

**RSA Identity Governance and Lifecycle
Database Setup and Management Guide
V7.0.1**



Contact Information

RSA Link at <https://community.rsa.com> contains a knowledgebase that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

Trademarks

RSA, the RSA Logo, and EMC are either registered trademarks or trademarks of EMC Corporation in the United States and/or other countries. All other trademarks used herein are the property of their respective owners. For a list of EMC trademarks, go to www.emc.com/legal/emc-corporation-trademarks.htm#rsa.

License agreement

This software and the associated documentation are proprietary and confidential to EMC, are furnished under license, and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright notice below. This software and the documentation, and any copies thereof, may not be provided or otherwise made available to any other person.

No title to or ownership of the software or documentation or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software and the documentation may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by EMC.

Third-party licenses

This product may include software developed by parties other than RSA. The text of the license agreements applicable to third-party software in this product may be viewed by launching the RSA Identity Governance and Lifecycle product and selecting the About menu.

Note on encryption technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when using, importing or exporting this product.

Distribution

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license. EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." EMC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright © 2016 EMC Corporation. All Rights Reserved. Published in the USA.

September 2016

Contents

Preface	4
About This Guide	4
Documentation Set	4
Support and Service	4
Chapter 1: Introduction	5
Overview	5
Database Parameter Values Worksheet	6
Download Installation Files for Your Database	7
Database Hardware Requirements	7
Requirements for Database Deployment (Non-RAC)	7
Requirements for Database Deployment (RAC)	8
Hardware Requirements	8
Information Required for Installation	9
Getting Sample Database Configuration Scripts	10
Chapter 2: Set Up the Database	11
Overview	11
Deployment Summary	11
Prepare the Oracle Database Instance (If Upgrading from a Previous Product Version)	12
Prepare the Oracle Database Instance (For a New Product Installation)	16
Create the Required Objects	20
Create Tablespaces	21
Create the Export/Import Database Directory	24
Map the Export/Import Directory	25
Create a Database User Password Profile	25
Create User Schemas	26
Configure the User Schema Privilege Grants	27
Create a Report Context	29
Complete the Required Database Instance Configuration	29
Installing Oracle Statspack to Enhance Database Diagnostics Capabilities	30
Verify Correct Database Configuration	30
Updating the Database for RSA Identity Governance and Lifecycle Upgrades	31
Chapter 3: Maintaining the Database	32
Back Up the Customer-Supplied Database	32
Importing AVUSER Schema/Data for a Customer-Supplied Database Restoration/Load	33
Validate Compatibility of the Database Import	35
Removing User Schemas from the Database	35

Preface

About This Guide

This guide provides instructions on how to set up an Oracle database in your infrastructure (or customer-supplied database) for RSA Identity Governance and Lifecycle V7.0.1. This guide assumes that the reader is authorized and qualified to configure and maintain the Oracle database.

Documentation Set

The latest product documentation is always available at <https://community.rsa.com/community/products/governance-and-lifecycle>.

Document	Description
Release Notes	What's new in the release, fixed issues, known issues and workarounds.
Installation Guide	Product installation instructions.
Upgrade and Migration Guide	Instructions for upgrading your product version and data.
Database Setup and Management Guide	Instructions for setting up and managing a customer-supplied Oracle database for RSA Identity Governance and Lifecycle.
Online Help	All concepts and instructions you need to configure and use the product.
Public Database Schema Reference	The public view of the database schema.

Support and Service

You can access community and support information on RSA Link at <https://community.rsa.com/community/products/governance-and-lifecycle>. RSA Link contains a knowledgebase that answers common questions and provides solutions to known problems, product documentation, community discussions, and case management.

You can also access the RSA Identity Governance and Lifecycle community at <https://community.rsa.com/community/products/governance-and-lifecycle/client-partner-community>. This private community is only available to RSA Identity Governance and Lifecycle customers, partners and internal RSA staff.

Chapter 1: Introduction

- [Overview](#)
- [Download the Installation Files for Your Database](#)
- [Hardware Requirements](#)
- [Requirements for Database Deployment \(Non-RAC\)](#)
- [Requirements for Database Deployment \(RAC\)](#)
- [Getting Sample Database Configuration Scripts](#)

Overview

Note: The Oracle database you maintain is referred to as the "remote database" in some sections in this document. The term is synonymous with the "customer-supplied database" term used in this guide and in other installation and upgrade documents. Both terms refer exclusively to a database you provide for your RSA Identity Governance and Lifecycle installation.

This chapter describes how to set up a database instance (RAC cluster option included) used by RSA Identity Governance and Lifecycle (application server) in a two-tier installation scenario where you supply and maintain an Oracle database in your hardware infrastructure.

It provides sample script download instructions and the steps required to configure or upgrade the database. The database must be configured before you install RSA Identity Governance and Lifecycle software.

RSA Identity Governance and Lifecycle is designed to use most (but not all) default Oracle database instance configuration settings and installation options. This document does not attempt to document every Oracle configuration setting or installation option for Oracle but instead covers only those settings or installation requirements over and above a default Oracle installation. Any modifications to the Oracle installation or configuration beyond the Oracle defaults and what is noted in this document may cause functional or performance issues with RSA Identity Governance and Lifecycle. Please contact RSA Support if you want to change any installation option or configuration beyond what is documented here to ensure a supportable configuration. Review [Prepare the Oracle Database Instance](#) for specific details on RSA Identity Governance and Lifecycle requirements.

Important: The SQL commands and configurations documented here should be executed using a database account with sysdba privileges in the database.

This document describes the database objects, tables, and directories that must be created. These include the database user schemas that RSA Identity Governance and Lifecycle uses:

- RSA Identity Governance and Lifecycle user. The default name is AVUSER.
- RSA Identity Governance and Lifecycle reporting engine user. The default name is AVDWUSER.
- RSA Identity Governance and Lifecycle public database schema user. The default name is ACMDB.
- RSA Identity Governance and Lifecycle RSA Identity Governance and Lifecycle Statistics Report user. The data source is AVPERF. (This is required only if Oracle Statspack is installed on the database and you want to include Statspack data in RSA Identity Governance and Lifecycle Statistics Reports.) Failure to configure and use Statspack will limit reporting information for database diagnostics. The default oracle user name is PERFSTAT. See [Installing Oracle Statspack to Enhance Database Diagnostics Capabilities](#) for more information.

This document references the user schemas by their default names. You can use non-default user schema names; however, that will require additional configuration when installing RSA Identity Governance and Lifecycle.

Database Parameter Values Worksheet

Print the worksheet below, record parameter values (case-sensitive), and provide the worksheet to the person who installs RSA Identity Governance and Lifecycle. The installer require these values for the installation.

RSA Identity Governance and Lifecycle schema user (AVUSER):

RSA Identity Governance and Lifecycle reporting engine schema user (AVDWUSER):

RSA Identity Governance and Lifecycle public database schema user (ACMDB):

AVUSER password:

AVDWUSER password:

ACMDB password:

AVPERF username/password:

Oracle Database SID:

(RAC) TNSName that references the SCAN name:

Oracle service name:

Oracle listener Hostname:

Oracle Listener Port:

Download Installation Files for Your Database

Before You begin

- You must have a valid license for RSA Identity Governance and LifecycleV7.0.1.
- Ensure that your remote Oracle 12.1.0.2.0 database has all up-to-date Oracle patches installed.

Procedure

1. Log in to RSA Link at <https://community.rsa.com/community/products/governance-and-lifecycle>.
2. Download the following files to the RSA Appliance or Soft Appliance machine.
 - instantclient-basiclite-linux-x64-12.1.0.2.0.zip
 - instantclient-sqlplus-linux-x64-12.1.0.2.0.zip

Database Hardware Requirements

The system on which the Oracle database is installed must adhere to the following requirements:

- 8 GB of free disk space in the /tmp directory
- No more than one root user (UID=0) configured
- The root user must have a umask of 0022
- The system must have sufficient disk space and correct permissions set as described in this guide.

Requirements for Database Deployment (Non-RAC)

Important: Synchronize the database system clock to the system clock on the application server where RSA Identity Governance and Lifecycle is installed.

- Hardware requirements for the database server:
 - Memory: Minimum 16GB, 32GB recommended for production servers, minimum 4GB, 8GB recommended for development servers
 - Disk: Minimum 300GB, 1TB+ recommended based on data load requirements (RAID 5 SAS drive configuration or better is highly recommended for performance)
 - Processors: Dual Quad Core XEON processors (E5420 or higher)
 - Network: Dual-load balanced gigabit NICs recommended

- An Oracle database (12cR1 version 12.1.0.2 for RAC or non-RAC implementation) running on a database machine that meets hardware requirements at the deployment site
- A database account with sysdba privileges to be able to perform the database system configuration and validation tasks described in this chapter
- A UNIX root user to create appropriate physical directories and users on the server

Requirements for Database Deployment (RAC)

Important: Synchronize the database system clock to the system clock on the application server where RSA Identity Governance and Lifecycle is installed.

Hardware Requirements

- Cluster instances: At least two database cluster instances required for load balancing and high availability
- Memory: Minimum 16GB, 32GB recommended for production servers, minimum 4GB, with 8GB recommended for development servers
- Processors: Dual Quad Core XEON processors (X3430, 2.4 GHz or higher)
- Disk: Minimum 100GB local to each node to house Oracle Home and local files. SAS drive configuration or better highly recommended for performance.
- Storage Area Network (SAN):
 - Disks: Minimum of 8 SATA/SAS drives, for a total of 2TB+ of disk space
 - Volume Configuration: Recommended RAID 5 for development and production servers (with RAID5 and RAID 0+1 recommended for production servers with higher performance demands)
 - LUN configuration: At least three LUNs should be configured and assigned iSCSI initiator names:
 - CRS1: CRS/Cluster Voting Disk, 10GB recommended
 - FS1: Shared file storage space, 200GB recommended
 - VOL1: Main tablespace volume, the remainder of the 2TB.

RSA recommends that an ACFS (ASM Clustered File System) share is created on the “FS1” LUN using “asmca” (Oracle ASM Configuration Assistant) from the “ASM Cluster File Systems” tab with the following options:

- General Purpose File System
- Register MountPoint (Our examples use /mnt/acfs-fs1.)

Note: When Oracle CRS starts up, this volume is automatically mounted. No modifications to the /etc/fstab file are required.

- Public Network: Dual-load balanced gigabit NICs recommended
- Private Network:
 - Network speed: Minimum 1Gb NIC, with 10Gb recommended for production servers
 - Frame size: Jumbo frames set on the switch

Information Required for Installation

Information the database system administrator requires for installation:

- Oracle service name
- Oracle (SCAN) listener Hostname (This is a virtual host address that is served by the cluster as a whole. It should be located in the DNS.)
- Oracle (SCAN) Listener Port (This is a virtual port.)
- Public virtual hostnames and IP addresses for all nodes in cluster
- Oracle Database SID template (For example, "avdb_" will result in instances with names avdb_1, avdb_2, etc.)
- Private hostnames and IP addresses for all nodes on the private interconnect
- Public hostname of Storage Area Network (SAN)
- Private hostname of Storage Area Network (SAN) on the private interconnect
- The name of the iSCSI Discovery initiator on the SAN
- The names of the iSCSI initiators for the other LUNs on the SAN

Getting Sample Database Configuration Scripts

You can find the database sample configuration scripts located in the DatabaseSamples directory in the aveksa-supplement-*.zip file from the packages directory for this RSA Identity Governance and Lifecycle release. The file includes scripts that include SQL used to set up the database (table spaces, user schemas, and directories on the database server). The files may be generic or contain SQL for a specific version of Oracle.

Scripts:

- aveksa_sample_tablespace_ASM_scripts.sql — Includes sample "create tablespace" instructions for use with ASM.
- aveksa_sample_tablespace_file_scripts.sql — Includes sample "create tablespace" instructions for use with a file system.
- aveksa_sample_ora12_db_scripts.sql — Includes the various commands to create the Aveksa users with the required grants and settings on Oracle 12cR1.
- aveksa_sample_sys_scripts.sql — Includes the required command samples to complete the database configuration.
- aveksa_db_check.sql — Includes the required SQL to verify minimum file size requirements.
- aveksa_db_password_lifetime.sql — Includes a script to determine information about database user password expiration dates.

Note: If you are using an Oracle RAC implementation, ensure that the directories referenced in the scripts point to clustered file systems.

Chapter 2: Set Up the Database

- [Overview](#)
- [Deployment Summary](#)
- [Prepare the Oracle Database Instance \(If Upgrading from a Previous Product Version\)](#)
- [Prepare the Oracle Database Instance \(For a New Product Installation\)](#)
- [Create the Required Objects](#)
- [Complete the Required Database Instance Configuration](#)
- [Installing Oracle Statspack to Enhance Database Diagnostics Capabilities](#)
- [Verify Correct Database Configuration](#)
- [Updating the Database](#)

Overview

RSA Identity Governance and Lifecycle requires an Oracle database instance pre-configured with the requirements outlined in this chapter. One requirement is that database users are created with necessary Oracle grant privileges. RSA Identity Governance and Lifecycle creates all of its required database objects such as tables, views, and packages when it is initially started and fully populates the database schema. Those objects may also be modified by upgrades of RSA Identity Governance and Lifecycle software through the RSA Identity Governance and Lifecycle migration process.

Deployment Summary

The deployment process includes the following steps:

1. Preparing the Oracle database instance:
 - a. Using a designated instance name.
 - b. Using a designated character set.
 - c. Ensuring a minimum memory configuration for both the SGA (system global area) and PGA (program global area). You can use ASMM to set either SGA/PGA or use AMM (not recommended) to set `memory_target` and `memory_max_target`. The settings are mutually exclusive.
 - d. Configuring the database instance with all required packages.

See [Prepare the Oracle Database Instance \(If Upgrading from a Previous Product Version\)](#) or [Prepare the Oracle Database Instance \(For a New Product Installation\)](#).

2. Creating the Aveksa objects:
 - a. Creating a set of table spaces with the specified naming conventions.
 - b. Creating a set of database directories.
 - c. Mapping the database directories to specified physical directories.
 - d. Creating a database password profile.
 - e. Configuring the user schemas with all required settings.
 - f. Creating a context for RSA Identity Governance and Lifecycle report filtering values.
See [Create the Aveksa Objects](#).
3. Completing the required database instance configurations.
 - Disabling the nightly job that gathers schema statistics.See [Complete the Required Database Instance Configuration](#).
See [Verify Correct Database Configuration](#).
4. Verify correct database configuration.
See [Verify Correct Database Configuration](#).
5. Update the database.
See [Updating the Database](#).

Prepare the Oracle Database Instance (If Upgrading from a Previous Product Version)

Do the following on the database server:

1. Create or identify the instance name that is going to be used.
2. Ensure that the database instance uses the Unicode (AL32UTF8) character set.

You can validate the character set by running the following SQL:

```
select * from NLS_DATABASE_PARAMETERS where parameter='NLS_
CHARACTERSET' ;
```

Output: NLS_CHARACTERSET AL32UTF8

RSA Identity Governance and Lifecycle will fail to start if this character set is not set for the database instance.

This character set is not the default when configuring Oracle. `NLS_LENGTH_SEMANTICS` is required to be `BYTE`, do not change this setting to `CHAR`.

Note: The following steps in this section assume that your database is initialized using an spfile and not a pfile. To determine if your database is using a pfile or an spfile, you can run the following commands via SQL*Plus. If the first command returns a value for "ifile," then this value is the name and location of the pfile for your system. If an "spfile" value is found, then this is the name and location of the spfile for your system.

```
show parameter ifile
show parameter spfile
```

One of these commands returns a value.

Convert the a pfile into an spfile if your system is using a pfile. For example:

```
shutdown immediate;
startup pfile=<ORACLE_HOME>/dbs/init<ORACLE_SID>.ora
create spfile='<ORACLE_HOME>/dbs/spfile<ORACLE_SID>.ora' FROM
pfile='<ORACLE_HOME>/dbs/init<ORACLE_SID>.ora';
shutdown immediate;
startup;
```

3. Configure memory management settings for Oracle.

Setting ASMM Values

RSA recommends that memory that can be dedicated to the Oracle instance is allocated in a 2:1 ratio between SGA and PGA. For example, if a host with 128GB of RAM is dedicated to run a single instance of Oracle, allocate 8GB of RAM for the operating system, and set SGA to 80GB, and PGA to 40GB (which would divide the remaining 120GB between SGA and PGA).

- a. Determine the memory allocation by running the following SQL:

```
show parameter sga_max_size;
show parameter sga_target;
show parameter pga_aggregate_target;
```

- b. Change the parameters with the recommended values by running the following commands:

```
alter system set sga_max_size=<sga_value> scope=spfile
alter system set sga_target=<sga_value> scope=spfile;
alter system set pga_aggregate_target=<pga_value> scope=spfile;
shutdown immediate;
startup;
```

- c. Make sure that AMM is not enabled by verifying that the following parameters are not greater than 0:

```
show parameter memory_target
```

```
show parameter memory_max_target
```

- d. If required, unset the parameter(s) by executing the following command and then restart the database:

```
alter system reset memory_target scope=spfile;
```

```
alter system reset memory_max_target scope=spfile;
```

Setting AMM Values

While using AMM has not been tested by RSA and some experts DO NOT recommend using AMM management with Linux HugePages, the following commands will help you set Oracle initialization parameters:

If X is the amount memory that can be dedicated to Oracle instance, RSA recommends that you set Oracle initialization parameters as follows:

Parameter Name	Value
memory_target	X
memory_max_target	X
sga_target	$X * 2/6$
pga_aggregate_target	$X * 1/6$

For example, if a host with 128GB of RAM is dedicated to run a single instance of Oracle, allocate 8GB of RAM for the operating system, and set memory_target and memory_max_target to 120GB, SGA minimum value to 40GB, and PGA minimum value to 20GB.

- a. Determine the memory allocation by running the following SQL:

```
show parameter memory_target
show parameter memory_max_target
show parameter sga_target
show parameter pga_aggregate_target
```

- b. Change the parameters with the recommended values by running the following commands:

```
alter system set memory_target=<memory_value> scope=spfile;
alter system set memory_max_target=<memory_value> scope=spfile;
alter system set sga_target=<sga_value> scope=spfile;
alter system set pga_aggregate_target=<pga_value> scope=spfile;
shutdown immediate;
startup;
```

4. Configure adequate space for all system tablespaces (see [Configure Undo, Temp, and Redo Sizes](#)).
5. Make sure that the database instance is configured with the XML_DB package. You can verify that XML DB has been installed by running the following SQL:

```
select comp_name from dba_registry where comp_name like '%XML%';
```

You should see results like the following:

```
COMP_NAME
Oracle XML Database
```

If this package does not exist, it can be installed with the database configuration assistant (dbca) or manually by executing the instructions found in the Oracle documentation at the following location:

```
http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14259/appaman.htm#CACIBCBA
```

6. Unset the following database optimizer settings that were previously set:


```
alter system reset optimizer_index_caching scope=spfile;
alter system reset optimizer_index_cost_adj scope=spfile;
alter system reset optimizer_adaptive_features scope=spfile;
```
7. The value of the cursor sharing parameter `cursor_sharing`, which determines which kinds of SQL statement can share the same cursors. Run the following command:


```
alter system set cursor_sharing=force scope=spfile;
```
8. Configure the database to accommodate a minimum of 500 processes by running the following

command:

```
alter system set processes=500 scope=spfile;
```

If your database serves multiple application server nodes, multiply the process number by the number of nodes.

If you set “sessions” in your previous product release, let Oracle use the default by running the following command:

```
alter system reset sessions scope=spfile;
```

9. Configure the log_buffer setting to 200 MB and the log_checkpoint_interval setting to 180 MB:

```
alter system set log_buffer=209715200
```

```
alter system set log_checkpoint_interval=188743680
```

The default settings for these support online transactional processing (OLTP) systems and not Data Warehousing systems. The larger settings are required to support data collections (which are more Data Warehouse style activities) by RSA Identity Governance and Lifecycle.

10. Configure the creation of deferred segments for tables to false, and also unset the db_file_multiblock_read_count parameter if it was changed. This is required for the Oracle 12c R1 Enterprise version.

```
alter system set DEFERRED_SEGMENT_CREATION = FALSE
```

```
alter system reset db_file_multiblock_read_count scope=spfile;
```

11. Validate that database requirements are reflected in your base Oracle startup by validating the spfile used by your database instance. Run this command:

```
SELECT NAME, Value FROM gv$parameter order by name
```

12. Restart the database server.

Prepare the Oracle Database Instance (For a New Product Installation)

Do the following on the database server:

1. Create or identify the instance name that is going to be used.
2. Ensure that the database instance uses the Unicode (AL32UTF8) character set.

You can validate the character set by running the following SQL:

```
select * from NLS_DATABASE_PARAMETERS where parameter='NLS_
CHARACTERSET';
```

Output: NLS_CHARACTERSET AL32UTF8

RSA Identity Governance and Lifecycle will fail to start if this character set is not set for the database instance.

This character set is not the default when configuring Oracle. NLS_LENGTH_SEMANTICS is required to be BYTE, do not change this setting to CHAR.

Note: The following steps in this section assume that your database is initialized using an spfile and not a pfile. To determine if your database is using a pfile or an spfile, you can run the following commands via SQL*Plus. If the first command returns a value for "ifile", then the this value is the name and location of the pfile for your system. If an "spfile" value is found, then this is the name and location of the spfile for your system.

```
show parameter ifile
show parameter spfile
```

One of these commands returns a value.

Convert the a pfile into an spfile if your system is using a pfile. For example:

```
shutdown immediate;
startup pfile=<ORACLE_HOME>/dbs/init<ORACLE_SID>.ora
create spfile='<ORACLE_HOME>/dbs/spfile<ORACLE_SID>.ora' FROM
pfile='<ORACLE_HOME>/dbs/init<ORACLE_SID>.ora';
shutdown immediate;
startup;
```

3. Configure memory management settings for Oracle.

Setting ASMM Values

RSA recommends that the memory that can be dedicated to Oracle instance is allocated in a 2:1 ratio between SGA and PGA. For example, if a host with 128GB of RAM is dedicated to run a single instance of Oracle, allocate 8GB of RAM for the operating system, and set SGA to 80GB, and PGA to 40GB (which would divide the remaining 120GB between SGA and PGA).

- a. Determine the memory allocation by running the following SQL:

```
show parameter sga_max_size;
show parameter sga_target;
show parameter pga_aggregate_target;
```

- b. Change the parameters with the recommended values by running the following commands:

```
alter system set sga_max_size=<sga_value> scope=spfile
alter system set sga_target=<sga_value> scope=spfile;
alter system set pga_aggregate_target=<pga_value> scope=spfile;
shutdown immediate;
startup;
```

- c. Make sure that AMM is not enabled by verifying that the following parameters are not greater than 0:

```
show parameter memory_target
show parameter memory_max_target
```

- d. If required, unset the parameter(s) by executing the following command and then restart the database:

```
alter system reset memory_target scope=spfile;
alter system reset memory_max_target scope=spfile;
```

Setting AMM Values

While using AMM has not been tested by RSA and some experts DO NOT recommend using AMM management with Linux HugePages, the following commands will help you set Oracle initialization parameters:

If X is the amount memory that can be dedicated to Oracle instance, RSA recommends that you set Oracle initialization parameters as follows:

Parameter Name	Value
memory_target	X

Parameter Name	Value
memory_max_target	X
sga_target	X * 2/6
pga_aggregate_target	X * 1/6

For example, if a host with 128GB of RAM is dedicated to run a single instance of Oracle, allocate 8GB of RAM for the operating system, and set memory_target and memory_max_target to 120GB, SGA minimum value to 40GB, and PGA minimum value to 20GB.

- a. Determine the memory allocation by running the following SQL:

```
show parameter memory_target
show parameter memory_max_target
show parameter sga_target
show parameter pga_aggregate_target
```

- b. Change the parameters with the recommended values by running the following commands:

```
alter system set memory_target=<memory_value> scope=spfile;
alter system set memory_max_target=<memory_value> scope=spfile;
alter system set sga_target=<sga_value> scope=spfile;
alter system set pga_aggregate_target=<pga_value> scope=spfile;
shutdown immediate;
startup;
```

4. Configure adequate space for all system tablespaces (see [Configure Undo, Temp, and Redo Sizes](#)).
5. Make sure that the database instance is configured with the XML_DB package. You can verify that XML DB has been installed by running the following SQL:

```
select comp_name from dba_registry where comp_name like '%XML%';
```

You should see results like the following:

```
COMP_NAME
Oracle XML Database
```

If this package does not exist, it can be installed with the database configuration assistant (dbca) or manually by executing the instructions found in the Oracle documentation at the following location:

```
http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14259/appaman.htm#CACIBCBA
```

6. Configure the database to accommodate a minimum of 500 processes by running the following command:

```
alter system set processes=500 scope=spfile;
```

If your database serves multiple application server nodes, multiply the process number by the number of nodes.

7. Configure the log_buffer setting to 200 MB and the log_checkpoint_interval setting to 180 MB:

```
alter system set log_buffer=209715200
```

```
alter system set log_checkpoint_interval=188743680
```

The default settings for these support online transactional processing (OLTP) systems and not Data Warehousing systems. The larger settings are required to support data collections (which are more Data Warehouse style activities) by RSA Identity Governance and Lifecycle.

8. Configure the creation of deferred segments for tables to false.

```
alter system set DEFERRED_SEGMENT_CREATION = FALSE
```

9. Validate that database requirements are reflected in your base Oracle startup by validating the spfile used by your database instance. Run this command:

```
SELECT NAME, Value FROM gv$parameter order by name
```

10. Restart the database server.

Create the Required Objects

Create all required table spaces, user schemas, and directories on the database server.

Create Tablespaces

The standard RSA Identity Governance and Lifecycle database setup has eight tablespaces, four for data and four for indices. RSA Identity Governance and Lifecycle uses the well known tablespace names when creating the various database objects, like tables and views, within the database. The table below reflects the recommended sizes for the tablespaces. See the `aveksa_sample_tablespace_file_scripts.sql` script for an example of creating these tablespaces using the file system, or see the `aveksa_sample_tablespace_ASM_scripts.sql` script for an example of creating these tablespaces using ASM with the recommended sizes. These sample scripts must be edited for your configuration before you use them.

These tablespaces must be configured with adequate storage settings as appropriate for your installation. The table below provides the initial, extended, and fixed size recommendations for the tablespaces.

If you do not wish to use auto extend in your database, use the fixed size as a recommended initial size.

Tablespace Name	Initial Size	Extend Size	Fixed Size
DATA_256K	25M	25M	500M
DATA_1M	200M	200M	10G
DATA_25M	25M	25M	500M
DATA_50M	200M	200M	10G
INDX_256K	100M	100M	10G
INDX_1M	100M	100M	10G
INDX_25M	400M	200M	10G
INDX_50M	100M	100M	10G

You can configure your database with different tablespace names or fewer tablespaces. However, because RSA Identity Governance and Lifecycle uses the eight known tablespaces names internally, additional configuration is required when installing or upgrading RSA Identity Governance and Lifecycle.

Important: The RSA Identity Governance and Lifecycle configuration requires that you map your tablespaces to the names known to RSA Identity Governance and Lifecycle.

Configure Undo, Temp, and Redo Sizes

RSA Identity Governance and Lifecycle requires minimum sizes for Oracle's undo, temp and redo logs to perform effectively. Temp and undo each require a minimum size of 96 GB. Redo requires a minimum size of 2.4 GB. The tables below provide examples of the database commands to increase sizes for the database files for both Oracle ASM and file system implementations. The examples below assume you know the filesystem paths for your Oracle installation and database instance. Ensure there is adequate disk space for the changes.

- Temp Logs. The examples in the following table demonstrate how to increase the temp filespace (default 32 GB) to 96 GB by adding two additional temp files of 32 GB each.

ASM	<pre>ALTER TABLESPACE "TEMP" ADD TEMPFILE '+DG01' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M; ALTER TABLESPACE "TEMP" ADD TEMPFILE '+DG01' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M;</pre>
File System	<pre>ALTER TABLESPACE "TEMP" ADD TEMPFILE '{ORACLE_BASE}/oradata/{DB_ UNIQUE_NAME}/temp02.dbf' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M; ALTER TABLESPACE "TEMP" ADD TEMPFILE '{ORACLE_BASE}/oradata/{DB_ UNIQUE_NAME}/temp03.dbf' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M;</pre>

- Undo Logs. The examples in the following table demonstrate how to increase the undo filespace (default 32GB) to 96GB by adding two additional temp files of 32GB each.

ASM	<pre>ALTER TABLESPACE "UNDOTBS1" ADD DATAFILE '+DG01' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M; ALTER TABLESPACE "UNDOTBS1" ADD DATAFILE '+DG01' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M;</pre>
File System	<pre>ALTER TABLESPACE "UNDOTBS1" ADD DATAFILE '{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}/undotbs02.dbf' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M; ALTER TABLESPACE "UNDOTBS1" ADD DATAFILE '{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}/undotbs03.dbf' SIZE 128M REUSE AUTOEXTEND ON NEXT 256M MAXSIZE 32767M;</pre>

- Redo Logs. To obtain information about your systems redo files execute the following SQL:

ASM	<pre>// Extend the existing redo logs. alter Database Clear Logfile '+DG01/avdb/onlinelog/group_ 1.263.763560403'; alter Database Drop Logfile Group 1; alter Database Add Logfile Thread 1 Group 1 '+DG01/avdb/onlinelog/group_1' Size 800m; alter Database Clear Logfile '+DG01/avdb/onlinelog/group_ 2.262.763560421'; alter Database Drop Logfile Group 2; alter Database Add Logfile Thread 1 Group 2 '+DG01/avdb/onlinelog/group_2' Size 800m; alter Database Clear Logfile '+DG01/avdb/onlinelog/group_ 3.261.763560439'; alter Database Drop Logfile Group 3; alter Database Add Logfile Thread 1 Group 3 '+DG01/avdb/onlinelog/group_3' Size 800m; // Add three additional redo logs. alter Database Add Logfile Thread 1 Group 4 '+DG01/avdb/onlinelog/group_4' Size 800m; alter Database Add Logfile Thread 1 Group 5 '+DG01/avdb/onlinelog/group_5' Size 800m; alter Database Add Logfile Thread 1 Group 6 '+DG01/avdb/onlinelog/group_6' Size 800m;</pre>
File System	<pre>// Extend the existing redo logs. alter Database Clear Logfile '{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME} /GROUP_1'; alter Database Drop Logfile Group 1; alter Database Add Logfile Thread 1 Group 1 '{ORACLE_BASE}/oradata/ {DB_UNIQUE_NAME}/group_1' Size 800m; alter Database Clear Logfile {ORACLE_BASE}/oradata/{DB_UNIQUE_NAME} /GROUP_2'; alter Database Drop Logfile Group 2; alter Database Add Logfile Thread 1 Group 2 '{ORACLE_BASE}/oradata/</pre>

```

{DB_UNIQUE_NAME}/group_2' Size 800m;

alter Database Clear Logfile '{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}
/REDO03.LOG';

alter Database Drop Logfile Group 3;

alter Database Add Logfile Thread 1 Group 3 '{ORACLE_BASE}/oradata/
{DB_UNIQUE_NAME}/group_3' Size 800m

// Add three additional redo logs.

alter Database Add Logfile Thread 1 Group 4 '{ORACLE_BASE}/oradata/
{DB_UNIQUE_NAME}/group_4' Size 800m

alter Database Add Logfile Thread 1 Group 5 '{ORACLE_BASE}/oradata/
{DB_UNIQUE_NAME}/group_5' Size 800m

alter Database Add Logfile Thread 1 Group 6 '{ORACLE_BASE}/oradata/
{DB_UNIQUE_NAME}/group_6' Size 800m

```

```
select * from v$logfile;
```

RSA Identity Governance and Lifecycle requires sufficient redo space to function properly. It requires six redo logs of 800 MB each at a minimum. By default Oracle database is configured with three redo logs of 50 MB each. The examples in the following table demonstrate how to increase the size of each redo logs to 800 MB by dropping the original three redo logs and adding three of 800 MB. After extending the default three oracle redo logs, add an additional three logs of 800 MB.

Create the Export/Import Database Directory

The Export/Import directory is used for the storage and retrieval of database backups (if RSA Identity Governance and Lifecycle provided backup and restore scripts are used). This directory is optional if you are using your own service tools or scripts to backup and restore the database. Otherwise, this directory is mandatory.

You must create the directory in the filesystem on the database server before you begin database instance configuration. See [Map the Export/Import Directory](#) for more information. Create the directory, for example: `/home/aveksa/AveksaExportImportDir`

Directory Permissions

The database user must be provided Read-Write permissions to the Export/Import directory. It is accessed by the user running the associated backup/restore scripts.

Directory Size Allotment

For the Export/Import directory, make sure the disk where the directory is located can accommodate at least the estimated size of your exported database backup.

RAC Database Implementation Only

In an Oracle RAC installation, the Export/Import directory must point to clustered file system locations. The following instructions use an example environment variable, `CLUSTERED_FILE_SHARE`, which can be freely substituted for whatever common location is defined for your installation.

1. Make sure the mount point is owned by oracle (and not root). For example:

```
chown -R oracle:oinstall /mnt/acfs-fs1/*
```

2. Set the environment variable on all nodes to the mount point. For example:

```
export CLUSTERED_FILE_SHARE=/mnt/acfs-fs1
cd ${CLUSTERED_FILE_SHARE}
```

3. Create the directory on the file share. This is only required from one node. For example:

```
mkdir AveksaExportImportDir
```

4. Create symbolic links to the mount point. This creates the expected directory location within the `/home/aveksa` root. For example:

```
ln -s ${CLUSTERED_FILE_SHARE}/AveksaExportImportDir
/home/oracle/AveksaExportImportDir
```

Map the Export/Import Directory

See the `aveksa_sample_ora12_db_scripts.sql` for installations using Oracle 12cR1 (12.1.0.2) for examples on how to map the directory:

Map the directory variable to the physical directories previously created as described in [Create the Aveksa Export/Import Database Directory](#):

```
create or replace directory AVEKSA_EXPORTIMPORT_DIRECTORY as
'/home/oracle/AveksaExportImportDir';
```

Create a Database User Password Profile

This section describes how to create a database profile for the users that connect to the database.

Oracle 12cR1 has a default password expiration of 180 days. If a database user password were to expire, RSA Identity Governance and Lifecycle would fail to connect to the database.

Important: If you choose to have a password policy that expires for the RSA Identity Governance and Lifecycle users, you will have to reconfigure the user database password settings when passwords expire. The sample file `aveksa_db_password_lifetime.sql` shows how to obtain the password lifetime information for the RSA Identity Governance and Lifecycle database users.

See the `aveksa_sample_ora12_db_scripts.sql` for installations using Oracle 12cR1 (12.1.0.2, 64-bit) script for examples on how to configure a database user password profile:

Enter the following command to create the profile:

```
Create Profile ACMPROFILE LIMIT PASSWORD_LIFE_TIME UNLIMITED;
```

Important: RSA does not require that you include the Oracle SYS user in the profile. The Oracle SYS user password, therefore, will expire at some point. RSA recommends you do the following: change the password in the database and the application, reset the password to its current value, or include the SYS user in the profile.

Create User Schemas

See the `aveksa_sample_ora12_db_scripts.sql` script for installations using Oracle 12cR1 (12.1.0.2) for examples on how to configure user schemas:

Create the following user schemas:

AVUSER (the RSA Identity Governance and Lifecycle user)

AVDWUSER (the RSA Identity Governance and Lifecycle reporting engine)

ACMDB (the RSA Identity Governance and Lifecycle public database schema)

If you are not using the default user schema names (AVUSER and others), substitute your user schema names in the scripts provided. Observe Oracle password guidelines when creating the user schemas. For example, although Oracle accepts a '\$' character in a password, it does not recommend that you use the character. See Oracle's password guidelines for more information .

Note: Additional configuration is required when installing RSA Identity Governance and Lifecycle with non-default schema names.

Command examples:

```
Create USER AVUSER identified by <password> profile ACMPROFILE;
ALTER USER AVUSER DEFAULT TABLESPACE DATA_1M TEMPORARY TABLESPACE
TEMP;
Create USER AVDWUSER identified by <password> profile ACMPROFILE;
```

```
ALTER USER AVDWUSER DEFAULT TABLESPACE DATA_1M TEMPORARY  
TABLESPACE TEMP;
```

```
Create USER ACMDB identified by <password> profile ACMPROFILE;
```

```
ALTER USER ACMDB DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE  
TEMP ACCOUNT UNLOCK;
```

Record the passwords in the worksheet provided in [Database Parameter Values Worksheet](#). The passwords are required for the RSA Identity Governance and Lifecycle installation.

Configure the User Schema Privilege Grants

See the `aveksa_sample_oracle12_db_scripts.sql` script for installations using Oracle 12cR1 (12.1.0.2) for examples on how to configure the AVUSER, AVDWUSER, and ACMDB user schemas.

Privilege grants:

- AVUSER

```
grant unlimited tablespace to AVUSER;
```

```
grant create session to AVUSER; (Used for Application server access)
```

```
grant create table to AVUSER; (Database object used for RSA Identity Governance  
and Lifecycle runtime and migration)
```

```
grant create view to AVUSER; (Database object used for RSA Identity Governance  
and Lifecycle runtime and migration)
```

```
grant create trigger to AVUSER; (Database object used for RSA Identity  
Governance and Lifecycle runtime and migration)
```

```
grant create sequence to AVUSER; (Database object used for RSA Identity  
Governance and Lifecycle runtime and migration)
```

```
grant create synonym to AVUSER; (Database object used for RSA Identity  
Governance and Lifecycle runtime and migration)
```

```
grant create procedure to AVUSER; (Database object used for RSA Identity  
Governance and Lifecycle runtime and migration)
```

```
grant create type to AVUSER; (Database object used for RSA Identity Governance  
and Lifecycle runtime and migration)
```

```
grant create job to AVUSER; (Database object used for RSA Identity Governance  
and Lifecycle runtime and migration)
```

```
grant read, write on directory AVEKSA_EXPORTIMPORT_DIRECTORY to  
AVUSER; (Optional. Used for import/export of Oracle database dumps.)
```

```
grant select on dba_free_space to AVUSER; (Provides database information for
Statistics Reports)
```

```
grant select on gv_$parameter to AVUSER; (Provides database information for
Statistics Reports)
```

```
grant execute ON XDB.DBMS_XMLPARSER TO AVUSER; (Used to process XML data
attributes and documents)
```

```
grant execute ON SYS.DBMS_XMLGEN TO AVUSER; (Used to process XML data
attributes and documents)
```

```
grant execute ON SYS.DBMS_LOB TO AVUSER; (Used to process large sql
statements)
```

```
grant execute ON SYS.DBMS_SQL TO AVUSER; (Used to process large sql
statements)
```

```
grant execute ON SYS.DBMS_SCHEDULER TO AVUSER; (Data model manipulation
for custom attributes)
```

The following four grants allow AVUSER to access the tables necessary for XA recovery and to avoid the "XA exception javax.transaction.xa.XAException, XAException.XAER_RMERR" message when the datasource is configured for XA recovery:

```
grant select on sys.dba_pending_transactions to AVUSER;
```

```
grant select on sys.pending_trans$ to AVUSER;
```

```
grant select on sys.dba_2pc_pending to AVUSER;
```

```
grant execute on sys.dbms_xa to AVUSER;
```

- AVDWUSER

```
grant unlimited tablespace to AVDWUSER; (Optional)
```

```
grant create session to AVDWUSER;
```

```
grant create synonym to AVDWUSER;
```

```
grant create procedure to AVDWUSER;
```

- ACMDB

```
grant unlimited tablespace to ACMDB; (Optional)
```

```
grant create session to ACMDB;
```

```
grant create synonym to ACMDB;
```

```
grant create procedure to ACMDB;
```

Note: Object privileges must be granted directly to the user. Use of Oracle roles for privileges may not be sufficient for DBMS objects.

Additional grant information:

- Avekxa leverages the SYS.UTL_FILE Oracle package. Permissions to this package must be granted to AVUSER. The package is used as part of the collection processes. If permissions are revoked for this package items such as the collections will fail.
- Avekxa Statistics Reports requires grants on some packages to obtain useful diagnostic information. These are dba_free_space, gv_\$parameter, and SYS.UTL_INADDR.
- Grants to execute the packages SYS.UTL_FILE and SYS.DBMS_LOB by the Oracle schema XDB should also be verified as they are a requirement of the XML Database functionality of Oracle. On some systems the packages DBMS_XMLPARSER and DBMS_XSLPROCESSOR must be recompiled following the new grants.

```
grant execute on SYS.UTL_FILE to XDB;
```

```
grant execute on SYS.DBMS_LOB to XDB;
```

```
alter PACKAGE xdb.DBMS_XMLPARSER compile body;
```

```
alter PACKAGE xdb.DBMS_XSLPROCESSOR compile body;
```

- A grant to execute the XDB.DBMS_XMLPARSER package is required by AVUSER for XML processing and metadata operations.

```
grant execute ON XDB.DBMS_XMLPARSER TO AVUSER;
```

Create a Report Context

The report context holds the filtering values for RSA Identity Governance and Lifecycle reports.

Procedure:

```
create or replace context AV_REPORT_CONTEXT using  
AVUSER.Reporting_Pkg;
```

Complete the Required Database Instance Configuration

See avekxa_sample_sys_scripts.sql for a sample script on how to perform this step. This script must be edited for your configuration before you use it. Complete database instance configuration as sysdba.

Procedure

Disable the daily job that gathers schema statistics. The RSA database executes its own database statistics collections and can conflict with this out-of-box Oracle setting.

```
execute dbms_auto_task_admin.disable(client_name => 'auto optimizer stats
collection', operation => NULL, window_name => NULL);
```

To verify, execute the following SQL:

```
SELECT client_name, status FROM dba_autotask_client where client_name =
'auto optimizer stats collection';
```

Confirm that the job is disabled as show in the example output:

```
CLIENT_NAME STATUS
-----
auto optimizer stats collection DISABLED
```

Installing Oracle Statspack to Enhance Database Diagnostics Capabilities

Oracle Statspack collects and compiles performance and execution statistics data that RSA Identity Governance and Lifecycle can include in reports generated by its Aveksa Statistics Report feature.

Installation of Statspack is optional but highly recommended; it provides diagnostics data that can indicate the causes of database performance issues that may arise.

You install Statspack on the database using the `screate.sql` script provided by Oracle. This script is executed as 'sys' on the database. The script prompts you for information and creates the STATSPACK schema owner, privilege grants, and objects. See Oracle documentation for complete information on Statspack installation and Statspack capabilities.

For example, to run the script in SQL*Plus:

Create a “perfstat” table space using the `screate.sql` script for Oracle 12.1.0.2.

```
SQL> connect / as sysdba
SQL> @?/rdbsms/admin/spcreate
```

The default username is “perfstat.” See the *Installation Guide* for information on changing the username and the Statspack user password in RSA Identity Governance and Lifecycle.

RSA recommends that the perfstat user is also configured with the same ACMPROFILE as configured for database users (as described in [Create a Database User Password Profile](#)), to prevent issues regarding password expiration.

As sys dba, execute the sql:

```
alter user perfstat profile ACMPROFILE;
```

Verify Correct Database Configuration

Use the following commands to verify that the database used by RSA Identity Governance and Lifecycle has been configured correctly:

Verify that the import/export described in [Deployment Summary](#) is defined:

```
select * from all_directories where directory_name in ('AVEKSA_
EXPORTIMPORT_DIRECTORY');
```

Verify that the tablespaces described in [Create Tablespaces](#) exist:

```
SELECT * FROM USER_TABLESPACES WHERE TABLESPACE_NAME IN ('DATA_
256K', 'DATA_1M', 'DATA_25M', 'DATA_50M', 'INDX_256K', 'INDX_
1M', 'INDX_25M',
'INDX_50M');
```

Validate the XML package exists:

```
select username from all_users where username='XDB';
```

Validate the schemas described in exist:

```
select * from all_users where username in ('AVUSER', 'AVDWUSER',
'ACMDB');
```

Verify the timezone settings within the database. As sys dba, execute the following SQL statements:

1. SELECT DBTIMEZONE FROM DUAL;
2. SELECT avuser.Utilities_Pkg.Get_DBTimezone_Value FROM DUAL;

If the values from those two queries are not exactly the same, execute the following SQL statements:

1. alter database set time_zone='<value you got from the previous second query>';
2. shutdown immediate;
3. startup;

Verify that the value has in fact been updated to the value you have specified by executing the SQL statement:

```
SELECT DBTIMEZONE FROM DUAL;
```

Updating the Database for RSA Identity Governance and Lifecycle Upgrades

If you are upgrading from a pre-5.0 version of RSA Identity Governance and Lifecycle, you must update these additional privileged grants (required for Aveksa Statistics Report generation) for AVUSER:

```
grant select on dba_free_space to AVUSER;
grant select on gv_$parameter to AVUSER;
```

Chapter 3: Maintaining the Database

- [Back Up the Customer-Supplied Database](#)
- [Importing AVUSER Schema/Data for a Customer-Supplied Database Restoration/Load](#)
- [Removing User Schemas from the Database](#)

Back Up the Customer-Supplied Database

It is often useful to back up your database regularly or before making major changes to your system. You perform the backup procedure for the Oracle database on an database server machine using Oracle's expdp data pump utility. It creates a .dmp file of the AVUSER schema.

The dump is essentially a snapshot of the database containing all of the application data and some environment data about a particular system environment. In the case where you intend to import a dump from one machine to another (which must be running the same RSA Identity Governance and Lifecycle version from which the dump was created), you may be required to perform additional configuration on the target machine.

Important: The dumps are upwardly compatible between Oracle versions. They are not backwards compatible when used to import to an older version of Oracle.

Before You Begin

- Ensure that the AVEKSA_EXPORTIMPORT_DIRECTORY directory has been created as described in [Set Up the Database](#)
- Ensure that the database process has Read-Write permissions to the directory.

Procedure

1. Shut down the RSA Identity Governance and Lifecycle server before you export a database.
2. Shut down AFX if it is installed.
3. Run the following command from the database server machine:

```
expdp avuser/<password>@<Oracle_SID> DumpFile=<FileName>.dmp  
Directory=AvekSA_ExportImport_Directory Schemas=avuser  
LogFile=<FileName>.log
```

Where:

Expdp is the Oracle data pump utility.

avuser/<password> is the connection string.

`DumpFile` is the output file name; here set with a date stamp.

`Directory` is an internal Oracle directory object mapped to a physical UNIX directory. It would typically be the `AveksaExportImportDir` directory created when the customer-provided database was set up.

`Schemas` is the database, `avuser` for example.

`LogFile` is the name of the log file generated for the export.

Important: Starting with the 7.0.1 product release, a new data encryption handling mechanism is in place which uses on disk (outside of the DB) data in conjunction with DB data to perform data encryption. When exporting DB data and moving it to a new installation it is no longer sufficient just to export/"pack up" the DB data and reimport it. The encryption key data stored on disk must also be "packed up" (see Step 2) and moved/reinstalled along with the DB data.

4. Zip up the master key data in the master key storage directory. This directory is identified by the `rsavialg.security.keydir` environment variable. The default directory is `/home/oracle/security`. You will unzip the master key data as part of the steps to import the database described in [Importing AVUSER Schema/Data for a Customer-Supplied Database Restoration/Load](#).

Note: In a clustered environment, if separate copies of the key data are stored on each cluster node, then only one copy of a key data (any node's key data) needs to be backed up as all areas should contain the same key data. However, when reinstalling the data, if a node has a local directory specified for storage of keys, then the key data should be reinstalled to each of these local directories (as specified by the `rsavialg.security.keydir` environment variable which each node has set).

Importing AVUSER Schema/Data for a Customer-Supplied Database Restoration/Load

You can import a AVUSER schema/data back up for restoring a database or loading it on a new machine. You perform the import procedure for the RSA Identity Governance and Lifecycle Oracle database on the database server machine using Oracle's `impdp` data pump utility. It uses a `.dmp` file of the AVUSER schema created from the export process. If you intend to import a dump from one machine to another, you may be required to perform additional configuration on the target machine. The target machine must be running the same RSA Identity Governance and Lifecycle version as the version from which the dump was created.

The server nodes may need to be updated; this is particularly true when moving clustered environments. Or directory specific locations configured for collectors may need to change. After you have imported the database, you must validate that the data is compatible with your database. for instructions. See "Validate Compatibility of the Database Import" in the *Database Setup and Management Guide* for instructions.

Important: The dumps are upwardly compatible between Oracle versions. They are not backwards compatible when used to import to an older version of Oracle.

Note: Ensure that the AVEKSA_EXPORTIMPORT_DIRECTORY directory has been created as described in "Set Up the Database" in the *Database Setup and Management Guide* and the database process has Read-Write permissions to the directory.

Procedure

1. Shut down RSA Identity Governance and Lifecycle.

Important: Starting with the 7.0.1 product release, a new data encryption handling mechanism is in place which uses on disk (outside of the DB) data in conjunction with DB data to perform data encryption. When exporting DB data and moving it to a new installation it is no longer sufficient just to export/"pack up" the DB data and reimport it. The encryption key data stored on disk must also be "packed up" and moved/reinstalled along with the DB data (see Step 2).

2. Unzip the master key data in the master key storage directory.

This is the data that was zipped up in the procedure described in "Export the AVUSER Schema/Data for an RSA Database Backup." (The master key storage directory is identified by the `rsavialg.security.keydir` environment variable. The default directory is `/home/oracle/security`.)

Note: In a clustered environment, if separate copies of the key data are stored on each cluster node, then only one copy of a key data (any node's key data) needs to be backed up as all areas should contain the same key data. However, when reinstalling the data, if a node has a local directory specified for storage of keys, then the key data should be reinstalled to each of these local directories (as specified by the `rsavialg.security.keydir` environment variable which each node has set).

3. Remove the avuser user from the database:

```
drop user AVUSER cascade;
```

4. Create the avuser user:

```
Create USER AVUSER identified by <password> profile ACMPROFILE;  
ALTER USER AVUSER DEFAULT TABLESPACE DATA_1M TEMPORARY TABLESPACE TEMP;
```

5. Specify avuser privilege grants as specified .

6. Import the schema/data:

```
impdp avuser/<password>@<Oracle_SID> DumpFile=<SomeFileName>.dmp  
Directory=AvekSA_ExportImport_Directory Schemas=avuser  
LogFile=<SomeFileName>.log
```

7. If the database does not require migration, then run these commands as avuser to refresh database statistics:

```
EXEC DBMS_STATS.GATHER_SCHEMA_STATS ('AVUSER') ;  
EXEC DATABASE_STATISTICS.AFTER_IMPORT ;
```

If the database does require migration, you will be prompted to migrate the database when you access RSA Identity Governance and Lifecycle.

8. Restart RSA Identity Governance and Lifecycle.

Validate Compatibility of the Database Import

After you import the database on your system, determine whether the database dump is compatible for your database. Determine whether these system settings from the imported dump are set as follows:

- isAppliance = No
- isRemoteDB = Yes
- isSoftAppliance = Yes

To determine these values, run the following SQL as avuser:

```
select * from T_SYSTEM_SETTINGS where PARAMETER like 'is%';
```

If the values above are not set to the correct values, run the following SQL to set them to the correct values:

```
update T_SYSTEM_SETTINGS set VALUE='N' where PARAMETER='isAppliance';  
update T_SYSTEM_SETTINGS set VALUE='yes' where PARAMETER='isRemoteDB';  
update T_SYSTEM_SETTINGS set VALUE='yes' where PARAMETER='isSoftAppliance';
```

If the exported database file is from an appliance and it is imported into a remote database, ensure that the system setting for “RemoteDB” on the target system that uses the remote database is enabled. After saving the RemoteDB setting you must restart the RSA Identity Governance and Lifecycle application server.

Removing User Schemas from the Database

All RSA Identity Governance and Lifecycle data is contained in three user schemas. You can, as the database administrator, remove these schemas required if you want to create a new database instance.

Procedure

```
drop user AVUSER cascade;  
drop user AVDWUSER cascade;  
drop user ACMDB cascade;  
drop user PERFSTAT cascade;
```

Note: Drop that user name for your installation if it differs from the default “perfstat” user name used when statspack package was set up on the database.

You can now proceed to create the database as described in [Set Up the Database](#).