ~]# lsluns
Scanning for LUNs on adapter 0.0.0010
   at port 0x500507680122a201:
      0x0000000000000000
      0x0002000000000000
   at port 0x500507680122aecd:
      0x0000000000000000
      0x0002000000000000
Scanning for LUNs on adapter 0.0.0020
   at port 0x500507680112a201:
      0x0000000000000000
      0x0002000000000000
   at port 0x500507680112aecd:
      0x0000000000000000
      0x0002000000000000
 [root@orastgu1 ~]# lsscsi
[0:0:0:0]    disk    IBM      2145             0000  /dev/sda
[0:0:0:2]    disk    IBM      2145             0000  /dev/sdc
[0:0:1:0]    disk    IBM      2145             0000  /dev/sde
[0:0:1:2]    disk    IBM      2145             0000  /dev/sdg
[0:0:2:1]    disk    IBM      2145             0000  /dev/sdi
[0:0:2:3]    disk    IBM      2145             0000  /dev/sdk
[0:0:3:1]    disk    IBM      2145             0000  /dev/sdm
[0:0:3:3]    disk    IBM      2145             0000  /dev/sdo
[1:0:0:0]    disk    IBM      2145             0000  /dev/sdb
[1:0:0:2]    disk    IBM      2145             0000  /dev/sdd
[1:0:1:0]    disk    IBM      2145             0000  /dev/sdf
[1:0:1:2]    disk    IBM      2145             0000  /dev/sdh
[1:0:2:1]    disk    IBM      2145             0000  /dev/sdj
[1:0:2:3]    disk    IBM      2145             0000  /dev/sdl
[1:0:3:1]    disk    IBM      2145             0000  /dev/sdn
[1:0:3:3]    disk    IBM      2145             0000  /dev/sdp
```

If your LUNs are not yet configured, you can use the YaST zfcp LUN configuration tool (using SUSE Linux Enterprise Server) or the `/etc/zfcp.conf` configuration (using Red Hat) to make the LUNs available to the Linux system (for more information, see Chapter 24, “Working with disks”, in *The Virtualization Cookbook for IBM Z Volume 1: IBM z/VM 7.2*, SG24-8147):

```
[root@orastgu1 ~]# cat /etc/zfcp.conf
   0.0.0010 0x500507680122a201 0x0000000000000000
   0.0.0010 0x500507680122a201 0x0002000000000000
   0.0.0010 0x500507680122aecd 0x0000000000000000
   0.0.0010 0x500507680122aecd 0x0002000000000000
   0.0.0020 0x500507680122a201 0x0000000000000000
   0.0.0020 0x500507680122a201 0x0002000000000000
   0.0.0020 0x500507680122aecd 0x0000000000000000
   0.0.0020 0x500507680122aecd 0x0002000000000000
```

In a two-node Oracle RAC configuration, you can require up to four separate paths to each LUN when you use FCP/SCSI and multipathing. Multiple separate LUNs can share the same path, but separate paths are required for each multipath path going to the same worldwide LUN ID.

Unlike ECKD/DASD configurations, a partition is not required for FCP/SCSI LUNs. As a preferred practice, configure the disk LUNs with multipathing to ensure high availability and better performance by spreading the I/O workload across more channel paths or host bus adapters (HBAs).

The first step in configuring is ensuring that the multipath distribution packages are installed. The exact version of multipath varies by the Linux distribution that you are running.

Example 6-2 shows the steps to verify that multipath RPMs are installed.

Example 6-2 Verifying multipath RPMs

```
#SUSE Linux Enterprise Server 12+:
orainst1:/etc # rpm -qa | grep multipath
multipath-tools-0.4.9-0.83.2
#Red Hat 7+:
[root@orastgu1 ~]# rpm -qa | grep multipath
device-mapper-multipath-0.4.9-119.el7.s390x
device-mapper-multipath-libs-0.4.9-119.el7.s390x
```

Next, check that the `multipathd` service is configured to start on system start by running the following command and reviewing its output:

- Red Hat 7

  ```
  [root@orastgu1 ~]# systemctl | grep multipathd
  multipathd.service         loaded active running   Device-Mapper Multipath
  Device Controller
  ```

- SUSE Linux Enterprise Server 12

  Using SUSE Linux Enterprise Server 12, ensure `boot.multipath` is started and configured to start on system restart by running the following commands and reviewing their output:

  ```
  # chkconfig multipathd on
  # chkconfig boot.multipath on
  # chkconfig --list multipathd
  boot.multipath 0:off 1:off 2:off 3:off 4:off 5:off 6:off B:on
  ```

You can start `boot.multipath` and `boot.multipathd` on SUSE Linux Enterprise Server 11 systems by running the following commands:

```
# /etc/init.d/boot.multipath start
# /etc/init.d/multipathd start
```

Now that multipath is configured, the next step is to gather the unique worldwide identifier (WWID) for each of the LUNs that you plan to use for your Oracle Database files.

Notice in Example 6-3 that the LUN `WWIDs /dev/sda` and `/dev/sdc` feature the same WWIDs. These LUNs are the same physical disk LUN, but are configured with an alternative path to provide higher availability.

Example 6-3 Gathering LUN WWIDs

```
# /lib/udev/scsi_id --whitelisted
36005076303ffcbbf000000000000ef00
# /lib/udev/scsi_id --whitelisted
36005076303ffcbbf000000000000ef01
# /lib/udev/scsi_id --whitelisted
36005076303ffcbbf000000000000ef00
# /lib/udev/scsi_id --whitelisted
36005076303ffcbbf000000000000ef01
--device=/dev/sda
--device=/dev/sdb
--device=/dev/sdc
--device=/dev/sdd
```

After the LUN WWIDs are determined, configure `/etc/multipath.conf` with aliases and define any storage array-specific settings to help improve disk I/O performance. Example 6-4 shows a sample `/etc/multipath.conf` setting.

Example 6-4 Sample multipath.conf file

```
# check with your storage vendor as well for any updates defaults {
dev_loss_tmo 90 #zSeries specific, no. of secs wait before marking path bad
failback "immediate"
# failback "manual" #use manual with RAC clusters to prevent ping-pong effect
fast_io_fail_tmo 5 #zSeries specific, length time to wait before failing I/O
getuid_callout "/lib/udev/scsi_id --whitelisted --device=/dev/%n"
#  getuid_callout "/sbin/scsi_id -g -u -s /block/%n" #use with Red Hat 5.x,SUSE
Linux Enterprise Server 10
#  uid_attribute "ID_SERIAL" #use uid_attribute instead of getuid for SUSE Linux
Enterprise Server 11SP3+ & RH 7
   max_fds "max" #Red Hat 6.3+, SUSE Linux Enterprise Server 11SP3+
#  max_fds 8192  #Use for Older SUSE Linux Enterprise Server and Red Hat distros
no_path_retry "queue" #XiV recommendation if no paths available, use fail with RAC
# no_path_retry "fail" # use fail if using Oracle RAC
path_selector "round-robin 0" #round-robin for SUSE Linux Enterprise Server 11 SP2
and Red Hat 6.x and older
# path_selector "service-time 0" #use service-time for SUSE Linux Enterprise
Server 11 SP3+ and Red Hat 7+
path_grouping_policy "multibus" # SUSE Linux Enterprise Server 11 SP1+ and Red Hat
6.x # path_grouping_policy "failover" # SUSE Linux Enterprise Server 10, Red Hat
5.x
}
```

```
path_checker "tur" # Determines state of path, XiV, polling_interval 30 # Time in
seconds to check paths prio "const"
queue_without_daemon "yes"
DS8000, verify with vendor
(Red Hat 5.x,SUSE Linux Enterprise Server 11sp1 & older)
Linux distros (SUSE Linux Enterprise Server 11sp2+ RH 6+)
#  rr_min_io 100
#  rr_min_io_rq 1
#  rr_min_io_rq 4
#  rr_min_io_rq 15  #IBM XiV
}
blacklist  {
   devnode "^(dasd)[0-9]*"
}
multipaths {
   multipath {
wwid alias
36005076303ffcbbf000000000000ef00
   ASMFCP1
```

For the changes for `multipathd` to take effect for SUSE Linux Enterprise Server 11, run the commands that are shown in Example 6-5.

Example 6-5 SUSE Linux Enterprise Server 11 - starting and stopping multipathd

```
# service multipathd stop
 Shutting down multipathd done
 #  /sbin/multipath -F
 #  /sbin/multipath -v2 -l
 #  service multipathd start
 Starting multipathd done
```

Note: It is not a preferred practice to use aliases for the system root or boot device.

A multipath YaST module can be installed from the SLE HA Extension that is included with a SUSE Linux Enterprise Server for System z subscription. The YaST multipath module makes configuring the `multipath.conf` file much easier.

For the changes for `multipathd` to take effect for Red Hat 7, run the commands that are shown in Example 6-6.

Example 6-6 Red Hat 7 - starting and stopping multipathd

```
[root@orastgu1 ~]# systemctl stop multipathd.service
     // NO Message, multipathd is stopped //
               #  multipath -F (-F     flush all unused multipath device maps)
               #  /sbin/multipath -v2
((-v  level. verbosity, print all paths and multipaths. : 2 +  print all info :
detected paths, coalesced paths (ie multipaths) and device maps))
               #  systemctl start multipathd
                    // NO Message, multipathd is started //
```

It is also a preferred practice to run **`zipl`** and **`mkinitrd`** after making changes to `/etc/multipath.conf`.

6.3 Oracle ASM

If you use ASM, you must set up the file permissions so that the Oracle user has read/write access to the ASM storage volumes.

The `/etc/udev/rules.d/12-dm-permissions.rules` file is used to map the multipath alias with the Linux user ID that is used to install the Oracle Grid Infrastructure code. Some sites use grid for this user ID, and other sites use the Linux Oracle ID for the Oracle Grid Infrastructure and the Oracle RDBMS binary files.

If you do not have a template copy of the `/etc/udev/rules.d/12-dm-permissions.rules` file, you can find a copy of this file in the following directories:

- Red Hat: `/usr/share/doc/device-mapper-1.02.xxx/12-dm-permissions.rules`
- SUSE Linux Enterprise Server systems: `/usr/share/doc/packages/device-mapper/12-dm-permissions.rules`

Configure the `/etc/udev/rules.d/12-dm-permissions.rules` file, as shown in Example 6-7, by assigning the multipath alias with the needed Oracle file permissions. It is a preferred practice to use the DM-NAME that is tied to the `multipath.conf` alias, but you can also use the `mpath-WWID` if you are not using the multipath aliases.

Example 6-7 Sample /etc/udev/rules.d/12-dm-permissions.rules file

```
# MULTIPATH DEVICES
#
# Set permissions for all multipath devices
#ENV{DM_UUID}=="mpath-36005076303ffcbbf000000000000ef01", OWNER:="oracle",
GROUP:="dba", MODE:="660"
ENV{DM_NAME}=="ASM*", OWNER:="oracle", GROUP:="dba", MODE:="660"
```

6.4 ECKD and DASD disk configuration

If you are configuring your disk storage for Oracle with ECKD/DASD, several enhancements are available in z/VM that allow for increased disk performance for Oracle databases, namely PAV and HyperPAV and Live Guest Relocation (LGR).

The scope of this section is configuring the DASD for Oracle Database to help use PAV/HyperPAV and LGR in your Oracle environments.

If you want to configure HyperPAV or PAV, the first step is to verify and configure that the Hardware Management Console (HMC) IOCP configuration for the DASD aliases is done.

For more information about configuring HyperPAV aliases in the IOCP, see *How to Improve Performance with PAV: Development stream (Kernel 2.6.35)*.

For more information about accomplishing this task, see IBM Documentation.

HyperPAV/PAV is supported by all in service Linux distributions.

At the z/VM layer, after the IOCP is defined, the next step is to attach the base DASD volumes and the associated aliases, and then, verify that z/VM can see the devices.
Complete the following steps to add the base and aliases devices:

1. Attach the base DASD device by running the following command:

   ```
   CP ATTACH 7408 *
   ```

2. Attach the alias by running the following command:

   ```
   CP ATTACH 74D0 *
   ```

3. Verify that HyperPAV aliases can be seen by z/VM:

   ```
   q pav all
   HYPERPAV BASE 7408 ON 7408 0X7408 ASSIGNED 7408 POOL 65 HYPERPAV BASE 7409 ON
   7409 0X7409 ASSIGNED 7409 POOL 65 HYPERPAV ALIAS 74D0 ASSIGNED 74D0 POOL 65
   HYPERPAV ALIAS 74D1 ASSIGNED 74D1 POOL 65
   ```

To be persistent across IPLs of z/VM, the base and aliases devices must be defined in the Linux guest's z/VM USER directory.

Aliases are not exclusive to the base device for which they are defined. An alias can be used for any base device in the same logical subsystem on the storage system. In our example, any base device that begins with `75XX` can use any of the `75XX` format aliases. HyperPAV alias definitions must be defined before any DEVICE definitions in the guest's USER directory.

Example 6-8 shows a partial user directory entry with HyperPAV alias definitions.

Example 6-8 Sample excerpt of USER DIRECT file showing HyperPAV aliases

```
USER ORARH1 1RV7QL00 64G 256G AG
   INCLUDE CMSUSER
   COMMAND DEFINE HYPERPAVALIAS 74D0 FOR BASE 7408
   COMMAND DEFINE HYPERPAVALIAS 74D1 FOR BASE 7408
```

6.4.1 Activating DASD and aliases with SUSE Linux Enterprise Server 12

The YaST DASD module (YaST | Hardware | DASD) or the **`dasd_configure`** command that is shown in Example 6-9 can be used to bring online a base device and the aliases. These commands are run to make the DASD and aliases persistent across restarts by creating a corresponding UDEV rule in the `/etc/udev/rules.d` directory.

Example 6-9 SUSE Linux Enterprise Server 12 dasd_configure example

```
dasd_configure 0.0.7408 1 0
dasd_configure 0.0.7409 1 0
dasd_configure 0.0.74d0 1 0
dasd_configure 0.0.74d1 1 0
```

You should then test and verify that HyperPAV is set up correctly from the Linux system by running **`lsdasd -u`**. If you do not see any aliases, HyperPAV is not configured properly.

6.4.2 Activating DASD and aliases with Red Hat Enterprise Linux 7

With Red Hat 7.x, you can use the `/etc/dasd.conf` file to automatically bring online a DASD base device. The corresponding aliases also can be enabled through this file. Each line pertains to the device (base or alias) to bring online (see Example 6-10).

Example 6-10 Red Hat 7 /etc/dasd.conf example

```
0.0.7408
0.0.7409
0.0.74d0
0.0.74d1
```

As a final test of whether HyperPAV is configured correctly for a Linux guest, as shown in Example 6-11, run the **`lsdasd -u` command** to confirm that your base devices include all of the aliases that you defined.

Example 6-11 lsdasd -u

```
Bus-ID Name UID ====================================================================
0.0.7408 dasdt IBM.750000000NA461.1554.08.00000000000027200000000000000000
0.0.7409 dasdce IBM.750000000NA461.1554.09.00000000000027200000000000000000
0.0.74d0 alias IBM.750000000NA461.1554.xx.00000000000000000000000000000000
0.0.74d1 alias IBM.750000000NA461.1554.xx.00000000000000000000000000000000
```

After the DASD volumes are presented to Linux, the next step is to format and partition the ECKD/DASD devices. All physical DASD disk volumes *must* be partitioned.

Example 6-12 shows the commands that you can use to format and partition the DASD base volumes.

Note: Formatting can result in loss of data. Make sure that you use the correct device names.

Example 6-12 Formatting a DASD volume

```
# /sbin/dasdfmt -f /dev/dasdt -b 4096 -p -y
# /sbin/dasdfmt -f /dev/dasdce -b 4096 -p -y
```

The following parameters are available:

- **`-f`** specifies the DASD volume to format
- **`-b`** flag specifies the block size (which should be 4096)
- **`-p`** flag shows a progress bar
- **`-y`** flag specifies start formatting without user confirmation

7

Configuring networks on LinuxONE for Oracle

This chapter discusses our experiences and best practices for configuring Oracle 19c database network configurations on IBM LinuxONE systems.

Many different Oracle, Linux, and virtualization technologies are available that can be used to configure Oracle 19c databases with a network for performance and availability.

This chapter is not meant to provide every possibility; rather, it provides some of the more common alternatives and our best practices in setting up these network alternatives for Oracle 19c databases.

This chapter includes the following topics:

7.1 Oracle high availability best practices

Oracle Maximum Availability Architecture (MAA) is a set of best practices that was developed by Oracle to help with a high availability (HA) architecture. It uses the Oracle high availability and disaster technologies.

MAA provides data protection and availability by minimizing or eliminating planned and unplanned downtime at the technology layers, including hardware and software components. It delivers protection and high availability for many types of failure, whether from hardware failures that cause data corruption, or anything ranging from human error or software malfunctions. It also includes HA features to help an installation when nature-related disasters affect a broad geographic area.

Oracle provides a comprehensive and integrated set of HA technologies that enable rapid recovery from failures and minimize planned downtimes. In this section, we focus on HA options that are related to the Oracle Database.

7.1.1 Oracle Real Application Clusters

Oracle Real Application Clusters (RAC) is a cluster database with a shared cache architecture that provides highly scalable and available database solutions for all business applications.

Introduced on Oracle Database Version 9 (9i), it is one of the key options for HA that installations can choose when running the Oracle Database. RAC is a chargeable licensable option of the Oracle Database Enterprise Edition.

Oracle RAC on Linux running on LinuxONE provides multiple Linux guest operating systems for a database application to help limit application downtime. Every quarter, Oracle provides patch set updates with important security and database fixes. With Oracle RAC/RAC One Node, application downtime can be minimized by patching one system at a time (rolling), while the other Linux guest systems continue to run applications.

7.1.2 Oracle Real Application Clusters One Node

Oracle RAC One Node is a single instance of an Oracle RAC-enabled database that is running on one node in a cluster. It is a lower cost, HA solution that runs a database on one node of a cluster at a lower price point than full Oracle RAC.

RAC One Node provides automatic failover capability to other nodes in the cluster. Typically, planned maintenance operations can be carried out with minimal disruption to application users by relocating the application Virtual IP addresses (VIPs) while the instances are active and running.

Technologies, such as transparent application failover (TAF), can be used to seamlessly move an active running SQL statement from one Oracle RAC ONE node to another without disruption to the user.

One of the benefits of Oracle RAC One Node is that applications that might not be “RAC friendly” because of data block contention, are not as affected as much as a full multi node RAC Cluster solution. For example, an insert intensive application that uses many non-cached sequences might perform better running on one node versus being distributed across multiple nodes in a RAC cluster.

Another benefit of Oracle RAC One Node database is that it can be converted from RAC One Node mode to full RAC dynamically, by using the **`srvctl convert`** command.

7.1.3 Oracle Data Guard/Active Data Guard

Data Guard is an Oracle Database offering that provides the management, monitoring, and automation software to create and maintain one or more synchronized copies of a production database. It provides HA for mission critical applications.

It is a HA/DR solution that provides fast automatic failover, in the case of database failures, node failures, corruption, and media failures. The standby databases can be used for read-only access, and reporting and testing and development purposes. Data Guard is included with Oracle Database Enterprise Edition.

Active Data Guard is an optional license component for Oracle Database Enterprise Edition. Active Data Guard adds advanced capabilities to extend basic Data Guard functions by allowing for databases to be opened at the DR site for read-only access use by applications.

Oracle Far Sync allows for zero data loss at any distance by first copying transaction logs initially to another DR site that is geographically close by using synchronous replication of the database to the nearby site. Then, this nearby site's data is replicated asynchronously to another standby DR site at a much great distance.

7.2 Networking

IBM LinuxONE features several options for designing the network between Logical Partitions (LPARs) and other servers on a network.

An Open System Adapter (OSA) is a physical network card in LinuxONE. An OSA can be dedicated to an Oracle Linux guest, shared among multiple Linux guests, or configured in a virtual switch (VSwitch) configuration. A VSwitch is beneficial for systems when many Linux guests sharing network infrastructure.

A *Hipersocket* is a high-speed, low-latency memory-to-memory network that traverses only internally within LinuxONE. Hipersockets are advantageous for workloads whereby a large amount of data must be transferred quickly between LPARs if enough CPU capacity is available to support the network transfers.

For example, an Oracle RAC system that uses a Hipersockets with CPU constraints might encounter packet loss, as CPU helps drive the fast memory to memory Hipersocket network. Whereas a data feed from a transactional system to a data warehouse using a Hipersocket might be fast, the use of the fast Hipersocket interface is not as important where consistent network latency exists.

Figure 7-1 shows some of these network concepts to consider in a network design for a development, test, and production environments.

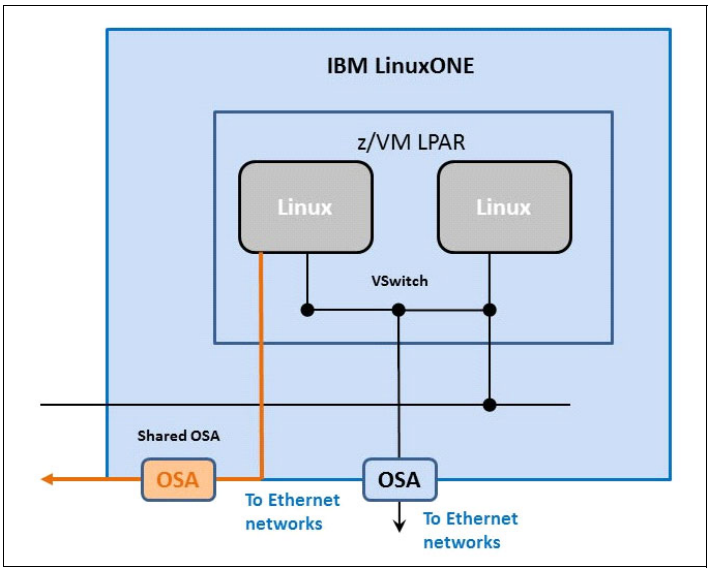

Figure 7-1 Swap VDisk configuration priorities

7.2.1 Oracle high availability networking options

IBM LinuxONE features the following options for supporting the Oracle Interconnect between Oracle nodes in an Oracle RAC or Oracle RAC One Node cluster configuration:

Link Aggregation (Active/Active) Allows up to eight OSA-Express adapters to be aggregated by using a virtual switch (VSwitch) type of configuration.

Linux Bonding Whereby Linux provides the high availability by using two or more Linux interfaces in an active/backup or active/active configuration.

Linux interfaces `eth1` `eth2` are configured to create a highly available bonded interface, `bond0`.

Oracle High Availability IP (HAIP) With Oracle 19c, you can have up to four Private interconnect interfaces, whereby Oracle's HAIP functions are used to balance network load across the Oracle RAC/RACONE interconnect network interfaces.

7.2.2 Oracle RAC interconnect z/VM Link Aggregation

Figure 7-2 shows a typical IBM LinuxONE topology, with multiple Oracle RAC clusters, that uses multiple z/VM VSwitch with Link Aggregation.

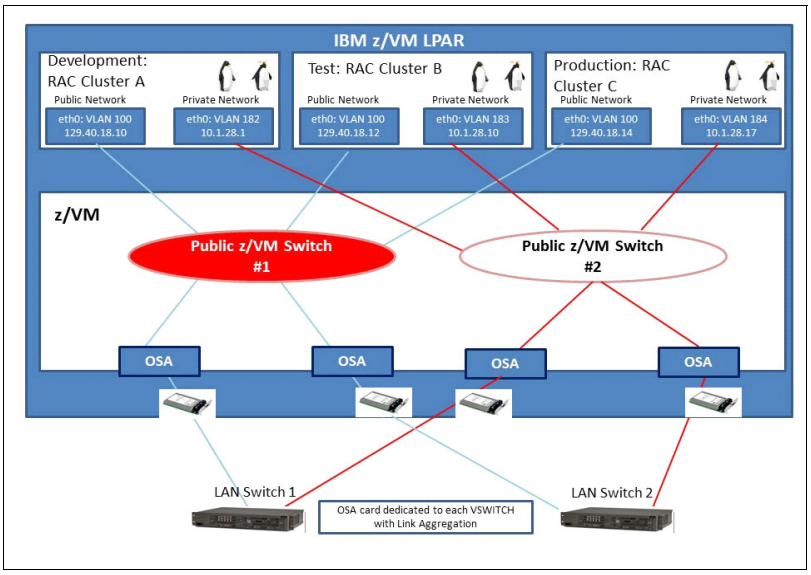

Figure 7-2 Oracle Interconnect with z/VM Link Aggregation

7.2.3 LinuxONE Network options for Oracle Databases

A LinuxONE System for Oracle can use the following hardware network options:

- OSA-Express, at 1, 10, or 25 Gbps network speeds
- Hipersocket memory-to-memory network (cannot span physical machines)
- RoCE Express (IP Mode Only), at 10 or 25 Gbps

A RHEL 7 or SUSE 12 Linux uses standard Linux `ifcg-<interface-name>.cfg` configuration files to define the attributes for the Linux system. You define the network configuration in the interface file with the associated sub-channel and ports for the network device to which the Linux guest is to be configured.

Figure 7-3 shows a typical Linux configuration file for an OSA-Express network port interface.

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth4
HWADDR=02:00:00:90:C8:E4
SUBCHANNELS=0.0.1040,0.0.1041,0.0.1042
NETTYPE=qeth
OPTIONS="portno=0 layer2=1"
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
IPADDR=10.25.60.82
PREFIX=24
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV4_DNS_PRIORITY=100
IPV6INIT=no
NAME=eth4
UUID=3c47b5b2-07cc-49d3-b32a-320e2dd2d8ac
DEVICE=eth4
ONBOOT=no
MTU=8992
```

Figure 7-3 Typical Linux interface configuration file

Figure 7-4 shows the network interface that is defined with a OSA Express 25Gbps card by using the Linux **qeth** command.

```
# lsqeth eth4
Device name                   : eth4
------------------------------------------------------------------------------
        card_type             : OSD_25GIG
        cdev0                 : 0.0.1040
        cdev1                 : 0.0.1041
        cdev2                 : 0.0.1042
        chpid                 : BE
        online                : 1
        portno                : 0
        state                 : UP (LAN ONLINE)
        priority_queueing     : always queue 0
        buffer_count          : 128
        layer2                : 1
        isolation             : none
        bridge_role           : none
        bridge_state          : inactive
        bridge_hostnotify     : 0
        bridge_reflect_promisc : none
        switch_attrs          : unknown
```

Figure 7-4 Linux Interface defined using OSA-Express 25Gb Card

Consider the following points regarding the use an OSA-Express network interface in an Oracle RAC configuration:

- Can use with a VSWITCH
- When CPU constrained, use dedicated OSA (non VSWITCH)
- Micro-code updates dynamic (most cases)
- Uses less CPU use than RoCE
- Can force network traffic to an external switch
- Many customers running Oracle RAC and supported
- Less bandwidth than RoCE (IP only)

Consider the following points regarding the use of a RoCE (IP only) network interface in an Oracle RAC configuration:

- Faster bandwidth and performance
- Cannot use with a VSwitch
- Microcode updates not concurrent (use Oracle HAIP or bonding for HA)
- Higher CPU usage than OSA
- Express
- Cannot force packets to go out to the network switch or firewall

7.2.4 Oracle RAC interconnect considerations

The design of the Oracle RAC/One Node interconnect is vital to the availability of an Oracle cluster. Oracle requires that the Cluster interconnect to be configured on a private, dedicated LAN or VLAN (tagged or untagged), which is non-routable and isolated from other non-interconnect traffic.

Note: The benefit of a VSwitch approach is that multiple RAC clusters can share the network infrastructure. It is strongly recommended to have a minimum of two VSwitch for an Oracle RAC configuration (public and private).

IBM LinuxONE environments, which run in two separate LPARs on the same LinuxONE machine, can use multiple Hipersocket interfaces that can be load balanced with Oracle's HAIP load balancing capability. Be careful with CPU capacity and sharing OSA Hipersocket interfaces, or packet loss can occur.

It is recommended to have the same Linux network interface name on all nodes of the Oracle RAC cluster. By default, some operating systems change the network interface names. Users must confirm that the interface name is consistent between nodes. For example, all nodes can be named `eth0` for public and `eth1` for RAC private.

For environments that are CPU bound, or cluster interconnect sensitive, the dedicated OSA cards approach (outside of a VSwitch) works well to ensure the cluster interconnect continues to operate efficiently during periods of high CPU load.

Another important consideration for the Oracle private interconnect is to use a large Maximum Transmission Unit (MTU) size of 8992 bytes. A typical Oracle data block is 8 K. By increasing the MTU size from 1500 standard to 8992, many network reassemblies can be avoided.

If your Oracle RAC cluster is across multiple LinuxONE machines, a network switch that support Jumbo packets is recommended.

Figure 7-5 shows a typical network interface that is configured with a large 8992 MTU size.

```
eth4: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 8992
        inet 10.25.60.82  netmask 255.255.255.0  broadcast 10.25.60.255
        inet6 fe80::ff:fe2c:3b7d  prefixlen 64  scopeid 0x20<link>
        ether 02:00:00:2c:3b:7d  txqueuelen 3000  (Ethernet)
        RX packets 409839  bytes 11475492 (10.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 7  bytes 586 (586.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Figure 7-5 Oracle Interconnect with large MTU size

7.2.5 LinuxONE network performance considerations

In this section, we describe some of the network tuning parameters that can be tuned depending on the workload.

The standard Oracle recommended parameters for Linux that are outlined in Chapter 2, “Oracle 19c database new features” on page 5 must be adhered to unless high network traffic is expected for an application.

If an application does encounter “`gc cr block lost`” in an Oracle diagnostic report, or lost packets from a Linux **`ifconfig`** or **`netstat -an`**, the network tuning tips that are presented next might help to resolve the issue or provide better network performance.

We used the suggested kernel parameters to help tune high network bandwidth workloads in the `/etc/sysctl.conf` parameters file.

To help prevent SYN packet loss, `net.ipv4.tcp_max_syn_backlog = 10000` can be increased. When changing **`tcp_max_syn_backlog`**, set `net.core.somaxconn` as well.

Another network tuning option is to increase the CPU input packet queue length from the default of 1000 (**`netdev_max_backlog`**).

Per *Troubleshooting gc block lost and Poor Network Performance (Doc ID 563566.1)*, increasing **`wmem`** parameters can help reduce Oracle data block loss and improve network performance in certain cases.

For Oracle RAC stability when defining multiple network interfaces for Oracle HAIP, Strict Reverse Path filtering must be disabled on the Oracle HAIP private interconnect interfaces. Strict mode is the default mode to prevent IP spoofing from Distributed Denial-of-service attacks. Having strict mode enabled on private interconnect of an Oracle RAC database cluster can disrupt interconnect communication.

Figure 7-6 show some of the `/etc/sysctl` Linux kernel Network Parameters for Oracle Workloads that can be used to improve network performance.

```
# Per IBM Network Performance Guide
net.ipv4.tcp_max_syn_backlog = 10000
net.core.somaxconn = 1024
net.core.netdev_max_backlog = 25000
# per best Practice Guide for multiple network Interconnects Running RAC
net.ipv4.conf.conf.enccw0/0/a800.rp_filter.rp_filter = 2
net.ipv4.conf.conf.enccw0/0/a900.rp_filter.rp_filter = 2
```

Figure 7-6 Sample Linux network performance and availability parameters

7.3 Client example: Oracle MAA RAC and Data Guard

Oracle MAA best practices also can be used to provide an Oracle centric HA and DR solution with IBM LinuxONE. In this scenario, the Oracle Database, RAC, and Data Guard are deployed on IBM LinuxONE (for the database portion of the deployment).

Oracle MAA provides a comprehensive architecture for reducing time that is for scheduled and unscheduled outages. An Oracle MAA solution consists of two identical sites: the primary site contains the RAC database, the secondary site contains Physical Standby database or both Physical and Logical one on RAC.

Active Data Guard is used for online reporting (extra licensed) at the DR site. Data Guard switchover and failover functions allow the roles to be traded between sites.

Figure 7-7 shows a typical customer configuration running on IBM LinuxONE that uses Oracle Data Guard to another DR site.

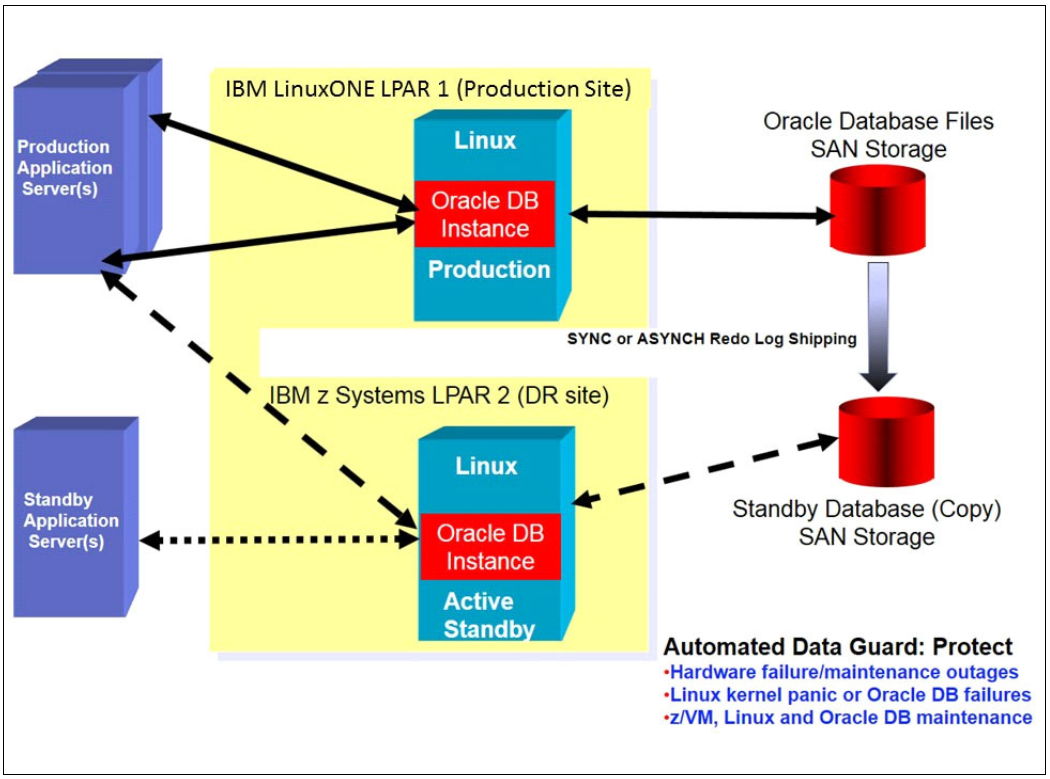

Figure 7-7 Oracle MAA example on IBM LinuxONE

7.4 Summary and recommendations

In this chapter, we described the main Oracle Database options for HA. We also described a few of the HA options for IBM LinuxONE.

IBM LinuxONE servers have the highest availability rating of any commercially available server that is available in the market today. IBM LinuxONE can perform dynamic reconfigurations to spare out another processor, and use redundant (RAIM) memory and another I/O or network path. With these key high availability features, IBM LinuxONE has the highest commercially availability rating of any server, according to ITIC's 2020 Global Server Hardware and Server OS Reliability survey.

IBM's LinuxONE virtualization technology has been in the marketplace for over 50 years, and deployed in Oracle development since the first Oracle 9i database was certified on the platform. Decades of experience in error correction and testing methods help ensure that not only is the hardware highly available, but the software is designed that way as well.

Combining the highest available server infrastructure of LinuxONE, with Oracle's maximum availability architecture creates a robust and powerful combination for running business critical Oracle workloads.

IBM LinuxONE is a good fit for Oracle High Availability because of the system's dynamic change capabilities. Adding memory, CPU, I/O or network resources if required, can be dynamically turned on and added to a system with microcode without the need to shut down an application or system.

8

Installing Oracle

In this chapter, we discuss installing Oracle.

This chapter includes the following topics:

8.1 Installing Oracle Grid Infrastructure for a stand-alone server

Oracle Grid Infrastructure for a stand-alone server is a version of Oracle Grid Infrastructure that supports single instance databases. Oracle Grid Infrastructure for a stand-alone server includes Oracle Restart and Oracle Automatic Storage Management (ASM).

Oracle Restart monitors and restarts Oracle Database instances, Oracle Net Listeners, and Oracle ASM instances.

Oracle ASM manages a small number of storage pools that are called *ASM Disk Groups*. Database-related files are assigned to ASM Disk Groups and ASM manages the layout and data organization.

Oracle combined the these two infrastructure products into a single set of binaries that is installed into an Oracle Restart home. To use Oracle ASM or Oracle Restart, we must install Oracle Grid Infrastructure for a stand-alone server before we install and create the database.

The following documents were referenced for installing Oracle Grid Infrastructure for a standalone server:

- *Oracle Database Installation Guide, 19c for Linux*, E96432-11
- *RPM Checker when Installing Oracle Database 19c on Linux on System z* (Doc ID 2553465.1)
- *Requirements for Installing Oracle 19c on SUSE Linux Enterprise Server 12 on IBM: Linux on System z* (Doc ID 2628889.1)
- *Requirements for Installing Oracle 19c RDBMS on RH7 on IBM: Linux on System z* (Doc ID 2628892.1)
- *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 and 12.1 on IBM: Linux on System z under Red Hat 6* (Doc ID 1377392.1)

8.1.1 Installation steps

The following steps were completed for Oracle Grid Infrastructure (19c) for a stand-alone server installation on a RHEL 7.7 Linux guest under z/VM 7 running on a LinuxONE server (for more information about this set up, see Chapter 3, “Setting up Linux to install Oracle Database” on page 11):

1. Validate Linux guest requirements.
2. Validate required Disk Space.
3. Validate required Linux RHEL 7.x RPMs for Oracle 19c.
4. Disable Linux Transparent Huge Pages.
5. Update Linux Kernel parameters.
6. Hugepage setup for Oracle Databases.
7. Create users or groups.
8. Create user directories for Oracle product installations.
9. udev set up for Storage for ASM.
10. Install Grid infrastructure for Standalone server binary.
11. Verify Grid infrastructure for Standalone server binary.

Table 8-1 lists the chapters and sections that provide more information about the installation steps.

Table 8-1 References for the installation steps

Steps	Reference	Comments
Validate Linux guest requirements	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.2, “General requirements and information for Oracle Database on LinuxONE” on page 15.
Validate required Disk Space	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.3, “Server minimum requirements” on page 15.
Validate required Linux RHEL 7.x RPMs for Oracle 19c	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.3.1, “Preinstallation checking” on page 26.
Disable Linux Transparent HugePages	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.5.2, “Disable Transparent HugePages” on page 32.
Update Linux Kernel parameters	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.3, “Server minimum requirements” on page 15.
Linux large page Setup for Oracle Databases	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.5.1, “Linux large page setup” on page 32.
Create users / groups	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.4, “Creating the database installation owner user” on page 19.
Create user directories for Oracle product installations	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.4, “Creating the database installation owner user” on page 19.
udev set up for Storage for ASM	Chapter 3, “Setting up Linux to install Oracle Database” on page 11	See 3.2.8, “Storage options for Oracle Database” on page 22.

8.1.2 Validate Linux guest requirements

Complete the following steps to validate the Linux guest requirements to install Oracle 19c Grid Infrastructure.

1. Run the following command to acquire the basic information about the Linux release:

   ```
   # cat        /etc/redhat-release
   ```

   Figure 8-1 shows the output of the command.

```
[root@rhel770 etc]# cat redhat-release
Red Hat Enterprise Linux Server release 7.7 (Maipo)
[root@rhel770 etc]#
```

Figure 8-1 Output of the cat redhat-release command

In this example, Linux Release 7.7 for 19c is certified by Oracle.

2. Run the following command to verify your Linux kernel version:

```
# cat        /proc/version
```

Figure 8-2 shows the output of the command.

```
[root@rhel770 etc]# cat /proc/version
Linux version 3.10.0-1062.el7.s390x (mockbuild@s390-037.build.eng.bos.redhat.com) (gcc version 4.8.5
 20150623 (Red Hat 4.8.5-39) (GCC) ) #1 SMP Thu Jul 18 16:27:07 EDT 2019
[root@rhel770 etc]#
```

Figure 8-2 Output of the cat /proc/version command

3. Run the following command to verify your Linux guest virtual RAM size:

```
# cat        /proc/meminfo | grep MemTotal
```

Figure 8-3 shows the output of the command.

```
[root@rhel770 /]# cat        /proc/meminfo | grep MemTotal
MemTotal:        3072820 kB
[root@rhel770 /]#
```

Figure 8-3 Output of the cat /proc/meminfo | grep MemTotal command

In this example, 3 GB of virtual memory is fine.

4. Check your swap space by running the following command:

```
# swapon -s
```

Figure 8-4 shows the output of the command.

```
[root@rhel770 /]# swapon -s
Filename                                Type            Size    Used    Priority
/dev/dm-1                               partition       1441788 0       -2
[root@rhel770 /]#
```

Figure 8-4 Output of the # swapon -s command

If you start seeing excessive swap used for the physical disk device, investigate the workload, Oracle memory settings, or Linux Virtual memory size.

8.1.3 Validating required disk space for software binary files

Consider the following points:

- Approximately 20 GB of disk space is required for Oracle Grid Infrastructure for a stand-alone server.
- Approximately 10 GB of disk space is required for the Oracle Database binaries.
- For the `/tmp` directory, 1 GB of space is recommended.
- In addition, at least 8 GB is needed to download the Oracle Software files.

8.1.4 Validating the required Linux RHEL 7.x RPMs for Oracle 19c

The Oracle 19c RPM checker checks that the required RPMs for Oracle Grid and Database installations are available in the Linux guest. Any missing RPMs are automatically installed.

You can download the suitable RPM checker as described in *RPM Checker when Installing Oracle Database 19c on Linux on System z* (Doc ID 2553465.1).

In our example, we downloaded the following RPM checker for RHEL 7.7 from MOS, which can be validated by running the **ls** command that is shown in Figure 8-5:

`ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm`

```
[root@rhel770 19c]# ls -alF ora*
-rw-r--r--. 1 root root 4784 Mar 26 12:31 ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
[root@rhel770 19c]#
```

Figure 8-5 Command to check the download RPM checker

Then, we installed the extracted RPM as root user to verify the Linux RPM requirements.

If any of the required RPMs are missing, error messages are shown during the installation of the RPM checker, as shown in Figure 8-6.

```
[root@rhel770 19c]# yum install ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to regist
er.
Examining ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm: ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
Marking ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm to be installed
Resolving Dependencies
--> Running transaction check
---> Package ora-val-rpm-RH7-DB-19c.s390x 0:19.0.1-1 will be installed
--> Processing Dependency: compat-libcap1 >= 1.10-7 for package: ora-val-rpm-RH7-DB-19c-19.0.1-1.s39
0x
rhel-WSC                                                                  | 2.8 kB  00:00:00
--> Processing Dependency: gcc >= 4.8.5-16 for package: ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
--> Processing Dependency: gcc-c++ >= 4.8.5-16 for package: ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
```

Figure 8-6 Missing RPMs error messages

If `yum` is configured, you can install the RPMs automatically, as shown in Figure 8-7.

```
 libpng                  s390    2.1.5.15-7.el7_2    rhel-WSC                  209 k
 libstdc++               s390    4.8.5-39.el7        rhel-WSC                  326 k
 libstdc++-devel         s390    4.8.5-39.el7        rhel-WSC                  1.5 M
 libstdc++-devel         s390x   4.8.5-39.el7        rhel-WSC                  1.5 M
 libuuid                 s390    2.23.2-61.el7       rhel-WSC                   84 k
 libxcb                  s390    1.13-1.el7          rhel-WSC                  215 k
 pam-devel               s390x   1.1.8-22.el7        rhel-WSC                  184 k
 zlib                    s390    1.2.7-18.el7        rhel-WSC                   92 k

Transaction Summary
================================================================================
Install  1 Package (+41 Dependent packages)

Total download size: 45 M
Installed size: 99 M
Is this ok [y/d/N]: y
```

Figure 8-7 YUM automatic installation

If all the required RPMs are installed, the panel that is shown in Figure 8-8 appears.

```
[root@rhel770 19c]# yum install ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm
Loaded plugins: langpacks, product-id, search-disabled-repos, subscription-manager
This system is not registered with an entitlement server. You can use subscription-manager to regist
er.
Examining ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm: ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x
ora-val-rpm-RH7-DB-19c-19.0.1-1.s390x.rpm: does not update installed package.
Error: Nothing to do
[root@rhel770 19c]#
```

Figure 8-8 The required RPMs are installed

8.1.5 Linux Transparent HugePages

Oracle recommends disabling Transparent HugePages because they can create performance issues for Oracle Database single instances.

Oracle continues to recommend the use of standard HugePages for Linux.

Complete the following steps to disable Transparent HugePages:

1. Back up the `/etc/zipl.conf` file.
2. Set the Linux parameter `transparent_hugepage=never` to the `[Linux]` boot parameter.
3. Run the **`zipl -VV`** command to generate a boot image.
4. Restart your Linux guest with `reboot`:

```
# cd /etc/
# cp zipl.conf     zipl.backup
# vi zipl.conf        (add linux parm transparent_hugepage=never  to [Linux]
boot option)
# zipl -VV           (make sure no errors)
# reboot               (for changes to take effect)
```

8.1.6 Updating Kernel parameters

Some default kernel parameters values must be updated for Oracle processes to run correctly. For more information about updating and setting up the kernel parameters value in `etc/sysctl.conf`, see Table 3-2 on page 17.

8.1.7 HugePage setup

Linux large pages are particularly beneficial for systems where the database's Oracle SGA is greater than 8 GB or if many database connections exist that greater than 50 GB.

For more information, see 3.5.1, "Linux large page setup" on page 32.

8.1.8 Configuring Host name

Ensure that the host name is correctly configured and an entry exists in the `/etc/hosts` file for the host. You also must ping by using your host name. For more information, see 3.2.7, "Host name" on page 21.

8.1.9 Creating users and groups

Create the needed groups and users. For the stand-alone grid infrastructure, we created one user (`oracle`). If your environment requires separate duties, create the `grid` user as well.

The following commands can be used to create the required groups and users. Group numbers can be set to meet local requirements:

```
#/usr/sbin/groupadd   -g 54321 oinstall
#/usr/sbin/groupadd   -g 54322 dba
#/usr/sbin/groupadd   -g 54323 oper
  #/usr/sbin/groupadd   -g 54327 asmdba
  #/usr/sbin/groupadd   -g 54328 asmoper
  #useradd --uid 54321 --gid oinstall --groups dba,oper,asmdba,asmoper     oracle
```

Figure 8-9 shows an example of running these commands.

```
[root@rhel770 scripts]# /usr/sbin/groupadd  -g 54323 oper
[root@rhel770 scripts]# useradd --uid 54321 --gid oinstall --groups dba,oper,asmdba,asmoper  oracle
[root@rhel770 scripts]# id oracle
uid=54321(oracle) gid=54321(oinstall) groups=54321(oinstall),54322(dba),54327(asmdba),54328(asmoper)
,54323(oper)
[root@rhel770 scripts]#
```

Figure 8-9 Creating groups and users commands

Use the following command to set the passwords for the `oracle` user:

```
# passwd oracle
```

8.1.10 Setting shell limits for the Oracle user

For more information about setting up the security limits for the `oracle` user, see 3.2.5, “User login security and limits configuration” on page 19 and use the following commands:

```
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
oracle soft stack 10240
oracle hard stack 32768
```

8.1.11 Creating user directories for product installations

Create the Oracle base and the Oracle inventory directories per the Oracle Optimal Flexible Architecture (OFA) recommendations. Specify the correct owner, group, and permissions for these directories by running the following commands:

```
# mkdir -p /u01/app/oracle
# mkdir -p /u01/app/oraInventory
# chown -R oracle:oinstall /u01/app/oracle
# chown -R oracle:oinstall /u01/app/oraInventory
# chmod -R 775 /u01/app
```

If you are planning to use VNC for GUI display, also install the RPMs `tigervnc*`, `openmotif`, `xterm`, `xsetroot`, and `xorg-x11-xauth` by running the following commands:

```
# yum -y install tigervnc* openmotif xterm xsetroot xorg-x11-xauth
# service firewalld stop
```

```
# systemctl disable firewalld
```

As the `oracle` user, add the following value in the `/home/oracle/.bash_profile`:

```
umask 022
```

8.1.12 Setting up udev for storage for ASM

We use the udev rules to specify the available storage devices for the Grid infrastructure. For more information about how to set up udev rules to use with ASM, see 3.2.8, “Storage options for Oracle Database” on page 22.

In our example, we use multi-pathed LUNs that are aliased as `orad1` and `orad2`. Figure 8-10, Figure 8-11, and Figure 8-12 on page 109 show the setup with permissions.

```
root@rhel770:/etc/udev/rules.d
orad2 (36005076309ffc6bb0000000000002011) dm-4 IBM     ,2107900
size=20G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='service-time 0' prio=0 status=active
  |- 0:0:0:1074872352 sdc 8:32   active undef running
  |- 1:0:0:1074872352 sdd 8:48   active undef running
  |- 0:0:1:1074872352 sdg 8:96   active undef running
  |- 1:0:1:1074872352 sdh 8:112  active undef running
  |- 0:0:2:1074872352 sdk 8:160  active undef running
  |- 1:0:2:1074872352 sdl 8:176  active undef running
  |- 0:0:3:1074872352 sdo 8:224  active undef running
  |- 1:0:3:1074872352 sdp 8:240  active undef running
  |- 0:0:4:1074872352 sds 65:32  active undef running
  |- 1:0:4:1074872352 sdt 65:48  active undef running
  |- 0:0:5:1074872352 sdw 65:96  active undef running
  `- 1:0:5:1074872352 sdx 65:112 active undef running
orad1 (36005076309ffc6bb0000000000002010) dm-3 IBM     ,2107900
size=20G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='service-time 0' prio=0 status=active
  |- 0:0:0:1074806816 sda 8:0    active undef running
  |- 1:0:0:1074806816 sdb 8:16   active undef running
  |- 0:0:1:1074806816 sde 8:64   active undef running
  |- 1:0:1:1074806816 sdf 8:80   active undef running
  |- 0:0:2:1074806816 sdi 8:128  active undef running
  |- 1:0:2:1074806816 sdj 8:144  active undef running
  |- 0:0:3:1074806816 sdm 8:192  active undef running
  |- 1:0:3:1074806816 sdn 8:208  active undef running
  |- 0:0:4:1074806816 sdq 65:0   active undef running
  |- 1:0:4:1074806816 sdr 65:16  active undef running
  |- 0:0:5:1074806816 sdu 65:64  active undef running
  `- 1:0:5:1074806816 sdv 65:80  active undef running
[root@rhel770 rules.d]#
```

Figure 8-10 Setting up udev for storage for ASM (1 of 3)

```
[root@rhel770 rules.d]# more 12-dm-permissions.rules
ENV{DM_NAME}=="orad*", OWNER:="oracle", GROUP:="oinstall", MODE:="660"
[root@rhel770 rules.d]#
```

Figure 8-11 Setting up udev for storage for ASM (2 of 3)

```
[root@rhel770 rules.d]# ls -lh /dev/dm-3
brw-rw----. 1 oracle oinstall 253, 3 Mar 27 16:01 /dev/dm-3
[root@rhel770 rules.d]# ls -lh /dev/dm-4
brw-rw----. 1 oracle oinstall 253, 4 Mar 27 16:01 /dev/dm-4
[root@rhel770 rules.d]#
```

Figure 8-12 Setting up udev for storage for ASM (3 of 3)

8.1.13 Oracle Grid Infrastructure for a standalone server installation

Starting with Oracle Grid Infrastructure 12c Release 2 (12.2), the installation and configuration of Oracle Grid Infrastructure software is simplified with image-based installation.

Use the previously downloaded Oracle Grid Infrastructure image file for a stand-alone server installation.

Figure 8-13 shows the image file that is downloaded and available.

```
[oracle@rhel770 ~]$ ls -alF /u01/code/19c/*Gri*
-rw-r--r--. 1 oracle oinstall 2519769777 Jun  6  2019 /u01/code/19c/V982652-01_19cGrid.zip
[oracle@rhel770 ~]$
```

Figure 8-13 Oracle Grid Infrastructure image file

Create the `grid home` directory, and extract the image files in this directory by running the following commands:

```
# mkdir -p /u01/app/oracle/product/19.0.0/grid
# cd  /u01/app/oracle/product/19.0.0/grid
# unzip -q /u01/code/19c/V982652-01_19cGrid.zip
```

Figure 8-14 shows the output of these commands.

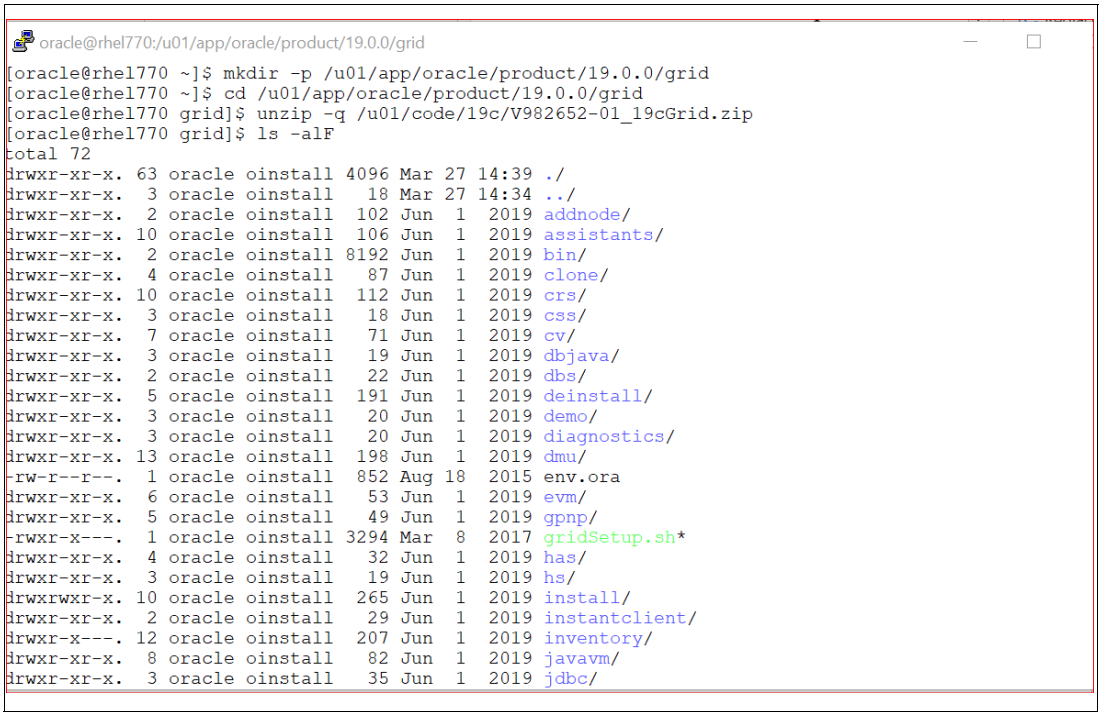

Figure 8-14 Extracting the image files into the grid home directory

We can use the **runcluvfy** utility to check that the server meets the installation requirements. Use the following command:

```
# ./runcluvfy.sh stage -pre hacfg
```

Figure 8-15 shows the output of the command.

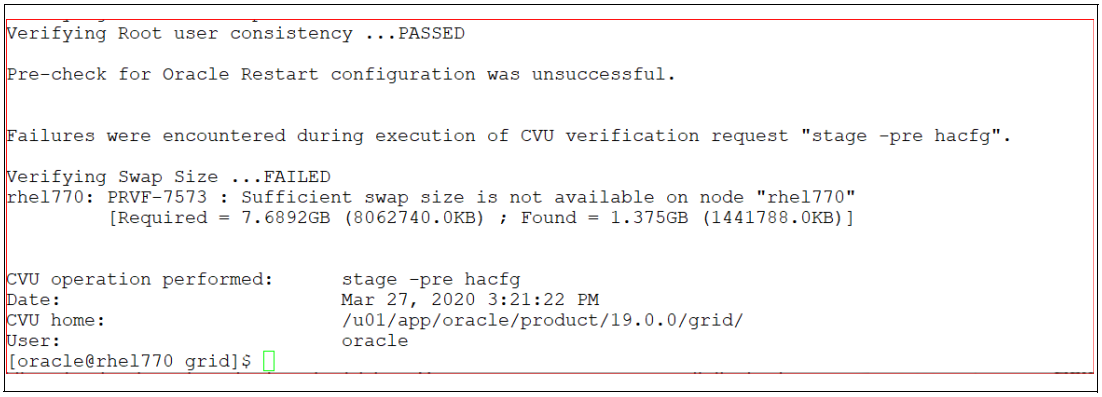

Figure 8-15 Checking whether the server meets the installation requirements.

In our example, the verification failed with swap size error, but this error can be safely ignored.

Now, we are ready to start the installation. We can use the Graphical interactive method or, if we do not have `x windows` facility in the environment, we can use the response file and perform a silent installation.

8.1.14 GUI installation

As a Grid installation user (in our example, as `oracle` in a x window, such as a VNC terminal to display [GUI]), run **`gridSetup.sh`**, as shown in Figure 8-16.

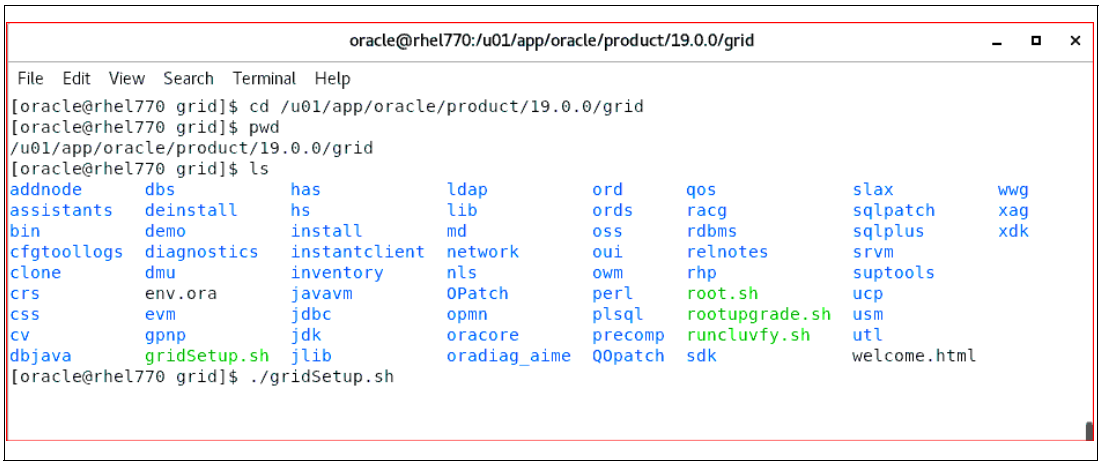

Figure 8-16 Running gridSetup.sh

Complete the following steps:

1. In the Select Configuration Option window, choose the **Configure Oracle Grid Infrastructure for a Standalone Server (Oracle Restart)** option (see Figure 8-17).

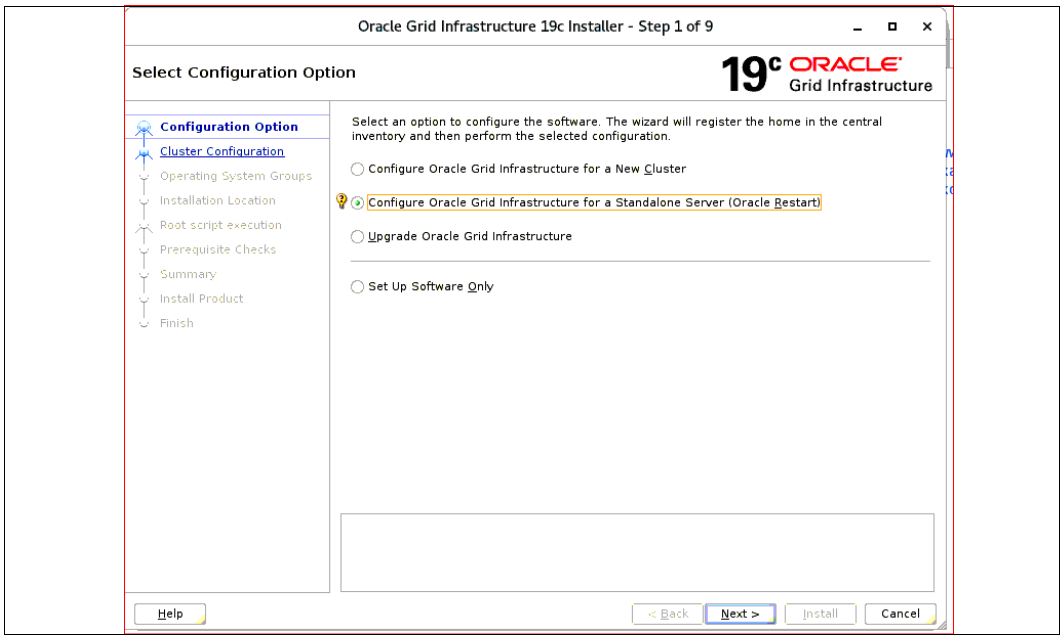

Figure 8-17 Select Configuration Option window

2. In the Create ASM Disk Group window, specify the settings that are listed in Table 8-2.

Table 8-2 Create ASM Disk Group window settings

Field name	Value	Comments
Disk group name	DATA	
Redundancy	External	
Allocation Unit Size	4 MB	
Select Disks	`/dev/mapper/orad1`	
Disk Discovery Path	`/dev/mapper/ord*`	

3. In the Specify ASM Password window, choose the **Use same passwords for these accounts** option and specify the password that conformed to Oracle specifications.
4. In the Operating System Groups window, specify the settings that are listed in Table 8-3.

Table 8-3 Operating System Groups window settings

Field name	Value
Oracle ASM Administrator (OSASM) Group	`dba`
Oracle ASM DBA (OSDBA) Group	`asmdba`
Oracle ASM Operator (OSOPER for ASM) Group optional	`oinstall`

Ignore invalid group chosen messages.

5. In the next window, specify the value for the Oracle base Directory. This value is the location where all the diag information for all the Oracle products are stored. In our example, we specified `/u01/app/oracle_base`.
6. In the Create Inventory window, specify the value for the Oracle Inventory Directory. This value is the location where Oracle stores the inventory of all the products that are installed. In our example, we specified `/u01/app/oraInventory`.
7. Choose to automatically run the configuration scripts as `root` user and specify the password for `root` user in the Root Script execution configuration window.

   The installation process performed prerequisite checks and reported the following errors:

   – Swap size error
   – Package cvuqdisk-1.0.10-1 missing error.

   The swap size error can be ignored. For the Package missing error, choose the **Fix and Check again** option. This choice created the fix and provided the information about where the fix available and how to run it, as shown in Figure 8-18 on page 113.

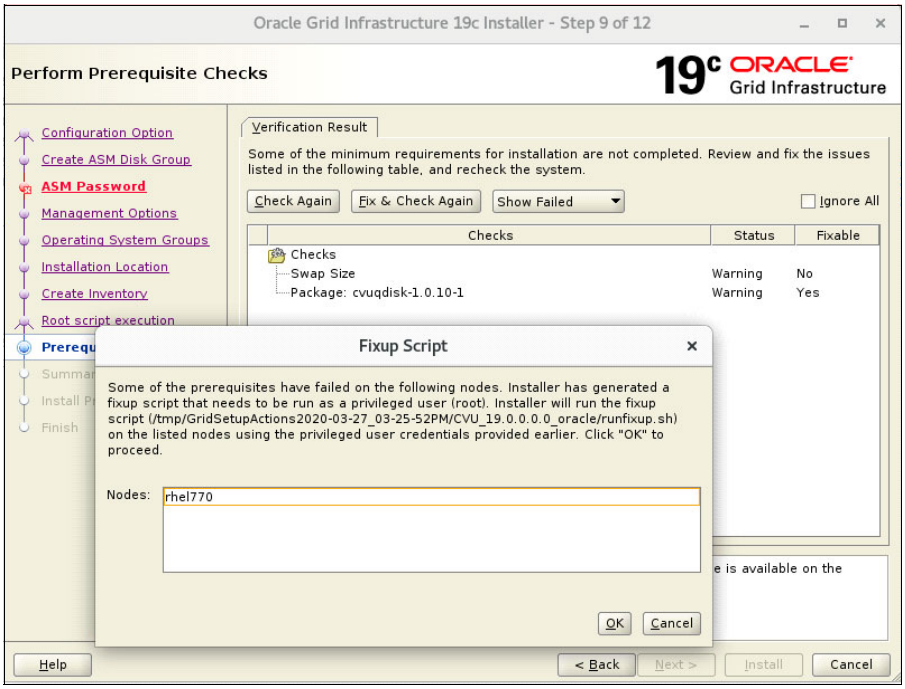

Figure 8-18 Fix and Check Again option

8. Run as `root` the generated **`runfixup.sh`** script.
9. The Summary window is displayed. Select **Install**.
10. When the installer prompt for running the scripts as `root` appears, allow it to run.
11. After the installer is complete and the Grid Infrastructure is configured, the successful completion window is displayed, as shown in Figure 8-19.

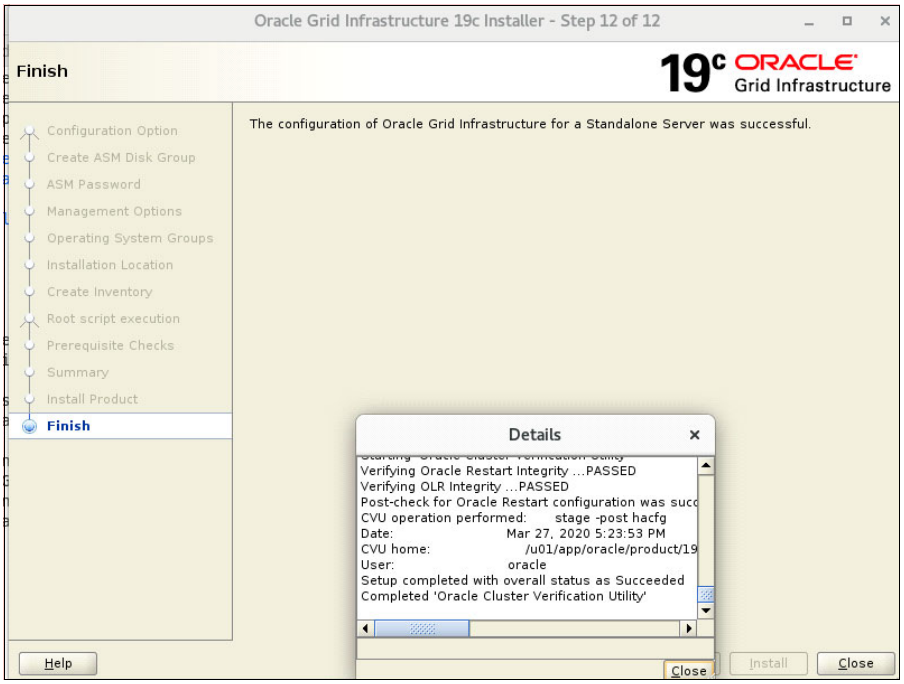

Figure 8-19 Installation complete

8.1.15 Silent installation

Typically, the installer runs in interactive mode, by prompting to provide information in graphical user interface (GUI) windows. We can also run the installer from a command prompt by using “Silent mode” and providing the values in a response file.

We can include all of the responses 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 windows. Instead, it displays progress information in the terminal that you used to start it.

The response file `gridsetup.rsp` is available at the grid installation or response folder. In our example, at `/u01/app/oracle/product/19.0.0/grid/install/response`.

Complete the following steps:

1. Make a backup copy of the response file (in our example, as `gridsetup.rsp.orig`), as shown in Figure 8-20 and Figure 8-21.

```
[oracle@rhel770 response]$ pwd
/u01/app/oracle/product/19.0.0/grid/install/response
[oracle@rhel770 response]$ ll
total 40
-rw-r-----. 1 oracle oinstall 34367 Jan 19  2019 gridsetup.rsp
-rw-r-----. 1 oracle oinstall  1541 May 20  2016 sample.ccf
[oracle@rhel770 response]$ cp gridsetup.rsp gridsetup.rsp.orig
[oracle@rhel770 response]$
```

Figure 8-20 Backing up the response file

```
##                                                                              ##
## To register and configure 'Grid Infrastructure for Standalone server'        ##
##  - Fill out sections A,B and G                                               ##
##                                                                              ##
```

Figure 8-21 Sections to be completed

2. In the `gridsetup.rsp` file, specify the values as shown in Figure 8-22 - Figure 8-31 on page 117.

```
################################################################################
#                                                                              #
#                           SECTION A - BASIC                                  #
#                                                                              #
################################################################################
```

Figure 8-22 Section A - Basic

```
#-------------------------------------------------------------------------------
# Specify the location which holds the inventory files.
# This is an optional parameter if installing on
# Windows based Operating System.
#-------------------------------------------------------------------------------
INVENTORY_LOCATION=
```

Figure 8-23 Specifying the Inventory location

– INVENTORY_LOCATION=/u01/app/oraInventory

```
#------------------------------------------------------------------------------
# Specify the installation option.
# Allowed values: CRS_CONFIG or HA_CONFIG or UPGRADE or CRS_SWONLY or HA_SWONLY
#   - CRS_CONFIG  : To register home and configure Grid Infrastructure for cluster
#   - HA_CONFIG   : To register home and configure Grid Infrastructure for stand alone server
#   - UPGRADE     : To register home and upgrade clusterware software of earlier release
#   - CRS_SWONLY  : To register Grid Infrastructure Software home (can be configured for cluster
#                   or stand alone server later)
#   - HA_SWONLY   : To register Grid Infrastructure Software home (can be configured for stand
#                   alone server later. This is only supported on Windows.)
#   - CRS_ADDNODE : To add more nodes to the cluster
#   - CRS_DELETE_NODE : To delete nodes to the cluster
#------------------------------------------------------------------------------
oracle.install.option=
```

Figure 8-24 Installation.option

– oracle.install.option=HA_CONFIG

```
#-------------------------------------------------------------------------------
# Specify the complete path of the Oracle Base.
#-------------------------------------------------------------------------------
ORACLE_BASE=
```

Figure 8-25 Oracle Base location

– ORACLE_BASE=/u01/app/oracle_base

```
###############################################################################
#                                                                             #
#                            SECTION B - GROUPS                               #
#                                                                             #
#   The following three groups need to be assigned for all GI installations.  #
#   OSDBA and OSOPER can be the same or different.  OSASM must be different   #
#   than the other two.                                                       #
#   The value to be specified for OSDBA, OSOPER and OSASM group is only for   #
#   Unix based Operating System.                                              #
#   These groups are not required for upgrades, as they will be determined    #
#   from the Oracle home to upgrade.                                          #
#                                                                             #
###############################################################################

#-------------------------------------------------------------------------------
# The OSDBA_GROUP is the OS group which is to be granted SYSDBA privileges.
#-------------------------------------------------------------------------------
oracle.install.asm.OSDBA=

#------------------------------------------------------------------------------
# The OSOPER_GROUP is the OS group which is to be granted SYSOPER privileges.
# The value to be specified for OSOPER group is optional.
# Value should not be provided if configuring Client Cluster - i.e. storageOption=CLIENT_ASM_STORAGE.
#------------------------------------------------------------------------------
oracle.install.asm.OSOPER=

#------------------------------------------------------------------------------
# The OSASM_GROUP is the OS group which is to be granted SYSASM privileges. This
# must be different than the previous two.
#------------------------------------------------------------------------------
oracle.install.asm.OSASM=
```

Figure 8-26 Section B to specify ASM groups

– oracle.install.asm.OSDBA=asmdba, oracle.install.asm.OSOPER=oinstall, and oracle.install.asm.OSASM=dba

```
#                                SECTION G - ASM                                 #
#                                                                                #
##################################################################################


#-------------------------------------------------------------------------------
# Password for SYS user of Oracle ASM
#-------------------------------------------------------------------------------
oracle.install.asm.SYSASMPassword=oracle

#-------------------------------------------------------------------------------
# The ASM DiskGroup
#
# Example: oracle.install.asm.diskGroup.name=data
#
#-------------------------------------------------------------------------------
oracle.install.asm.diskGroup.name=DATA

#-------------------------------------------------------------------------------
# Redundancy level to be used by ASM.
# It can be one of the following
#   - NORMAL
#   - HIGH
#   - EXTERNAL
#   - FLEX#   - EXTENDED (required if oracle.install.crs.config.ClusterConfiguration=EX
# Example: oracle.install.asm.diskGroup.redundancy=NORMAL
#
#-------------------------------------------------------------------------------
oracle.install.asm.diskGroup.redundancy=EXTERNAL

#-------------------------------------------------------------------------------
# Allocation unit size to be used by ASM.
# It can be one of the following values
#   - 1
#   - 2
#   - 4
#   - 8
#   - 16
# Example: oracle.install.asm.diskGroup.AUSize=4
# size unit is MB
#
#-------------------------------------------------------------------------------
oracle.install.asm.diskGroup.AUSize=4
```

Figure 8-27 ASM parameters for installation

– oracle.install.asm.SYSASMPassword=oracle, oracle.install.asm.diskGroup.name=DATA, oracle.install.asm.diskGroup.redundancy=EXTERNAL, and oracle.install.asm.diskGroup.AUSize=4

```
#-------------------------------------------------------------------------------
# List of disks to create a ASM DiskGroup
# (Use this variable only if failure groups configuration is not required)
# Example:
#     For Unix based Operating System:
#     oracle.install.asm.diskGroup.disks=/oracle/asm/disk1,/oracle/asm/disk2
#     For Windows based Operating System:
#     oracle.install.asm.diskGroup.disks=\\.\ORCLDISKDATA0,\\.\ORCLDISKDATA1
#
#-------------------------------------------------------------------------------
oracle.install.asm.diskGroup.disks=/dev/mapper/orad1
```

Figure 8-28 ASM disks

– `oracle.install.asm.diskGroup.disks=/dev/mapper/orad1`

```
#-------------------------------------------------------------------------------
# The disk discovery string to be used to discover the disks used create a ASM DiskGroup
#
# Example:
#     For Unix based Operating System:
#     oracle.install.asm.diskGroup.diskDiscoveryString=/oracle/asm/*
#     For Windows based Operating System:
#     oracle.install.asm.diskGroup.diskDiscoveryString=\\.\ORCLDISK*
#
#-------------------------------------------------------------------------------
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/orad*
```

Figure 8-29 ASM Disk Discovery String

– `oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/orad*`

```
#
oracle.install.asm.diskGroup.diskDiscoveryString=/dev/mapper/orad*

#-------------------------------------------------------------------------------
# Password for ASMSNMP account
# ASMSNMP account is used by Oracle Enterprise Manager to monitor Oracle ASM instanc
es
#-------------------------------------------------------------------------------
oracle.install.asm.monitorPassword=oracle

#
```

Figure 8-30 ASM monitor password

– `oracle.install.asm.monitorPassword=oracle`

```
###############################################################################
#                                                                             #
#                    Root script execution configuration                      #
#                                                                             #
###############################################################################

#------------------------------------------------------------------------------
# Specify the root script execution mode.
#
#   - true  : To execute the root script automatically by using the appropriate configuration methods.
#   - false : To execute the root script manually.
#
# If this option is selected, password should be specified on the console.
#------------------------------------------------------------------------------
oracle.install.crs.rootconfig.executeRootScript=true

#------------------------------------------------------------------------------
# Specify the configuration method to be used for automatic root script execution.
#
# Following are the possible choices:
#   - ROOT
#   - SUDO
#------------------------------------------------------------------------------
oracle.install.crs.rootconfig.configMethod=ROOT
```

Figure 8-31 Automatic root script execution parameters

– `oracle.install.crs.rootconfig.executeRootScript=true` and `oracle.install.crs.rootconfig.configMethod=ROOT`

3. After the required values are entered, run the following command to start the installation in silent mode:

```
./gridSetup.sh -silent -ignorePrereqFailure -responseFile
/u01/app/oracle/product/19.0.0/grid/install/response/gridsetup.rsp
```

Figure 8-32 - Figure 8-34 show the output of the command.

```
[oracle@rhel770 grid]$ ls
addnode      css         demo         gpnp         instantclient  jlib     nls          ord     perl     racg      rootupgrade.sh  sqlplus   utl
assistants   cv          diagnostics  gridSetup.sh inventory      ldap     OPatch       ords    plsql    rdbms     runcluvfy.sh    srvm      welcome.html
bin          dbjava      dmu          has          javavm         lib      opmn         oss     precomp  relnotes  sdk             suptools  wwg
clone        dbs         env.ora      hs           jdbc           md       oracore      oui     QOpatch  rhp       slax            ucp       xag
crs          deinstall   evm          install      jdk            network  oradiag_aime owm     qos      root.sh   sqlpatch        usm       xdk
[oracle@rhel770 grid]$ pwd
/u01/app/oracle/product/19.0.0/grid
[oracle@rhel770 grid]$  ./gridSetup.sh -silent -ignorePrereqFailure -responseFile /u01/app/oracle/product/19.0.0/grid/install/response/gridsetup.rsp
```

Figure 8-32 Grid setup script execution command

```
[oracle@rhel770 grid]$  ./gridSetup.sh -silent -ignorePrereqFailure -responseFile /u
01/app/oracle/product/19.0.0/grid/install/response/gridsetup.rsp
Launching Oracle Grid Infrastructure Setup Wizard...

[WARNING] [INS-30011] The SYS password entered does not conform to the Oracle recomm
ended standards.
   CAUSE: Oracle recommends that the password entered should be at least 8 character
s in length, contain at least 1 uppercase character, 1 lower case character and 1 di
git [0-9].
   ACTION: Provide a password that conforms to the Oracle recommended standards.
[WARNING] [INS-30011] The ASMSNMP password entered does not conform to the Oracle re
commended standards.
   CAUSE: Oracle recommends that the password entered should be at least 8 character
s in length, contain at least 1 uppercase character, 1 lower case character and 1 di
git [0-9].
   ACTION: Provide a password that conforms to the Oracle recommended standards.
[WARNING] [INS-41808] Possible invalid choice for OSASM Group.
   CAUSE: The name of the group you selected for the OSASM group is commonly used to
 grant other system privileges (For example: asmdba, asmoper, dba, oper).
   ACTION: Oracle recommends that you designate asmadmin as the OSASM group.
[WARNING] [INS-32047] The location (/u01/app/oraInventory) specified for the central
 inventory is not empty.
   ACTION: It is recommended to provide an empty location for the inventory.

 Enter password for 'root' user:
```

Figure 8-33 Entering password for root to run script

```
 Enter password for 'root' user:
[WARNING] [INS-13014] Target environment does not meet some optional requirements.
   CAUSE: Some of the optional prerequisites are not met. See logs for details. grid
SetupActions2020-03-30_07-50-02PM.log
   ACTION: Identify the list of failed prerequisite checks from the log: gridSetupAc
tions2020-03-30_07-50-02PM.log. Then either from the log file or from installation m
anual find the appropriate configuration to meet the prerequisites and fix it manual
ly.
The response file for this session can be found at:
 /u01/app/oracle/product/19.0.0/grid/install/response/grid_2020-03-30_07-50-02PM.rsp

You can find the log of this install session at:
 /tmp/GridSetupActions2020-03-30_07-50-02PM/gridSetupActions2020-03-30_07-50-02PM.lo
g
You can find the log of this install session at:
 /u01/app/oraInventory/logs/UpdateNodeList2020-03-30_07-50-02PM.log

Successfully Setup Software with warning(s).
Moved the install session logs to:
 /u01/app/oraInventory/logs/GridSetupActions2020-03-30_07-50-02PM
[oracle@rhel770 grid]$
[oracle@rhel770 grid]$
```

Figure 8-34 Grid setup run completed

4. Check the logs to make sure that any warnings or errors regarding prerequisites are understood and addressed.

Now, the Oracle Grid Infrastructure for a single instance is completed.

8.1.16 Oracle Grid Infrastructure for a single instance verification

After installing Oracle Grid Infrastructure for a single instance, use the `asmcmd` command-line utility to test the Oracle ASM installation.

Use ASMCMD to list the disk groups for the Oracle ASM instance:

```
$ORACLE_HOME/bin/asmcmd lsdg
```

If the Oracle ASM instance is running, ASMCMD connects by default as the SYS user with SYSASM privileges and is available, as shown in Figure 8-35 and Figure 8-36.

```
[oracle@rhel770 ~]$ env | grep ORA
ORACLE_SID=+ASM
ORACLE_BASE=/u01/app/grid_base
ORACLE_HOME=/u01/app/oracle/product/19.0.0/grid
[oracle@rhel770 ~]$ cd $ORACLE_HOME
[oracle@rhel770 grid]$ asmcmd lsdg
State    Type    Rebal  Sector  Logical_Sector  Block       AU  Total_MB  Free_MB  Req_mir_free_MB
Usable_file_MB  Offline_disks  Voting_files  Name
MOUNTED  EXTERN  N         4096            4096   4096  4194304     20480    20380                0
         20380              0             N  DATA/
[oracle@rhel770 grid]$
```

Figure 8-35 Output of asmcmd to display disk groups

```
[oracle@rhel770 grid]$ asmcmd
ASMCMD> ls
DATA/
ASMCMD> cd DATA
ASMCMD> ls
ASM/
orapwasm
ASMCMD> cd ASM
ASMCMD> ls
ASMPARAMETERFILE/
PASSWORD/
ASMCMD> cd ASMPARAMETERFILE
ASMCMD> ls
REGISTRY.253.1036171411
ASMCMD>
```

Figure 8-36 asmcmd samples

Also, we can use the **crsctl** commands to check the status of “has” and other resources, as shown in Figure 8-37.

```
[oracle@rhel770 grid]$ crsctl check has
CRS-4638: Oracle High Availability Services is online
[oracle@rhel770 grid]$ crsctl status res -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------------------
ora.DATA.dg
               ONLINE  ONLINE       rhel770                  STABLE
ora.LISTENER.lsnr
               ONLINE  ONLINE       rhel770                  STABLE
ora.asm
               ONLINE  ONLINE       rhel770                  Started,STABLE
ora.ons
               OFFLINE OFFLINE      rhel770                  STABLE
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.cssd
      1        ONLINE  ONLINE       rhel770                  STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.evmd
      1        ONLINE  ONLINE       rhel770                  STABLE
--------------------------------------------------------------------------------
[oracle@rhel770 grid]$
```

Figure 8-37 Output of crsctl command

8.2 Oracle Database binary file installation

The image file for Oracle Database that is shown in Figure 8-38 is downloaded and available.

```
[oracle@rhel770 19c]$ ls -alF *19cDB*
-rw-r--r--. 1 oracle oinstall 2835755051 Jun  6  2019 V982648-01_19cDB.zip
[oracle@rhel770 19c]$
```

Figure 8-38 Oracle Database image file

Unset any `ORACLE_BASE`, `ORACLE_HOME`, and `ORACLE_SID` values, and then, create the `oracle` home directory. Extract the image files into that `oracle` home directory by running the following commands:

```
# mkdir -p /u01/app/oracle/product/19.0.0/dbhome_1
# cd  /u01/app/oracle/product/19.0.0/dbhome_1
# unzip -q /u01/code/19c/V982648-01_19cDB.zip
```

Figure 8-39 shows the output of these commands.

```
[oracle@rhel770 19c]$ env | grep ORA
[oracle@rhel770 19c]$ mkdir -p /u01/app/oracle/product/19.0.0/dbhome_1
[oracle@rhel770 19c]$ cd /u01/app/oracle/product/19.0.0/dbhome_1
[oracle@rhel770 dbhome_1]$ unzip -q /u01/code/19c/V982648-01_19cDB.zip
[oracle@rhel770 dbhome_1]$ ls
addnode     cv           drdaas          javavm  network  ords     racg           sqlj      wwg
apex        data         dv              jdbc    nls      oss      rdbms          sqlpatch  xdk
assistants  dbjava       env.ora         jdk     odbc     oui      relnotes       sqlplus
bin         dbs          has             jlib    olap     owm      root.sh        srvm
clone       deinstall    hs              ldap    OPatch   perl     runInstaller   suptools
crs         demo         install         lib     opmn     plsql    schagent.conf  ucp
css         diagnostics  instantclient   md      oracore  precomp  sdk            usm
ctx         dmu          inventory       mgw     ord      QOpatch  slax           utl
[oracle@rhel770 dbhome_1]$
```

Figure 8-39 Output of image file installation

8.2.1 GUI installation of Oracle Database

As a Oracle install user (in our example, `oracle`), open a VNC terminal and run **`runInstaller`**, as shown in Figure 8-40.

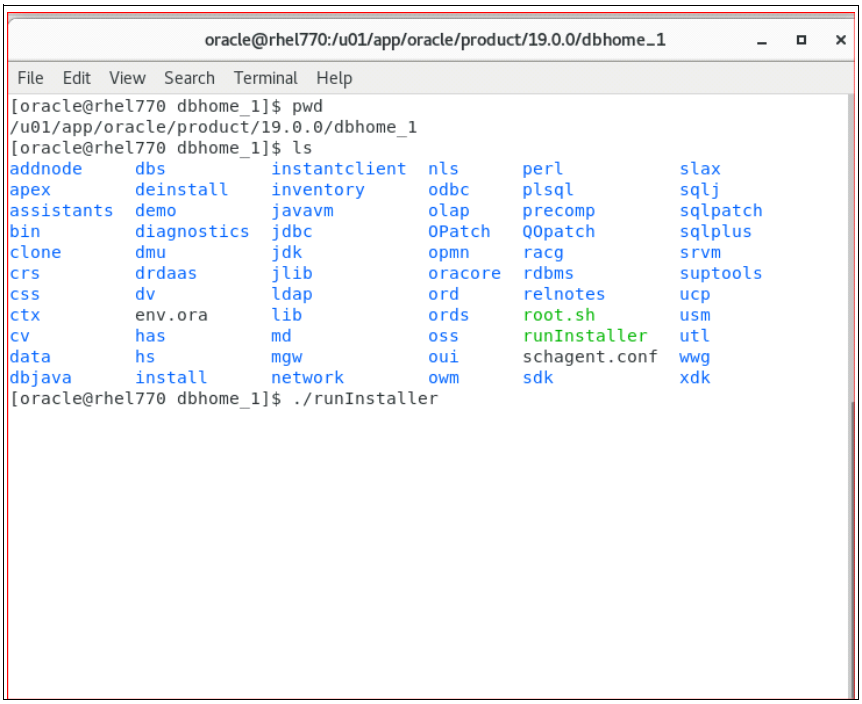

Figure 8-40 Running the runInstaller command in a VNC terminal

Complete the following steps:

1. In the Select Configuration Option window, select **Set Up Software Only**, as show in Figure 8-41.

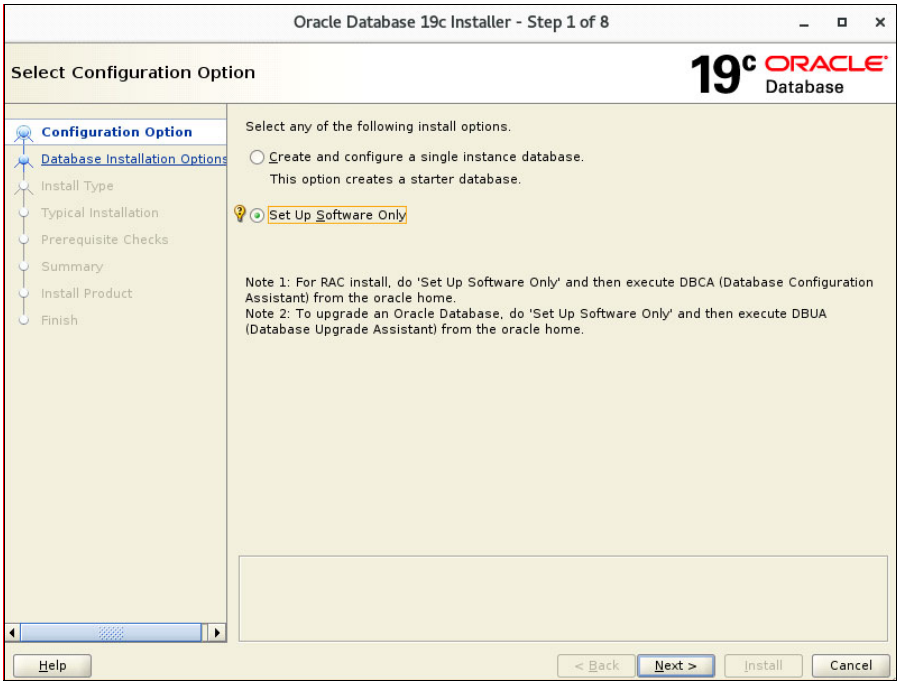

Figure 8-41 Select Configuration Option window

2. In the Select Database Installation Option window, select **Single Instance database installation**.
3. In the next Select Database Edition window, click **Enterprise Edition**.
4. In the next window, specify the value for the Oracle base directory. This value is the location where all the diag information for all the Oracle products are stored. In our example, we specified `/u01/app/oracle_base`.
5. In the following Privileged Operating System Groups window, specify **dba** for all groups, except **oper** for the Database operator group to keep things simple.
6. The installation process performed prerequisite checks and reported a Swap size error, which we ignored.
7. The summary window is displayed. Select **Install**.

8. When the Install Product window prompts you to run the `root.sh` script to be run as root, run it and then, click **OK** (see Figure 8-42).

```
[root@rhel770 ~]# /u01/app/oracle/product/19.0.0/dbhome_1/root.sh
Performing root user operation.

The following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /u01/app/oracle/product/19.0.0/dbhome_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.
Oracle Trace File Analyzer (TFA - Standalone Mode) is available at :
    /u01/app/oracle/product/19.0.0/dbhome_1/bin/tfactl

Note :
1. tfactl will use TFA Service if that service is running and user has been granted access
2. tfactl will configure TFA Standalone Mode only if user has no access to TFA Service or TFA is not
 installed

[root@rhel770 ~]#
```

Figure 8-42 Running the root.sh script

9. The Finish window opens, which indicates that the registration of Oracle Database was successful (see Figure 8-43).

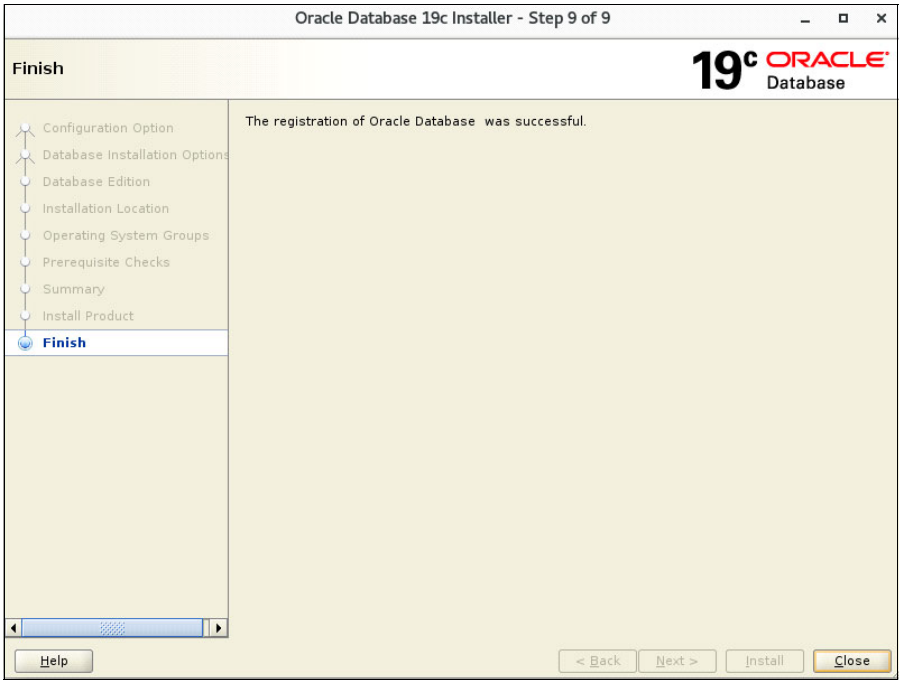

Figure 8-43 Finish window

8.2.2 Silent installation

Typically, the installer runs in interactive mode, by prompting to provide information in graphical user interface (GUI) screens. We can also run the installer from a command prompt using Silent mode and providing the values in a response file.

We can include all of the responses 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.

The response file **`db_install.rsp`** is available at the oracle `install/response` folder; in our example, at the following location:

```
/u01/app/oracle/product/19.0.0/dbhome_1/install/response
```

Make a backup copy of the response file and in our case as `db_install.rsp.orig`, as shown in Figure 8-44.

```
[oracle@rhel770 dbhome_1]$ cd install
[oracle@rhel770 install]$ ls
acls            envVars.properties  mesg            pilot           sizes
chainedInstall  files.lst           modmakedeps.pl  response        utl
config          jlib                orabasetab      root_schagent.sh
[oracle@rhel770 install]$ cd response/
[oracle@rhel770 response]$ ls
db_install.rsp
[oracle@rhel770 response]$ cp db_install.rsp db_install.rsp.orig
[oracle@rhel770 response]$ vi db_install.rsp
```

Figure 8-44 Backup copy of the response file

In the `db_install.rsp` file, we specified the following values:

```
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=oinstall
INVENTORY_LOCATION=/u01/app/oraInventory
ORACLE_BASE=/u01/app/oracle_base
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=oper
oracle.install.db.OSBACKUPDBA_GROUP=dba
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
```

After the required values are entered, we used the following command to start the installation in silent mode:

```
./runInstaller -silent -ignorePrereqFailure -responseFile
/u01/app/oracle/product/19.0.0/dbhome_1/install/response/db_install.rsp
```

Check the logs to make sure no prerequisite warnings or errors exist.

The Oracle Database for a single instance is completed.

8.3 Creating the Oracle Database

The next step is to create the Oracle Database and for that you can use the `dbca` utility in GUI mod.

1. Set the ORACLE_HOME and ORACLE_BASE directory. Then, in a VNC client terminal, run the **dbca** command. This process starts the database creation process.
2. In the following Select Database Operation window select the **Create a database** option, as shown in Figure 8-45.

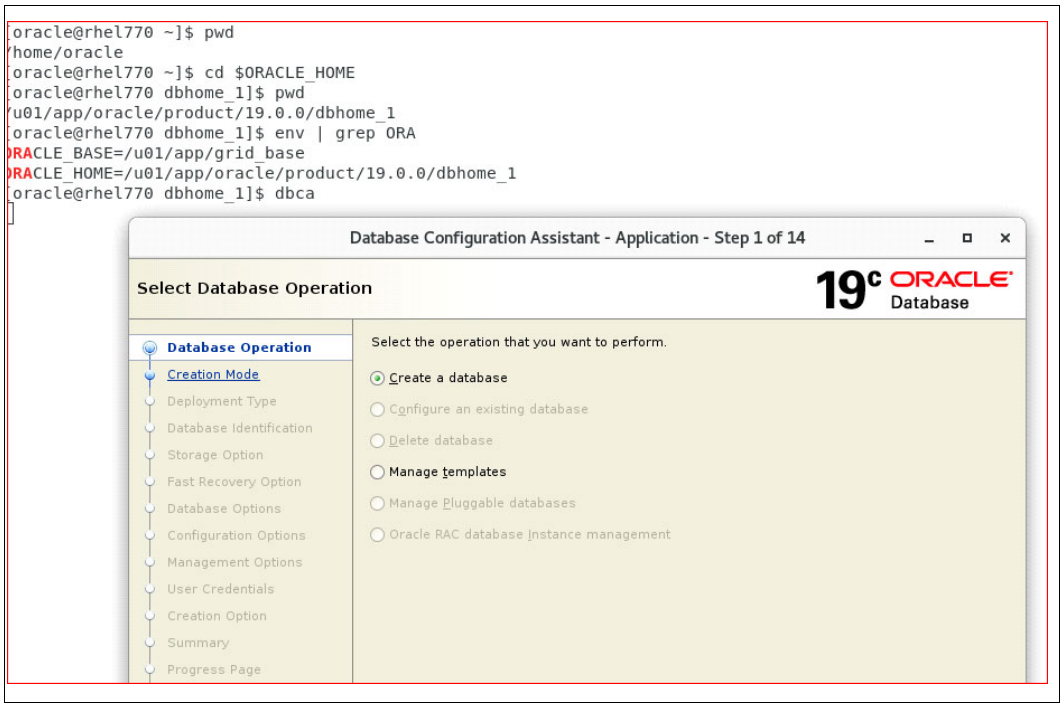

Figure 8-45 Creating a database

3. In the Select Database Creation Mode window, choose the **Advanced configuration** option.
4. In the Select Database Deployment Type window, choose the **Oracle Single Instance database** option for the database type and then, choose the **General Purpose or Transaction Processing**" template for the database.

In the Specify Database Identification Details window, we chose the following options, as show in Figure 8-46.

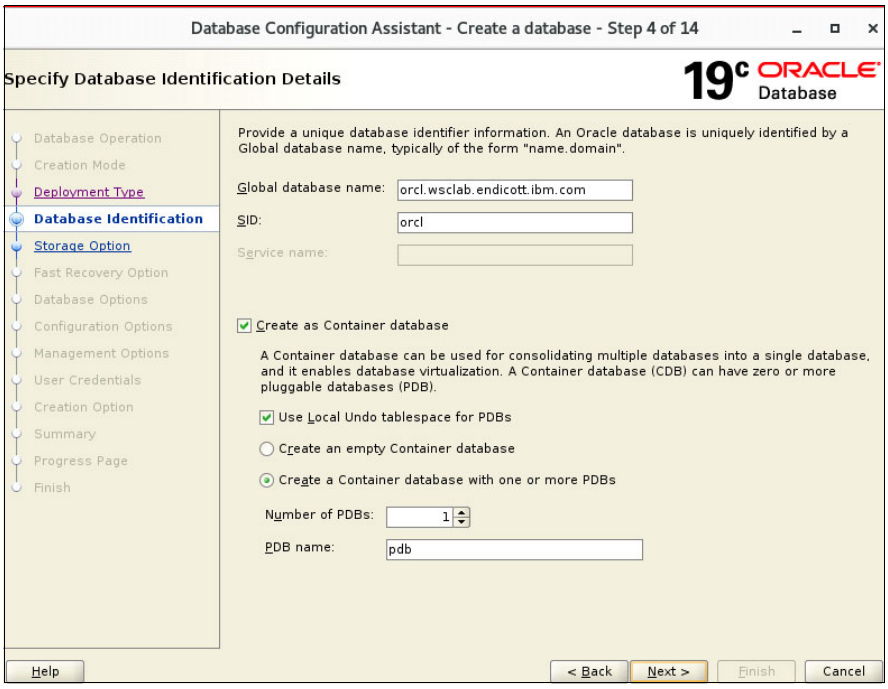

Figure 8-46 Specify Database Identification Details window

5. In the Select Database Storage Option window we choose the ASM for Database files storage type and selected the **Use Oracle-Managed Files (OMF)** option.
6. For the Select Fast Recovery Option, we did not specify the Fast Recovery Area and did not enable archiving.
7. For the Specify Network Configuration Details window, we chosen the default listener that is configured. In our example, **LISTENER**.
8. In the Select Oracle Data Vault Config Option window, we did not choose any option.
9. In the Specify Configuration Options window, we accepted the default values.
10. In the Specify management Options window, we accepted default EM values.
11. In the Specify Database user Credentials window, we choose **Use the same administrative passwords for all accounts** and specified the password that conformed to Oracle specifications.
12. In the Select Database Creation Option window, we selected **Create Database**.
13. In the Summary window, we selected **Finish** and the database creation process started.

    The Progress Page shows the progress of the database creation steps.

After some time, the `orcl` database was created.

You can use the **`crsctl`** command to see that the newly created database `orcl` is in online mode:

```
# /u01/app/oracle/product/19.0.0/grid/bin/crsctl status res -t
```

Figure 8-47 shows the output of the command.

```
[oracle@rhel770 ~]$ /u01/app/oracle/product/19.0.0/grid/bin/crsctl status res -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------------------
ora.DATA.dg
               ONLINE  ONLINE       rhel770                  STABLE
ora.LISTENER.lsnr
               ONLINE  ONLINE       rhel770                  STABLE
ora.asm
               ONLINE  ONLINE       rhel770                  Started,STABLE
ora.ons
               OFFLINE OFFLINE      rhel770                  STABLE
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.cssd
      1        ONLINE  ONLINE       rhel770                  STABLE
ora.diskmon
      1        OFFLINE OFFLINE                               STABLE
ora.evmd
      1        ONLINE  ONLINE       rhel770                  STABLE
ora.orcl.db
      1        ONLINE  ONLINE       rhel770                  Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /dbhome_1,STABLE
--------------------------------------------------------------------------------
[oracle@rhel770 ~]$
```

Figure 8-47 Output of crsctl command status

You can also use the **sqlplus** command to see that `orcl` database is running, as shown in Figure 8-48.

```
[oracle@rhel770 logs]$ env | grep ORA
ORACLE_SID=orcl
ORACLE_BASE=/u01/app/oracle_base
ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1
[oracle@rhel770 logs]$ sqlplus /'as sysdba'

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Apr 1 15:14:51 2020
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Produc
tion
Version 19.3.0.0.0
[oracle@rhel770 logs]$
```

Figure 8-48 Output of sqlplus command

9

Oracle RAC installation

In this chapter, we discuss installing Oracle RAC.

This chapter includes the following topics:

9.1 Installing Oracle Grid Infrastructure for a RAC Cluster

To install Oracle RAC 19c, you must have Oracle Grid Infrastructure (Oracle Clusterware and Oracle ASM) 19c installed on your cluster. The Oracle Clusterware version must be equal to or greater than the Oracle RAC version that you plan to install.

The following documents were referenced for installing Oracle Grid Infrastructure and RAC:

- *Oracle Grid Infrastructure Installation and Upgrade Guide, 19c for Linux*, E96272-10
- *Oracle Real Application Clusters Installation Guide, 19c for Linux and UNIX*, E96277-03
- *Oracle Database Installation Guide, 19c for Linux*, E96432-11
- *Oracle Automatic Storage Management Administrator's Guide, 19c,* E96198-05
- *Oracle Clusterware Administration and Deployment Guide, 19c*, E95727-08
- *Oracle Real Application Clusters Administration and Deployment Guide, 19c,* E95728-07
- *RPM Checker when Installing Oracle Database 19c on Linux on System z* (Doc ID 2553465.1)
- *Requirements for Installing Oracle 19c on SUSE Linux Enterprise Server 12 on IBM: Linux on System z* (Doc ID 2628889.1)
- *Requirements for Installing Oracle 19c RDBMS on RH7 on IBM: Linux on System z* (Doc ID 2628892.1)
- *How to Manually Configure Disk Storage devices for use with Oracle ASM 11.2 and 12.1 on IBM: Linux on System z under Red Hat 6* (Doc ID 1377392.1)
- *Getting Started - 12c Grid Infrastructure / Oracle Database - IBM: Linux on System z (s390x)* (Doc ID 1574412.1)

9.1.1 Steps

The following steps were completed for Oracle Grid Infrastructure (19c) for a stand-alone server installation on a RHEL 7.7 Linux guest under z/VM 7 running on a LinuxONE server. We completed the installation on our first node, `adpnoe1`. For more information about that installation, see Chapter 8, “Installing Oracle” on page 101:

1. Validate Linux guest requirements.
2. Validate required Disk Space.
3. Validate required Linux RHEL 7.x RPMs for Oracle 19c.
4. Disable Linux Transparent Huge Pages.
5. Update Linux Kernel parameters.
6. Set up HugePages for Oracle Databases.
7. Create users and groups.
8. Create user directories for Oracle product installations.
9. Set up udev for Storage for ASM.
10. Install Grid infrastructure for Standalone server binary.
11. Verify Grid infrastructure for Standalone server binary.

9.1.2 Setting up udev for storage for ASM

Although we are not using `asmlib`, we use the udev rules to specify the available storage devices for Grid infrastructure. For more information about how to set up udev rules to use for ASM, see 3.2.8, “Storage options for Oracle Database” on page 22.

In this section, we describe the LUNs that we used for database data storage.

We used ECKD disks LUNs that are aliased as `ASM0800`, `ASM0801`, `ASM0802`, and `ASM0802`. Figure 9-1 and Figure 9-2 show the udev rules setup with permissions at both of our nodes.

```
[root@adpnode1 ~]# more /etc/udev/rules.d/90-udev-oracle.rules
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0800",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0800"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0801",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0801"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0802",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0802"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0803",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0803"
[root@adpnode1 ~]#
```

Figure 9-1 Node1 udev rules file

```
[root@adpnode2 ~]# more /etc/udev/rules.d/90-udev-oracle.rules
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0800",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0800"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0801",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0801"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0802",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0802"
ACTION=="add|change",KERNEL=="dasd*1",ENV{ID_PATH}=="ccw-0.0.0803",OWNER="oracle",GROUP="dba",MODE="0660",SYMLINK+="ASM0803"
[root@adpnode2 ~]#
```

Figure 9-2 Node2 udev rules file

9.1.3 Network considerations

For more information, see 7.2, “Networking” on page 93 for the network setup for Oracle on LinuxONE.

The public network names are in a domain name server (DNS). Running **`nslookup`** shows the following information:

```
Name:   adpnode1.wsclab.washington.ibm.com         Name:
adpnode2.wsclab.washington.ibm.com
Address: 9.82.21.32                                                  Address: 9.82.21.32
9.82.21.32              adpnode1.wsclab.washington.ibm.com                    adpnode1
9.82.21.34              adpnode2.wsclab.washington.ibm.com                    adpnode2
```

For this installation, the private interconnect uses a single network on subnet 192.0.1.0. The `/etc/hosts` entries on both nodes contain the following settings for the private network:

```
192.0.1.32              adpnode1-priv.wsclab.washington.ibm.com            adpnode1-priv
192.0.1.34              adpnode2-priv.wsclab.washington.ibm.com            adpnode2-priv
```

The VIPs are in a DNS and are in the same subnet as the public IPs:

```
9.82.21.33             adpnode1-vip.wsclab.washington.ibm.com              adpnode1-vip
9.82.21.35             adpnode2-vip.wsclab.washington.ibm.com              adpnode2-vip
```

The Single Client Access Name (SCAN) name is with the following three IPs in DNS and in the same subnet as the public IPs and VIPs:

```
nslookup rac-scan1
[root@adpnode1 ~]# nslookup rac-scan1
Server:         9.82.22.2
Address:        9.82.22.2#53
Name:   rac-scan1.wsclab.washington.ibm.com
Address: 9.82.21.70
Name:   rac-scan1.wsclab.washington.ibm.com
Address: 9.82.21.71
Name:   rac-scan1.wsclab.washington.ibm.com
```

```
Address: 9.82.21.72
```

The public IPs, VIPs, and SCAN IPs must be in the same network subnet. The subnet mask that is used in our example is 255.255.252.0, which means all the IPs are in the same subnet.

9.1.4 Oracle Grid Infrastructure configuration

Starting with Oracle Grid Infrastructure 12c Release 2 (12.2), installation and configuration of Oracle Grid Infrastructure software is simplified with image-based installation. We followed the same processes to install the grid infrastructure for a stand-alone server installation.

We used the `runcluvfy` utility at the grid home to check that both nodes of the cluster meet the installation requirements. You can direct the output to a file to look at any errors.

As the `oracle` user, run the following command:

```
# ./runcluvfy.sh stage -pre crsinst -n adpnode1, adpnode2 -verbose >
/home/oracle/cluvfy.txt
```

Figure 9-3 shows the output of the command.

```
Pre-check for cluster services setup was unsuccessful on all the nodes.


Failures were encountered during execution of CVU verification request "stage -pre crsinst".

Verifying Swap Size ...FAILED
adpnode2: PRVF-7573 : Sufficient swap size is not available on node "adpnode2"
          [Required = 7.6892GB (8062740.0KB) ; Found = 1021.8555MB
          (1046380.0KB)]

adpnode1: PRVF-7573 : Sufficient swap size is not available on node "adpnode1"
          [Required = 7.6892GB (8062740.0KB) ; Found = 1.0133GB (1062500.0KB)]
```

Figure 9-3 Cluster verification preinstallation check

In our case, the verification failed with the following errors, which can be safely ignored:

```
Verifying Swap Size ...FAILED
Verifying Package: cvuqdisk-1.0.10-1 ...FAILED
Verifying /dev/shm mounted as temporary file system ...FAILED
```

Now, we are ready to start the installation. We can use the Graphical interactive method or, if we do not have `x windows` facility in the environment, we can use the response file and perform a silent installation.

9.1.5 GUI installation

As a Grid install user, in our case as `oracle` in a x window, such as a VNC terminal to display (GUI), start `gridSetup.sh` from the first node (in our case, `adpnode1`), as shown in Figure 9-4.

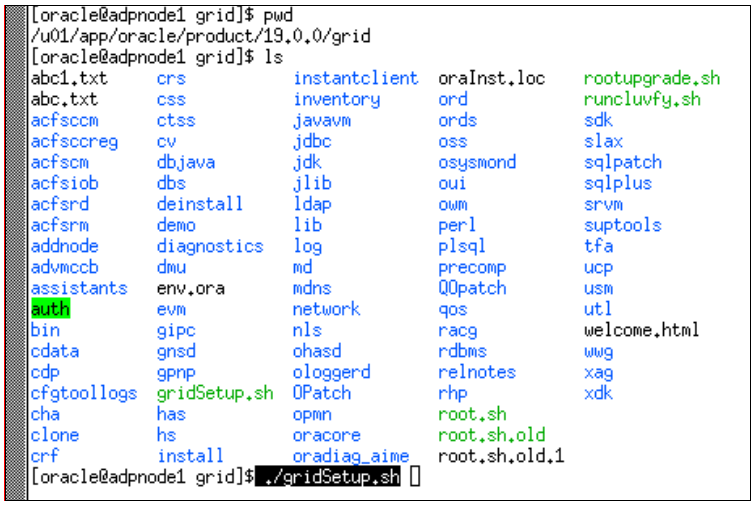

Figure 9-4 Invocation of gridsetup GUI

Complete the following steps:

1. In the Select Configuration Option window, we chose the **Configure Oracle Grid Infrastructure for a New Cluster** option.
2. In the Select Cluster Configuration Option window, we chose the **Configure an Oracle Standalone Cluster** option.
3. In the Grid Plug and Play Information window, we chose the **Create Local SCAN** option. We specified the information that is listed in Table 9-1.

Table 9-1 Scan name information

Field name	Value
Cluster name	`adpnode-cluster`
SCAN name	`rac-scan1`
SCAN port	`1521`

4. In the Cluster Node Information window, we entered the list of nodes to be managed by Oracle Grid Infrastructure with their Public Hostname and Virtual Hostname. Then, we set up the SSH connectivity by specifying the Oracle user ID for the operating system user name and its password.
5. In the Specify Network Interface Usage window, we specified the private network interface as **ASM & Private** and the public interface as **Public**.
6. In the Storage Option Information window for placing the OCR and Voting disk files, we chose the **Use Oracle Flex ASM for storage** option.
7. For Create Grid Infrastructure Management Repository, we choose **No**. (In Oracle 19c, this requirement is not necessary.)

8. In the Create ASM Disk Group window, we specified the information that is listed in Table 9-2.

Table 9-2 ASM Disk group name

Field name	Value
Disk group name	OCR
Redundancy	External
Allocation Unit Size	4 MB
Select Disks	/dev/ASM0800
Disk Discovery Path	/dev/ASM*

9. In the Specify ASM Password window, we choose the **Use same passwords for these accounts** option and specified the password that conformed to Oracle specifications.
10. In the Operating System Groups window, we specified the information that is listed in Table 9-3.

Table 9-3 ASM groups information

Field name	Value
Oracle ASM Administrator (OSASM) Group	dba
Oracle ASM DBA (OSDBA) Group	asmdba
Oracle ASM Operator (OSOPER for ASM) Group optional	oinstall

We ignored invalid group chosen messages.

11. In the next window, we specified the value for Oracle base Directory. This value is the location where all the diag information for all the Oracle products is stored. In our case, we specified `/u01/app/oracle_base`.
12. In the Create Inventory window, we specified the value for Oracle Inventory Directory. This value is the location where Oracle stores the inventory of all the installed products. In our case, we specified `/u01/app/oraInventory`.
13. We chose not to automatically run the configuration scripts as `root` user in the Root Script execution configuration window.
14. The installation process performed prerequisite checks and reported the following errors:
    - Swap size warning
    - RPM Package Manager database information
    - `/dev/shm` mounted as temporary file system warning

    Because these errors can be ignored, we selected the **Ignore All** option and clicked **Next**.
15. The Summary window displayed and we selected **Install**.

16. After some time, the installer prompted us to run two scripts as `root` at the cluster nodes, as shown in Figure 9-5.

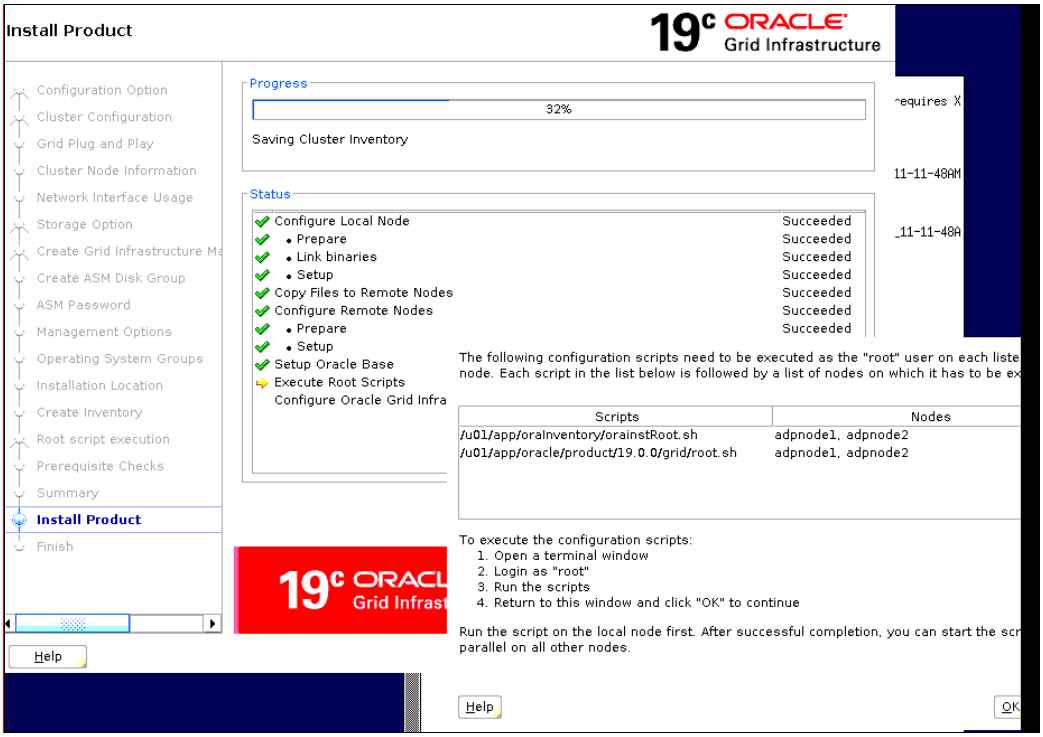

Figure 9-5 Scripts execution as root user

17. We ran `orainstRoot.sh` at both nodes as `root` user, as shown in Figure 9-6.

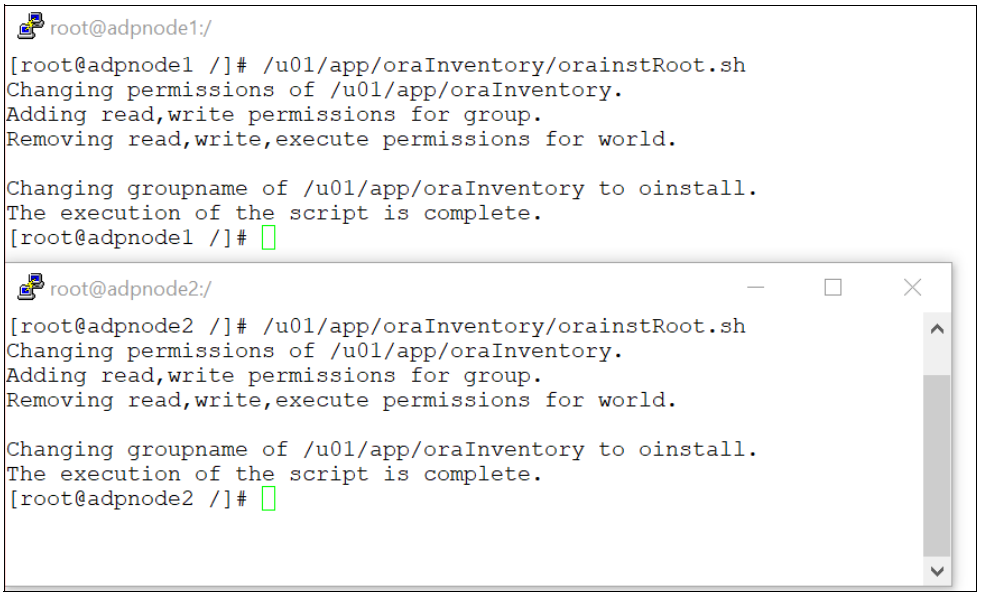

Figure 9-6 orainstRoot.sh execution at both nodes

18. We ran `root.sh` at both nodes as `root` user, as shown in Figure 9-7 and Figure 9-8.

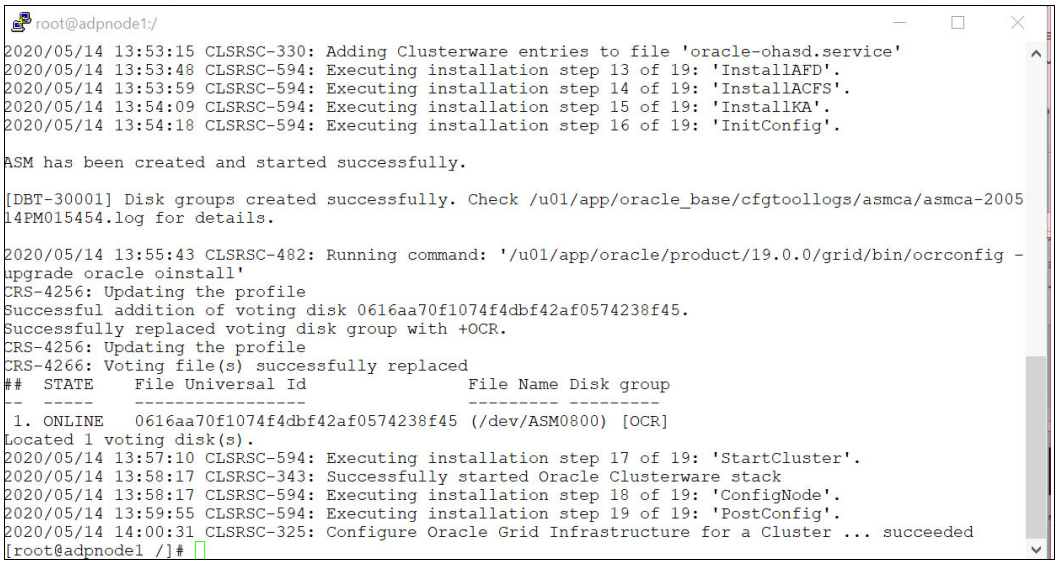

Figure 9-7 Results of root.sh execution at Node 1

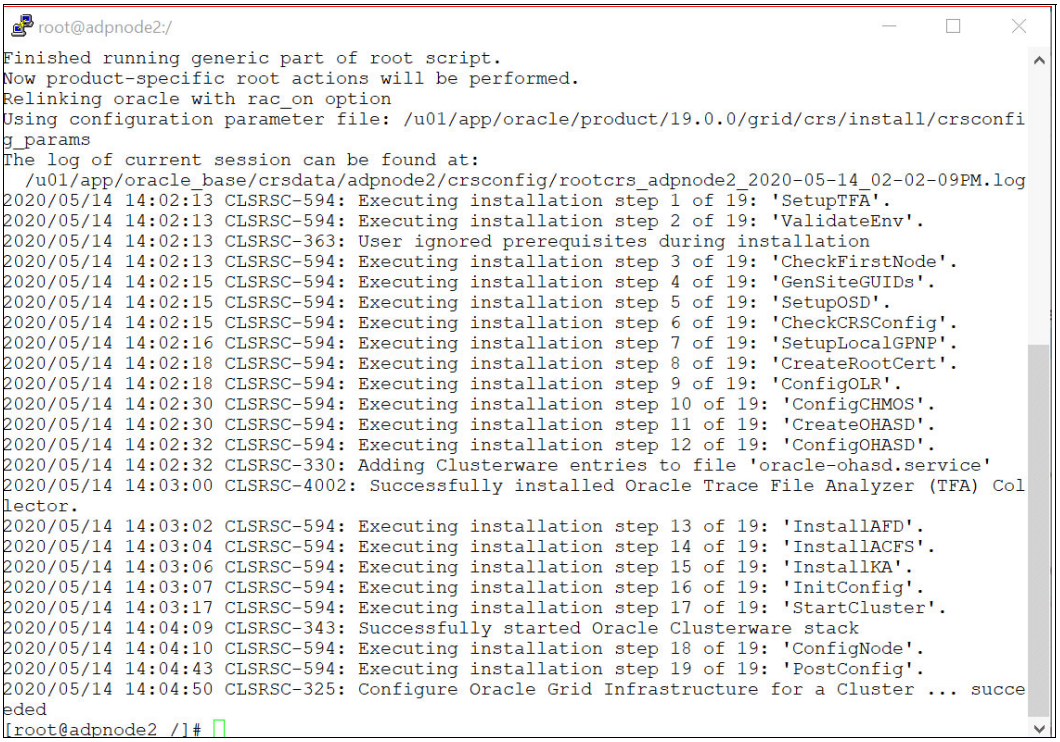

Figure 9-8 Results of reoot.sh execution at Node 2

19. After the installation completed and the grid infrastructure was configured, the successful completion window was displayed.

20. We used the **`crsctl check has`** command to verify the cluster in `adpnode1`, as shown in Figure 9-9.

```
[root@adpnode1 grid]# bin/crsctl check cluster -all
**************************************************************
adpnode1:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
**************************************************************
adpnode2:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
**************************************************************
[root@adpnode1 grid]#
```

Figure 9-9 Output of crsctl check cluster command

The output of the command is shown in Figure 9-10 and Figure 9-11 on page 138.

```
root@adpnode1:~
[root@adpnode1 ~]# /u01/app/oracle/product/19.0.0/grid/bin/crsctl status res -t
--------------------------------------------------------------------------------
Name           Target  State        Server                   State details
--------------------------------------------------------------------------------
Local Resources
--------------------------------------------------------------------------------
ora.LISTENER.lsnr
               ONLINE  ONLINE       adpnode1                 STABLE
               ONLINE  ONLINE       adpnode2                 STABLE
ora.net1.network
               ONLINE  ONLINE       adpnode1                 STABLE
               ONLINE  ONLINE       adpnode2                 STABLE
ora.ons
               ONLINE  ONLINE       adpnode1                 STABLE
               ONLINE  ONLINE       adpnode2                 STABLE
--------------------------------------------------------------------------------
Cluster Resources
--------------------------------------------------------------------------------
ora.ASMNET1LSNR_ASM.lsnr(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 STABLE
      2        ONLINE  ONLINE       adpnode2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.DATA.dg(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 STABLE
      2        ONLINE  ONLINE       adpnode2                 STABLE
      3        ONLINE  OFFLINE                               STABLE
ora.FRA.dg(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 STABLE
      2        ONLINE  ONLINE       adpnode2                 STABLE
      3        ONLINE  OFFLINE                               STABLE
ora.LISTENER_SCAN1.lsnr
      1        ONLINE  ONLINE       adpnode2                 STABLE
ora.LISTENER_SCAN2.lsnr
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.LISTENER_SCAN3.lsnr
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.OCR.dg(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 STABLE
      2        ONLINE  ONLINE       adpnode2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.adpnode1.vip
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.adpnode2.vip
      1        ONLINE  ONLINE       adpnode2                 STABLE
```

Figure 9-10 Output of crsctl status command: Part 1

```
      1        ONLINE  ONLINE       adpnode2                 STABLE
ora.asm(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 Started,STABLE
      2        ONLINE  ONLINE       adpnode2                 Started,STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.asmnet1.asmnetwork(ora.asmgroup)
      1        ONLINE  ONLINE       adpnode1                 STABLE
      2        ONLINE  ONLINE       adpnode2                 STABLE
      3        OFFLINE OFFLINE                               STABLE
ora.cvu
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.orcl.db
      1        ONLINE  ONLINE       adpnode1                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /oracle/dbhome_1,STA
                                                             BLE
      2        ONLINE  ONLINE       adpnode2                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /oracle/dbhome_1,STA
                                                             BLE
ora.scan1.vip
      1        ONLINE  ONLINE       adpnode2                 STABLE
ora.scan2.vip
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.scan3.vip
      1        ONLINE  ONLINE       adpnode1                 STABLE
--------------------------------------------------------------------------------
[root@adpnode1 ~]#
```

Figure 9-11 Output of crsctl status command: Part 2

Now, the Oracle Grid Infrastructure installation is completed.

9.2 Oracle Database binary installation

The following image file for Oracle Database is downloaded and available:

`V982648-01_19cDB.zip`

Unset any ORACLE_BASE, ORACLE_HOME, and ORACLE_SID values, create the Oracle `home` directory, and extract the image files in this directory at the `adpnode1` by running the following commands:

```
# mkdir -p /u01/app/oracle/product/19.0.0/oracle/dbhome_1
# cd  /u01/app/oracle/product/19.0.0/oracle/dbhome_1
# unzip -q /u01/code/19c/V982648-01_19cDB.zip
```

9.2.1 GUI installation of Oracle Database

As a Oracle install user (in our case as oracle in a x window, such as VNC terminal to display [GUI]), start `runInstaller`, as shown in Figure 9-12.

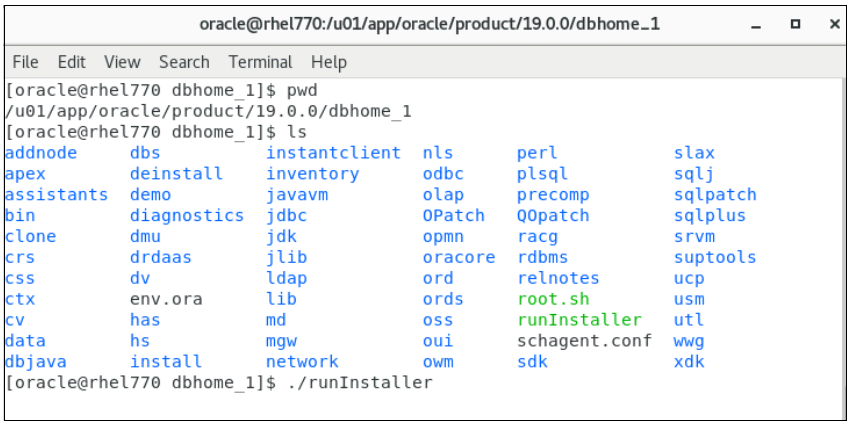

Figure 9-12 runInstaller GUI invocation

Complete the following steps:

1. In the Select Configuration Option window, choose the **Set Up Software Only** option.
2. In the Select Database Installation Option window. choose the **Oracle Real Application Clusters database installation** option
3. In the Select List of Nodes window, choose the nodes that you want to install. In our example, we selected both nodes: `adpnode1` and `adpnode2`.
4. In the Select Database Edition window, choose **Enterprise Edition**.
5. In the next window, specify the value for Oracle base Directory. This value is the location where all the diag information for all the Oracle products are stored. In our example, we specified `/u01/app/oracle_base`.
6. In the Privileged Operating System Groups window, specify **dba** for all groups, except **oper** for the Database operator group to keep things simple.
7. The installation process performs the prerequisite checks and reports the following errors:
   - Swap size warning.
   - RPM Package Manager database information.
   - `/dev/shm` mounted as temporary file system warning.

   Because these errors can be ignored, select **Ignore All** and then, click **Next**.
8. The Summary window is displayed. Select **Install**.

9. The Install Product window prompts you to run the `root.sh` script to be run as `root` at both nodes (see Figure 9-13 and Figure 9-14). Run it and then, click **OK**.

```
root@adpnode1:/
[root@adpnode1 /]# /u01/app/oracle/product/19.0.0/oracle/dbhome_1/root.sh
Performing root user operation.

The following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /u01/app/oracle/product/19.0.0/oracle/dbhome_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.
[root@adpnode1 /]#
```

Figure 9-13 root.sh execution at Node 1

```
root@adpnode2:/
[root@adpnode2 /]# /u01/app/oracle/product/19.0.0/oracle/dbhome_1/root.sh
Performing root user operation.

The following environment variables are set as:
    ORACLE_OWNER= oracle
    ORACLE_HOME=  /u01/app/oracle/product/19.0.0/oracle/dbhome_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.
[root@adpnode2 /]#
```

Figure 9-14 root.sh execution at Node 2

Now, the Finish window shows that the registration of Oracle Database was successful.

9.3 Creating the Oracle Database

The next step is to create the Oracle Database. For this process, use the `dbca` utility in GUI mode and follow the process as described in this section.

In our environment, we have four ECKD disks for ASM and we used the ASM utility `asmca` to create a disk group DATA and FRA. We use those disk groups when we create the database.

9.3.1 Creating a multitenant RAC database with DBCA

Set the ORACLE_HOME and ORACLE_BASE directory and in a VNC client terminal, run the **`dbca`** command. Complete the following steps:

1. In the Select Database Operation window, choose the **Create a database** option.
2. In the Select Database Creation Mode window, choose the **Advanced configuration** option.
3. In the Select Database Deployment Type window, choose the **Oracle Real Application Cluster database** option for the database type and the **General Purpose or Transaction Processing**" template for the database.
4. In the Select List of Nodes window, choose the nodes you want to install. In our example, we selected both nodes: `adpnode1` and `adpnode2`.
5. In the Specify Database Identification Details window, chose the options that are shown in Figure 9-15.

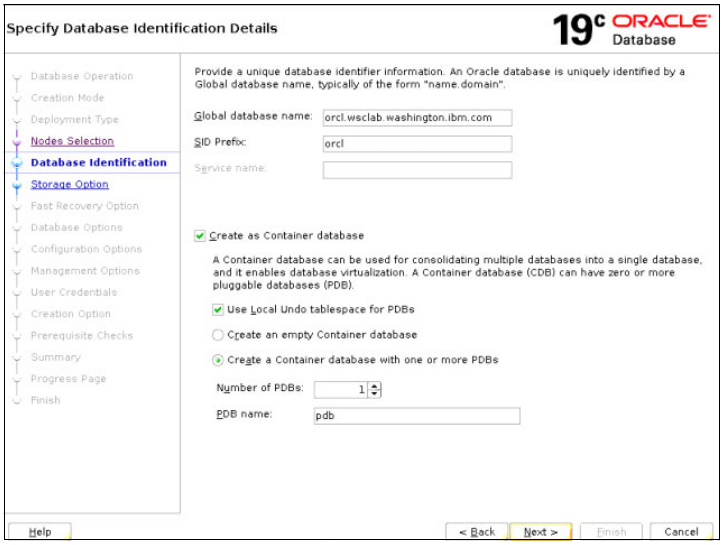

Figure 9-15 Database identification information

6. In the Select Database Storage Option window, choose **ASM** for the database files storage type with **+DATA** location and select the **Use Oracle-Managed Files (OMF)**" option.
7. In the Select Fast Recovery Option window, Fast Recovery Area is not specified and archiving is not enabled.
8. Do not select any options in the Specify management Options window.
9. Do not select any options in the Select Oracle Data Vault Config Option window.
10. Accept the default values in the Specify Configuration Options window.

11. In the Specify Database user Credentials window, select the **Use the same administrative passwords for all accounts** option and specify the password that conforms to Oracle specifications.
12. In the Select Database Creation Option window, select **Create Database**.
13. The installation process performs prerequisite checks and reports the following errors:
    - Swap size warning
    - `/dev/shm` mounted as temporary file system warning

    Because these errors can be ignored, select the **Ignore All** option and click **Next**.
14. In the Summary window, select **Finish** and the database creation process starts.

The Progress Page shows the progress of the database creation steps (see Figure 9-16). After some time, the `orcl` database is created.

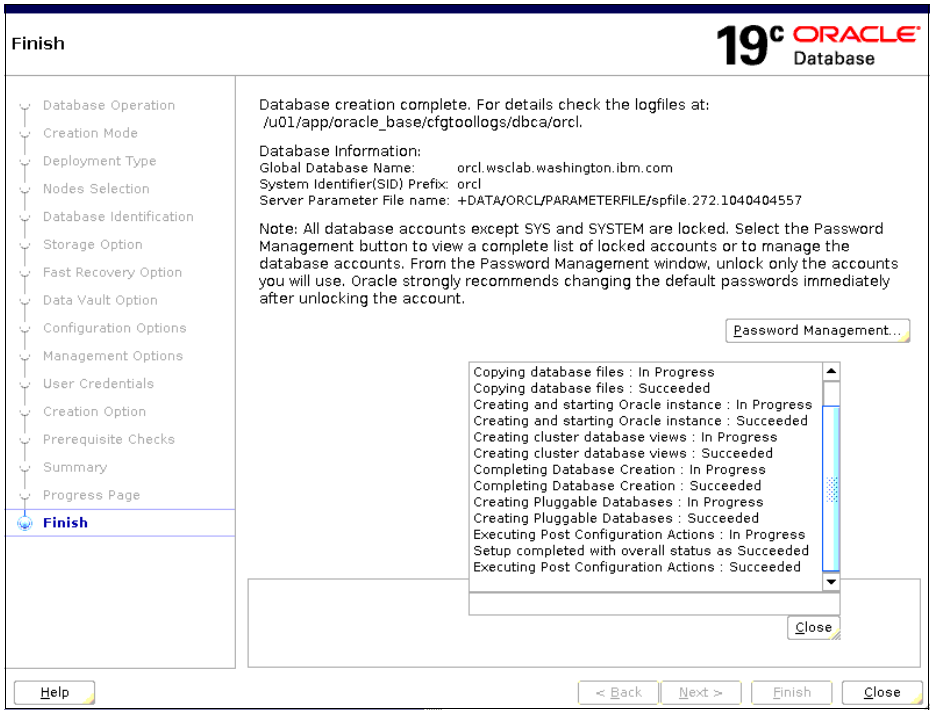

Figure 9-16 Output of crsctl command status

9.3.2 Checking the RAC cluster configuration

Several commands, such as **`crsctl`** and **`srvctl`**, can be used to check the configurations of a cluster. In this section, we discuss some of the frequently used variations of those commands.

You can use the **`crsctl`** command to see that the newly created database `orcl` is in online mode:

```
# /u01/app/oracle/product/19.0.0/grid/bin/crsctl status res -t
```

Figure 9-17 shows the output of the command.

```
      1        ONLINE  ONLINE       adpnode1                 STABLE
ora.orcl.db
      1        ONLINE  ONLINE       adpnode1                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /oracle/dbhome_1,STA
                                                             BLE
      2        ONLINE  ONLINE       adpnode2                 Open,HOME=/u01/app/o
                                                             racle/product/19.0.0
                                                             /oracle/dbhome_1,STA
                                                             BLE
```

Figure 9-17 Output of the crsctl command

You can also use the **srvctl** command to see that the newly created database `orcl` instances are running in the nodes:

```
# srvctl status database -d orcl
```

Figure 9-18 shows the output of the command.

```
[oracle@adpnode1 dbhome_1]$ pwd
/u01/app/oracle/product/19.0.0/oracle/dbhome_1
[oracle@adpnode1 dbhome_1]$ srvctl status database -d orcl
Instance orcl1 is running on node adpnode1
Instance orcl2 is running on node adpnode2
[oracle@adpnode1 dbhome_1]$
```

Figure 9-18 Output of the srvctl command

You also can use the **srvctl** command to see the status of the listener, as shown in Figure 9-19.

```
[oracle@adpnode1 dbhome_1]$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): adpnode2,adpnode1
[oracle@adpnode1 dbhome_1]$
```

Figure 9-19 Listener status

You can also use the **srvctl** command to see the status of the scan listeners, as shown in Figure 9-20.

```
[oracle@adpnode1 dbhome_1]$ srvctl status scan
SCAN VIP scan1 is enabled
SCAN VIP scan1 is running on node adpnode2
SCAN VIP scan2 is enabled
SCAN VIP scan2 is running on node adpnode1
SCAN VIP scan3 is enabled
SCAN VIP scan3 is running on node adpnode1
[oracle@adpnode1 dbhome_1]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node adpnode2
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node adpnode1
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node adpnode1
[oracle@adpnode1 dbhome_1]$
```

Figure 9-20 srvctl status output for scan

When the **lsnrctl** status is checked, the command displays which databases are listening, as shown in Figure 9-21.

```
Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date                14-MAY-2020 14:00:22
Uptime                    0 days 22 hr. 25 min. 8 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /u01/app/oracle/product/19.0.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/oracle_base/diag/tnslsnr/adpnode1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=9.82.21.32)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=9.82.21.33)(PORT=1521)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "+ASM_FRA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "+ASM_OCR" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "8a48f44a2a059470e0530ad680811882.wsclab.washington.ibm.com" has 1 instance(s).
  Instance "orcl1", status READY, has 1 handler(s) for this service...
Service "a5a33f051be829b6e05309521520eef0.wsclab.washington.ibm.com" has 1 instance(s).
  Instance "orcl1", status READY, has 1 handler(s) for this service...
Service "orcl.wsclab.washington.ibm.com" has 1 instance(s).
  Instance "orcl1", status READY, has 1 handler(s) for this service...
Service "orclXDB.wsclab.washington.ibm.com" has 1 instance(s).
  Instance "orcl1", status READY, has 1 handler(s) for this service...
Service "pdb.wsclab.washington.ibm.com" has 1 instance(s).
  Instance "orcl1", status READY, has 1 handler(s) for this service...
The command completed successfully
[oracle@adpnode1 dbhome_1]$
```

Figure 9-21 Output of lsnrctl status

You also can use the **sqlplus** command to see which databases are running, as shown in Figure 9-22.

```
SQL> col name format a10
SQL> select name,open_mode from v$containers;

NAME       OPEN_MODE
---------- ----------
CDB$ROOT   READ WRITE
PDB$SEED   READ ONLY
PDB        READ WRITE

SQL>
```

Figure 9-22 Output of sqlplus command

10

High availability

The IBM LinuxONE is a perfect choice for installations where high availability (HA) is a key customer criteria in their hardware infrastructure selection for Oracle Database deployments. In this chapter, we show how the highly regarded HA characteristics of IBM LinuxONE are an ideal complement for installations that are running the Oracle Database and use Oracle's specific HA features.

In this chapter, we describe many of the Oracle Database options that are available to customers as a standard feature or a chargeable License Fee Based option. For more information about all the Oracle options, see Oracle's documentation or online resources.

We also describe the LinuxONE features that play an important role for clients who are looking for a hardware server with a robust set of characteristics in this domain.

Finally, we provide examples of many scenarios where the Oracle Database that is running Linux on LinuxONE can provide an excellent HA enterprise-ready environment solution.

This chapter includes the following topics:

10.1 Oracle HA options

The topic of HA is important for installations that are running databases that process critical applications for the organizations that they serve. This subject affects almost every data center or cloud environment in the IT Industry. HA is top of mind for those customers who choose Oracle Database as their software in this area and have (or are considering) LinuxONE as their infrastructure hardware.

Oracle Maximum Availability Architecture (MAA) is a set of best practices that are developed by Oracle that help with a HA architecture. It uses the Oracle HA and disaster technologies. MAA is designed to provide data protection and availability by minimizing or eliminating planned and unplanned downtime at the technology layers, including hardware and software components. It delivers protection and HA for many types of failure, whether from hardware failures that cause data corruption, or anything ranging from human error to software malfunctions. It also includes HA features to help an installation when nature-related disasters affect a broad geographic area.

Oracle provides a comprehensive and integrated set of HA technologies that enable rapid recovery from failures and minimize planned downtimes. The focus of this section is on HA options that are related to the Oracle Database.

The available Oracle HA options are listed in Table 10-1.

Table 10-1 Oracle HA options

HA/DR feature	Benefit	Cost	Level of availability
Real Application Clusters One Node	Automated fail over (active/passive) for planned and unplanned outages	Oracle optional licensed feature	High Availability with minimal disruption to database (seconds0
Real Application Clusters	Continuous availability of cluster	Oracle optional licensed feature	High Availability with no disruption to database
Application Continuity	Protects applications from database session failures by replaying in-flight transactions	Licensed (included with RAC, RAC One Node, and Active Data Guard)	Disaster Recovery
Edition-Based Redefinition (EBR)	Enables online applications upgrades with uninterrupted availability of the applications	Included with Enterprise Edition	High Availability for maintenance
Oracle Flashback	Flashback provides efficient and quick error correction (logical and table rows)	Included with Enterprise Edition	High Availability for error correction
Online Reorganization and Redefinition	Provides flexibility to modify a table's physical attributes, while allowing users full access to the database	Included with Enterprise Edition	High Availability for database maintenance

HA/DR feature	Benefit	Cost	Level of availability
Data Guard/ Active Data Guard	Disaster-recovery solution that provides fast automatic failover, in the case of database or node failures.	Data Guard included with Enterprise Edition. Active Data Guard requires a license	Disaster Recovery
GoldenGate	Provides real-time data replication for heterogeneous environments	Licensed product	Disaster Recovery and Migration/Replication Services
Global Data Services (GDS)	GDS provides region-based workload routing, and replication lag-based workload routing for Active Data Guard	Included with GoldenGate or Active Data Guard	Disaster Recovery
Storage Base Replication Solutions	Storage replication uses disk consistency groups for databases and applications, to provide a simplified Disaster Recovery	Replication software cost (non-Oracle products)	Disaster Recovery
Oracle Site Guard	Oracle Site Guard enables administrators to automate complete site switchover or failover.	Included with Data Guard	Disaster Recovery
Recovery Manager (RMAN)	RMAN provides a foundation for backing up and recovering the Oracle Database.	Included with Oracle Database Enterprise Edition.	High Availability for back ups
Secure Backup (OSB)	OSB delivers centralized disk or backup	Included with Oracle Database Enterprise Edition	High Availability for back ups
Oracle Multitenant	Consists of a container database (CDB) for the metadata and the pluggable databases (PDB) which contain the application database (can be unplugged and plugged into different CDB databases)	Optional licensed feature of the Database	High Availability

HA/DR feature	Benefit	Cost	Level of availability
Oracle Sharding Sharded Database (SDB)	Sharding is a data tier architecture in which data is horizontally partitioned across independent databases.	No additional cost if every shard has an Oracle Active DG, GG, or RAC license. Else, limited to 3 primary shards with basic DG, standbys.	Scalability and High Availability

10.1.1 Oracle Real Application Clusters

Oracle Real Application Clusters (RAC) is a cluster database with a shared cache architecture that provides highly scalable and available database solutions for all business applications.

It was introduced on Oracle Database Version 9 (9i) several years ago. It is one of the key options for HA that installations can choose when running Oracle Database. RAC is a chargeable License Fee Based option of the Oracle Database Enterprise Edition.

Oracle RAC on Linux running on LinuxONE provides multiple Linux guest operating systems for a database application to help limit application downtime. Every quarter, Oracle provides patch set updates with important security and database fixes. Oracle RAC/RAC One Node application downtime can be minimized by patching one system at a time (rolling), while the other Linux guest systems continue to run applications.

10.1.2 Oracle RAC One Node

Oracle RAC One Node is a single instance of an Oracle RAC-enabled database that is running on one node in a cluster. It is an HA solution that runs a database on one node of a cluster at a lower price point than full Oracle RAC.

RAC One Node provides automatic failover capability to other nodes in the cluster. Typically, planned maintenance operations can be completed with minimal disruption to application users by relocating the application Virtual IP addresses (VIPs) while the instances are active and running.

Technologies, such as transparent application failover (TAF), can be used to seamlessly move an active running SQL statement from one Oracle RAC ONE node to another node without disruption to the user.

One of the benefits of Oracle RAC One Node is that applications that might not be RAC friendly are not as affected as a full multi-node RAC Cluster solution. For example, an insert intensive application that uses many non-cached sequences might perform better running on only one node versus being distributed across multiple nodes in a RAC cluster.

Another benefit of Oracle RAC One Node is that a RAC One Node mode can be converted to full RAC (and vice versa) by using the **`srvctl convert`** command.

10.1.3 Oracle Clusterware

Oracle Clusterware provides the foundation for running Oracle RAC/RAC One Node.

A cluster consists of multiple interconnected computers or servers that appear as though they are one server to users and applications. Oracle RAC uses Oracle Clusterware for the infrastructure to bind multiple servers so that they operate as a single system.

In the addition to be the integrated foundation for Oracle RAC, Oracle Clusterware 19c can be used to deliver HA for other applications. Also, Oracle Clusterware depends on interconnect technology for rapid dissemination of information between different Oracle instances.

10.1.4 Edition-based redefinition

Edition-based redefinition (EBR) is an Oracle Database feature that was originally introduced with Oracle Database 11gR2. It enables online applications upgrades with uninterrupted availability of the applications. When the installation of an upgrade is complete, the pre-upgrade application and the post-upgrade application can be used simultaneously. Therefore, a session can continue to use the pre-upgrade application until its user decides to end it and all new sessions can use the post-upgrade application. When there are no longer any sessions using the pre-upgrade application, it can be closed. EBR enables hot rollover from a pre-upgrade version to a post-upgrade version, with no downtime.

10.1.5 Online Reorganization and Redefinition

The Online Reorganization and Redefinition feature that is available in Oracle Database 19c provides the flexibility to modify a table's physical attributes and transform data and table structure, while allowing users full access to the database. This capability improves data availability, query performance, response time, and disk space utilization.

During an application upgrade, administrators can start the redefinition process and at intervals synchronize the interim table so that it includes the latest changes to the original table. The advantage is that the amount of time to complete the final redefinition step is reduced. Also, administrators can validate and use the data in the interim table before completing the redefinition process.

10.1.6 Oracle Flashback

Oracle Database Flashback Technologies are a set of data recovery solutions that allow reversing mistakes by selectively undoing the effects of a previous error. Flashback provides efficient and quick error correction. Flashback supports recovery at all levels, including the row, transaction, table, and entire database. It also provides a growing set of features to view and rewind data back and forth in time.

Oracle Flashback Database is related to data protection features that enable a user to rewind data back in time to correct any problems that are caused by logical data corruption or user errors within a designated time window. Oracle Flashback provides an efficient alternative to point-in-time recovery and does not require a backup of the database to be restored first.

Flashback Database and restore points are effective in traditional database recovery situations and also can be useful during database upgrades, application deployments, and testing scenarios when test databases must be quickly created and re-created. Flashback Database also provides an efficient alternative to rebuilding a failed primary database after a Data Guard failover.

Note: Oracle Flashback does not replace a regular database backup and recovery procedure.

10.1.7 Oracle Data Guard/Active Data Guard

Data Guard is an Oracle Database offering that provides the management, monitoring, and automation software to create and maintain one or more synchronized copies of a production database. It also provides HA for mission critical applications.

It is a HA and Disaster Recovery (DR) solution that provides fast automatic failover, in the case of database failures, node failures, corruption, and media failures. The standby databases can be used for read-only access, reporting purposes, and testing and development purposes. Data Guard is included with Oracle Database Enterprise Edition.

Active Data Guard is an optional license component for Oracle Database Enterprise Edition. Active Data Guard adds advanced capabilities to extend basic Data Guard functions by allowing databases to be opened at the DR site for read-only access use by applications.

Oracle Far Sync allows for zero data loss at any distance by first copying transaction logs to another DR site that is geographically close by using synchronous replication of the database to the nearby site. Then, this nearby site's data is replicated asynchronously to another standby DR site at a greater distance.

10.1.8 Application Continuity

Application Continuity (AC) protects applications from database session failures because of instance, server, storage, network, or any other related component, and even complete database failure. AC replays affected "in-flight" requests so that the failure appears to the application as a slightly delayed execution, which masks the outage to the user.

AC is a feature that is available with the Oracle RAC, Oracle RAC One Node, and Oracle Active Data Guard.

If an entire Oracle RAC cluster fails, which makes the database unavailable, AC replays the session, including the transaction, following an Oracle Active Data Guard failover. Use of AC with a standby database requires Data Guard Maximum Availability mode (zero data loss) and Data Guard Fast Start Failover (automatic database failover).

10.1.9 Oracle GoldenGate

Oracle GoldenGate is a software package for real-time data integration and replication in heterogeneous IT environments. It also enables HA solutions, real-time data integration, transactional change data capture, data replication, transformations, and verification between operational and analytical enterprise systems.

Active–active HA

In a GoldenGate active-active configuration, the source and destination databases are available for reading and writing. This configuration yields a distributed configuration in which any workload can be balanced across any participating database. This provides high availability and data protection if an individual site fails. It also provides an excellent way to perform zero downtime maintenance.

Oracle GoldenGate 19c is the newest version of this Oracle product. While maintaining excellent performance with simplified configuration and management, it features a tighter integration with Oracle Database, support for cloud environments, expanded heterogeneity, and enhanced security.

10.1.10 Global Data Services

Oracle GoldenGate and Oracle Active Data Guard allow for the distribution of application workloads with replicated databases. When applications are spread across multiple databases in multiple data centers, it can sometimes be challenging to efficiently optimize databases with the best performance based on server loads and network latencies.

Oracle Global Data Services (GDS) provides the following key capabilities for a set of replicated databases that are globally distributed or located within the same data center:

- Region-based workload routing
- Connect-time Load balancing
- Runtime load balancing advisory for Oracle-integrated clients
- Inter-database Service failover
- Replication lag-based workload routing for Active Data Guard
- Role-based global Services for Active Data Guard
- Centralized workload management framework

Oracle GDS/Global Service Manager (GSM) is available for Linux running on LinuxONE, starting with Oracle 12c. Oracle GDS is included with an Active Data Guard or a Golden Gate replication license.

10.1.11 Storage-based replication

Oracle databases typically use a principle of dependent-write I/O through a series of synchronized writes to the data files and logs. No dependent-write is issued until the predecessor write completes. This feature allows those systems to be restorable if a power failure occurs.

For an Oracle consistent backup/DR solution in “hot” or online mode, the database must be synchronized with the online database logs. This configuration is typically done by using the **`alter database begin`** and **`alter database end backup`** commands. When these Oracle backup commands are issued, data blocks for the table spaces are flushed to disk and the data file headers are updated with the last checkpoint System Change Number (SCN).

Updates of the checkpoint SCN to the data file headers are not performed while in this backup mode. Oracle does not update these data file headers while in backup mode. When these files are copied, the non-updated SCN signifies to the database that recovery is required.

For a storage-based “consistent” Oracle replication strategy, co-ordination between the storage-based disk backup solution and the Oracle Database is needed.

Oracle Data Guard is the only certified and supported way (by Oracle) to protect Oracle Database files from Database block corruption. If an Oracle data block is corrupted on disk, a storage-based replication can replicate the corrupted blocks to the DR site.

Disk-based storage replication is supported for replicating Linux OS file systems, Oracle binaries, and any non-database file systems that are required to be kept in sync with the database.

10.1.12 Oracle Site Guard

Oracle Site Guard is a DR solution that enables administrators to automate complete site switchover or failover. It orchestrates the coordinated failover of Oracle Data Guard with Application Servers, which can be integrated with a storage-based replication technique.

Oracle Site Guard integrates with underlying replication mechanisms that synchronize primary and standby environments and protect mission critical data. It includes built-in support for Oracle Data Guard for Oracle Database. Oracle Site Guard supports typical storage replication solutions that are used with LinuxONE, but requires some scripting to integrate with a SAN storage-based replication solution.

10.1.13 Oracle Recovery Manager

Oracle Recovery Manager (RMAN) provides a comprehensive foundation for efficiently backing up and recovering the Oracle Database. Providing block-level corruption detection during backup and restore, RMAN optimizes performance and space consumption during backup with file multiplexing and backup set compression. It works with Oracle Secure Backup and third-party media management products for tape backup.

RMAN handles all underlying database procedures before and after backup or restore, which frees dependency on operating system and SQL*Plus scripts.

10.1.14 Oracle Secure Backup

Oracle Secure Backup (OSB) delivers centralized disk and backup management for the entire IT environment. It consists of the following offerings, both of which are integrated with Oracle RMAN:

- Centralized backup management and high-performance and heterogeneous data protection in distributed UNIX, Linux, Windows, and network-attached storage (NAS) environments.
- An OSB loud Module that provides integrated Oracle Database backup to Amazon Cloud (Internet) storage.

10.1.15 Oracle Multitenant

Oracle Database 19c offers a new chargeable License Fee Based option that is named Oracle Multitenant. This option provides simplified consolidation that requires no changes to your applications.

In this new architecture, a multitenant container database can hold many pluggable databases. An administrator manages the multitenant container database, but application code connects to one pluggable database.

Oracle announced that from 19c, three users can be defined as PDBs, without licensing multitenant option. For more information, see Table 1-2, “Consolidation”, of the *Oracle Database Licensing Information User Manual*.

Oracle Multitenant consists of a container database (CDB) that features all of the metadata for the Oracle system. No application data schemas are configured in the CDB.

The pluggable database (PDB) contains the application-specific database and can be unplugged and plugged into different CDB databases.

Oracle Multitenant allows the management of many databases as one and can provide several benefits from an administration perspective. For example, in a multiple PDB configuration, many pluggable databases can be backed up with one Oracle RMAN configuration. This feature reduces the time that is necessary to back up many databases if they were independent.

Oracle Multitenant in a HA scenario allows for the ability to unplug and move a pluggable database from one container database to another. This feature is useful in a HA environment where a pluggable database is moved to another CDB running on another guest.

In the case of an Oracle upgrade, the database upgrade downtime is reduced as the schema updates are mostly made to the system table spaces, which are a component of the CDB and not to the metadata components of the PDB.

You can upgrade PDBs by unplugging a PDB from an earlier release CDB, plugging it into a later release CDB, and then, upgrading that PDB to the later release.

For more information about the new functions of the Oracle 19c Multitenant options, see this Oracle web page.

10.2 HA building blocks for Oracle Database on Linux running on LinuxONE

This section describes how Oracle HA options interact with Linux running on LinuxONE. The many Oracle HA options are an ideal complement for the superb HA features that are provided by LinuxONE.

Figure 10-1 shows a complete HA scenario that consists of the following major components:

- Oracle Database and its HA options
- Linux operating system
- LinuxONE system

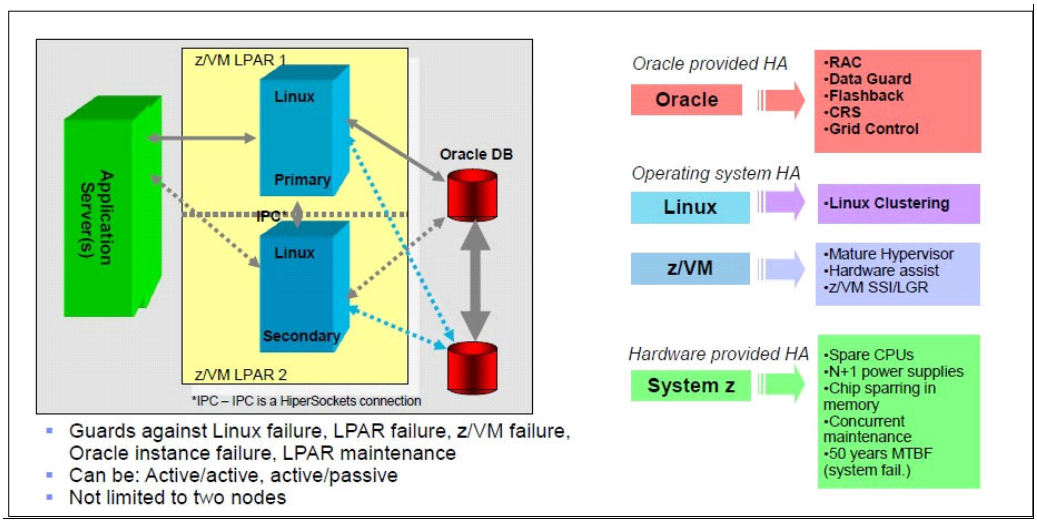

Figure 10-1 Building blocks of an HA environment for Oracle Database on Linux running on LinuxONE

10.3 Hardware-provided HA

IBM LinuxONE has high reliability through various redundant hardware options, which provides a mean time between failure of over 50 years.

According to the *ITIC 2020 Global Server Hardware OS Reliability Report*[1]:

> IBM Z mainframe class servers running Z/OS or Linux, once again had the distinction of being ranked best in class for reliability, accessibility, performance, and security among all server platforms. The Z mainframes had highest reliability/uptime ratings across the board in terms of actual minutes of unplanned per server/per annum downtime. An 86% majority of IBM Z mainframe users achieved at least five nines or 99.999% reliability and of that percentage. Specifically, IBM Z servers exhibit true fault tolerance experiencing just 0.62 - less than one minute of unplanned per server, per annum annual downtime due to inherent problems with the server or its component parts.

For example, IBM LinuxONE includes transparent CPU sparing; that is, if a CPU encounters a hardware failure, a spare CPU can be quickly brought online without affecting the applications or database.

IBM LinuxONE has Redundant Array of Independent Memory (RAIM) memory modules. RAIM memory prevents a memory module from affecting the availability of a system or application.

[1] https://itic-corp.com/blog/2020/05/itic-2020-reliability-poll-ibm-lenovo-hpe-huawei-mission-critical-servers-deliver-highest-uptime-availiability/

In addition, IBM LinuxONE includes numerous other RAS features. The following hardware-provided HA features are a standard offering:

- Concurrent maintenance: At a minimum of two processor books, hardware parts of the processor can be replaced without a Power on Reset with the normal continuation of operations.
- Chip Sparring in Memory: An error detection and correction mechanism on IBM LinuxONE that allows error detection during instruction execution and transparent error correction of spare processors, which were configured to the system.
- N+1 Processor supplies: The ability to have two sets of redundant power supplies. Each set of the power supplies has its individual power cords or pair of power cords, depending on the number of Bulk Power Regulator (BPR) pairs installed.

10.3.1 Operating system HA

Several operating systems can run on an IBM LinuxONE. In this section, we describe those systems that are relevant for an environment in which Oracle Databases are deployed.

Linux clustering

Red Hat Enterprise Linux Servers and SUSE Enterprise Linux Server are the two Linux distributions that are certified by Oracle to run its database on IBM LinuxONE. Oracle Clusterware works by moving application VIPs between Linux guest nodes. Several Linux operating system-based solutions also can be used to provide HA for applications.

SUSE Linux Enterprise High Availability Extension is included with every subscription. This extension offers you a clustered file system for LinuxONE. It also delivers all of the essential monitoring, messaging and cluster resource management features in an integrated suite of open source technologies.

RHEL High Availability is a collection of software that is deployed on top of the Red Hat Enterprise Linux operating system, policies, requirements, and limitations that are applicable to RHEL and RHEL High Availability.

Red Hat supports RHEL High Availability clusters that contain 2 - 4 nodes running as IBM LinuxONE members.

These Linux-based HA solutions work in a similar manner, whereby Oracle Database is configured to run on a Linux guest. If planned maintenance or an unplanned system event occurs, the HA solution fails over to another idle Linux guest.

The Linux HA solution shuts down the source system database and unmounts the Oracle file systems on the source system. Next, the Linux HA solution mounts the necessary Oracle file systems, starts the application VIPs on the server, and then, starts the Oracle listener on this VIP IP address on the failover server. The last step is to start the database on the failover Linux guest such that the applications can now connect to the failover server.

Depending on activity load in the database and other factors, a failover can be as quick as a few seconds to the new server, or up to several minutes if many insert or update operations require rolling back.

Hypervisor (z/VM)

IBM z/VM offers a base to use IBM virtualization technology on IBM LinuxONE. Fully tested and deployed in Oracle development environments, Oracle products are certified with z/VM.

IBM z/VM provides a highly secure and scalable enterprise cloud infrastructure. It also provides an environment for efficiently running multiple diverse critical applications on IBM LinuxONE with support for more virtual servers than any other platform in a single footprint.

IBM z/VM virtualization technology is designed to allow for the capability to run hundreds to thousands of Linux servers on a single LinuxONE footprint.

IBM for z/VM is a virtualization management solution that helps dramatically simplify the administration and management of z/VM and virtual Linux servers that are running on IBM LinuxONE. IBM Wave integrates seamlessly with z/VM and enterprise Linux environments to help administrators view, organize, automate and manage resources in an optimized and standardized manner.

IBM Wave provides easy provisioning and management capabilities of Oracle Linux Virtual Machines in a Cloud implementation on LinuxONE.

For more information about IBM Wave, see IBM Wave for z/VM.

10.3.2 Oracle-provided HA

Oracle Clusterware provides HA for an application by transferring the applications connectivity information with VIP. When a node goes down (planned or unplanned), the application connectivity information or VIP is moved to another node in the cluster to accept the application's connections.

When the VIP is moved from one Linux host to another, Oracle broadcasts the new IP to the network and the newly associated network media access control or MAC address with an ARP command.

Applications can seamlessly route database connections to the new server without incurring downtime by connecting to the highly available VIP, as opposed to a hard-coded Linux server IP address.

Oracle Clusterware interfaces directly with the hardware technology that is available on LinuxONE.

10.4 IBM LinuxONE with Oracle Database: Considerations and examples for HA

In this section, we provide some considerations and examples of HA.

10.4.1 Networking

IBM LinuxONE has several options for designing the network between Logical Partitions (LPARs) and other servers on a network.

An Open System Adapter (OSA) is a physical network card in LinuxONE. An OSA can be dedicated to an Oracle Linux guest, which is shared among multiple Linux guests, or configured in a Virtual Switch (VSWITCH) configuration. Virtual Switches are beneficial for systems with many Linux guests sharing network infrastructure.

IBM HiperSockets is a technology that provides high-speed TCP/IP connectivity within a central processor complex. It eliminates the need for any physical cabling or external networking connection between servers running in different LPARs.

The communication is through the system memory of the processor; therefore, servers are connected to form an “internal LAN.”

IBM HiperSockets are advantageous for workloads in which a large amount of data must be transferred quickly between LPARs if enough CPU capacity exists to support the network transfers.

Figure 10-2 shows some of these network concepts to consider in a network design for development, test, and production environments.

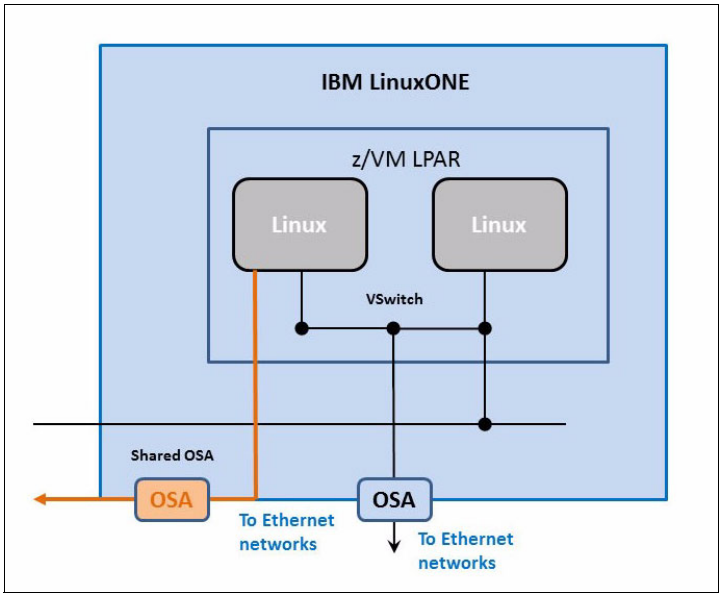

Figure 10-2 IBM LinuxONE servers network considerations

10.4.2 Oracle HA networking options

IBM LinuxONE includes the following options for supporting the Oracle Interconnect between Oracle nodes in an Oracle RAC or Oracle RAC OneNode cluster configuration:

- Link Aggregation (active/active): Allows up to eight OSA-Express adapters to be aggregated by using a VSWITCH type of configuration.
- Linux Bonding: Whereby Linux provides the HA by using two or more Linux interfaces in an active/backup or active/active configuration; for example, Linux interfaces `eth1` and `eth2` are configured to create a highly available bonded interface that is named `bond0`.
- Oracle HA IP (HAIP): With Oracle 11gR2+, you can now have up to four private interconnect interfaces in which Oracle's HAIP function is used to balance network load across the Oracle RAC/RACONE interconnect network interfaces.

10.4.3 Oracle RAC Interconnect z/VM Link Aggregation

Figure 10-3 shows a typical IBM z Server topology, with multiple Oracle RAC clusters, that uses multiple Virtual Switches with Link Aggregation.

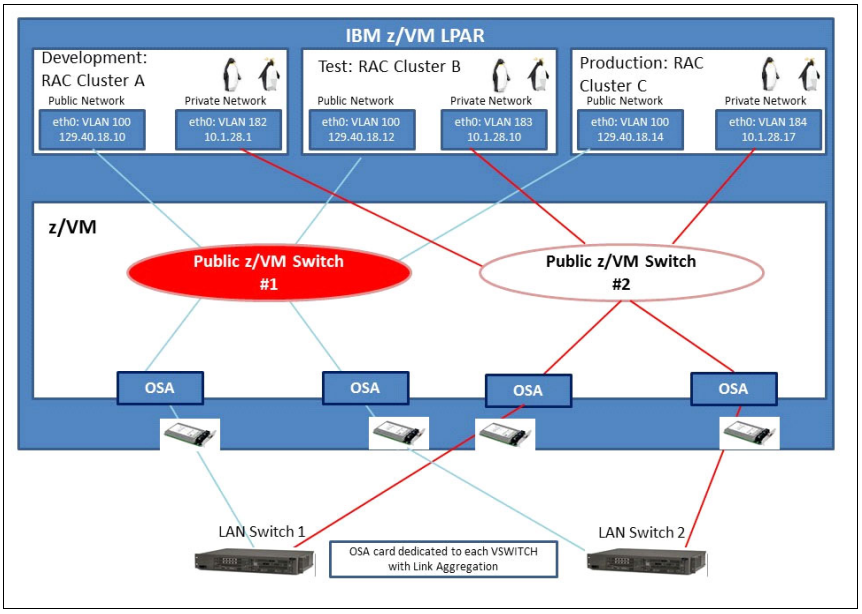

Figure 10-3 Oracle Interconnect with z/VM Link Aggregation

10.4.4 Oracle RAC interconnect considerations

The design of the Oracle RAC/OneNode interconnect is vital to the availability of an Oracle cluster. Oracle requires that the Cluster interconnect is configured on a private, dedicated LAN or VLAN (tagged or untagged), which is nonroutable and isolated from other non-interconnect traffic.

The benefit of a Virtual Switch approach is that multiple RAC clusters can share the network infrastructure. It is strongly recommended to have a minimum of two Virtual Switches for an Oracle RAC configuration (public and private).

IBM LinuxONE environments that run in two separate LPARs can use multiple HiperSocket interfaces, which can be load balanced with Oracle's HAIP load balancing capability.

For environments that are CPU bound or cluster interconnect sensitive, the dedicated OSA cards approach (outside of a Virtual Switch) works well to ensure that the cluster interconnect continues to operate efficiently during periods of high CPU load.

10.4.5 I/O channel failover considerations

IBM LinuxONE supports multiple logical channel subsystem (LCSS) images that are mapped onto a physical channel subsystem (see Figure 10-4).

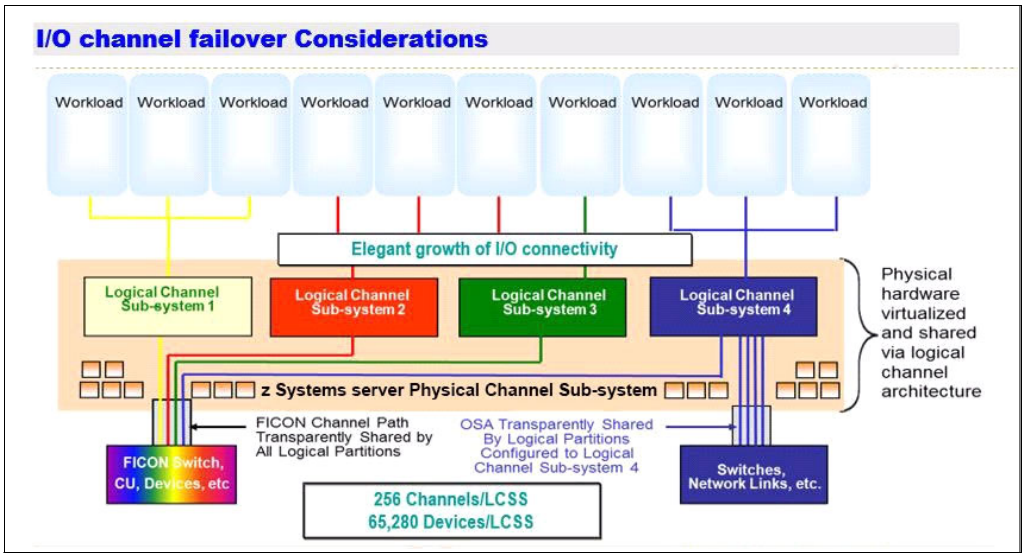

Figure 10-4 Example of an I/O channel failover architecture

Consider the following points:

- The physical hardware, such as the IBM FICON® channels and Open System Adapter (OSA) cards, are shared by using the logical channel architecture.
- The processors that run the channel subsystem are called the System Assist Processors (SAP). More than one SAP can be running the channel subsystem. The SAP drives the LinuxONE Server's I/O channel subsystem, which serves a collection of more than 1,000 high-speed buses.

  The SAP relieves the operating system and the general-purpose CPs of much of the work that is required to run I/O operations.

  Specifically, the SAP schedules and starts each I/O request; that is, it locates an available channel path to the requested I/O device and starts the I/O operation. The SAP does not handle the data movement between central storage (CS) and the channel.
- Channels are the communication path from the CSS to the control units and I/O devices. They are represented by the black rectangles in Figure 10-4; one for IBM FICON, one for OSA.
- Channel Path Identifier (CHPID) is a value that is assigned to each channel path that uniquely identifies that path. A maximum of 256 CHPIDs can be used in an LCSS.

- Within the physical I/O subsystem, channel paths and I/O adapters (for example, Open System Adapter Ethernet cards) can be shared, potentially by all logical subsystems. As shown in Figure 10-4 on page 159, the FICON Channel is being shared by all 4 LCSS, and thus all partitions.
- If necessary (for example, for performance reasons), a path or an adapter can be restricted to one LCSS or a subset of the LCSSs. A path or an adapter can be restricted to one LCSS or a subset of the LCSSs (as shown with the OSA on the right in Figure 10-4 on page 159), which can be used only by LCSS 4 or the blue LPARs that are shown at the top of Figure 10-4 on page 159.

More information

In this section, we provide more information about the LCSS, its maximums in terms of LPARs, channels, and physical devices it supports, and why we must create the LCSS concept to map to the physical channel subsystem.

IBM LinuxONE has a unique channel architecture that is designed to provide powerful and flexible support for the most demanding I/O performance and high-volume workloads. This channel architecture is managed through the foundation technology called LCSS.

Each IBM LinuxONE server can have up to four of these LCSSs and each can support 15 LPARs. Each LPAR can address 256 data channels and 64,000 I/O devices. Therefore, a single LinuxONE system can handle over 1,000 data channels and a quarter of million I/O devices.

The LCSS's architecture is further enhanced by the LinuxONE Multiple Image Facility. This Multiple Image Facility allows all 15 LPARs that share a common LCSS to directly access each I/O device without having to forward the request through an intermediate partition, as is the case with UNIX architectures. The direct result of a shorter path length is improved I/O performance.

LinuxONE Channel Spanning allows each device (disk or tape units that are attached by using Fibre Channel technology) to appear to be on any LCSS. The result is that it can be accessed by any partition on IBM LinuxONE. Therefore, greater I/O flexibility and simplified operational management can be achieved.

LinuxONE III supports the following components:

- Up to six LCSSs
- Up to 15 LPARs per LCSS
- Up to 256 Channels per LCSS
- Up to 256 CHPID per LCSS
- Up to 64 K Physical Devices per LCSS
- Up to 1536 CHPIDs made available for entire system
- Up to 1536 Physical Channels for entire system

Often in computing, virtualization allows more logical resources than physical. In this case, the physical resources exceed the logical channel addressing ability of the LCSS.

The architecturally defined channel-path identification number, called the CHPID, must be maintained without change. The CHPID value is defined as an 8-bit binary number that results in a range of unique CHPID values 0 - 255.

Since the inception of the precursor S/370 XA channel-subsystem architecture in the late 1970s, this 8-bit CHPID was maintained without change because of its pervasive use in the IBM z/VM operating system.

For example, the CHPID value is maintained in many internal programming control blocks, is displayed in various operator messages, and is the object of various system commands, programming interfaces, and so on, all of which must be redesigned if the CHPID value was increased to more than an 8-bit number to accommodate more than 256 channel paths.

Therefore, another level of channel-path-addressing indirection was created because of the I/O subsystem that allows more than 256 physical channel paths to be installed and uniquely identified without changing the former 8-bit CHPID value and the corresponding programming dependencies on the CHPID.

The new channel-path-identification value, called the *physical-channel identifier* (PCHID), is a 16-bit binary number 0 - 65,279, which uniquely identifies each physically installed channel path.

With the current IBM LinuxONE servers, a maximum of 1024 external channel paths out of the 65 K (for example, ESCON, FICON, OSA) and 48 internal channel paths (for example, Internal Coupling and IQDIO hyperlink) are each assigned a unique PCHID value of 0 - 2,047.

10.4.6 Government client example: Oracle MAA RAC and Data Guard

Oracle's MAA best practices also can be used to provide an Oracle-centric HA and DR solution with IBM LinuxONE. In this scenario, the Oracle Database, RAC, and Data Guard are deployed on IBM LinuxONE (for the database portion of the deployment).

Oracle MAA (see Figure 10-5) provides a comprehensive architecture for reducing time for scheduled and unscheduled outages. An Oracle MAA solution consists of two identical sites: the primary site contains the RAC database, the secondary site contains Physical Standby database or both Physical and Logical one on RAC. Active Data Guard is used for online reporting (extra licensed) at the DR site. Data Guard switchover and failover functions allow the roles to be traded between sites.

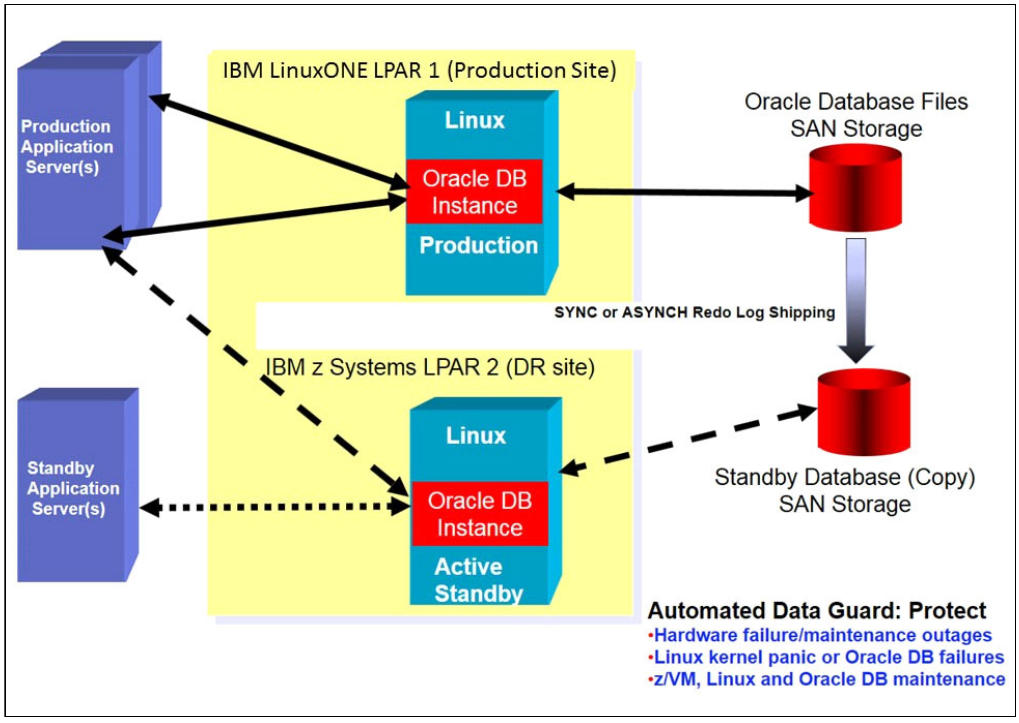

Figure 10-5 Public sector client MAA example

10.5 Oracle Maximum Availability architecture

In this section, we provide some considerations and examples of Oracle MAA architecture on LinuxONE.

10.5.1 Oracle MAA concept and architecture blueprints definitions

Oracle MAA is a set of best practices blueprints for the integrated use of Oracle High Availability (HA) technologies by using combination of Oracle MAA features that are listed in 10.1, “Oracle HA options” on page 146.

Blueprints architectures help to define different levels of service Level Agreements (SLAs) for the Oracle Databases.

MAA blueprints, split in four tiers (Bronze, Silver, Gold, and Platinum), consider and discuss the various failure scenarios that can affect any database or application:

- Bronze Reference Architecture is suitable for databases in which simple restart or restore from backup is “HA enough”.
- Silver Reference Architecture is designed for databases that cannot afford to wait for a cold restart or a restore from backup if an unrecoverable database outage occurs.
- Gold Reference Architecture is well suited for service level requirements that cannot tolerate site outages.
- Platinum Reference Architecture is designed to make unplanned outages and planned maintenance transparent to the user.

Figure 10-6 shows the architecture tiers.

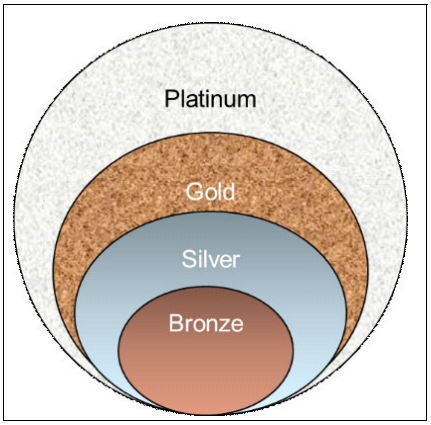

Figure 10-6 Oracle MAA architecture tiers

Architecture tiers overview

In this section, we describe the features of each Oracle MAA architecture tier.

Bronze Tier

This tier features the basic features of a single instance HA environment. Non-critical DBs (development, test) can run on Single Instance, rather than on a cluster.

Acceptable timeframe of recovery for an unplanned data corruption event: A few hours or an hour of downtime might be acceptable in cases where a human mistake occurs that corrupts the data.

Uses the Oracle stack, which is composed of the following components:

- Oracle RMAN
- Oracle OSB
- Oracle Database backup cloud service
- Real-time data protection; zero data loss recovery appliance (ZDLRA)
- Oracle flashback technology to recover from logical corruption
- Online data reorganization and redefinition

Silver Tier

This tier includes the features Active/Active database clustering.

More critical to the overall business applications: need for a high availability solution that provides near instantaneous recovery from unexpected outages is necessary. It expands on the Bronze MAA tier capabilities by adding active/active clustering, automatic storage management (ASM), and Application Continuity.

Uses the Oracle Bronze stack, plus the following component:

- RAC)
- Transparent Automatic Failover (TAF)
- ASM

Gold Tier

This tier includes physical replication, zero data loss, and fast failovers. It uses the Oracle Silver Tier stack, plus Oracle Active Data Guard.

Platinum Tier

This tier features the highest uptime for all outages, and zero data loss.

It uses Oracle's top level of HA features and includes the Oracle Gold Tier stack, plus Active-active HA: GOLDENGATE.

10.5.2 Oracle MAA architecture blueprints on LinuxONE

In this section, we describe the Oracle MAA architecture blueprints that are available on LinuxONE.

Bronze Tier

This tier features the basic features of a single instance HA environment.

Oracle MAA Bronze expects a Single Instance Database Server and two RMAN backups, one local and one mirrored remotely. It features the following architecture:

- No HA: Data is back-up locally and safely mirrored remotely.
- DR: Remote backup of data.

Benefits:

- Cheapest solution architecture, but even protected against data loss
- Security and resiliency with IBM LinuxONE platform

Figure 10-7 shows the Bronze Tier on LinuxONE.

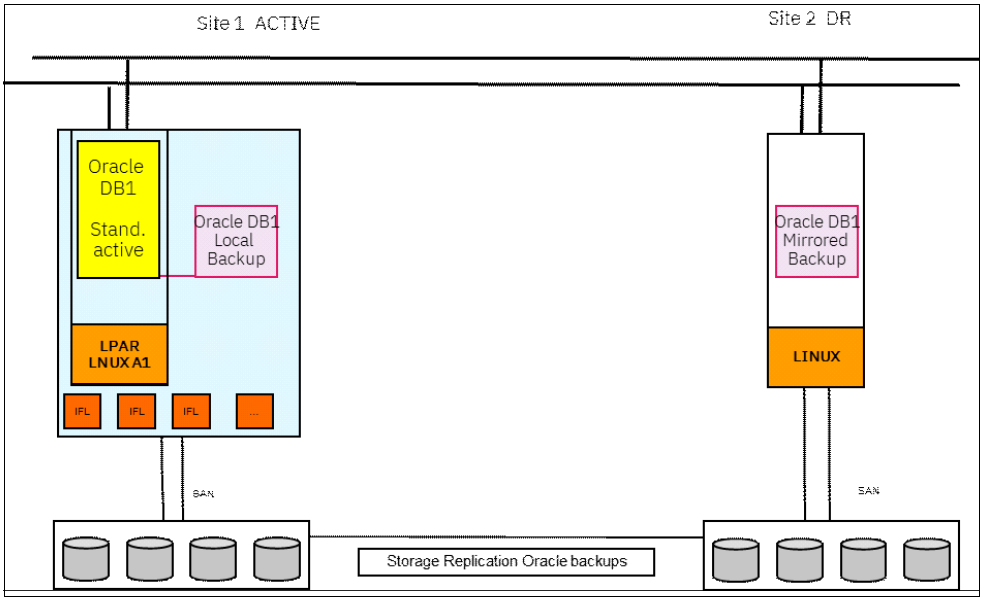

Figure 10-7 Bronze Tier on LinuxONE

Silver Tier

This tier features active/active database clustering. Oracle MAA Silver expects an active RAC across two different boxes and includes the following architecture:

- HA: Oracle RAC ACTIVE in same PROD Data Center. Data is back-up locally and safely mirrored remotely.
- DR: Remote backup of data

Benefits

- HA of Oracle DB Server by using Oracle RAC
- Security and resiliency with IBM LinuxONE platform

Figure 10-8 shows the Silver Tier on LinuxONE.

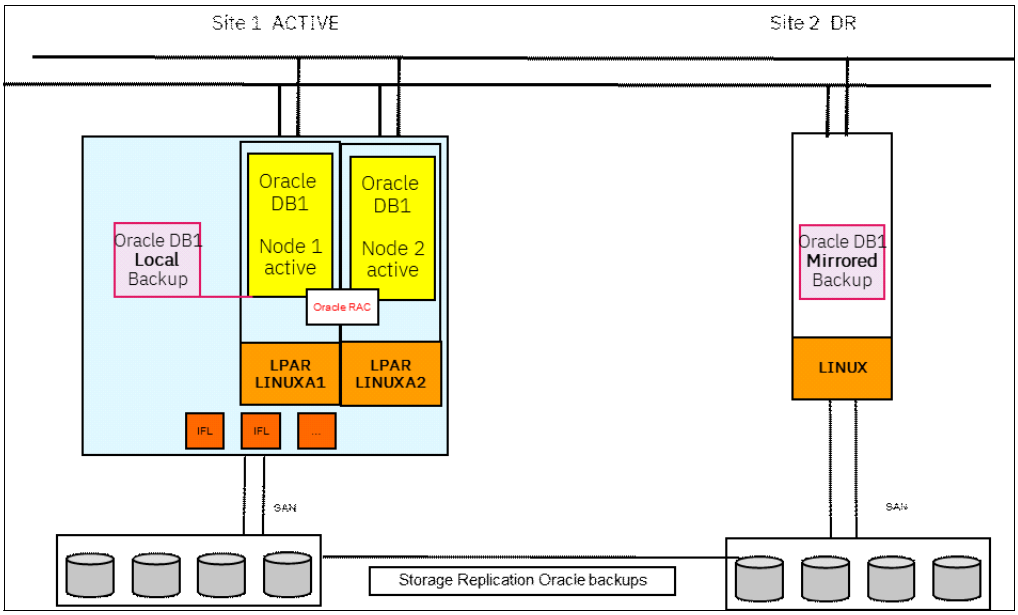

Figure 10-8 Silver Tier on LinuxONE

Gold Tier

This tier features physical replication, zero data loss, and fast failovers. Oracle MAA Gold expects an Active RAC across two different boxes and a Standby Data Guard DB across two different Data centers. It includes the following architecture:

- HA: Oracle RAC ACTIVE in same PROD data center.
- DR: Oracle STANDBY DB in the DR data center. Oracle Active Data Guard can be used. Far sync is required for long distances. This feature requires more licences. Data is back-up locally and remotely.

Benefits:

- High level of HA/DR with Oracle RAC and Data Guard.
- Data Guard used to quickly failover if a disaster occurs.
- Security and resiliency with IBM LinuxONE platform

Figure 10-9 shows the Gold Tier on LinuxONE.

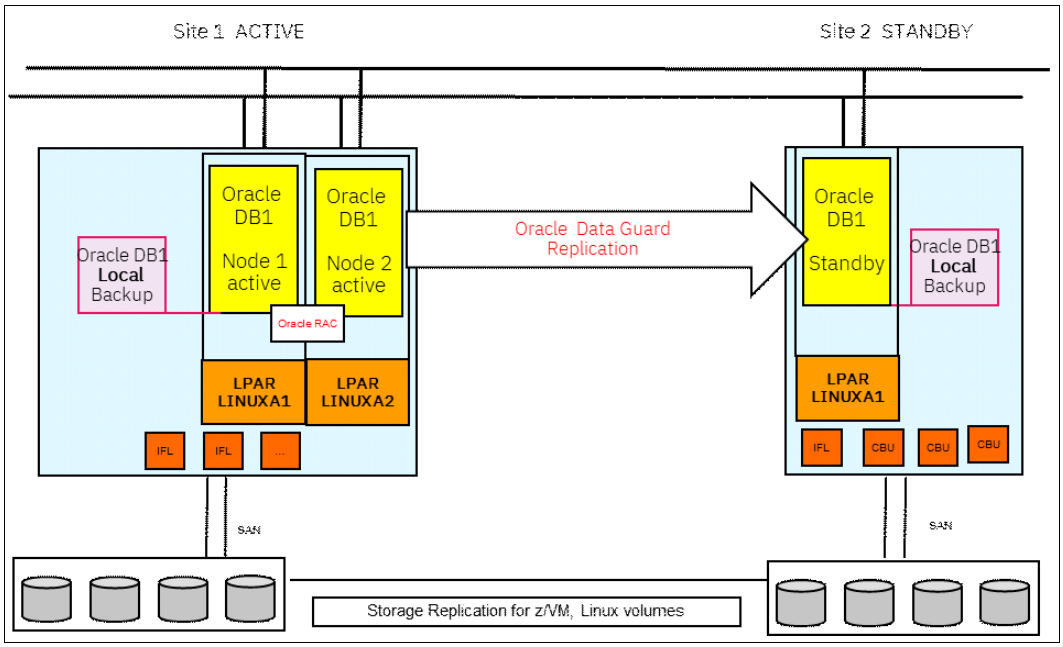

Figure 10-9 Gold Tier on LinuxONE

Platinum Tier

This tier features the highest uptime for all outages, and zero data loss. Oracle MAA Platinum expects an Active RAC across two different boxes, another RAC replicated in the same PRODUCTION center, and a standby Data Guard DB across two different data centers.

This tier features the following architecture:

- HA: Oracle RAC ACTIVE in same PROD data center. Replication of this RAC is in same data center.
- DR: Oracle STANDBY DB in the DR data center. Oracle Active Data Guard can be used. Far sync is required for long distances. This feature requires more licences. Data is back-up locally and remotely

Benefits:

- Highest level of availability and resiliency with LinuxONE, Oracle RAC, and Golden Gate Replication.
- Reduces the RTO time to zero for database upgrades, patch sets, and even application upgrades.
- Security and resiliency with LinuxONE platform.

Figure 10-10 on page 167 shows the Platinum Tier on LinuxONE.

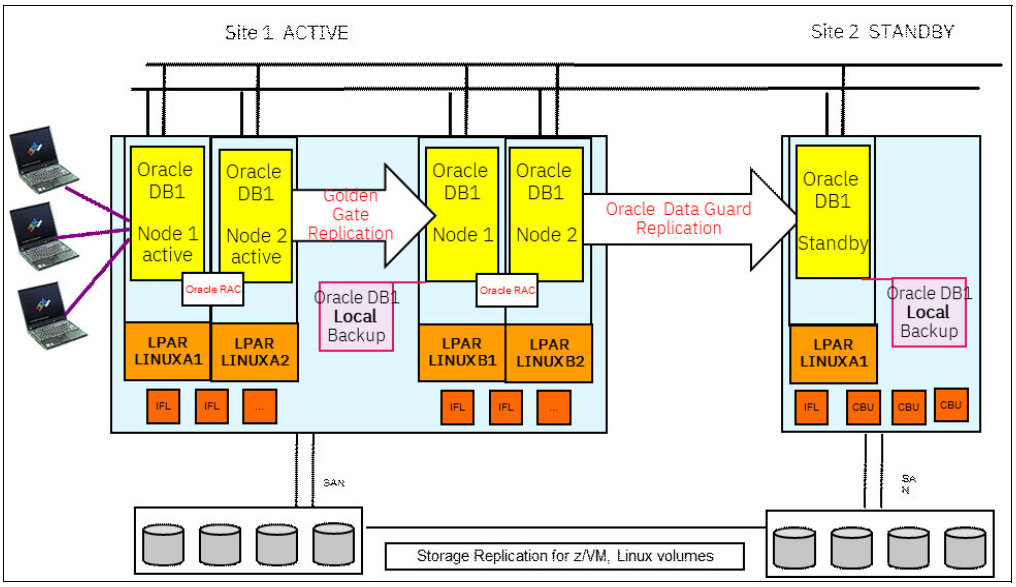

Figure 10-10 Platinum Tier on LinuxONE

10.6 Summary and recommendations

In this chapter, we described the main Oracle Database options for HA. We also described a few of the HA options for IBM LinuxONE.

IBM LinuxONE servers feature the highest availability rating of any commercially available server on the market today. LinuxONE's ability to perform dynamic reconfigurations to spare out another processor, use RAIM Memory, and use another I/O/network path is true HA.

LinuxONE's virtualization technology has been in the marketplace for over 50 years. It was deployed in Oracle development since the first 10gR2 database was certified on the platform. Decades of experience in error correction and testing methodologies help ensure that the hardware is highly available and that the software also is designed that way.

Combining the highest available server infrastructure of LinuxONE with Oracle's maximum availability architecture creates the most powerful and robust combination that is available for running business-critical Oracle workloads.

IBM LinuxONE also is a good fit for Oracle HA because ofLinuxONE's Dynamic capabilities. More memory, CPU, I/O, or network resources can be dynamically enabled and added to a system microcode without the need to shutdown an application.

LinuxONE with Oracle's HA can protect an application and a business from unplanned events, which provides excellent business value through Reliability, Availability, Scalability, and Oracle's HA options.

The LinuxONE's hardware stack is fully certified with Oracle for the entire Oracle Database Enterprise Edition solution stack to run virtualized under z/VM.

Oracle continues to demonstrate the delivery of their quarterly patch set updates and critical patches for Red Hat and SUSE Linux distributions for LinuxONE currently with other platforms. This commitment to the currency of critical patches helps add to HA capabilities of running the Oracle Database with IBM LinuxONE.

Customers who deployed Oracle solutions with IBM LinuxONE enjoy lower total cost of ownership over the entire product lifecycle. They also achieve near immediate return on investment by using its crucial embedded security, near flawless system availability, and reliability.

When it comes to HA, LinuxONE delivers unmatched performance, scalability, and mission-critical reliability in physical, virtual, and cloud environments.

11

Performance tuning overview for IBM z/VM, Linux, Oracle on LinuxONE, and IBM Z

For many years now, server consolidation, virtualization, and workloads as services in the cloud environment were the driving factors for the IT infrastructure industry.
IBM LinuxONE is a platform that is designed for the consolidation of discrete servers that are running business application workloads using Oracle Databases.

It is designed for virtualizing physical hardware, such as processing units, memory, network and I/O devices, and facilitating Oracle databases as services in the cloud environment. Virtualized environments are created in LinuxONE or IBM Z by using hardware hypervisor and software hypervisor.

The physical machine can be divided into logical partitions (LPARs) and each LPAR can run one to thousands of Linux guests to host Oracle databases. Each Linux guest also can run one or many Oracle databases on them. Applications (such as Oracle) can be deployed to use simultaneous multithreading (SMT) to increase the processing efficiency and throughput. Single-instruction, multiple-data (SIMD) elevates performance enhancements in the analytics area.

These levels of sophistication and flexibility to run Oracle databases on LinuxONE and IBM Z, along with the ability for the systems administrator to overcommit certain resources (such as CPU and memory), necessitates the need to closely monitor the IBM System server for optimized usage and efficiency. Performance can be monitored on LinuxONE and IBM Z at various levels, such as the hypervisor, hosting operating system, middleware, and workload levels.

In this chapter, we provide some of the more widely used tools that are used to monitor the system, the performance of LinuxONE, and IBM Z servers running Oracle databases.

This chapter includes the following topics:

11.1 Linux monitoring tools

In a LinuxONE or Linux on System z server that is running Oracle databases, many tools are proposed from the Linux community.

The following Linux tools are available for monitoring guests that are running (this list is ever growing):

- pidstat: Per process statistics
- slabtop: Kernel memory pool consumption
- lsof: Check file flags of open files
- blktrace: Low-level disk I/O analysis
- Hyptop: Cross guest CPU consumption monitor
- Iptraf: Network traffic monitor
- Dstat: Configurable live system overview
- Irqstats: Check IRQ amount and CPU distribution
- smem: Per process/per mapping memory overview
- Jinsight: Java method call stack analysis
- Htop: T op on steroids
- Strace: System call statistics
- Ltrace: Library call statistics
- Kernel tracepoints: Get in-depth timing inside the kernel
- Vmstat: Virtual memory statistics
- Sysstat: Full system overview
- Iostat: I/O-related statistics
- Dasdstat: Disk statistics
- scsi statistics: Disk statistics
- Perf: Hardware counters, tracepoint-based evaluations, profiling to find hotspots
- Valgrind: In-depth memory/cache analysis and leak detection
- Java Health Center: High-level java overview and monitoring
- Java Garbage Collection and Memory visualizer: In-depth GC analysis
- Netstat: Network statistics and overview
- Socket Statistics: Extended socket statistics
- top/ps: Process overview
- Icastats/lszcrypt: Check usage of crypto-hardware support
- Lsluns/multipath: Check multipath setup
- Lsqeth: Check hardware, checksumming and buffer count
- Ethtool: Check offloading functions
- Collectl: Full system monitoring
- Ftrace: Kernel function tracing
- TTng: Open source tracing framework for Linux
- Ziomon: Analyze FCP setup and I/O
- Systemtap: Another kernel tracing infrastructure
- Wireshark/Tcpdump: Analyze network traffic in depth
- Iotop: Order processes by disk I/O
- Iftop: Per connection traffic overview

For more information about how to use the tools and commands, see *Linux on IBM System z: Performance Measurement and Tuning*, SG24-6926.

Multiple resources are available on the web for Linux system monitoring tools.

11.1.1 CPU performance

For CPU performance, consider the following points:

- Ensure that the recommended Linux Kernel Parameters for Oracle are set on the Linux guest.
- Ensure that Simultaneous Multithreading (SMT) is activated at the LPAR level to increase the processing efficiency and throughput.
- Use IBM Think® to configure Oracle threads parallelism: the default value is:

  `PARALLEL_MAX_SERVERS = (CPU_COUNT x PARALLEL_THREADS_PER_CPU x 10)`

Note: If too many query server processes are running, memory contention (paging), I/O contention, or excessive context switching can occur because contention can reduce system throughput to a level lower than if parallel execution were not used.

Also, you can use Oracle Consumer Group to limit processes for certain types of users/jobs.

Starting a performance determination for CPU

If CPU bottleneck occurs, use the **vmstat** command to check whether the percentage idle is 0 (including I/O wait of 0) *and* a run queue greater than the number of virtual processors in the Linux guest. Then, you might be out of CPU on that specific Linux guest at that time. Both conditions should be met.

Run the **vmstat** command to identify the run queue, process wait, and memory swap, as shown in Figure 11-1.

 # vmstat 3 (on virtual 2 way machine)

```
procs -----------memory---------- ---swap-- -----io---- -system-- -----cpu------
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 4  0 276900 286468   1164 468472    0    0     5    26    7    8  0  0 100 0  0  <- 1st Value is avg.
 1  0 276896 284772   1256 468900    0    0   267    76  257  760 43  7 49  1  0    from last startup
 2  0 276888 272052   1392 470320    0    0   475   107  218  439 47  4 47  1  2    i.e. Typically ignore
 3  0 275672   8988   1228 464564  277 42971 1224 47888 1332  350 67 11  0 15  6
 2  0 273636   8884    652 489576  524    3   889 20575  397  321 59  4 37  0  1
 1  0 271560   8580    788 536964  599    5   984 29069  470  255 61  3 34  1  1
 1  0 267576   8732   1068 591056 1412    0  3772 31208  796  696 50 11 22 16  1
 6  5 283124   6168    240 586176  299 5451  2148 17865 1220  528 15 24  6 53  1
 0  8 307192   5840    432 614808  437 8451 12868 26735 1249  575 14 21  2 59  4
16 12 307192   6668    136 572948    3   17 46792   701 1744  963  0 87  0 13  1
15 15 307192   7796    120 570384    0    0 13271     0  393  188  0 99  0  0  1
```

- r –run queue –how many processes currently waiting for CPU
- b – how many processes waiting in uninterruptible sleep
- Steal time (**st**) is the percentage of time a virtual CPU waits for a real CPU while the hypervisor is servicing another virtual processor.

- When sustained load average/run queue is > 4x number of virtuals with no cpu idle then this may be an indication of lack of cpu.
- In this example Linux swap (lack of Linux guest memory) seems to be the main culprit.

Figure 11-1 Using the vmstat command

As shown in Figure 11-1, the `r` and `b` columns show that contentions exist around the CPU. Also, the `si` and `so` columns show swap activity.

Work with your system administrator to increase Linux guest or LPAR memory to avoid swapping. Then, continue to monitor the run queue and processes wait.

Also, commands, such as **top**, **hyptop**, and **nmon** can be used to follow the CPU activities.

For more information, see the following resources:

- *SUSE Linux Enterprise Desktop 15 SP1: System Analysis and Tuning Guide*
- *Red Hat Enterprise Linux 7: Performance Tuning Guide*

11.1.2 Memory

It is recommended to use Linux Large Pages for Oracle Database Memory. The use of large (1 MB) pages is preferred and requires a specific system configuration.

In general, a 10 - 15% savings can be gained by the reduction in CPU use and having more memory for applications that is used in Linux Page Table.

If z/VM is used, z/VM manages the allocated memory that is then shared between the virtual Linux servers. When memory storage is over allocated and all virtual servers do not fit into this memory storage, it is called *overcommit*, or *level of over commitment*. Therefore, when the memory storage is overcommitted, pages are paged to the paging subsystem on Expanded Storage or disk by the z/VM Control Program.

It is not recommended to over allocate memory for production environments.

For DEV and QA environment, memory storage over allocation can be used 15% - 25% memory overcommitment.

However, with the memory capacity of the IBM System servers, memory storage over allocation software capabilities become useless because they require more tuning and administration time to monitor them.

Starting a performance determination for memory

Linux swapping is occurring with `vmstat` (the `si` and `so` columns that are shown in Figure 11-1 on page 171), or the system is showing swap pages in use specifically to physical disk volumes.

Also, check whether any evidence exists of the Out of Memory Killer (OOM) in the `/var/log/messages` directory. If evidence exists, a potential of a lack of virtual memory problem can occur on that Linux guest at that time.

Also, a command (such as **`: top, slabtop`**) can be used for more investigations.

11.1.3 I/O

With z/VM used, ECKD/DASD disk devices can be used for the Oracle Database files. Ensure HyperPAV is enabled for optimal performance (if available for your storage sub-system). Approximately 10 - 20 HyperPAV aliases per Logical Control Unit (LCU) showed best performance.

Also, more I/O paths (aliases) improves I/O performance for Oracle workloads.

Verify HyperPAV alias usage by using the `dasdstat` utility. Enable the `dasdstat` if necessary.

ECKD/DASD is recommended for all z/VM Control Program system devices and other data characterized as important for performance. Ensure enough devices are available so that device usage is less than 20% device busy.

For large Oracle databases, FCP is the most reasonable in terms of cost and manageability. When database sizes are measured in terabytes or petabytes, it is better to allocate disk space in terms of large LUNs on FCP.

Using a Linux File System then XFS is recommended for Oracle Database file.

Set the Oracle parameter for file system database files: `filesystemio_options=setall`, which provides asynchronous and direct I/O (avoids linux file system cache).

Ensure that multipath is activated and used to reduce the Linux Read-Ahead for LVM file systems (which improve performances) by using the **`: lvchange -r none <lv device name>`** command.

Starting a performance determination for I/O

Monitor the I/O performance by using the **`iostat -x`** command (see Figure 11-2).

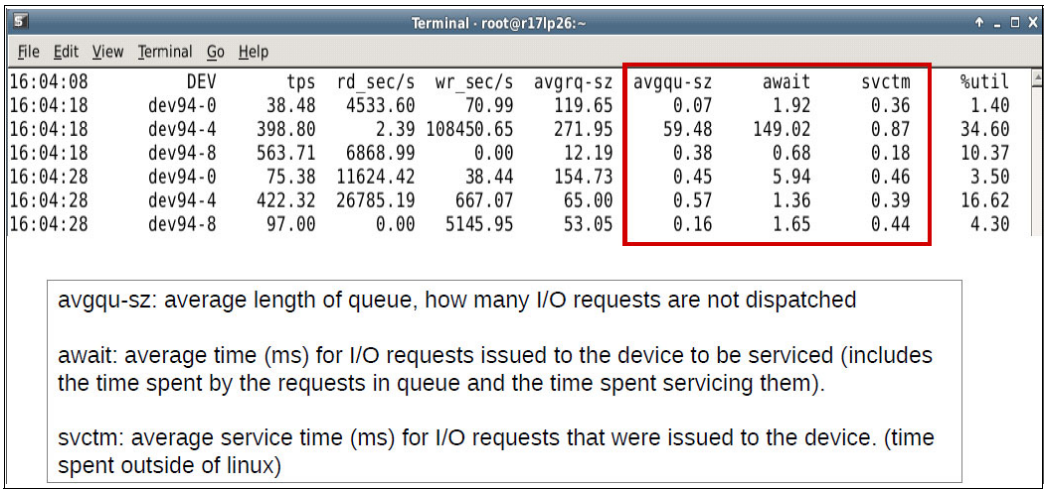

Figure 11-2 Monitoring I/O performance by using the iostat -x command

If **`iostat -x`** shows high service times (`svctm`), review the storage layer. If a high await (`r_await` or `w_await`) and low service time are found, the Linux layer also must be investigated. The `ziomon` (open systems storage) or `dasdstat` (ECKD/DASD) can be used to further drill down and isolate I/O performance related issues.

For more information, see *Oracle Database: Database Performance Tuning Guide 19c*.

11.1.4 Networking

Choose the correct, suitable network MTU size between the different network components per to the network architecture of the solution.

It is recommended to use dedicated OSA port (non-VSWITCH) for Linux guests with network sensitive or transactional workloads.

For the Linux kernel, the network queue length must be increased from the default size of 1000 to at least 2000 by using `sysctl`:

```
#sysctl -w net.core.netdev_max_backlog=2000
```

Starting a performance determination for the network

Check whether any network packet loss was occurred by using **`ifconfig -a`** command.

For more information, see this Red Hat web page.

11.1.5 Starting a performance determination for Oracle Database

For Oracle, review the Oracle AWR or Statspack Top 10 foreground wait events, as shown in Figure 11-3. Any high wait event (taking more time than DB CPU) must be investigated to look for opportunities for tuning improvements.

Top 10 Foreground Events by Total Wait Time

Event	Waits	Total Wait Time (sec)	Avg Wait	% DB time	Wait Class
latch: ges resource hash list	52,094,691	548.8K	10.53ms	67.0	Other
library cache: mutex X	30,904,972	68.8K	2.23ms	8.4	Concurrency
db file sequential read	96,100,654	52.7K	548.45us	6.4	User I/O
log file sync	58,346,117	51.6K	883.68us	6.3	Commit
DB CPU		49.2K		6.0	
latch: enqueue hash chains	2,573,108	10.8K	4.22ms	1.3	Other
latch free	1,946,407	5363.1	2.76ms	.7	Other
enq: TX - row lock contention	154,307	2000.4	12.96ms	.2	Application
enq: HW - contention	15,904	1183.8	74.43ms	.1	Configuration
cursor: pin S	618,501	1115.9	1.80ms	.1	Concurrency

Figure 11-3 Oracle AWR or Statspack Top 10 foreground wait events

Improvement might be made in the design, tuning the code, or potential software bugs. The Oracle ADDM report can be ran to further drill down into wait events with recommendations from the database for improving performance.

11.2 Oracle Database tools

Running the databases at optimal performance is a primary responsibility for an Oracle DBA. Achieving this goal requires familiarity with the available tools and the ability to choose the correct tool to use.

11.2.1 Oracle Diagnostics Pack

Oracle Diagnostics Pack is a comprehensive solution to manage the performance of Oracle Database environments. This pack includes the following popular features:

- Automatic Workload Repository (AWR)

  The AWR is a repository that collects statistics at predetermined intervals about the workloads within the database. The AWR provides a historical reference for performance changes over time, including establishment of performance baselines.

- Automatic Database Diagnostic Monitor (ADDM)

  This feature is an automated tool that focuses on the most intensive Oracle Database operations. It drills down proactively to determine the root cause of any performance problems.

- Active Session History (ASH)

  The ASH samples session activity every second and stores the information in views. Many DBAs use the `v$active_session_history` view to isolate performance problems with individual database sessions.

11.2.2 Oracle Tuning Pack for Oracle Database

The Oracle Tuning Pack for Oracle Database automates the database tuning process and includes the following key features:

- SQL Tuning Advisor

  One or more SQL statements can be entered that provide output in the form of specific advice or recommendations about how to optimize SQL for better performance, along with a rationale for each recommendation and its expected benefit. A recommendation might be to collect the statistics about objects, create indexes, restructure the SQL statements, or create SQL profiles.

  An automatic mode facility is available that allows the database to automatically implement recommendations. The most frequent use of this feature often is real-time SQL monitoring. If a production environment experiences a performance issue, this method is the most efficient way for a DBA to determine what SQL statements are running while the problem is occurring.

- SQL Access Advisor

  This feature enables you to optimize data access paths of SQL queries by recommending the suitable set of materialized views and view logs, indexes, and partitions for a specific SQL workload. It gives advice about how to optimize schema design to maximize query performance.

  This feature receives input from various sources, including AWR, to analyze a workload. It also provides recommendations about index creation and deletion, partition creation, and materialized views creation.

11.2.3 Oracle Enterprise Manager Cloud Control

Oracle Enterprise Manager Cloud Control is a popular tool for Oracle customers to manage databases and for provisioning. By using this tool, users can create or upgrade Oracle databases on Linux guests, and configure the network.

Cloud Control also provides an interface for performance advisors and for database utilities, such as SQL*Loader and Recovery Manager (RMAN). To manage Oracle databases, Database Control or Cloud Control can be used. These tools are intuitive, web-based tools from Oracle.

Database Control is included with every database installation and can be used to monitor a single database instance or a clustered database. Cloud Control can monitor many databases that are running on multiple environments from a single console.

In addition to database monitoring, other applications, such as Siebel, PeopleSoft, and specific non-Oracle products (for example, IBM WebSphere® Application Server) also can be monitored from a single Cloud Control console. Although the Cloud Control Server does not run on LinuxONE servers, the agents run on the Linux guests and communicate with the Cloud Control.

11.2.4 STATSPACK

If the users are not licensed to use Oracle Diagnostics Pack and Database Tuning Pack, they can use the STATSPACK utility, with less functions. Data is stored in tables that are owned by PERFSTAT.

11.2.5 Oracle Real Application Testing

The Oracle Real Application Testing includes the following features:

- Database Replay

  This feature enables you to capture the database workload on a production system, and replay it on a test system with the same timing and concurrency as the production system on the same or later release of Oracle Database.

- SQL Performance Analyzer

  This feature enables you to assess the effect of system changes on SQL performance by identifying SQL statements that regressed, improved, or remained unchanged.

11.2.6 Oracle Orion tool

The Oracle Orion tool is used to predict the I/O performance of an Oracle Database. The tool is part of Oracle Database installation and simulates Oracle Database I/O workloads by using the same I/O software stack as Oracle.

Orion also can simulate the effect of striping that is performed by Oracle Automatic Storage Management. It can run tests by using different I/O loads to measure performance metrics, such as MBPS, IOPS, and I/O latency.

If a database was created on a storage device, the storage alternatively can be calibrated by using the PL/SQL routine `dbms_resource_manager.calibrate_io()`.

11.2.7 Downloadable Oracle tools

The following downloadable tools are provided by Oracle on their support website. These tools also help to monitor and tune Oracle databases that are running on LinuxONE:

- SQL TExplain (SQL T)

  By using this powerful tool, users can diagnose SQL statements that are performing poorly. SQLT also helps to expedite SQL tuning process.

  For more information, see MOS Note 215187.1.

- Light Onboard Monitor (L TOM)

  LTOM provides real-time automatic problem detection and data collection. It proactively collects the necessary diagnostic traces in real time while the database or system performance problem is occurring.

  For more information, see MOS Note 352363.1.

- Remote Diagnostic Agent (RDA)

  The data that is captured provides Oracle Support with a comprehensive picture of the customer's environment, which aids in problem diagnosis.

  For more information, see MOS Note 414966.1.

- OSWatcher

  This tool gathers system profile data, including `vmstat`, `top`, and `iostat`, and works similar to a “sar”.

  For more information, see Oracle MOS Note 461053.1.

One of the deciding factors when considering the use of any of these tools is to ensure that the tool can collect usage metrics of the database. The tool also should identify specific database problems and solutions as they occur.

A

More information about the Oracle on LinuxONE solution

Appendix A provides more sources of information to assist planning deployments.

This appendix includes the following topics:

IBM z15 (8562) Technical Guide: An IBM Redbooks publication

This publication is available for download from this IBM Redbooks web page.

Abstract

This IBM Redbooks publication describes the features and functions the latest member of the IBM Z platform, the IBM z15 Model T02 (machine type 8562). It includes information about the IBM z15 processor design, I/O innovations, security features, and supported operating systems.

The z15 is a state-of-the-art data and transaction system that delivers advanced capabilities, which are vital to any digital transformation. The z15 is designed for enhanced modularity, which is in an industry standard footprint.

This system excels at the following tasks:

- Making use of multicloud integration services
- Securing data with pervasive encryption
- Accelerating digital transformation with agile service delivery
- Transforming a transactional platform into a data powerhouse
- Getting more out of the platform with IT Operational Analytics
- Accelerating digital transformation with agile service delivery
- Revolutionizing business processes
- Blending open source and Z technologies

Maximizing Security with LinuxONE: An IBM Redpaper publication

This IBM Redpaper publication is available for download from this IBM Redbooks web page.

Abstract

LinuxONE is a hardware system that is designed to support and use the Linux operating system based on the value of its unique underlying architecture. LinuxONE can be used within a private and multi-cloud environment to support a range of workloads and service various needs.

On LinuxONE, security is built into the hardware and software.

This IBM Redpaper publication gives a broad understanding of how to use the various security features that make the most of and complement the LinuxONE hardware security features, including the following examples:

- Hardware accelerated encryption of data, which is delivered with near-zero overhead by the on-chip Central Processor Assist for Cryptographic Function (CPACF) and a dedicated Crypto Express adapter.
- Virtualization and industry-leading isolation capabilities with PR/SM, EAL 5+ LPARs, DPM, KVM, and IBM z/VM.

- The IBM Secure Service Container technology, which provides workload isolation, restricted administrator access, and tamper protection against internal threats, including from systems administrators.
- Other technologies that use LinuxONE security capabilities and practical use cases for these technologies.

Oracle Database 19c and Oracle RAC 19c on IBM Z and IBM LinuxONE

This white paper is available for download at this IBM Support web page.

Abstract

This paper consolidates the information necessary for planning and implementing Oracle Database 19c (19.x.x.x) single instance database or Oracle Database 19c Real Application Clusters (RAC) on Linux on System z and LinuxONE servers. The material presented can also be applied to the IBM Z platform when running Linux operating systems.

Related publications

The publications that are listed in this section are considered particularly suitable for a more detailed discussion of the topics that are covered in this book.

Other publications

The following publications are relevant as further information sources:

- *Maximum Availability with Oracle Database 19c*:

  https://www.oracle.com/a/tech/docs/maximum-availability-wp-19c.pdf

- *Maximum Availability Architecture - Best Practices for Oracle Database 19c*:

  https://www.oracle.com/a/tech/docs/tip4847-maa-best-practices-for-database.pdf

- *Oracle Database: Unplugging, Plugging, and Upgrading a PDB to a New CDB*:

  https://docs.oracle.com/en/database/oracle/oracle-database/19/spupu/unplugging-plugging-and-upgrading-pdb-new-cdb.pdf

- SUSE Linux Enterprise Server High Availability Extension:
  - https://www.suse.com/products/systemz/features/
  - https://www.suse.com/products/highavailability/

- Support Policies for RHEL High Availability Clusters - IBM System Z/VM Guests as Cluster Members:

  https://access.redhat.com/articles/3130081

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Redbooks

Oracle 19c on IBM LinuxONE

(0.2”spine)
0.17”<->0.473”
90<->249 pages

SG24-8487-00

ISBN 0738459518

Printed in U.S.A.