# Oracle® Retail Merchandising System

Installation Guide
Release 12.0.12
E29467-03

May 2013

ORACLE

Oracle® Retail Merchandising System Installation Guide, Release 12.0.12



Primary Author: Wade Schwarz

Contributors: Kris Lange

## Value-Added Reseller (VAR) Language

### Oracle Retail VAR Applications

The following restrictions and provisions only apply to the programs referred to in this section and licensed to you. You acknowledge that the programs may contain third party software (VAR applications) licensed to Oracle. Depending upon your product and its version number, the VAR applications may include:

(i) the **MicroStrategy** Components developed and licensed by MicroStrategy Services Corporation (MicroStrategy) of McLean, Virginia to Oracle and imbedded in the MicroStrategy for Oracle Retail Data Warehouse and MicroStrategy for Oracle Retail Planning & Optimization applications.

(ii) the **Wavelink** component developed and licensed by Wavelink Corporation (Wavelink) of Kirkland, Washington, to Oracle and imbedded in Oracle Retail Mobile Store Inventory Management.

(iii) the software component known as **Access Via™** licensed by Access Via of Seattle, Washington, and imbedded in Oracle Retail Signs and Oracle Retail Labels and Tags.

(iv) the software component known as **Adobe Flex™** licensed by Adobe Systems Incorporated of San Jose, California, and imbedded in Oracle Retail Promotion Planning & Optimization application.

You acknowledge and confirm that Oracle grants you use of only the object code of the VAR Applications. Oracle will not deliver source code to the VAR Applications to you. Notwithstanding any other term or condition of the agreement and this ordering document, you shall not cause or permit alteration of any VAR Applications. For purposes of this section, "alteration" refers to all alterations, translations, upgrades, enhancements, customizations or modifications of all or any portion of the VAR Applications including all reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

# Contents

# Send Us Your Comments

Oracle Retail Merchandising System, Installation Guide, Release 12.0.12

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

> **Note:** Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

# Preface

Oracle Retail Installation Guides contain the requirements and procedures that are necessary for the retailer to install Oracle Retail products.

## Audience

This Installation Guide is written for the following audiences:

- Database administrators (DBA)
- System analysts and designers
- Integrators and implementation staff

## Related Documents

For more information, see the following documents in the Oracle Retail Merchandising System Release 12.0.12 documentation set:

- *Oracle Retail Merchandising System Release Notes*
- *Oracle Retail Sales Audit User Guide*
- *Oracle Retail Trade Management User Guide*

## Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

https://support.oracle.com

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

## Review Patch Documentation

When you install the application for the first time, you install either a base release (for example, 12.0) or a later patch release (for example, 12.0.12). If you are installing the base release, additional patch, and bundled hot fix releases, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch and bundled hot fix releases can contain critical information related to the base release, as well as information about code changes since the base release.

## Oracle Retail Documentation on the Oracle Technology Network

Documentation is packaged with each Oracle Retail product release. Oracle Retail product documentation is also available on the following Web site:
http://www.oracle.com/technology/documentation/oracle_retail.html

(Data Model documents are not available through Oracle Technology Network. These documents are packaged with released code, or you can obtain them through My Oracle Support.)

Documentation should be available on this Web site within a month after a product release.

## Conventions

**Navigate:** This is a navigate statement. It tells you how to get to the start of the procedure and ends with a screen shot of the starting point and the statement "the Window Name window opens."

```
This is a code sample
    It is used to display examples of code
```

# 1

# Preinstallation Tasks

**Note:** The RMS 12.0.12 patch contains all the bundled hot fix releases since RMS 12.0.11, plus "delta" defect fixes since the last bundled hot fix. The following defect fixes make up the RMS 12.0.12 patch:

- 12.0.11.1
- 12.0.11.2
- 12.0.11.3
- 12.0.11.4
- 12.0.11.5
- 12.0.11.6
- 12.0.11.7
- 12.0.12.delta (defect fixes since bundled hot fix release 12.0.11.7)

With this release, patching is more flexible because the patch can be started at any hot fix level. For example, if Release 12.0.11.5 has already been applied, then it is only necessary to apply 12.0.11.6, 12.0.11.7 and 12.0.12.delta defect fixes.

This installation guide explains how to install all bundled hot fix releases plus the delta defect fixes.

# Check Database Server Requirements

General Requirements for a database server running RMS include:

| Supported on: | Versions Supported: |
|---|---|
| Database Server OS | OS certified with Oracle Database 10gR2 Enterprise Edition. Options are:<br>▪ AIX 5.3, AIX 6.1 (Actual hardware or LPARs)<br>▪ Solaris 10 SPARC (Actual hardware or Logical Domains)<br>▪ HP-UX 11.23 (PA-RISC)<br>▪ Oracle Linux 4 Update 4 for x86-64<br>▪ Red Hat Enterprise Linux 4 Update 4 for x86-64<br>▪ Oracle Linux 5 for x86-64 (Actual hardware or Oracle virtual machine).<br>▪ Red Hat Enterprise Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). |
| Database Server 10gR2 | Oracle Database 10g Release 2 Enterprise Edition (minimum 10.2.0.5 patchset required) with the following patches and components:<br>**Patches:**<br>▪ 10.2.0.5 patchset (8202632)<br>**Components:**<br>▪ Oracle Database 10g<br>▪ Oracle Partitioning<br>▪ Oracle Net Services<br>▪ Oracle Call Interface (OCI)<br>▪ Oracle Programmer<br>▪ Oracle XML Development Kit<br>▪ Optional Database Vault<br>**Other components:**<br>▪ Perl 5.0 or later<br>▪ X-Windows interface<br>▪ ANSI compiant C-compiler (certified with OS and database version). |
| Database Server OS | OS certified with Oracle Database 11gR2 Enterprise Edition. Options are:<br>▪ AIX 5.3, AIX 6.1 (Actual hardware or LPARs)<br>▪ Solaris 10 SPARC (Actual hardware or Logical Domains)<br>▪ Oracle Linux 5 for x86-64 (Actual hardware or Oracle virtual machine).<br>▪ Red Hat Enterprise Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). |

| Supported on: | Versions Supported: |
|---|---|
| Database Server 11gR2 | Oracle Database 11g Release 2 (11.2.0.2) Enterprise Edition with the following oneoff patches:<br>▪ 10170431 - CTWR CONSUMING LOTS OF CPU CYCLES<br>Apply the following patch to RDBMS home if ASM is used.<br>▪ 11808931 - MERGE REQUEST ON TOP OF 11.2.0.2.0 FOR BUGS 10410054 10422126<br>**Components:**<br>▪ Oracle Partitioning<br>▪ Optional Database Vault<br>▪ Examples CD<br>**Other components:**<br>▪ Perl 5.0 or later<br>▪ X-Windows interface<br>▪ ANSI compliant C-compiler (certified with OS and database version). |

## Check Application Server Requirements

General requirements for an application server capable of running RMS include:

| Supported on: | Versions Supported: |
|---|---|
| Application Server OS | OS certified with Oracle Application Server 10g 10.1.2.3. Options are:<br>▪ AIX 5.3, AIX 6.1 (Actual hardware or LPARs)<br>▪ Solaris 10 SPARC (Actual hardware or Logical Domains)<br>▪ HP-UX 11.23 (PA-RISC)<br>▪ Oracle Linux 4 Update 4 for x86-64<br>▪ Red Hat Enterprise Linux 4 Update 4 for x86-64<br>▪ Oracle Linux 5 for x86-64 (Actual hardware or Oracle virtual machine).<br>▪ Red Hat Enterprise Linux 5 for x86-64 (Actual hardware or Oracle virtual machine). |
| Application Server | Oracle Application Server Forms and Reports 10g version 10.1.2.3 with the following patches:<br>▪ 4601861 (NEED TO EXPOSE NZOS_SETIOSEMANTICS) |

## Check Web Browser and Client Requirements

General requirements for client running RMS include:

| Product | Version |
|---|---|
| JRE Plug-in | ORACLE (Sun) JRE Plug-in 1.4.1+ or<br>5.x Update 11 (1.5.0_11) or 6.x Update 12 (1.6.0_12) |
| Operating System | Windows 2000 or XP |
| Processor | Pentium processor |
| Display resolution | 1024x768 resolution |
| PC Configuration | minimum 256 MB RAM, 450 MHz |
| Browser | Microsoft Internet Explorer 7.0 or 8.0 |

## Supported Oracle Retail Products

**Note:** RMS is dependent on RPM database objects and stored procedures for initial item pricing and requires that this portion of RPM is always deployed with RMS. Without this, RMS would require customization and Oracle Retail does not provide guidance for this type of implementation. In addition to initial price there are other areas where dependencies exist such as vendor funded markdowns, vendor funded promotions, and margin visibility.

| Product | Version |
|---|---|
| Oracle Retail Price Management (RPM) | 12.0.12 |
| Oracle Retail Allocation | 12.0.12 |
| Oracle Retail Invoice Matching (ReIM) | 12.0.12 |
| Oracle Retail Store Inventory Management (SIM) | 12.0.12 |
| Oracle Retail Warehouse Management System (RWMS) | 12.0.12 |
| Oracle Retail Data Warehouse (RDW) | 12.0.10 |
| Oracle Retail Strategic Store Solutions (ORSSS) | 12.0.12 |
| Oracle Retail Demand Forecasting (RDF) | 12.1.3 |
| Oracle Retail Grade | 12.1.3 |
| Oracle Retail Predictive Applications Server (RPAS) | 12.1.3 |

## Supported Oracle Retail Integration Technologies

| Integration Technology | Version |
|---|---|
| Oracle Retail Extract, Transform and Load (RETL) | 12.0.11 |
| Oracle Retail Integration Bus (RIB) | 12.0.12 |
| Oracle Retail Service Layer (RSL) | 12.0.12 |

# Supported Oracle Applications

| Requirement | Version |
|---|---|
| Oracle E-Business Suite | 11.5.10 or 12.0.2 |

# 2

# RAC and Clustering

Real Application Cluster Database and Oracle Application Server Clustering for Oracle Retail Merchandising System has been validated to run only on Linux:

The Oracle Retail products have been validated against a 10.2.0.3 RAC database. When using a RAC database, all JDBC connections should be configured to use OCI connections rather than THIN connections. It is suggested that when using OCI connections, the Oracle Retail products database be configured in the tnsnames.ora file used by the Oracle Application Server installations.

Clustering for Oracle Application Server 10.1.3 is managed as an Active-Active cluster accessed through a hardware Load Balancer. It is suggested that a VirtualHost be added to the OAS 10.1.3 reflecting the Virtual Server Name configured in the load balancer. It is also suggested that the OC4J select method be configured to prefer the use of local OC4J instances. The Oracle Retail products are currently not validated to be distributable at the application level in an OAS 10.1.3 cluster.

Clustering for Oracle Application Server 10.1.2.2 is managed as an Active-Active cluster accessed through a hardware Load Balancer. It is suggested that the Web Cache installation included with OAS 10.1.2.2 be configured to reflect all application server Mid-Tier installations. Validation has been completed utilizing a RAC 10.2.0.3 Oracle Internet Directory database with the OAS 10.1.2.2 cluster.

**References for Configuration:**

- Oracle® Application Server High Availability Guide 10g Release 3 (10.1.3) Part Number B15977-02
- Oracle® Application Server High Availability Guide 10g Release 2 (10.1.2) Part Number B14003-05
- Oracle® Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide 10g Release 2 (10.2) Part Number B14197-03

# 3

# Database Installation Tasks

Before you apply the RMS 12.0.12 patch:

- Make a backup of all your objects and database schema.
- Check that RMS 12.0.11 is installed.
- Review the enclosed RMS 12.0.12 Patch Release Notes (rms-12012-rn.pdf).
- Review each of the enclosed defect documents.
- Note: The RMS 12.0.12 patch contains all the bundled hot fix releases plus the delta from the last patch to this release. The following make up the RMS 12.0.12 patch
  - 12.0.11.1
  - 12.0.11.2
  - 12.0.11.3
  - 12.0.11.4
  - 12.0.11.5
  - 12.0.11.6
  - 12.0.11.7
  - 12.0.12.delta (delta between 12.0.11.7 and 12.0.12 release)

Patching is made more flexible by allowing the ability to start the patch at any hot fix level. For example, if 12.0.11.5 has already been applied in the environment, then applying 12.0.11.6, 12.0.11.7 and 12.0.12.delta is only necessary.

The installation instructions explain how to install all bundled hot fixes plus the delta.

Before copying over any files:

- Note whether customizations have been made to the module. If so, then the customizations must be reapplied over the new version of the module (or the fix may need to be applied to the custom version of the code).
- Copy the original files to a different directory before copying over them in case they need to be referred to at a later date.

> **Note:** These instructions refer to RMS12DEV as the Oracle owning schema.

## Copy from CD Directory

1. Copy the rms12012patch.zip file from the CD / to a newly created staging directory on your UNIX server.
2. Unzip the file by entering:

   ```
   unzip rms12012patch.zip
   ```

# Apply RMS 12.0.11.1 Bundled Hot Fix Release

Run the following scripts only if the RMS 12.0.11.1 bundled hot fix release has not been installed in the environment:

## Alter RIB Objects

1. Change directories to STAGING_AREA/rms12.0.11.1/rib_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> patch120111rib.sql`
3. Check the log file patch120111rib.log for any errors.

## Alter RMS Tables

1. Change directories to STAGING_AREA/rms12.0.11.1/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120111dbcs.sql`
3. Check the log file patch120111dbcs.log for any errors.

## Alter RMS Triggers

1. Change directories to STAGING_AREA/rms12.0.11.1/triggers.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120111triggers.sql`
3. Check the log file patch120111triggers.log for any errors.

## Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.1/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120111rms.sql`
3. Check the log file patch120111rms.log for any errors. Validating the invalid objects will be completed at the end.

## Update Data for RMS

**Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.1/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120111ctl.sql`
3. Check the log file patch120111ctl.log for any errors.

## Alter Pricing Tables

1. Change directories to STAGING_AREA/rms12.0.11.1/pricing/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @pricing120111dbcs.sql`
3. Check the log file pricing120111dbcs.log for any errors.

### Alter Pricing Objects

1. Change directories to STAGING_AREA/rms12.0.11.1/pricing/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120111rms.sql
   ```
3. Check the log file pricing120111rms.log for any errors.

## Apply RMS 12.0.11.2 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1 has been installed and RMS 12.0.11.2 bundled hot fix release has not been installed in the environment.

### Alter RMS Tables

1. Change directories to STAGING_AREA/rms12.0.11.2/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112dbcs.sql
   ```
3. Check the log file patch120112dbcs.log for any errors.

### Alter RMS Triggers

1. Change directories to STAGING_AREA/rms12.0.11.2/triggers.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112triggers.sql
   ```
3. Check the log file patch120112triggers.log for any errors.

### Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.2/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112rms.sql
   ```
3. Check the log file patch120112rms.log for any errors. Validating the invalid objects will be completed at the end.

### Update Data for RMS

> **Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.2/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112ctl.sql
   ```
3. Check the log file patch120112ctl.log for any errors.

## Insert Language Data

> **Note:** These scripts are only for customers who have a primary or secondary language of any of the following: Korean, Russian. The scripts are UTF-8 encoded. We recommend installing them into a database that has been set to UTF-8.

**If you have a primary or secondary language of Korean:**

1. Change directories to INSTALL_DIR/rms12.0.11.2/data/KO.
2. Set the SQL*Plus session so that the encoding component of the NLS_LANG is UTF8. For example,AMERICAN_AMERICA.UTF8
3. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112ctl_KO.sql
   ```
4. Check the log file patch120112ctl_KO.log for any errors.

**If you have a primary language of Russian:**

1. Change directories to INSTALL_DIR/rms12.0.11.2/data/RU.
2. Set the SQL*Plus session so that the encoding component of the NLS_LANG is UTF8. For example,AMERICAN_AMERICA.UTF8
3. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112ctl_RU_prim.sql
   ```
4. Check the log file patch120112ctl_RU_prim.log for any errors.

**If you have a secondary language of Russian:**

1. Change directories to INSTALL_DIR/rms12.0.11.2/data/RU.
2. Set the SQL*Plus session so that the encoding component of the NLS_LANG is UTF8. For example,AMERICAN_AMERICA.UTF8
3. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120112ctl_RU_sec.sql
   ```
4. Check the log file patch120112ctl_RU_sec.log for any errors.

## Alter Pricing types

1. Change directories to STAGING_AREA/rms12.0.11.2/pricing/types.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120112types.sql
   ```
3. Check the log file pricing120112types.log for any errors.

## Alter Pricing Tables

1. Change directories to STAGING_AREA/rms12.0.11.2/pricing/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120112dbcs.sql
   ```
3. Check the log file pricing120112dbcs.log for any errors.

### Alter Pricing Objects

1. Change directories to STAGING_AREA/rms12.0.11.2/pricing/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120112rms.sql
   ```
3. Check the log file pricing120112rms.log for any errors.

## Apply RMS 12.0.11.3 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1 and RMS 12.0.11.2 have been installed and RMS 12.0.11.3 bundled hot fix release has not been installed in the environment:

### Alter RIB Objects

1. Change directories to STAGING_AREA/rms12.0.11.3/rib_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> patch120113rib.sql
   ```
3. Check the log file patch120113rib.log for any errors.

### Alter RMS Tables

1. Change directories to STAGING_AREA/rms12.0.11.3/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120113dbcs.sql
   ```
3. Check the log file patch120113dbcs.log for any errors.

### Alter RMS Triggers

1. Change directories to STAGING_AREA/rms12.0.11.3/triggers.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120113triggers.sql
   ```
3. Check the log file patch120113triggers.log for any errors.

### Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.3/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120113rms.sql
   ```
3. Check the log file patch120113rms.log for any errors. Validating the invalid objects will be completed at the end.

### Update Data for RMS

> **Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.3/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120113ctl.sql
   ```
3. Check the log file patch120113ctl.log for any errors.

### Alter Pricing Tables

1. Change directories to STAGING_AREA/rms12.0.11.3/pricing/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @pricing120113dbcs.sql`
3. Check the log file pricing120113dbcs.log for any errors.

### Alter Pricing Objects

1. Change directories to STAGING_AREA/rms12.0.11.3/pricing/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @pricing120113rms.sql`
3. Check the log file pricing120113rms.log for any errors.

## Apply RMS 12.0.11.4 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1, RMS 12.0.11.2, and RMS 12.0.11.3 have been installed and RMS 12.0.11.4 bundled hot fix release has not been installed in the environment:

### Alter RMS Tables

1. Change directories to STAGING_AREA/rms12.0.11.4/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120114dbcs.sql`
3. Check the log file patch120114dbcs.log for any errors.

### Alter RMS Triggers

1. Change directories to STAGING_AREA/rms12.0.11.4/triggers.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120114triggers.sql`
3. Check the log file patch120114triggers.log for any errors.

### Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.4/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120114rms.sql`
3. Check the log file patch120114rms.log for any errors. Validating the invalid objects will be completed at the end.

### Update Data for RMS

> **Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.4/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch120114ctl.sql`
3. Check the log file patch120114ctl.log for any errors.

# Apply RMS 12.0.11.5 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1, RMS 12.0.11.2, RMS 12.0.11.3, and RMS 12.0.11.4 have been installed and RMS 12.0.11.5 bundled hot fix release has not been installed in the environment:

## Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.5/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120115rms.sql
   ```
3. Check the log file patch120115rms.log for any errors. Validating the invalid objects will be completed at the end.

## Update Data for RMS

> **Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.5/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120115ctl.sql
   ```
3. Check the log file patch120115ctl.log for any errors.

## Alter Pricing Tables

1. Change directories to STAGING_AREA/rms12.0.11.5/pricing/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120115dbcs.sql
   ```
3. Check the log file pricing120115dbcs.log for any errors.

## Alter Pricing Objects

1. Change directories to STAGING_AREA/rms12.0.11.5/pricing/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @pricing120115rms.sql
   ```
3. Check the log file pricing120115rms.log for any errors.

# Apply RMS 12.0.11.6 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1, RMS 12.0.11.2, RMS 12.0.11.3, RMS 12.0.11.4 and RMS 12.0.11.5 have been installed and RMS 12.0.11.6 bundled hot fix release has not been installed in the environment:

## Alter RIB Objects

1. Change directories to STAGING_AREA/rms12.0.11.6/rib_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> patch120116rib.sql
   ```
3. Check the log file patch120116rib.log for any errors.

## Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.6/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120116rms.sql
   ```
3. Check the log file patch120116rms.log for any errors. Validating the invalid objects will be completed at the end.

## Update Data for RMS

**Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.6/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120116ctl.sql
   ```
3. Check the log file patch120116ctl.log for any errors.

# Apply RMS 12.0.11.7 Bundled Hot Fix Release

Run the following scripts only if RMS 12.0.11.1, RMS 12.0.11.2, RMS 12.0.11.3, RMS 12.0.11.4, RMS 12.0.11.5 and RMS 12.0.11.6 have been installed and RMS 12.0.11.7 bundled hot fix release has not been installed in the environment:

## Alter RMS Tables

1. Change directories to STAGING_AREA/rms12.0.11.7/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120117dbcs.sql
   ```
3. Check the log file patch120117dbcs.log for any errors.

## Alter RMS Triggers

1. Change directories to STAGING_AREA/rms12.0.11.7/triggers.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120117triggers.sql
   ```
3. Check the log file patch120117triggers.log for any errors.

## Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.11.7/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120117rms.sql
   ```
3. Check the log file patch120117rms.log for any errors. Validating the invalid objects will be completed at the end.

## Update Data for RMS

> **Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.11.6/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   ```
   SQL> @patch120117ctl.sql
   ```
3. Check the log file patch120117ctl.log for any errors.

# Apply the Delta from RMS 12.0.11.7 to 12.0.12

Run the following scripts only after the RMS 12.0.11.1, RMS 12.0.11.2, RMS 12.0.11.3, RMS 12.0.11.4, RMS 12.0.11.5, RMS 12.0.11.6, and RMS 12.0.11.7 bundled hot fix releases have been installed in the environment:

## Alter RIB Objects

1. Change directories to STAGING_AREA/rms12.0.12.delta/rib_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> patch12012rib.sql`
3. Check the log file patch12012rib.log for any errors.

## Alter RMS Database Objects

1. Change directories to STAGING_AREA/rms12.0.12.delta/db_objects.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch12012rms.sql`
3. Check the log file patch12012rms.log for any errors.

## Update Data for RMS

**Note:** Ensure that you have your NLS_LANG set to UTF-8. For example AMERICAN_AMERICA.UTF8

1. Change directories to STAGING_AREA/rms12.0.12.delta/data.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @patch12012ctl.sql`
3. Check the log file patch12012ctl.log for any errors.

## Alter Pricing Tables

1. Change directories to STAGING_AREA/rms12.0.12.delta/pricing/dbcs.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @pricing12012dbcs.sql`
3. Check the log file pricing12012dbcs.log for any errors.

## Validate all Invalid Objects

**Note:** Deadlocked objects may appear when running this script. This is expected. Run the script until no invalid objects remain.

1. Change directories to STAGING_AREA/utility.
2. Log in to SQL*Plus as RMS12DEV and run the following command:
   `SQL> @inv_obj_comp.sql`
3. This script may need to be run more than once.

# Optional – Configure RMS with Database Vault and Transparent Data Encryption

To encrypt ReSA application data using Transparent Data Encryption (TDE), please refer to Appendix A.

To enable enhance application security for RMS using Database Vault, please refer to Appendix B.

# Update RETL for RDW

**Note:** These steps only need to be done if you are using RDW.

1. Change directories to STAGING_AREA/rms12.0.11.1/retl4rdw/rfx/src.
2. Copy all the files from this directory to INSTALL_DIR/retlforRDW/rfx/src.
   ```
   cp -R * INSTALL_DIR/retlforRDW/rfx/src
   ```
3. Change directories to STAGING_AREA/rms12.0.11.2/retl4RDW/rfx/src.
4. Copy all the files from this directory to INSTALL_DIR/retlforRDW/rfx/src.
   ```
   cp -R * INSTALL_DIR/retlforRDW/rfx/src
   ```
5. Change directories to STAGING_AREA/rms12.0.11.3/retl4RDW/rfx/src.
6. Copy all the files from this directory to INSTALL_DIR/retlforRDW/rfx/src.
   ```
   cp -R * INSTALL_DIR/retlforRDW/rfx/src
   ```
7. Change directories to STAGING_AREA/rms12.0.11.5/retl4RDW/rfx/src.
8. Copy all the files from this directory to INSTALL_DIR/retlforRDW/rfx/src.
   ```
   cp -R * INSTALL_DIR/retlforRDW/rfx/src
   ```
9. Change directories to STAGING_AREA/rms12.0.11.7/retl4rdw/rfx/src.
10. Copy all the files from this directory to INSTALL_DIR/retlforRDW/rfx/src.
    ```
    cp -R * INSTALL_DIR/retlforRDW/rfx/src
    ```

# Compile RMS Batch Libraries and Programs

**Note:** Warning messages may appear during the compilation of the batch. These warnings can be ignored if the batch executables are successfully generated.

## Set Environment Variables

1. Make sure the following variables are set:

   **Note:** INSTALL_DIR is the location where RMS 12 was installed.

   Make sure the path for make, makedepend, and the compiler are in $PATH environment variable.

   - MMHOME=INSTALL_DIR/rms
   - MMUSER=RMS Schema Owner
   - PASSWORD=RMS Schema Owner Password
   - ORACLE_HOME=Location of Oracle install
   - ORACLE_SID=The Oracle Sid for the RMS database

   **AIX:**

   - LIBPATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:$LDPATH
   - OBJECT_MODE=64
   - LINK_CNTRL=L_PTHREADS_D7

   **HP:**

   - SHLIB_PATH=$ORACLE_HOME/lib:$MMHOME/oracle/lib/bin:
   - $SH_LIBPATH

   **Solaris:**

   - LD_LIBRARY_PATH=$ORACLE_HOME/lib:
     $MMHOME/oracle/lib/bin:$LD_LIBRARY_PATH

   **Linux:**

   - LD_LIBRARY_PATH=$ORACLE_HOME/lib:
     $MMHOME/oracle/lib/bin:$LD_LIBRARY_PATH

## Compile Batch Libraries

1. Copy the files from STAGING_AREA/<release version>/batch/lib/src to INSTALL_DIR/rms/oracle/lib/src for all bundled hot fix releases in this order: rms12.0.11.1, rms12.0.11.2, rms12.0.11.3, rms12.0.11.5.
2. Change directories to INSTALL_DIR/rms/oracle/lib/src.
3. Create library dependencies.
   a. For UNIX, run the following command:
      `make -f retek.mk depend 2>&1 | tee libdpnd.log`
   b. For Linux, run the following command:
      `make -f retek.mk -r depend 2>&1 | tee libdpnd.log`
   c. Check the libdpnd.log file for errors.
4. Create the batch libraries.
   a. For UNIX, run the following command:
      `make -f retek.mk retek rms resa 2>&1 | tee libretek.log`
   b. For Linux, run the following command:
      `make -f retek.mk -r retek rms resa 2>&1 | tee libretek.log`
   c. Check the libretek.log file for errors.

5. Install the batch libraries.
   ```
   make -f retek.mk install
   ```
   The batch libraries should now be in INSTALL_DIR/rms/oracle/lib/bin.

## Compile Batch Source Code

1. Copy the files from STAGING_AREA/<release version>/batch/proc/src to INSTALL_DIR/rms/oracle/proc/src for all bundled hot fix releases in this order: rms12.0.11.1, rms12.0.11.2, rms12.0.11.3, rms12.0.11.4, rms12.0.11.5, rms12.0.11.6, rms12.0.11.7, rms12.0.12.delta.
2. Copy all .ksh and .sql files from INSTALL_DIR/rms/oracle/proc/src to INSTALL_DIR/rms/oracle/proc/bin.
3. Change directories to INSTALL_DIR/rms/oracle/proc/src.
4. Create dependencies.
   a. For UNIX, run the following command:
      ```
      make -f mts.mk rms-depend recs-depend rtm-depend resa-depend 2>&1 | tee srcdpnd.log
      ```
   b. For Linux, run the following command:
      ```
      make -f mts.mk -r rms-depend recs-depend rtm-depend resa-depend 2>&1 | tee srcdpnd.log
      ```
   c. Check the srcdpnd.log file for errors.
5. Create batch programs.
   a. For UNIX, run the following commands in the order stated.
      ```
      make -f rms.mk PRODUCT_PROCFLAGS=dynamic=ansi ditinsrt
      make -f mts.mk rms-ALL recs-ALL resa-ALL rtm-ALL 2>&1 | tee srcall.log
      ```
   b. For Linux, run the following commands:
      ```
      make -f rms.mk -r PRODUCT_PROCFLAGS=dynamic=ansi ditinsrt
      make -f mts.mk -r rms-ALL recs-ALL resa-ALL rtm-ALL 2>&1 | tee srcall.log
      ```
   c. Check the srcall.log file for errors.
6. Install the batch programs.
   ```
   make -f mts.mk rms-install recs-install resa-install rtm-install
   ```
   The batch programs should now be in INSTALL_DIR/rms/oracle/proc/bin.

# 4

# Application Server Installation Tasks

## Copy from CD Directory

1. Copy the rms12012patch.zip file from the CD to a newly created staging directory on your UNIX server. This zip file also contains the database files.
2. Unzip the file by entering:
   `unzip rms12012patch.zip`

## Set Environment Variables

> **Note:** ORACLE_HOME is the location where Oracle Application Server 10g (10.1.2.0.2) has been installed

1. Set the DISPLAY variable to the IP address plus ":0.0" (for example: 10.1.1.1:0.0) of the application server.
2. Set the following variables:

   > **Note:** ORACLE_HOME is the location where Oracle Application Server 10g (10.1.2.0.2) has been installed

   - **All OS Platforms**
     - PATH=$ORACLE_HOME/bin:$ORACLE_HOME/opmn/bin:$ORACLE_HOME/dcm/bin:INSTALL_DIR/forms10gr2_scripts:$PATH
     - CLASSPATH=$ORACLE_HOME/jlib/importer:$ORACLE_HOME/jlib/debugger.jar:$ORACLE_HOME/jlib/utj.jar:$ORACLE_HOME/jlib/ewt3.jar:$ORACLE_HOME/jlib/share.jar:$ORACLE_HOME/jlib/dfc.jar:$ORACLE_HOME/jlib/help4.jar:$ORACLE_HOME/jlib/oracle_ice.jar:$ORACLE_HOME/jlib/jewt4.jar
     - FORMS_BUILDER_CLASSPATH=$CLASSPATH
     - FORMS_PATH=INSTALL_DIR/toolset/bin:INSTALL_DIR/rms/forms/bin:$ORACLE_HOME/forms
     - REPORTS_PATH=INSTALL_DIR/rms/reports/bin:$ORACLE_HOME/forms
     - TK_UNKNOWN==$ORACLE_HOME/guicommon/tk/admin
     - UP=<RMS schema owner>/<RMS schema password>@<RMS database>

       > **Note:** Verify that TNS is set up correctly by using the UP variable to successfully log in to the RMS 12 schema.
       >
       > **Example:** /u00/oracle> sqlplus $UP

- **Solaris**
  - LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/jdk/jre/lib/sparc:$ORACLE_HOME/jdk/jre/lib/sparc/native_threads
- **HP-UX**
  - SHLIB_PATH=$ORACLE_HOME/lib32:$ORACLE_HOME/lib:$ORACLE_HOME/jdk/jre/lib/PA_RISC:$ORACLE_HOME/jdk/jre/lib/PA_RISC/server
- **AIX**
  - LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/jdk/jre/lib
  - LIBPATH=$LD_LIBRARY_PATH
- **Linux**
  - LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32:$ORACLE_HOME/jdk/jre/lib

# RMS Toolset Installation

> **Note:** There are no RMS Toolsets included in the 12.0.12 patch. If you have already installed the latest 12.0.11 toolsets you can skip to the next section RMS Forms Installation.

1. Copy all libraries, forms and menus (.pll, .fmb, .mmb files) in the STAGING_AREA/toolset/src directory to the INSTALL_DIR/toolset/bin directory for all bundled hot fix releases.
2. Change directories to INSTALL_DIR/toolset/bin.
3. Run pll2plx10gr2_toolset to compile all Toolset .pll's.

> **Note:** If the pll2plx10gr2_toolset script is not used and the libraries are compiled individually, then they must be compiled in the following order (which is noted in the pll2plx10gr2_toolset script):

- messge45.pll
- ariiflib.pll
- stand45.pll
- calend45.pll
- find45.pll
- item45.pll
- tools45.pll
- mblock45.pll
- mview45.pll
- nav45.pll
- work45.pll
- itnumtype.pll
- hierfilter.pll
- rmslib.pll

4. Check to make sure that each .pll file has a corresponding .plx (to ensure that all .pll's compiled successfully).
5. Remove all newly created .plx files.
6. Run fmb2fmx10gr2_fm (in INSTALL_DIR/toolset/bin) to compile the Toolset reference forms.
7. Remove all newly created fm_*.fmx files (reference forms should not have executable files).
8. Run fmb2fmx10gr2 (in INSTALL_DIR/toolset/bin) to generate Toolset runtime forms – .fmx's.
9. Check to make sure that each non-reference form (.fmb file) has a corresponding .fmx file.

> **Note:** Disregard fm_*.fmx files should they be created. These files should be removed. They should NOT exist in the INSTALL_DIR/toolset/bin directory.

10. Remove all non-reference forms from INSTALL_DIR/toolset/bin; the following syntax leaves all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

```
> for PROG in `ls *.fmb | grep -v fm_`
> do PROGNAME=`echo $PROG`
> rm $PROGNAME
> done
```

11. Run mmb2mmx10gr2 (in INSTALL_DIR/toolset/bin) to generate Toolset runtime menus – .mmx's.
12. Check to make sure that each .mmb file has a corresponding .mmx file.

> **Note:** .err files may be created by the compilation scripts above. These files are logs of the compilation process and can be removed.

13. Remove all .mmb files from INSTALL_DIR/toolset/bin.

## RMS Forms Installation

1. Copy all the files from STAGING_AREA/<release version>/forms/src to INSTALL_DIR/rms/forms/src for all bundled hot fix releases in this order: rms12.0.11.1, rms12.0.11.2, rms12.0.11.3, rms12.0.11.4, rms12.0.11.5, rms12.0.11.6, rms12.0.11.7, rms12.0.12.delta.
2. Copy all libraries (.pll files) in the INSTALL_DIR/rms/forms/src directory to the INSTALL_DIR/rms/forms/bin directory.
3. Change directories to INSTALL_DIR/rms/forms/bin.
4. Run pll2plx10gr2_forms to compile all RMS .pll's.
5. Check to make sure that each .pll file has a corresponding .plx (to ensure that all .pll's compiled successfully). Remove all newly created .plx files.
6. Copy all forms (*.fmb files) in the INSTALL_DIR/rms/forms/src directory to the INSTALL_DIR/rms/forms/bin directory.
7. Run fmb2fmx10gr2_fm (in INSTALL_DIR/rms/forms/bin) to compile the RMS reference forms.
8. Remove all newly created fm_*.fmx files (reference forms should not have executable files).

9. Run fmb2fmx10gr2 (in INSTALL_DIR/rms/forms/bin) to generate RMS runtime forms – .fmx's.
10. Check to make sure that each non-reference form .fmb file has a corresponding .fmx file.

> **Note:** Disregard fm_*.fmx files should they be created. These files should be removed. They should NOT exist in the INSTALL_DIR/rms/forms/bin directory.

11. Remove all non-reference form forms from INSTALL_DIR/rms/forms/bin. The following syntax leaves all reference forms (fm_*.fmb) in the bin directory, while removing all other forms:

```
> for PROG in `ls *.fmb | grep -v fm_`
> do PROGNAME=`echo $PROG`
> rm $PROGNAME
> done
```

12. Copy all menus (*.mmb files) in the INSTALL_DIR/rms/forms/src directory to the INSTALL_DIR/rms/forms/bin directory.
13. Run mmb2mmx10gr2 (in INSTALL_DIR/rms/forms/bin) to generate RMS runtime menus – .mmx's.
14. Check to make sure that each .mmb file has a corresponding .mmx file.
15. Remove all .mmb files from INSTALL_DIR/rms/forms/bin.

> **Note:** .err files may be created by the compilation scripts above. These files are logs of the compilation process and can be removed.

# RMS Reports Installation

1. Copy all reports (*.rdf files) in the STAGING_AREA/<release version>/reports/src directory to the INSTALL_DIR/rms/reports/src directory for all bundled hot fix releases in this order: rms12.0.11.1, rms12.0.11.2.
2. Copy all .rdf files in the INSTALL_DIR/rms/reports/src directory to the INSTALL_DIR/rms/reports/bin directory.
3. Run rdf2rep10gr2 (in INSTALL_DIR/rms/reports/bin) to generate Reports runtime reports – .rep's.

   **Note:** The following error messages may appear when running rdf2rep10gr2; these errors can be ignored if report (.rep) generation was successful:

   REP-0759: One or more PL/SQL libraries have been modified since the reports was saved. The PL/SQL will be recompiled.

   REP-0202: Attempt to free a null pointer

   REP-0759 is generated by the r25conv program. The error appears any time a report is converted.

   REP-0202 is due to an Oracle bug with rwconverter and can be ignored

4. Check to make sure that each .rdf file has a corresponding .rep file.
5. Remove all .rdf files from INSTALL_DIR/rms/reports/bin.

   **Note:** .err files may be created by the compilation scripts above. These files are logs of the compilation process and can be removed.

# A

# Appendix: Setting up Transparent Data Encryption

12.0.12 RMS supports TDE (Transparent Data Encryption) on both 10.2.0.5 and 11.2.0.2.

Encryption can be setup at table column level for 10gR2. Alternatively if the database is on 11gR2, it can be setup at tablespace level. Depending on the requirements of your environment, determine which option to implement. Shown below are the examples of both methods:

## Configure a Wallet

1. Create a sqlnet.ora in $TNS_ADMIN of the database similar to the below entry:

```
ENCRYPTION_WALLET_LOCATION =
  (SOURCE = (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = /u00/oracle/admin/$ORACLE_SID/wallet)))
```

2. Create the directory as 'root':

```
# mkdir –p /u00/oracle/admin/<$ORACLE_SID>/wallet
```

For example:

```
# mkdir –p /u00/oracle/admin/dvsss03/wallet
# chown –R oracle:dba /u00/
# chmod –R 700 /u00/
```

3. As a user with the 'alter system' privilege, create the wallet:

```
SQL> ALTER SYSTEM SET ENCRYPTION KEY IDENTIFIED BY "password";
```

4. Confirm if the wallet is created and open (the TDE master encryption key has been created and inserted automatically):

SQL> SELECT * FROM V$ENCRYPTION_WALLET;

| WRL_TYPE | WRL_PARAMETER | STATUS |
|---|---|---|
| file | /u00/oracle/admin/dvsss03/wallet | OPEN |

In order for the Oracle database to process encrypted data, the wallet needs to be opened after database startup; it stays open until closed or database shutdown:

```
SQL> ALTER SYSTEM SET ENCRYPTION WALLEY OPEN IDENTIFIED BY "password";
```

5. On 10gR2, wallets can be configured to open at database startup time by implementing the following procedure and trigger as sysdba:
   a. run crt_wallet_prc.sql to create the procedure
   b. run crt_wallet_trg.sql to create the trigger
6. As of 11G, an auto-open wallet can be setup, which does not require a password to be given after database startup. For better security, there is also a local auto-open wallet which only auto-opens on the server it was created on:

```
$ cd /u00/oracle/admin/dvsss03/wallet
$ orapki wallet create –wallet . -auto_login_local
$ chmod 600 *
```

# Implement TDE Column Encryption (10gR2)

1. Connect to the database as the RMS schema owner and run enable_resa_tde.sql to encrypt the columns using the default algorithm. This default algorithm can be changed. Shown below is an example illustrating how columns can be encrypted by specifying the AES128 algorithm, turning off the creation of an additional message authentication code (MAC) of the cipher text with 'nomac':

   Example:

   ```
   alter table SA_TRAN_TENDER modify (CC_NO ENCRYPT using 'AES128' 'nomac');
   alter table SA_TRAN_TENDER_REV modify (CC_NO ENCRYPT using 'AES128' 'nomac');
   alter table SA_ERROR modify (ORIG_CC_NO ENCRYPT using 'AES128' 'nomac');
   alter table SA_ERROR_WKSHT modify (ORIG_CC_NO ENCRYPT using 'AES128' 'nomac');
   alter table SA_ERROR_REV modify (ORIG_CC_NO ENCRYPT using 'AES128' 'nomac');
   alter table SA_ERROR_TEMP modify (ORIG_CC_NO ENCRYPT using 'AES128' 'nomac');
   ```

2. Confirm the columns in tables are encrypted:

   ```
   SQL> select * from user_encrypted_columns;
   TABLE_NAME           COLUMN_NAME  ENCRYPTION_ALG     SALT__
   SA_TRAN_TENDER       CC_NO        AES 128 bits key   YES
   SA_TRAN_TENDER_REV   CC_NO        AES 128 bits key   YES
   SA_ERROR             ORIG_CC_NO   AES 192 bits key   YES
   SA_ERROR_WKSHT       ORIG_CC_NO   AES 192 bits key   YES
   SA_ERROR_REV         ORIG_CC_NO   AES 192 bits key   YES
   SA_ERROR_TEMP        ORIG_CC_NO   AES 192 bits key   YES
   ```

# Implement Tablespace Level Encryption (11gR2)

Once the wallet is configured, determine an encryption algorithm to be used for the encrypted tablespace and then create them. In this example, the recommended default of AES128 is used:

```
CREATE TABLESPACE RETEK_INDEX DATAFILE
'/u03/oradata/$ORACLE_SID/retek_index01.dbf' SIZE 100M AUTOEXTEND ON NEXT
100M MAXSIZE 3000M EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
ENCRYPTION DEFAULT STORAGE(ENCRYPT);

CREATE TABLESPACE RETEK_DATA DATAFILE
'/u03/oradata/$ORACLE_SID/retek_data01.dbf' SIZE 100M AUTOEXTEND ON MAXSIZE 3000M,
'/u03/oradata/$ORACLE_SID/retek_data02.dbf' SIZE 100M AUTOEXTEND ON MAXSIZE 3000M
SEGMENT SPACE MANAGEMENT AUTO ENCRYPTION DEFAULT STORAGE(ENCRYPT);

  CREATE TABLESPACE LOB_DATA DATAFILE
'/u03/oradata/$ORACLE_SID/lob_data01.dbf' SIZE 50M
  AUTOEXTEND ON NEXT 100M MAXSIZE 2000M EXTENT MANAGEMENT LOCAL SEGMENT SPACE
MANAGEMENT AUTO ENCRYPTION DEFAULT STORAGE(ENCRYPT);
```

Once the tablespaces have been created, the RMS data is ready to be imported/loaded.

> **Note:** After encryption at the tablespace level or at the column level is implemented, it is absolutely essential to backup the contents in the wallet directory, otherwise, if they are lost you will not be able to access the table and/or the tablespaces.

For additional information on TDE, please refer to:

http://www.oracle.com/technetwork/database/focus-areas/security/twp-transparent-data-encryption-bes-130696.pdf

# B

# Appendix: RMS Policies with Database Vault

RMS 12.0.12 supports enhanced data protection using Oracle Database Vault with Oracle Database 10g Release 2. It is important to install a dedicated RDBMS home for database with DV installed. Use the following procedures to install Oracle Database Vault and set up your RMS policies.

## Before Getting Started

- Make sure RMS 12.0.12 or higher is installed on a supported Oracle Database release
- Make sure the database has the Tablespace "TEMP" as a temporary Tablespace.

> **Note:** At this point Database Vault should NOT be installed in the Oracle Home

## Installation Steps

1. Unzip the file DBVault_RMS_scripts_Release.zip into a temporary directory.
2. Edit all the Database Vault API scripts by replacing RMS01 with the RMS application owner schema name.
3. Add your specific administrator's database accounts to the rule 'Allow Non RMS Users for CONNECT command rule' in the file RMS_rule.sql under the create_policies directory and remove existing ones if they do not apply.
4. Read all the Database Vault API script comments for any additional instructions.
5. Install Oracle Database Vault release 10.2.0.4 as documented in the Database Vault Installation Guide
6. Login to the database as Data Vault Manager
   a. Run the script setup/RMSDBA_USER.sql
   b. Run the script setup/RMSDBA_GRANTS_VAULT_MGR.sql
7. Login to the database as SYSDBA
   a. Run the script setup/RMSDBA_GRANTS_SYS.sql.
   b. Run the script setup/VAULT_MGR_PRIVILAGE.sql (Note: Please replace vault_mgr schema as per provided user)
8. Use RMSDBA for the following tasks:
   a. RMSDBA user is intended for on boarding of new RMS business users.
   b. Make sure you change the default password for RMSDBA user after you run this script
   c. Customers are encouraged to create personalized accounts for RMSDBA like: RMSDBA_SARKARS then add it to the RMSl Application Protection Realm authorizations.
   d. For tuning, RMSDBA_SARKARS for example, can be added to the Enterprise Manager Administrators and do tuning.
9. Login to the database with the Database Vault manager
   a. Run the script setup/TABLE.sql (Note: Please replace vault_owner schema as per provided user)

b. Run the script setup/FUNCTION.sql

c. Run the script setup/FUNCTION_PRIVILAGE.sql (Note:Please replace dvsys and vault_owner as per provided user)

10. Login to the database with the Database Vault Administrator (owner).

    Run the script **create_policies/CREATE_RMS_DBV_POLICIES.sql**

11. While testing the security policies if you need to remove them you Login to the database with the Database Vault Administrator (owner)

    Run the script **delete_policies/ DELETE_RMS_DBV_POLICIES.sql**

12. While testing the security policies if you need to diable them you Login to the database with the Database Vault Administrator (owner)

    Run the script **disable_policies/ DISABLE_RMS_DBV_POLICIES.sql**

13. While testing the security policies if you need to enable them you Login to the database with the Database Vault Administrator (owner)

    Run the script **enable_policies/ ENABLE_RMS_DBV_POLICIES.sql**

# Description

The following security policies are installed:

**RMS Application Protection Realm**: This realm protects against unauthorized access by privileged users to business data. RMS users are allowed access through RMS application. RMSDBA has no SELECT access to RMS Data but is authorized to the realm to be able to board new RMS users and grant them the role developer.

This RMS application protection is complemented by the SELECT Command Rule and the CONNECT Command Rule. The SELECT Command Rule prevents RMSDBA user from having SELECT access to RMS business data. The CONNECT Command Rule ensures business users access through RMS Processes.

This realm secures all RMS objects and the RMS Role DEVELOPER. Only the RMS owner RMS01 and the RMSDBA are authorized. If you create your named RMS DBAs like RMSDBA_SARKARS, you can add them to the realm authorization and the SELECT Command Rule restriction. Before running these scripts in your environment make sure to change RMS01 to your RMS application owner schema name.

**Select Command Rule**: This command restricts Select access to business data owned by the RMS Application. It specifically disallows RMSDBA SELECT on RMS data using the rule set 'RMSDBA no SELECT on RMS data'. This is because RMSDBA is tasked to do on boarding of new RMS application users and does not need access to business data.

**Connect Command Rule**: This command controls access to the RMS database using the security policy (rule set) "User Access to RMS Application" as follows:

1. Normal RMS application users are allowed access through the application tier.
2. RMSDBA and other DBAs are allowed backend access but prevented from accessing RMS Data thanks to realm protection and the Select command rule.

Following is the RMS Application Protection Matrix. It summarizes the policies that are installed by the API scripts and the additional protections that can be added to the production.

| Protection Type \ Authorized with Rule Set | RMS | RMSDBA | DBA & SYSTEM |
|---|---|---|---|
| Rms Realms | OWNER | OWNER | No Access |
| Select Command Rule | Not Restricted | Restricted Select Rule Set | No Access |
| Connect Command Rule | Rms Access Rule Set | Not Resctricted | Not Resctricted |

ORACLE

Customers are encouraged to review the Oracle Technology Network examples on how to add protections to the production environment. This is done by adding command rules that restrict SQL commands like Drop Table, Truncate Table. These examples are available at the following link:

`http://www.oracle.com/technology/deploy/security/database-security/database-vault/dbvb1.html`

# C

# Appendix: AIX Shared Library Bug Fix

The env_rdbms.mk file for Oracle 10g and higher includes defect 2143531, which was not fixed because there is a workaround. For the workaround, the following changes in bold and italics must need to be made to the $ORACLE_HOME/rdbms/lib/env_rdbms.mk file. Note that changes are made in both the BUILDLIB_WITH_CONTEXT and BUILDLIB_WITH_NO_CONTEXT functions.

```
-----------------------------------------
BUILDLIB_WITH_CONTEXT=generate_export_list() \
{ \
/bin/nm -X32_64 -B -h -g "$$1" | grep -v ' U ' | awk '{print $$3}' | \
egrep -v '^\.|^TOC' | sort | uniq ; \
}; \
generate_import_list() { \
LIB_NAME=$$1; \
IMP_FILE=$$2; \
\
cat ${ORACLE_HOME}/rdbms/lib/xa.imp | head -1 | awk '{print $$0, "." }' >
$${IMP_FILE}; \
/bin/nm -X32_64 -C -B -h -g $${LIB_NAME} | grep ' U ' | grep -v "::" | grep -v "("
| grep -v "\.cc" | awk '{print $$3}' | sed -e "s/\.//g
" | grep -v "^_" >> $${IMP_FILE}; \
}; \
\
generate_import_list ***"$(OBJS)"*** $(SHARED_LIBNAME).imp; \
generate_export_list $(OBJS) > $(SHARED_LIBNAME).exp; \
$(LD) -bnoentry -bM:SRE -bE:$(SHARED_LIBNAME).exp -bI:$(SHARED_LIBNAME).imp \
-o $(SHARED_LIBNAME) $(OBJS) -L$(ORACLE_HOME)/lib -lc_r -lm
***$(LLIBCLNTSH)$(MATHLIB)***
-----------------------------------------
BUILDLIB_NO_CONTEXT=generate_export_list() \
{ \
/bin/nm -X32_64 -B -h -g "$$1" | grep -v ' U ' | awk '{print $$3}' | \
egrep -v '^\.|^TOC' | sort | uniq ; \
}; \
generate_import_list() { \
LIB_NAME=$$1; \
IMP_FILE=$$2; \
\
cat ${ORACLE_HOME}/rdbms/lib/xa.imp | head -1 | awk '{print $$0, "." }' >
$${IMP_FILE}; \
/bin/nm -X32_64 -C -B -h -g $${LIB_NAME} | grep ' U ' | grep -v "::" | grep -v "("
| grep -v "\.cc" | awk '{print $$3}' | sed -e "s/\.//g
" | grep -v "^_" >> $${IMP_FILE}; \
}; \
\
generate_import_list ***"$(OBJS)"*** $(SHARED_LIBNAME).imp; \
generate_export_list $(OBJS) > $(SHARED_LIBNAME).exp; \
$(LD) -bnoentry -bM:SRE -bE:$(SHARED_LIBNAME).exp -bI:$(SHARED_LIBNAME).imp \
-o $(SHARED_LIBNAME) $(OBJS) -L$(ORACLE_HOME)/lib -lc_r -lm
***$(LLIBCLNTSH)$(MATHLIB)***
```