

**Oracle® Retail Predictive Application Server and
Applications**

Security Guide

Release 14.1

E55399-01

December 2014

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

Primary Author: Bernadette Goodman

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

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

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

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

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	ix
Preface	xi
Audience.....	xi
Documentation Accessibility	xi
Related Documents	xi
Customer Support	xii
Review Patch Documentation	xii
Improved Process for Oracle Retail Documentation Corrections	xii
Oracle Retail Documentation on the Oracle Technology Network	xiii
Conventions	xiii
 1 Overview	
Terminology	1-1
RPAS Concepts.....	1-1
RPAS Applications.....	1-1
Dependent Applications	1-2
RPAS Server	1-2
RPAS Classic Client	1-2
RPAS Fusion Client.....	1-2
Security Guides.....	1-2
Discussion of Dependencies on Underlying Platform.....	1-3
Client Deployments	1-3
Fusion Client Deployments	1-3
Architecture	1-3
Non-SSO Deployment.....	1-4
SSO Deployment	1-5
Deployment Recommendations	1-5
Dependent Applications	1-6
Classic Client Deployments.....	1-6
Architecture	1-6
Non-Weblaunch Deployment	1-6
WebLaunch Deployment.....	1-7
SSO Deployment.....	1-7
Non-SSO Deployment.....	1-8

Multi-Solution Deployments.....	1-9
2 Installing the RPAS Server	
Operating System Level Security	2-1
Clean Up File Ownership and Access.....	2-1
RPAS Server Precautions.....	2-1
Fusion Client Precautions.....	2-1
Secure User Accounts.....	2-2
Maintenance.....	2-3
3 Securing the Fusion Client	
Factors Affecting Security	3-1
Authentication.....	3-1
Single Sign On.....	3-1
Authorization.....	3-1
Auditing and Error Logging.....	3-1
Tracking Users.....	3-2
Error Logging.....	3-2
Managing Sensitive Data	3-2
Passwords.....	3-3
Configuration Post-Installation	3-3
4 Securing the Classic Client	
Authentication	4-1
Password Administration Workbook.....	4-1
Setting a Password Policy.....	4-1
Setting a Logon Policy.....	4-2
Password Storage.....	4-2
Authorization	4-2
Auditing	4-3
Managing Sensitive Data	4-3
5 Securing the RPAS Server	
User and Group Management	5-1
Locking User Accounts.....	5-1
Authorization	5-2
Workbook Security.....	5-2
Measure Level Security.....	5-2
Position Level Security.....	5-3
Setting Proper Resource Limits.....	5-4
Workbook Template Limits Views.....	5-5
Max Domain Session Limit View.....	5-5
Max User Session Limit View.....	5-5
Dimension Modification Rights View.....	5-5
Auditing.....	5-5
Managing Sensitive Data	5-6

Domain Daemon IP Filtering and Redundancy.....	5-6
Secure Socket Layer	5-6
Introduction	5-7
Setting Up SSL for the Fusion Client.....	5-7
Two-Way SSL Authentication Only	5-8
Setting up the OBIEE Connection over SSL	5-8
Setting up SSL for the Classic Client	5-9
Setting up SSL for the RPAS Server.....	5-9
Supporting Documentation	5-10
 6 Domain Creation and Maintenance	
Configuration Management	6-1
Dynamic Position Maintenance	6-2
RPAS Maintenance	6-3
 7 RPAS Integration	
Integrating User Dictionaries.....	7-1
Integrating Hierarchy and Dimension Data	7-1
Integrating Measure Data	7-1
ODI	7-2
RETL.....	7-3
 8 RPAS Hybrid Storage Architecture Option	
RPAS Data Mart Construction and Security Model.....	8-1
External Integration APIs	8-2
SSL Connection	8-3
SSL Overview	8-3
Set Up SSL on Oracle Server.....	8-3
Create Oracle Server Wallet	8-4
Update Oracle Server Network Configuration	8-5
Update Oracle Listener Configuration	8-6
Set up SSL on Oracle Client.....	8-6
Update Schema Info Configuration	8-6
Import Server CA Certificate.....	8-7
Update Oracle Client Network Configuration	8-8
Update Oracle Net Service Names	8-8
Test and Confirm SSL Connection	8-8
 9 Extending and Customizing Products	
Custom Libraries and Custom Template Libraries	9-1
Creating Custom Libraries and Custom Template Libraries	9-1

Send Us Your Comments

Oracle Retail Predictive Application Server and Applications Security Guide, Release 14.1

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 Online Documentation available on the Oracle Technology Network Web site. 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 <http://www.oracle.com>.

Preface

This document serves as a guide for administrators, developers, and system integrators who securely administer RPAS and RPAS applications. Installation and configuration for each product are covered in more detail in the each product's Installation Guide.

Audience

This document is intended to provide an overview of the security features of the RPAS Platform and applications built upon it. It contains a set of best practices for administrators, developers, and system integrators who perform the following functions:

- Manage the RPAS environment at the OS level.
- Install and configure the RPAS Server, Fusion Client, and/or Classic Client.
- Integrate RPAS domains with other domains or other products.
- Perform RPAS Administration tasks such as user management, permissions, and system limits.

This document is not intended to describe in detail the processes of deploying and maintaining an RPAS application; for detailed information on these topics, readers should consult the relevant end user documents. It is assumed that the readers have a general knowledge of administering the underlying technologies and applications.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

This document serves as a guide for administrators, developers, and system integrators who securely administer, customize, and integrate Oracle Retail Predictive

Application Server (RPAS) and RPAS applications. Installation and configuration for each product are covered in more detail in the each product's Installation Guide. Information on securing the following RPAS applications is included in this guide:

For more information, see the following documents in the RPAS documentation set:

- Oracle Retail Analytic Parameter Calculator for Regular Price Optimization
- Oracle Retail Assortment Planning
- Oracle Retail Category Management
- Oracle Retail Demand Forecasting
- Oracle Retail Item Planning
- Oracle Retail Merchandise Financial Planning
- Oracle Retail Regular Price Optimization
- Oracle Retail Replenishment Optimization
- Oracle Retail Size Profile Optimization

The following application is **not** included:

- Oracle Retail Analytic Parameter Calculator for Replenishment Optimization

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, 14.1) or a later patch release (for example, 14.1.1). If you are installing the base release or additional patches, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch releases can contain critical information related to the base release, as well as information about code changes since the base release.

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times not be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

<http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html>

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-02 is an updated version of a document with part number E123456-01.

If a more recent version of a document is available, that version supersedes all previous versions.

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/technetwork/documentation/oracle-retail-100266.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

The following text conventions are used in this document:

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

Overview

The Retail Predictive Application Server, or RPAS, is a platform that provides a set of common components used by a number of applications (or solutions). For these solutions, RPAS provides the infrastructure needed to store, process and produce information based on data input by the retailer.

This guide discusses security considerations around the deployment and operation of an RPAS Server deployment and interaction between that server and the set of clients deployed for the users of an RPAS application.

Where applicable, application-specific information about secure deployment of each RPAS application can be found in that application's Installation Guide.

RPAS itself does not have any special security requirements.

Terminology

The following section provides a brief introduction to RPAS and its terminology.

RPAS Concepts

- **RPAS:** A platform that provides a foundation to run solutions used for retail planning. RPAS provides those solutions with a common interface based on wizards, templates, workbooks and batch processes.
- **RPAS Solution:** An application running on top of RPAS that provides solutions for retail problems such financial planning or forecasting demand.
- **RPAS Domain:** A collection of server side directories and files containing the data and procedures required to execute a specific RPAS solution. Domains may be:
 - **Global:** accessible for all users.
 - **Local:** configured so that only a sub-set of data is accessible for specified users.

RPAS Applications

There are two ways of accessing information in a RPAS solution:

- **Classic Client:** A windows based thick client.
- **Fusion Client:** A web based client

In addition, Administrators can access the **Configuration Tools**. This is a Windows based set of utilities used to configure and maintain a RPAS Solution.

Dependent Applications

A series of applications are required to install and run the RPAS Server. Additional software is required to install the Fusion Client or the Classic Client required by users to access and manipulate the data. Full details can be found in chapter 1 (Introduction) of the *Oracle Retail Predictive Application Server Installation Guide*.

RPAS Server

Java

Java 1.7 JDK is required for the RPAS Server, the RPAS Configuration Tools (including domain creation and patching), and for the JDBC environment. For the latest security patches, refer to the *Oracle Retail Predictive Application Server Installation Guide* for your current version.

Other Applications

If installing the RPAS Server on a Unix or Linux platform, an unzip utility will be required. Perl will also be required for the upgrade process.

If installing the RPAS Server on a Windows platform, Cygwin will be required. For details, see Chapter 4: *Installing on a Windows Environment* in the *Oracle Retail Predictive Application Server Installation Guide*.

If the optional Hybrid Storage Architecture (HSA) functionality will be used, an Oracle Database 12c installation will be required. For details, see the Hybrid Storage Architecture chapter in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

RPAS Extension Libraries

For any implementers/customers who wish to compile RPAS C++ extension libraries (custom templates, functions, or expressions), the required C++ compiler versions are listed in the *Oracle Retail Predictive Application Server: RPAS Extension Development Guide* (Doc ID 1926977.1) on My Oracle Support.

RPAS Classic Client

If using the Classic Client with WebLaunch, users are also required to install the WebLogic Server. See Table 1–2: RPAS Classic Client Hardware and Software Requirements and Chapter 7: RPAS Classic Client Web Deployment in the *Oracle Retail Predictive Application Server Installation Guide* for more information.

RPAS Fusion Client

If using the Fusion Client, see Table 1–3: RPAS Fusion Client Hardware and Software Requirements in the *Oracle Retail Predictive Application Server Installation Guide* for more information on the Web Browser, Application Server, Supported Operating system, and Java requirements.

Security Guides

As well as the RPAS Security Guide, Security Guides exist for other applications such as the WebLogic server. Information on these is available on the Oracle Technology Network at the following URL:

<http://www.oracle.com/technetwork/documentation>

The following security guides are useful:

- *Oracle Retail Merchandising Security Guide*
- *Oracle Retail Advance Science Engine (ORASE) Security Guide*

Discussion of Dependencies on Underlying Platform

The following documents provide further information on RPAS Server dependencies:

- The *Oracle Retail Predictive Application Server Installation Guide* chapter 1 lists the hardware and software requirements for the RPAS Server. Table 1-1 is especially useful.
- Basic requirements of environment variables for running the RPAS Server are listed in the *Oracle Retail Predictive Application Server Installation Guide* chapter 3 (for Unix) or chapter 4 (for Windows).
- A more detailed discussion of RPAS Server environment variables, including the required path variables, plus variables covering Database tuning, Log level settings, Date and Time specifiers, and control of parallel processing is found in the RPAS Administration Guides, Appendix D.

Client Deployments

Users can connect to the RPAS solutions using one of two clients:

- The primary client used by RPAS applications is the Fusion Client. The Fusion Client is a web-based application that allows access to RPAS workbooks through interaction with the web server in users' browsers.
- RPAS also supports a legacy Classic Client. The Classic Client is a stand-alone desktop application deployable on Windows that interacts directly with an RPAS domain.

When deciding between the Fusion and Classic Clients, installers should take into account that the Fusion Client, based as it is upon standard Oracle technologies, allows greater assurance of a secure environment and greater flexibility in the nature of that environment. Details on deploying each of the clients can be found in the following sections:

- [Fusion Client Deployments](#)
- [Classic Client Deployments](#)

Fusion Client Deployments

This section contains information on how to secure a Fusion Client deployment.

Architecture

This is an Application Development Framework (ADF) based 3-tier web application. The Fusion Client is deployed on the WebLogic Server. It interacts with the RPAS Server deployed as daemon processes. Typically, WebLogic and RPAS Servers are deployed on separate machines. They support the AIX, HP-UX, Solaris, and Linux platforms.

The Fusion Client (running within WebLogic) and the RPAS Server are typically deployed behind a firewall. They communicate using a TCP/IP based protocol that supports encryption. More components are involved if using multiple WebLogic

managed servers for scalability (hardware or software load balancer), and for supporting single-sign-on (Web tier server, OAM, load balancer).

Single Sign-On (SSO) deployment requires perimeter authentication in the Web Tier. Oracle SSO architecture calls for an Oracle HTTP Server configured as a reverse proxy and an OAM WebGate component plugged into it for intercepting and enforcing authentication on all requests. The authentication is done using Oracle Access Manager (OAM) and Oracle Internet Directory (OID).

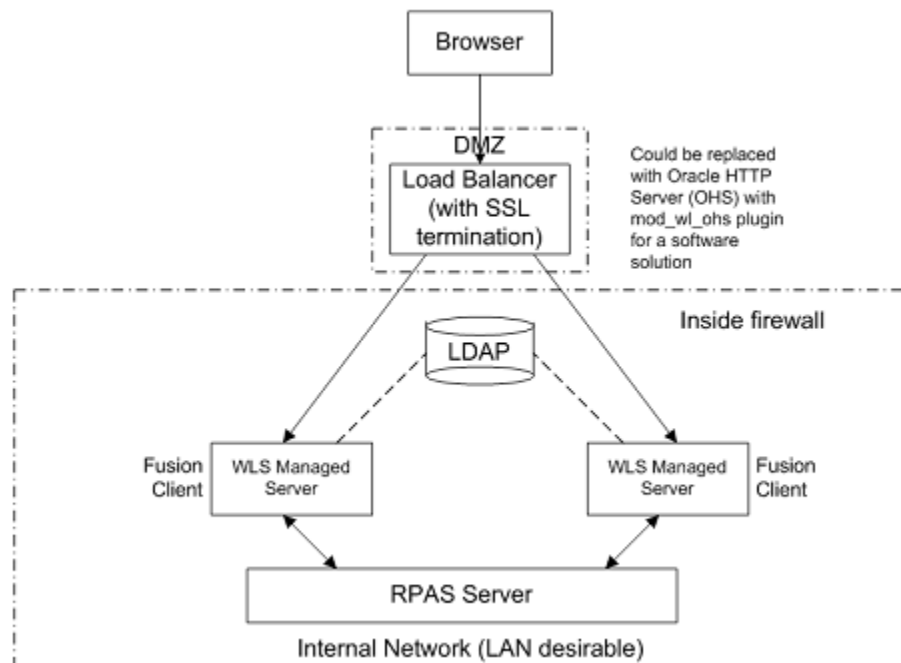
A web tier consisting of either a reverse-proxy web server or a hardware load balancer is recommended in non-SSO deployments as well. This provides better security management and an opportunity to reduce the performance overhead of SSL by implementing it in the web tier which is often better equipped to execute SSL endpoint functions than the WebLogic Server.

Non-SSO Deployment

This is a typical topology for deployments without SSO. The main features are as follows:

- A load balancer with SSL termination capability: it provides a public URL to prevent direct access to the internal corporate network where the application servers are deployed. It also provides load-balanced connection to multiple application servers.
- Application servers, such as Oracle WebLogic, are deployed inside the firewall. Multiple servers provide horizontal scaling.
- A single Fusion Client deployment can provide access to multiple RPAS solutions (ex. MFP and IP) which might be hosted on separate machines with their own DomainDaemon processes. These RPAS Servers are also deployed inside the firewall. The communication between the RPAS Fusion Client instances and the RPAS processes ideally takes place over a LAN.

Figure 1–1 Topology for Non-SSO Deployment

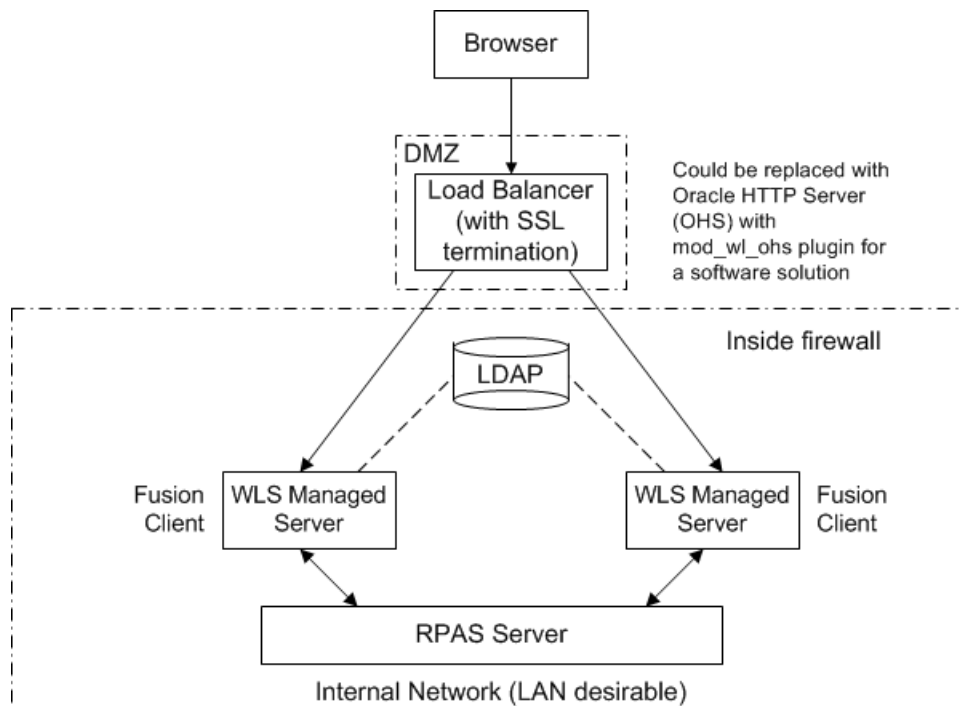


SSO Deployment

A typical SSO deployment has the following additional characteristics:

- Perimeter authentication enforced by the Oracle Access Manager (OAM) WebGate plug-in attached to an Oracle HTTP Server instance deployed in reverse-proxy configuration.
- Mod_wl_ohs is an OHS plug-in that funnels requests in a load-balanced way to the WebLogic managed servers.
- The identity store (here labeled as "LDAP") is deployed inside the firewall and is used by the WebLogic Servers and the Oracle Access Manager server.

Figure 1–2 Topology for SSO Deployment



Deployment Recommendations

- Use a web tier. In conjunction with other security measures (described below) this provides better security by allowing hiding of application data and configuration files behind a firewall.
- Deploy the WebLogic Managed Servers hosting the Fusion Client behind a firewall.
- SSL is required on the browser to Web Tier Internet connection. For performance reasons it is a good idea to do SSL termination at the web tier. Requests forwarded to the application servers can be unencrypted since the communication is behind the firewall.
- Enable the SSL listen ports on the WebLogic Servers and turn off the non-SSL ports.
- Install CA-signed SSL certificate on the WebLogic Domain.
- Implement and install a WebLogic Network Connection Filter on the WebLogic Servers to accept connections only from the web tier component. This is to prevent

access to the application from unauthorized sources in case the firewall is down for any reason.

- Disable all web access methods on the WebLogic Servers other than HTTP.
- Deploy the web tier server or Load Balancer in a DMZ. Browser requests are first received at the web tier server on a publicly accessible URL. It needs to have access to the application servers located behind a firewall.
- SSL is used for communication with the RPAS Server. It is generally recommended to use CA-signed SSL certificates (one for the fusion client and one for the RPAS server). In cases where the customer will always be in full control of the Fusion Client and RPAS Server setups, it is acceptable to use a self-signed root certificate as the certificate signing authority.

Dependent Applications

Security Guides are available for the following dependent applications:

- Oracle Access Manager
- Oracle HTTP Server
- Oracle Internet Directory
- Oracle WebLogic

These Security Guides may be found on the Oracle Technology Network at the following URL:

<http://www.oracle.com/technetwork/documentation>

Classic Client Deployments

This section contains information on how to secure a Classic Client deployment. Deployment can either be WebLaunch or non-WebLaunch. If the deployment is WebLaunch then users can then opt whether to use SSO.

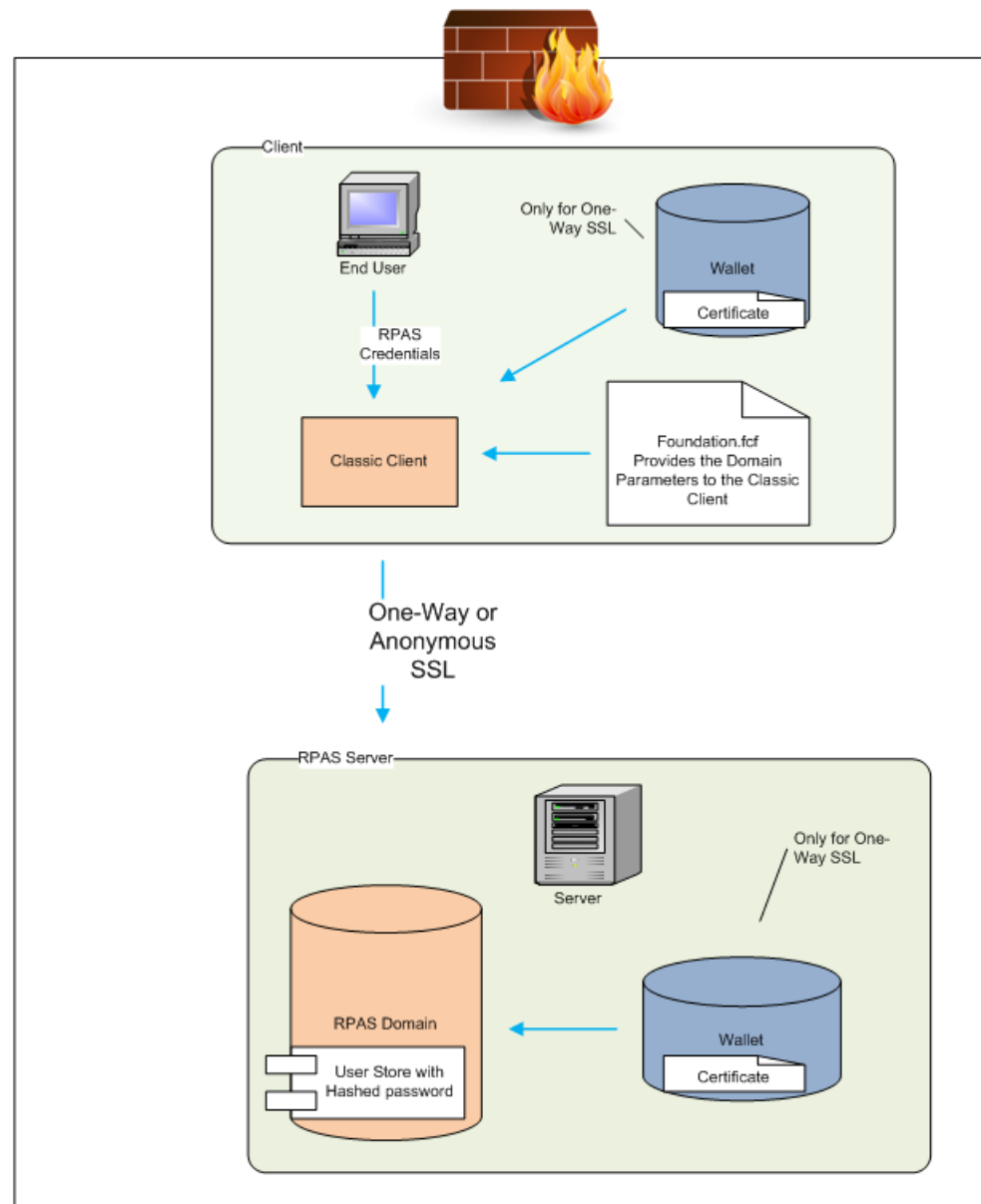
Architecture

The Classic Client is a thick client that is installed directly on an end-user's desktop. When interacting with the RPAS Server, the Classic Client uses either SSL 1 or SSL 3, depending on the configuration of the system. In order to establish a connection with an RPAS Server, the Classic Client must provide credentials in the form of a user name and password that are validated against the user store of the domain.

The list of RPAS domains to which a client can connect can be specified through a file named `foundation.fcf`. Connection information can also be distributed via a standalone installation kit, or remotely installed on end-user PCs through the WebLaunch interface.

Non-WebLaunch Deployment

In a non-WebLaunch deployment, connection information used by the Classic Client is read by the client from a file system resource named `foundation.fcf`. This file, which can be managed using the `eConfigure` utility provided as a part of the client installation package, contains information used by the client to create connections to a RPAS Server instance and the domain used by the instance. This information includes network address information and configuration information for the connection. The following diagram on the next page provides a high-level view of a Classic Client deployment without WebLaunch:

Figure 1–3 Non WebLaunch Deployment

WebLaunch Deployment

RPAS WebLaunch is a way to centralize the distribution of the Classic Client throughout an organization. It hosts the RPAS Classic Client installer on a web server and can install or update the Classic Client on the user's Windows PC directly from a web browser. Additionally, it can centralize the management of the list of domains that are available, removing the need for storing the foundation.fcf file locally. It is available in a SSO and non-SSO environment.

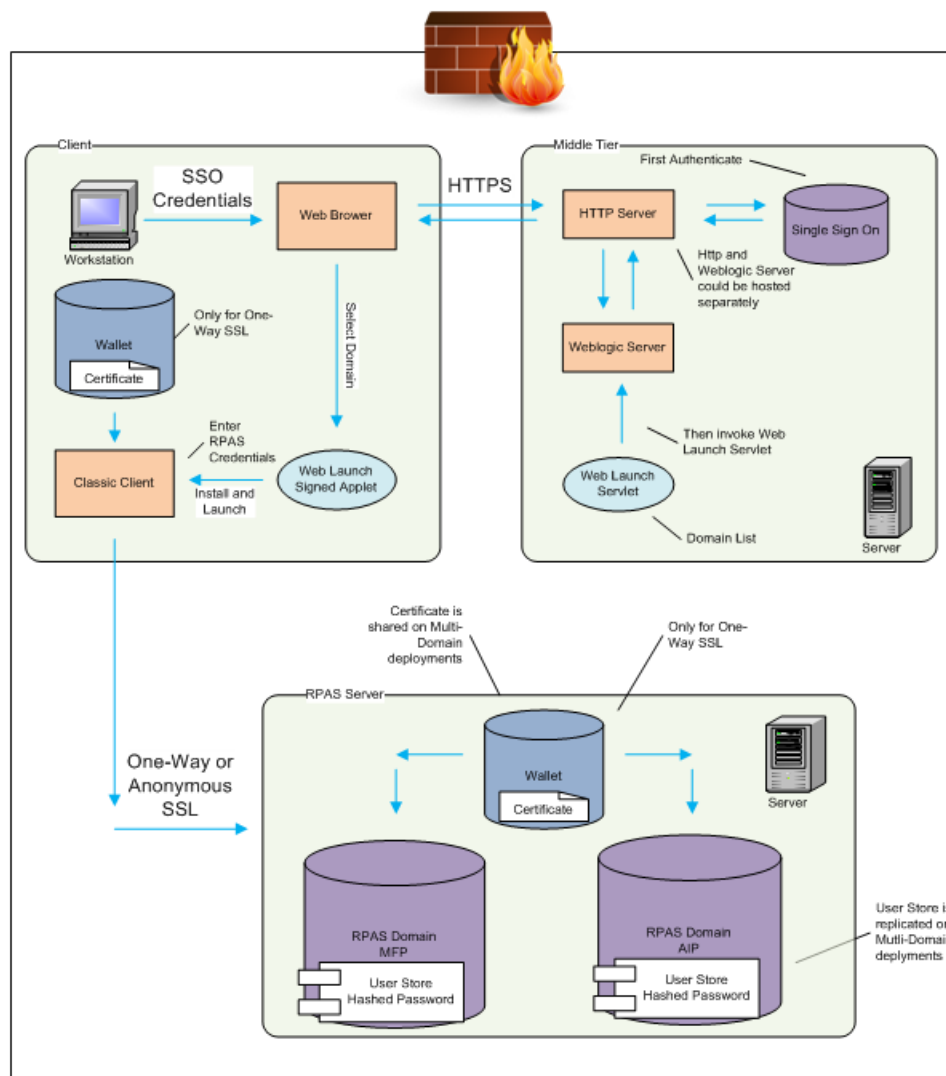
For more information on Web Launch deployment, see Chapter 7: *RPAS Classic Client Web Deployment* in the *Oracle Retail Predictive Application Server Installation Guide*.

SSO Deployment

RPAS WebLaunch can be deployed in an SSO environment which is similar to that of the Fusion Client. The SSO version of Web Launch allows remote-configuration of

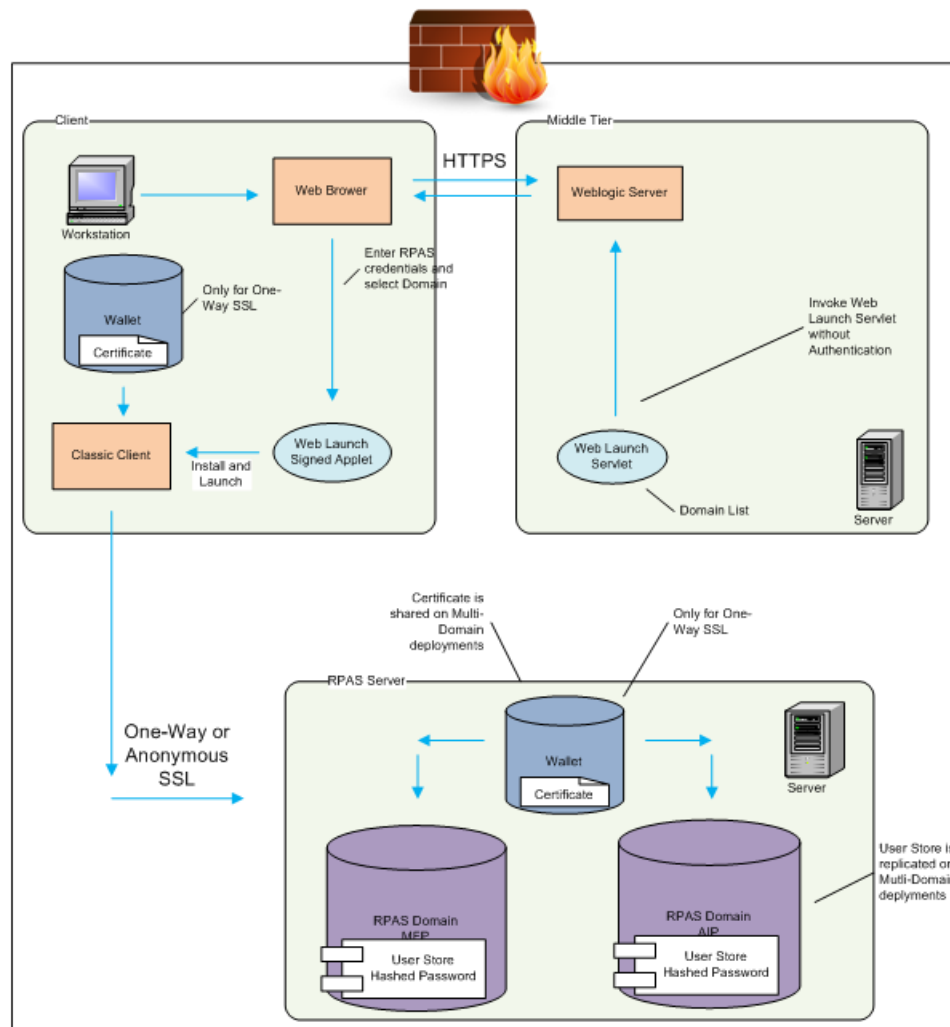
domains with their Classic Client version from the web browser by admin-privilege authorized SSO users. Other authorized SSO users with less privilege can install and launch the Classic Client. These SSO users are solely for the web interface and have nothing to do with RPAS users. The Classic Client will prompt for RPAS login once it is started. The diagram on the next page provides a high-level view of a Classic Client deployment with WebLaunch and SSO. It also displays a Multi-Domain deployment:

Figure 1-4 SSO Deployment



Non-SSO Deployment

RPAS Web Launch in a Non-SSO environment allows an RPAS user to install and launch the Classic Client and then connect to an pre-configured domain. Because of the lack of authentication available, the RPAS administrator must configure the domains on the back end by editing the domain properties file manually. The diagram on the next page provides a high-level view of a Classic Client deployment with WebLaunch but without SSO. It also displays a Multi-Domain deployment:

Figure 1–5 Non-SSO Deployment

Multi-Solution Deployments

In situations where more than one RPAS-based solution is deployed, these separate deployments may be setup to operate independently. In such cases, there are no additional security considerations beyond those of each application. However, it is also possible to configure applications such that they operate in a more integrated fashion.

See the [Domain Daemon IP Filtering and Redundancy](#) section for additional information.

Such deployments are called multi-solution deployments and they require additional consideration in terms of the degree of integration between applications. Of primary importance is the ability to replicate the user dictionary of one domain for use in another. In multi-solution deployments, creating a unified user dictionary will allow users to work within each of the domains without the need for managing separate credentials for each domain.

See the [Integrating User Dictionaries](#) section in the RPAS Integration chapter for information on shared user dictionaries.

Installing the RPAS Server

This chapter of the security guide deals with security factors related to the installation of the RPAS Server.

Operating System Level Security

As part of the RPAS installation there are several security considerations regarding Operating System file permissions, account creation, and folder permissions, among others. This section provides recommendations for operating system permissions and accounts for an RPAS installation, file ownership and access for the RPAS server and the Fusion Client, account creation guidelines, and overall operating system maintenance.

Clean Up File Ownership and Access

This section contains a short list of operating system security precautions to consider while installing the RPAS Server. This set of precautions is primarily intended for preventing unauthorized access to operating system files, whether they are sitting in a folder or in the process of being transferred. They are grouped into RPAS Server and Fusion Client precautions.

RPAS Server Precautions

The following are recommended when setting up the RPAS Server:

- Require ssh and scp or other secure methods to log in to a shell in the operating system hosting the application server when doing administrative tasks.
- Employ an internet firewall between the collection of the application server/RPAS server machines and the outside world.
- Eliminate telnet, ftp, rsh, rlogin, and rcp connections.
- Configure SSL access between the RPAS Server and Oracle Database, if using the HSA functionality.

Fusion Client Precautions

The following are recommended when setting up the Fusion Client.

- Ensure that the operating system user who installs the WebLogic Server and ADF runtime libraries and who creates the WebLogic domains is **not** the root user. Instead create another user - for the purposes of this document, the user 'oracle' is used as the example.

- Ensure that no files are created as root or any other user within the WebLogic Server installation directory.
- Ensure that the 'oracle' user is also the user who starts up the WebLogic Servers.
- Ensure that the Fusion Client is installed by the 'oracle' user.
- Ensure that no other user creates files in the Fusion Client installation directory and that all files are owned by the 'oracle' user. (A select few other users present in the installer user's group may be allowed to read the files).
- A permission of 640 is appropriate for all files under the Fusion Client installation directory. Files created by the Fusion Client installer have this permission by default.

Secure User Accounts

The following list provides general recommendations on how to strengthen the overall system security by configuring the Operating System (OS) accounts in a secure manner.

- Make sure that all OS accounts have passwords that cannot be guessed.
 - Enforce rules for passwords requiring a combination of upper and lower case letters, numerals and special characters.
 - Ensure that enforced password changes are required at regular intervals.
 - Use a password cracking tool (such as Crack or John-the-Ripper) at regular intervals. This will guard against people using passwords associated with them such a children's names or hobbies.
- Automatically disable accounts after a specified number of failed login attempts.
- Severely restrict the distribution of the root password and keep track of who has them:
 - Change the root password at frequent, regular intervals.
 - Change the root password as a matter of policy as soon as anyone with knowledge of it leaves the company.
- .netrc files weaken security.
- Root and root only should have uid "0".
- Check root ".*" files for security holes. Such files should have 700 or 600 permissions and nothing else.
- To avoid Trojan Horse programs, root user should always use full path names including aliases. Root should never have "." in its PATH.
- Oracle recommends that an RPAS OS account be created and given a default file creation permission of 700 (via umask).
 - This account should be used to install the RPAS binaries, execute the rpasInstall process, administer the daemons, and own the cron and batch processes. This will provide a hardened configuration where the files in RPAS_HOME, RIDE_HOME, and the RPAS domains are not accessible at the operating system level to anybody other than the rpas account.
 - An rpas user group can also be created to share this rpas administration privilege among multiple OS accounts, and in this case the umask should be set to 750 instead of 700.

Maintenance

Other tasks that should be carried out as part of the maintenance process include:

- Installing the latest operating system patches as they become available.
- Regularly audit user accounts. Delete or lock any accounts no longer required.

Securing the Fusion Client

This chapter discusses security for the RPAS Fusion Client.

Factors Affecting Security

Some factors affecting security are Authentication, Single Sign On, Authorization and Auditing and Error Logging.

Authentication

The application uses an external authentication model which uses pluggable plugged-in authentication providers in the Weblogic Server. This allows the retailer to use authentication providers such as Oracle Internet Directory. Users and groups are configured in the external authentication system, and are mapped to the Fusion Client roles. This can be done in the installer itself (using the `input.security.user` and `input.security.group` properties), or it can be done post-install using Oracle Enterprise Manager.

Single Sign On

Fusion Client supports deployment in an SSO environment using perimeter authentication. A gateway component installed in the Web Tier is required to enforce authentication on all requests. The Fusion Client installation process provides an option to deploy the Fusion Client in an SSO-supporting mode. It is necessary that SSO users are also present in the RPAS domain.

For more information, see the *Oracle Retail Predictive Application Server Installation Guide*.

Authorization

The Fusion Client provides role-based authorization. These roles are Application Development Framework (ADF) roles. The customer needs to map them to their enterprise users and groups as defined in (for example) LDAP. Initially this is done through the installer. Subsequently the Enterprise Manager tool found in the WebLogic application server can be used to administer the role mapping.

Auditing and Error Logging

This section covers auditing and error logging.

Tracking Users

The Fusion Client tracks users as they log in and log out. This is over and above any auditing done inside the customer's identity management product. To enable this, the log4j Category "common.security" is set by default to the INFO level. This provides an audit trail of user login and logout activities.

Error Logging

Error logging is configured in the file log4jConfig.xml that is present in the Fusion Client installation directory. In it the customer specifies the file system location of the log files. The customer also sets the logging levels for various categories (application areas).

The available levels are debug, info and error.

- **Debug** produces the most comprehensive logging, and is appropriate for capturing data for reporting defects.
- **Info** records a less comprehensive set of error reports. The file size is less than le sized file outputs and at the same time produces information for a first-level analysis of a defect. This is the recommended level for day to day use.
- **Error** produces the least logging and only records application errors (which might or might not be fatal to the application as a whole).

There are two types of logging in the Fusion Client: a log that reports activities in the overall session (called "rpas_fc.log" by default), and a log where performance timings are recorded (called "perf.log" by default). It is possible to change the location and names of these log files by configuration inside log4jConfig.xml.

The default values of the log file locations are not likely to match any location on the customer's file system. The customer must edit these values to point to secure locations.

It is also possible to roll over to new log files on a scheduled timing event, such as at the top of the hour or at the start of a new day. The method to configure such "rolling" logging is described in log4jConfig.xml.

RPAS Logging within User Session

The Fusion Client has a configuration property called "serverloglevel" whose value dictates the logging level of the RPAS Server process to which the user session is connected. The log file is called "rpas<sessionid>.log" and it is placed under the directory <RPAS-domain>/users/<userid>.

Managing Sensitive Data

The Fusion Client does not store any sensitive data. It is also able to protect itself against click-jacking, cross site scripting, session fixation, and cross site request forgery attacks.

However, sensitive data flows through the Fusion Client as it moves between the browser and the RPAS Server. To protect data in transit, the data is encrypted using SSL on both the front and back end. Setting up front-end SSL is the customer's responsibility. SSL setup on the back end is mandatory.

Passwords

The RPAS Fusion Client uses external authentication and as such relies on password policies of the external system such as Oracle internet directory (OID) to ensure secure authentication.

Passwords are never stored in the middle tier.

Configuration Post-Installation

There are a few parameters in the Fusion Client that the customer can modify post-install, that have a bearing on the application security profile. They are set to certain values that provide the maximum security possible. However, these factory settings may not necessarily work well in relation to a specific customer's needs. We enumerate the parameters here, explain what they mean, and the implications of changing them from the factory settings.

Table 3–1 Factory Settings

Name	Released Setting	Description
announcements.update.delivery	Never	This setting determines how often to check for new announcements. If set to never, a check for new announcements is only carried out at login time. If set to onaction , checks are carried out more frequently and can cause some loss of performance.
autocreate.concurrent.session	False	Automatically creates new concurrent RPAS session if set to true. Setting it to true without proper thought can lead to large numbers of abandoned user sessions consuming system resources.
commitstatus.polling.interval	30000	This is the time in milliseconds that must elapse between polling events issued by the browser to check for commit status changes. Lower values report client status more promptly but degrade performance.
guidedlaunch.max.size	50000	This is the size in bytes of the data provided in the body of the guided launch request. It should be no larger than the largest guided launch request.
location of the images directory	None	The files system location where images of products are stored. Ensure that this location cannot be accessed by unauthorized users.
perftiming.enabled	False	This setting enables enhanced logging to capture performance timings. It is useful to diagnose performance issues. It is suggested it is only enabled when required as the logging does result in a small performance penalty.
printexport.maximum.cells	200000	Print/export will work only if cells in question do not exceed this value in number. Higher value can cause the server to fail with Out-Of-memory error. It should be set to lowest acceptable value.

Table 3–1 (Cont.) Factory Settings

Name	Released Setting	Description
RPAS domain IP address and port number	None	The location of the RPAS domains. Ensure that the values cannot be changed by unauthorized users.
serverloglevel	Error	This is the logging level of the RPAS Server. It is set for the duration of the user session. Lower levels collect more information and may be required for tracking defects. However, this can severely degrade performance. It is therefore recommended that the log level is only set to the lower levels when there are repetitive defects to be identified.
session.timeout	35	This determines how long a session between the browser and the application server should be inactive before expiring. The value is in minutes. It should be set to the smallest value acceptable to users.

Securing the Classic Client

This chapter discusses security for the RPAS Classic Client.

Authentication

The RPAS Server handles the authentication for all users connecting via the classic client. The connection between the server and client does not time out and it is SSL protected, so the password is always encrypted when transmitted. Passwords are hashed using a configurable, de-optimized algorithm and stored in the user's private metadata database.

Users created via the batch-load functionality in usermgr will receive a temporary password that is applied to all users who are part of that batch load. This temporary password expires after the first time the user logs in. Any account that is not going to be claimed immediately should be locked by the administrator after user load.

A user can change their password at any time via the File->Change Password option. If they forget their password, the RPAS administrator can change their password either through the Edit User template or the usermgr utility. The user's password history prevents them from re-using a password within a certain time interval. However, when the password is changed via one of the administrator interfaces, the history is ignored. This allows RPAS administrators to reuse temporary passwords.

Password Administration Workbook

This section covers options concerning password security.

Setting a Password Policy

Using the Password Policy Measures Settings view, administrators can configure password complexity and settings in order to ensure the account security of users and other administrators. With this view, administrators can set the required password complexity, the number of allowable password attempts, the expiration time of a password, and the length of time a user is locked out of the system after failed password attempts.

Most companies have their own password policy, which the configurable parameters in the Password Administration Template should accommodate. If you need to create your own password policy, here are some guidelines to follow:

- Password security is directly related to password complexity. Requiring lowercase, uppercase, symbolic and numeric characters help to prevent common-use passwords and reduce the effectiveness of dictionary attacks.

- Increasing the minimum password length exponentially affects the number of attempts needed to derive a password. This is typically set to either six or eight, with eight being recommended for more secure environments.
- Password reuse and history settings work in conjunction to reduce the amount of time that an unknowingly exposed password can be exploited. Typical settings for these fields are forcing customers to change their passwords every three to six months. The history threshold should be enough to ensure that the user must have at least 6-12 unique passwords. (Example: If users must change password every three months, the history threshold should be set to 18-36 months).

The severity of the security policy should be related to the security of the environment and the data under the application's control. The end-user experience should also be taken into consideration when designing the policy. A password policy that is so strict that users are going to write their passwords down and keep them on their desk will do more harm than good.

These parameters can be set in the **Password Administration Policy Workbook**.

More information can be found in Chapter 6: *System Administration* in the *RPAS Administration Guide for the Classic Client*.

Setting a Logon Policy

Accounts may be configured to automatically lock out after a certain number of failed logon attempts. A domain administrator can configure the number of failed logon attempts and the duration of the lockout using the Password Policy Administration workbook.

Accounts may be marked as requiring the user to change the password. When this is set, users are prompted to change their password the next time they log in. Users cannot proceed using the RPAS client unless they successfully change their password. This is useful for new accounts that are created with a stock password. The domain administrator can set or clear this setting using the User Management utility or the Edit User workbook.

Password expiration may be enabled for the domain. The domain administrator may set the number of days after which passwords expire. After this time passes, users are prompted to change their password the next time they log on. Users cannot proceed using the RPAS client unless they successfully change their password.

A password reuse time can be set for the domain. This is often used in combination with password expiration to ensure that users do not change their password to a recently used password after the current one expires. The domain administrator may set the minimum number of days that may pass before users can reuse a previous password using the Password Policy Administration workbook.

Password Storage

In a global domain, passwords and the password policy are centralized in the master domain. The administration templates are not available in the local domains, and any attempt to add users to a local domain on the back end will result in an error. These settings are specific to the application domain, and cannot be shared with other domains.

Authorization

The Classic Client has no intrinsic authorization process. However, connections made from the Classic Client to instances of the RPAS Server must successfully complete an

authorization process before the client may interact with that server instance. Information on RPAS Server authorization can be found in the Securing the RPAS Server section.

Auditing

RPAS Classic Client can be started with a command line option `-loglevel {level}` to control the logging granularity. The log file for the client is created under the same directory where the executable "Foundation.exe" resides, and named "Client.log". This log level is also passed to the RPAS Server and controls the logging granularity of the session on the server side. The RPAS Server log level cannot be set below its minimum logging threshold in this manner (See RPAS Server Auditing for details).

Managing Sensitive Data

All communications between the Classic Client and RPAS server are protected by SSL. Passwords stored on the server are hashed, and the system administrator can configure the settings of the hashing policy.

Securing the RPAS Server

This chapter contains information on securing the RPAS server.

User and Group Management

RPAS allows administrators to put users into distinct groups. A group is similar to a traditional database role in that it allows the administrator to configure authorization settings for several users at once. The main difference, however, is that user and group have a hierarchical relationship where settings are always stored at the user level, and group is a rollup of user. User groups are typically assigned based on a common business role such as Planners, in order to facilitate managing the authorization settings at the group level.

The group that a user rolls up to is referred to as their primary group. A user can also be associated with other groups using the Other Groups property. The Other Groups property is not used for authorization purposes, but instead allows a user to save workbooks and formatting in a way that it is visible to users whose primary group is one of those Other Groups. This behavior is typically used by people that need to support other users rather than an end-user. An example of this would be a team whose job is to set up the formatting for all of the other project groups.

When a user is added, a position is created for them in the metadata dimension User. Similarly, when a group is added that group gets a position in the metadata dimension Group. Frequent adding and dropping of users and groups can eventually exhaust the list of available positions in these dimensions, and will require reindexing of these dimensions.

Additionally, when a user is added, a directory is created for them in the /users directory of the domain root. In global domains, this directory is created in the master, and all subdomains. This directory serves as a workbook repository, as well as a cache for some metadata such as MRU lists. When a user is deleted, these directories, as well as any workbooks created by that user, will be deleted with the user.

For more information for the Fusion Client, see *Chapter 6: User Maintenance* in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*. For more information for the Classic Client, see *Chapter 5: User Maintenance* in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

Locking User Accounts

User accounts can be marked as locked by the domain administrator. This prevents the user from logging on with the RPAS Client. The account remains locked until the administrator re-enables the account. The domain administrator can set or clear account lockouts by using the User Management utility or the Edit User workbook.

Authorization

This section deals with authorizing access.

Workbook Security

Currently, workbook access is either granted or denied. If users have been granted access to a workbook, they can open, modify, and commit the workbook. No distinction is made between read-write-commit, read-write, and read-only access. Workbook access is automatically granted to the user that built it, and it may be shared with multiple groups or the world.

Note: A user must have access to the workbook template in order to access the workbook, even if the workbook has world or group access rights.

Users with administrator status automatically have access to all workbook templates. By default, administrators have access to all workbooks that are saved with world access. If a workbook is saved with group access, administrators can only access the workbook if they are members of the default user group of the user who saved the workbook.

The Open dialog initially shows only workbooks owned by the current user and in domains for which the user has position level security access. This is not the same as workbook access, however, and a user may have access to workbooks saved by others in other domains by using View > Other Domains in the Open dialog by others Word or Group.

Another aspect of workbook security is the ability to set limits for the number of workbooks that a user can have saved at any given time. Limits can be set for a user per template, for a user group per template, or for a template for all users. The limits are evaluated in the above order, which means that a limit defined at user-template overrides any values defined at group-template or template. If the above limits are not defined, the default value is one billion.

The limits are checked when the workbook build process is initiated. When the limit is reached, an error message displays informing the user that the workbook build process cannot complete because the limit has been reached. The message also lets the user know what that limit is. The wizard process then terminates.

Administrative users have full access to all workbook templates, regardless of the access rights that other admin users may assign to them in the Security workbook. The administrative user can build the Security workbook to change the access right back, so the nominal assignment does not matter for administrative users.

Non-administrative users do not have access to Security template and User Administration template groups even if the administrator inadvertently assigns them access rights.

Measure Level Security

Measures have access rights; these are read-write, read-only, or denied. Measures that are read-write or read-only may be selected in the extra measures and insert measure dialogs. RPAS ensures that read-only measures are not editable by the user, and the presence of read-only measures does not affect the ability to commit a workbook.

Measure security can be specified and changed through the Security Administration workbook. The Measure Rights view allows Read Only, Deny, or Read/Write access to a measure to be specified for each user.

A workbook template can override the security of a measure, but it can only narrow the security of that measure. For example, a measure can have read-write access for a user and a template can specify that all users have read-only access to the measure when a workbook is built. However, if the measure security is read-only, the template can not expand the security of that measure to read-write. Measures that are explicitly made read-only by a workbook template are not expanded to read-write access by RPAS.

Note: Refer to the *Oracle Retail Predictive Application Server User Guide for the Fusion Client* or the *Oracle Retail Predictive Application Server User Guide for the Classic Client* for more information on the Measure Analysis workbook.

Position Level Security

Position Level Security allows access control for dimensions on a position-by-position basis. This capability is completely optional. If position level security is not explicitly defined and configured, all users in a domain have access to all positions in all hierarchies. After the position level security is defined, access to a position can be granted or denied for individual users, users in a group, or for all users.

Position level security can be defined at levels (dimensions) at or above base (such as class in the product hierarchy) in any hierarchy other than calendar. As positions are added at a level/dimension lower in the hierarchy than where the position level security is maintained, access to those positions is automatically granted if a user has access to the parent position. In other words, if security is maintained at the subclass level, users are automatically granted access to all the SKUs in a given subclass if they have access to that subclass. This includes those that were added after security was established.

Exactly one dimension in each hierarchy can be defined as the security dimension for the hierarchy. If a security dimension is defined for the hierarchy, all dimensions in the hierarchy have position level security enabled, but position security is set at or above the designated dimension. For instance, if the class dimension is designated as the security dimension, an administrator can maintain access to positions in the class dimension or at any level above class.

The enabling of position level security as well as the specification of the dimension at which position level security will be maintained are managed within the configuration used to define the domain. The RPAS Configuration Tools provide the ability to do this configuration within the Hierarchy Definition Tool. For more information on configuring position level security, consult the RPAS documentation:

- For the Fusion Client, see the information on Position Level Security in Chapter 8: *System Administration* in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- For the Classic Client, see the information on Position Level Security in Chapter 6: *System Administration* in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

Additionally, position level security can be enabled on a domain by using the hierarchyMgr utility. This utility allows specification of the security dimension without requiring modifications to the domain's configuration and the application of a

domain content patch through the `rpasInstall` process. For more information on the use of the `hierarchyMgr` utility, consult the RPAS Administration Guide.

After a security dimension is defined for a hierarchy, all users in the domain default to having access to all positions in any dimension in the hierarchy. Additionally, users automatically have access to newly added positions to a domain.

The Security Administration workbook is used to control position access for individual users, user groups, or all users (referred to as world or default access). Three views are provided in this workbook for each hierarchy with a defined security dimension. The default view controls access to positions for all users (for instance, Prod Security Default); one view controls access to positions by user group (for instance, Prod Security Group); and the last view controls access to positions by individual users (for instance, Prod Security User).

Access must be granted at all levels for a user to have access to a position. This means a position must have a value of **true** at the levels default/world, group, and user. The following table demonstrates how access is granted or denied based on all combinations of settings:

Table 5–1 Granting Access

Security Set by Position			Based on settings on left, user is Granted or Denied access
Denied = False			
Granted = True			
User	User Group	World	Resulting Access
Denied	Denied	Denied	Denied
Denied	Denied	Granted	Denied
Denied	Granted	Denied	Denied
Granted	Denied	Denied	Denied
Denied	Granted	Granted	Denied
Granted	Denied	Granted	Denied
Granted	Granted	Denied	Denied
Granted	Granted	Granted	Granted

Position-level security is used when a user selects positions in the wizard process before building a workbook. Only positions to which a user has access are available for selection in the 2-tree, which are then included in the build of the workbook.

Note that position-level security, when used for a global domain environment on the same dimension on which it is partitioned, is used to guide a user to the domain or domains that user has access to. If a user only has access to positions within a single local domain, that user will be guided there on New Workbook. If a user has access to more than one, that user will be asked and can choose based on partition-level positions.

Similarly, Open by default only lists workbooks from those domains, and a user is only shown alert counts from those domains.

Setting Proper Resource Limits

This section specifies how to set resource limits.

Workbook Template Limits Views

The Workbook Template Limit views are used to limit the number of workbooks that the user can have saved. Limits can be set for a user per template, for a user group per template, or for a template for all users. The limits are evaluated in the above order, which means a limit defined in a user-template overrides any values defined at group-template or template. If the above limits are not defined, the default value is one billion, but it is not displayed in the workbook.

The limits are checked when the user begins the workbook build process. If the limit has been reached, an error message appears that informs the user that the workbook build process cannot complete because the limit has been reached. The wizard process then terminates.

Max Domain Session Limit View

The Max Domain Session Limit view is used to limit the number of user sessions that can be attached to a single domain by all users of that domain. The limit is set at the domain level. In a global domain environment, the same limit is applied individually to each local domain and the master domain.

This limit is checked during user login. If the limit has been reached, an error message appears to inform the user that the login has failed because this limit has been reached.

Max User Session Limit View

The Max User Session Limit view is used to limit the number of concurrent user sessions that can be attached to a single domain by the same user at the same time. The limit is set per user so that the administrator can control the maximum number of concurrent sessions that are allowed for an individual user. In a global domain environment, the same limit is applied individually to each local domain and the master domain.

This limit is checked during user login. If the limit has been reached, an error message appears to inform the user that the login has failed because this limit has been reached.

Information on how to set these limits can be found in the following documentation:

- For the Fusion Client, see Chapter 8: *System Administration* in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- For the Classic Client see Chapter 8: *System Administration* in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

Dimension Modification Rights View

The Dimension Modification Rights view allows the administrator to determine which user defined dimensions, if any, a user can modify by using the Hierarchy Maintenance Workbook. The view contains a check box for each available user and dimension combination. A check mark in the cell indicates that the user is permitted to modify the specified user defined dimension. Check mark on regular dimension has no affect.

After changes are made to a user's dimension modification rights, they must be committed before they take effect.

Auditing

The RPAS_LOG_LEVEL environment variable establishes the minimum logging level used by the RPAS server. Both of the clients have the ability to override this log level (see the Auditing section of the desired client for details), however, if both this method

and `RPAS_LOG_LEVEL` are used, then the logging level with the most granularity is used.

Table 5–2 Auditing

Client Log Level	RPAS_LOG_LEVEL	Resulting Level
Debug	Info	Debug
Error	Info	Info
None	<not set>	None

That last example illustrates why you should always set the `RPAS_LOG_LEVEL` variable at the server. By setting this, you prevent the malicious user from setting a log level in the client that would prevent the server from tracking their activity.

Logs generated by `DomainDaemon` are stored in the current working directory for the `DomainDaemon` application. Once a user is authorized and their connection is moved to an `RpasDbServer` process, the log file for their activity is stored in their user directory in the domain.

Managing Sensitive Data

While `RPAS` can be configured to store any type of data, it is designed to be used with sales history, inventory, and other business related information with low security requirements. It is not intended to be used with any sensitive data such as personally identifiable information or credit card information. It does not have any mechanisms to protect this data such as encryption, and therefore should not be used in this manner.

Domain Daemon IP Filtering and Redundancy

`RPAS` supports the concept of blocking some IP addresses in a multi-homed server from being used by the `Domain Daemon`, thereby limiting the security exposure to external attacks. For example, a server might have a network port to connect to the internal network, and another network port to connect to external networks. In this scenario the `Domain Daemon` can select the port that accesses the internal network. This way the daemon is less vulnerable to external attacks.

The `Domain Daemon` also has the flexibility to serve multiple domains, and multiple domain daemons can be started on a single server (`Domain Daemon Redundancy`). For example, on a single server we can have `Domain Daemon DD1` servicing Domains `D1` and `D2`, plus `Domain Daemon DD2` also servicing Domains `D1` and `D2`. In this scenario the system will continue servicing requests even in the case where `DD1` gets compromised.

Secure Socket Layer

Secure Socket Layer (SSL), a protocol for securing network connections, is used by `RPAS` to provide secure communication between the `RPAS Client` and server processes.

This section covers the following topics:

- [Introduction](#)
- [Setting Up SSL for the Fusion Client](#)

- [Setting up SSL for the Classic Client](#)
- [Setting up SSL for the RPAS Server](#)
- [Supporting Documentation](#)

Introduction

SSL can provide two benefits to a networked application: Encryption and Authentication. Both of these benefits are typically handled by an exchange of SSL certificates. SSL certificates are based on the PKI (public-key infrastructure model), and consist of both public key certificates and private key (root) certificates. Private keys are stored in an application's key store, and are not shared with any other application. Public keys are distributed to other applications that you wish to communicate with, and are stored inside their trust stores.

SSL operates in three deployment models with varying degrees of security:

- **One-way SSL:** With one-way SSL, the server is required to present a certificate to the client, but the client is not required to present a certificate to the server. This creates an environment where the server will accept connections from any client, but the client can only connect to a single server.
- **Two-way SSL:** With two-way SSL, both the server and the client are required to present their certificates. This is typically used in an environment where a single client connects to a single server. It is possible to add additional clients and servers to this deployment model by creating additional certificates and propagating the public keys to the appropriate trust stores.
- **Anonymous SSL:** With anonymous SSL, neither the server nor the client is required to present certificates. The encryption algorithms used for the session are exchanged during the SSL handshake. This model provides encryption, but not authentication and is susceptible to man-in-the-middle attacks. Because of this, it should be limited to test environments only, and should never be implemented in production.

The `orapki` tool is used to create SSL certificates, as well as Oracle wallets. The Oracle wallet serves as both the key store and trust store for an application. Details for using the wallet and certificates can be found in the RPAS Classic Client and RPAS Fusion Client Administration Guides.

Certificates need to be signed in order to be used. In most cases, you should get your certificates signed by a Certificate Authority. Many companies have their own Certificate Authority software and can create their own trusted certificates. If this option is not available, then there are trusted vendors such as Entrust or Verisign that can provide signed certificates for you.

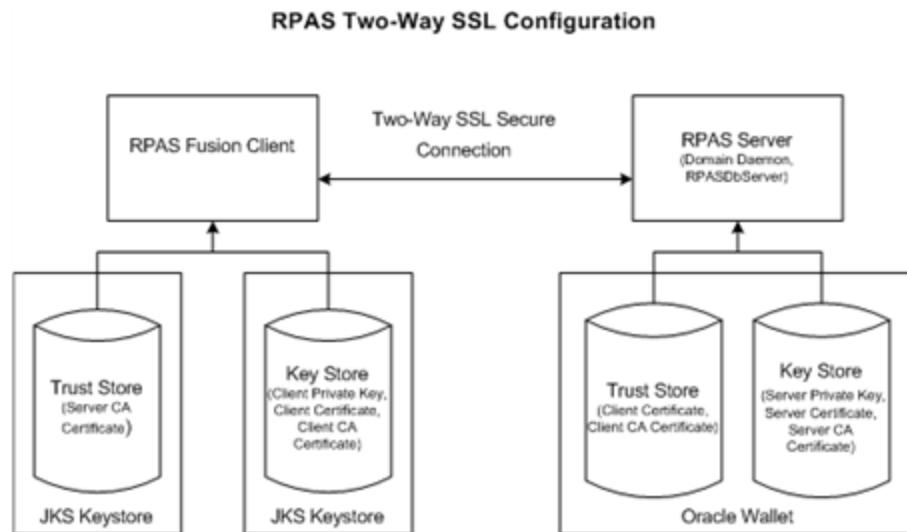
Another option is to use the `orapki` tool to create self-signed certificates. This option is ideally suited for a test environment, but should not be considered for a production environment.

Setting Up SSL for the Fusion Client

In a Fusion Client deployment, the connection between WebLogic and the RPAS server must be protected with two-way SSL. This establishes a trust chain between the WebLogic layer and the RPAS server that enables users to be authenticated externally. Since RPAS trusts connections coming from the Fusion Client, it does not need to perform redundant user authorization, which simplifies user management on the RPAS server.

The Fusion Client requires its key store and trust store to use the JKS format, which is managed by the keytool utility distributed with the standard JDK. The orapki utility can convert between the Oracle Wallet format and the JKS format, so you can either create all of your wallets with orapki, then convert the client side ones to JKS format, or use keytool to create the JKS wallets and import the certificates.

Figure 5–1 Two Way SSL Configuration



Two-Way SSL Authentication Only

The RPAS server also supports a special SSL mode which uses two-way SSL solely to authenticate the Fusion Client and establish the trust chain. All communication after the SSL handshake is done in plain-text. This option should only be used if SSL introduces performance issues into an environment, and should not be considered a secure option. Some current server hardware handles SSL at the hardware level, so the cases where this option provides benefit are rare.

Setting up the OBIEE Connection over SSL

Oracle Business Intelligence Enterprise Edition (OBIEE) reports can be displayed in the context of an RPAS workbook. An OBIEE server connection needs to be set up for this purpose. If the OBIEE server and Fusion Client server are communicating over the wide area network rather than over a private network, then it is necessary to encrypt the message traffic using one-way SSL. An SSL certificate needs to be deployed to the OBIEE server, and the Fusion Client needs to access the OBIEE server on the SSL listen port.

For the details on how to set up the SSL connection between the Fusion Client and the OBIEE server, see section 1.5.1, "How to Configure SSL and Create an Oracle BI EE Presentation Services Connection," of the *Oracle Fusion Middleware Developer's Guide for Oracle Business Intelligence Enterprise Edition*, version 11g Release 1, available at the following URL:

http://docs.oracle.com/cd/E23943_01/bi.1111/e10545/embedding_adf.htm

Following is a summary of the steps:

1. Deploy an SSL certificate to the WebLogic server hosting the OBIEE application.

The certificate can be a self-signed one if both the OBIEE and Fusion Client servers are in the control of the same business entity. Otherwise, the party controlling the OBIEE server is recommended to acquire and deploy a CA-signed SSL certificate.

2. If using a self-signed certificate, set the cn field of the identity to the host name of the OBIEE server. Turn off host name verification in the OBIEE server's SSL configuration. Enable use of JSSE SSL.
3. Turn on the SSL listen port.
4. Insert the self-signed certificate into the trust store of the JDK used by the WebLogic server on which the Fusion Client is deployed. Also, insert it into the trust store used by this WebLogic server.
5. In the OBIEE connection setup, specify https for the protocol field, and use the https URL of the OBIEE analytics application for the StaticResourcesLocation field. For more information, see the section on OBIEE connections in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

Setting up SSL for the Classic Client

Because the classic client is distributed in a many-to-one fashion, two-way SSL simply is not practical. Due to this, the classic client can connect to the RPAS server using either one-way or anonymous SSL.

One Way SSL

With One-Way SSL, the RPAS server authenticates itself to the client via a certificate. Thus, the client has assurance the server is valid

When setting up one-way SSL for the classic client, simply create a certificate for the server, export the public key certificate and put it in an Oracle Wallet. This path to this wallet should be referenced in either the foundation.fcf file, or the WebLaunch configuration. It is possible to have a single wallet out on a shared drive that all of the clients will reference, but be careful to secure the folder so that only authorized users can get access to the wallet.

Anonymous SSL

The classic client can connect to the RPAS server using anonymous SSL without creating or sharing any certificates. However, due to the lack of authentication and the susceptibility to man-in-the-middle attacks, this should only be done in a test environment.

Setting up SSL for the RPAS Server

Setting the SSL Type

When launching the DomainDaemon process, the SSL type needs to be specified on the command line. Which type you choose depends on which client you are using and the desired level of security.

Table 5–3 SSL Options

SSL Type	Client	Certificates Needed	Data Encryption	Recommended for Production
1	Classic	Server only	Y	Y
2	Fusion	Client and Server	Y	Y

Table 5–3 (Cont.) SSL Options

SSL Type	Client	Certificates Needed	Data Encryption	Recommended for Production
3	Classic	None	Y	N
4	Fusion	Client and Server	N	N

Setting Cypher Suite

In addition to setting the SSL Type, the Domain Daemon also allows you to specify the cipher suite to be used for all incoming connections. This enables a system administrator to change the suite if the one being used is compromised. The list of supported cipher suites should expand over time. When a cipher suite is deemed insecure, it will be deprecated in the next release. Whenever a deprecated cipher suite is selected, a warning message will be written to the console.

For 14.1, RPAS supports the following cipher suites:

SSL_RSA_WITH_RC4_128_MD5
Table 5–4 Cipher Suite Options

Algorithm	SSL Types	Whether Deprecated
SSL_RSA_WITH_RC4_128_MD5	1, 2, and 4	Y
SSL_RSA_WITH_3DES_CBC_SHA	1, 2, and 4	Y
SSL_RSA_WITH_RC4_128_SHA	1, 2, and 4	Y
SSL_RSA_WITH_AES_128_CBC_SHA	1, 2, and 4	N
SSL_DH_anon_RC4_128_MD5	3	Y
SSL_DH_anon_3DES_SHA	3	N

Supporting Documentation

Further information can be found in the following RPAS documentation:

- For the Fusion Client, see the information on implementing SSL in Chapter 8: *System Administration in the Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- For the Classic Client, see the information on implementing SSL in Chapter 8: *System Administration in the Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

Domain Creation and Maintenance

This chapter of the security guide covers domain creation and maintenance.

Configuration Management

The process of RPAS Application configuration can be performed by an RPAS Administrator, an application expert, a consultant or a third-party implementation team. In all cases, the process of creating or modifying the configuration of an RPAS Application is performed using a stand-alone Java application known as the RPAS Configuration Tools.

The RPAS Configuration Tools work with an XML representation of the content of a domain known as the domain configuration. Using the Configuration Tools, a domain configuration can be inspected and modified. The configuration is then used as an input to the `rpasInstall` process, which creates and modifies RPAS domains.

Because the RPAS Configuration Tools are supported only on the Windows platform, there is a need to manage the transfer of that configuration between the system being used for the configuration and the system on which the RPAS domain will be built and maintained.

Although the configuration itself does not contain any sensitive information, it does contain information about the meta-data of the domain and the processes used to maintain and modify that domain data. As such, it is prudent secure the representation of the domain contained within the configuration.

To that end, there are three areas in which the security of a configuration can be discussed. These areas are:

- Upon the system on which the configuration process is performed.
- Upon the system on which the RPAS domain is deployed.
- Upon the transfer of the configuration between the above two systems.

In each of these areas, precautions can be taken to maintain the integrity and confidentiality of the information represented within the configuration.

Securing the configuration system

As the RPAS Configuration Tools do not interact directly with an RPAS domain, they cannot be used to inspect or modify domain information. However, because the configuration describes information about the information in the domain and the processes used to maintain and modify that information, it should be viewed as proprietary information. As such it should be subjected to the appropriate considerations employed to protect other proprietary information present on user systems.

The considerations include safeguarding the physical security of systems that store proprietary information, encryption of storage devices for these systems and limiting risk of exposure through controlling access to the information contained within the configuration.

Securing the deployment system

The domain configuration is an input to the `rpasInstall` process which runs on the system on which the RPAS application is deployed. It therefore needs to be deployed to that system in order to build or modify the RPAS domain. As such, the information contained within it should be protected while it is present on the system.

The protection requirements for the information contained within the configuration are similar to the requirements for other proprietary information stored on the system. These include controlling access to the system and maintenance of system for managing rights of users on that system. The configuration itself, being a set of XML files, should be subject to file system protections to limit access to the files to appropriate users.

Securing the transfer of configurations

Configuration is performed on one or more users' individual systems. In order to build or update an RPAS domain with that configuration, it is necessary to transfer the configuration to the system upon which the domain will be deployed. As with any information transfer between systems, this transfer should be protected. Therefore, maintaining a secure environment for the configuration includes the use of secure file transfer protocols to protect the information during the transfer along with the safeguarding of the source and destination systems.

Dynamic Position Maintenance

The creation of positions within the dimensions of an RPAS domain is a process that is performed as part of an off-line process managed through the `loadHier` utility. However, the business processes performed by some RPAS applications make deferring position creation and management to an off-line process unacceptable.

Dynamic Position Maintenance (DPM) allows user to create and manage certain positions in an online process while working within a workbook. Users can create additional positions within constraints based on domain security settings and the workbook configuration and enforced by the RPAS Server instance.

Users can also modify and or delete existing positions created through DPM operations within constraints based on domain security settings and the workbook configuration and enforced by the RPAS Server instance.

Users are not allowed to modify or delete positions which the domain's security settings do not grant them access to; they may also not modify positions not allowed by the configuration of the workbook in which they are working. Finally, changes to formal positions managed through the `loadhier` process cannot be modified in any circumstances through DPM operations.

Enabling DPM functionality within a workbook involves the following process:

1. Configurator must enable DPM on particular dimensions on the domain.
2. Configurator must enable DPM on the specific workbook template.
3. Configurator or system administrator must ensure there is enough space to accommodate the volume of DPM position given by the bitsize of the dimension.

4. Administrator must give WRITE permission on that workbook template to the user.

When user creates DPM positions, they are treated as temporary positions; loadHier does not update these positions. A command line utility informalPositionMgr is available for the purpose of:

1. When user has finalized its information and wants to convert them to normal positions.
2. Application involves creating a very large number of DPM positions.

Like all RPAS server utilities. This command line utility should only have execution rights granted to system administrators.

RPAS Maintenance

Domain maintenance is a periodic operation that needs to be performed by the administrator. Its frequency depends on the degree to which the domain is subjected to hierarchy changes across time. Many of these operations can improve overall performance of data access operations - this can result in fewer contention issues which improves accessibility.

In addition, many of these operations involve removing data from the domain when that data is no longer needed by the operations being performed by the domain. This periodic cleansing serves to remove data from the system and addresses the need to retire data as a part of the data management life cycle. Some of the domain maintenance tasks that can be performed periodically are:

Purging unused and inactive hierarchy positions

All measure data within a domain is stored in either scalar or dimensional measures. As positions are introduced to the hierarchies of a domain, these positions become available for the storage of measure data. When a position is no longer needed by the domain, it can be purged. This purging, along with the use of the reindex domain, or optimize domain processes will result in the measure data associated with the retired positions being cleaned from the domain.

The purging process is performed by use of the loadHier utility purge operation. loadHier can be used to purge formal, informal, and user-defined positions from the listed hierarchies.

Cleanup of the input and processed directories

RPAS makes use of the loadhier and loadmeasure utilities to load information into the domain. These utilities read data in the form of text files that are staged to the input directory of the domain. Once the data in an input file is loaded, that file is moved to the processed sub-directory of the domain, where they are suffixed with a timestamp indicating the date and time of load.

Periodic clean up of these processed files is advisable because, over a period of time, these files can occupy sizable and valuable disk space. Furthermore, although all information contained within the files present in both in the input directory and processed sub-directory should be protected by file system security, removing files when they are no longer required removes their potential vulnerability should file system protections be compromised. User can maintain and use a script to delete these files from the input/processed folder periodically.

Reindexing domain arrays

Run the `reindexDomain` analyze option from the master domain on individual hier/dims periodically to check whether a particular hier/dim requires a bitsize increase or whether it needs to be defragged. If hierarchy operations are frequent enough and if the above check is not made, then the size of the hier/dim and the available list of physical ids may not be sufficient enough to accommodate and allocate for the incoming hierarchy load request. This can result in a loadhier failure.

`ReindexDomain` also reshapes arrays and a periodical run, in conjunction with the use of hierarchy purging, will remove inactive physical ids and can potentially reduce the size of the domain arrays and remove unneeded data from the domain.

Optimizing domain arrays

Run `optimizeDomain` periodically from master domain to improve performance and to minimize the space required by the domain data. `Optimize domain` has options to selectively defrag domain data based on database fragmentation and, in conjunction with hierarchy purging, to clean up domain data that is no longer required by the system.

A detailed description of `LoadHier`, `ReindexDomain`, and `OptimizeDomain` can be found in RPAS Admin Guide.

RPAS Integration

This chapter covers integrating information across multiple RPAS domains.

Integrating User Dictionaries

While user dictionaries cannot be shared across domains, they can be copied. This process involves exporting the users from one domain into either a users.xml or a users.db format. The users.xml file is easy to manually write or edit. However, because this file is plain-text, it cannot be used to store password information. When you import a users.xml file into a domain, you will be forced to specify a temporary password that will apply to all admin users, and another one that will apply to all non-admin accounts. These passwords automatically expire after their first use. Fusion Client deployments can skip this step by specifying the -noPassword option.

Since this approach is not automation-friendly, an administrator can pre-generate the temporary passwords by converting the user.xml file to a users.db format. This process will prompt for the passwords, then hashes them and stores them in an RPAS database. An automation process can then be set up to accept the users.db without prompting the user for anything. This step should only be used for Classic Client deployments.

Integrating Hierarchy and Dimension Data

Hierarchy information is not automatically kept in sync across domains. They can be manually sync'd up by exporting and importing hierarchy files. Domains with non-conforming hierarchies can still be synchronized by using filterHier from a master file to remove the non-conforming dimensions.

Files created by exportHier and filterHier inherit the user's default file permission (umask). The file loaded by loadHier requires only read permission but the domain's input directory and the "processed" directory under it requires write permission as loadHier will move the data files once it completes.

Integrating Measure Data

The RPAS platform stores data within an embedded BTree database located within the domain on the file system. As such, it is necessary to manage the integration of the data within an RPAS domain with other domains or with outside systems through a set of data import and export operations. The primary operations used for this are the loadhier and loadmeasure utilities for importing data and the exportHier and exportmeasure utilities for exporting data.

The RPAS platform supports the importing of data from and exporting of data to text files. These files provide an efficient method of moving large amounts of data into and out of an RPAS domain.

Because of the use of files in the load and export process, users must be aware of conventions regarding the files used for the process and how they interact with file system security. In order for RPAS utilities to import data, that data must be contained within appropriately formatted flat files staged to the input directory of the domain.

The user executing the utilities must have read and write privileges to the files used by the process. The name of the file resources used for these processes must conform to standards defined for the utility.

For more information on the data loading process, see the following documentation:

- For the Fusion Client, see Chapter 10: *Data Management* in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- For the Classic Client, see Chapter 8: *Data Management* in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

Transfer Data Utility

transferData is a regular command line utility which should only be given execution rights to system admin.

transferData requires both READ and WRITE on both the source and destination domain. transferData will acquire locks on the source domain. Therefore, online operations on the source will be affected.

For more information on the transfer data facility, see the following documentation:

- For the Fusion Client, see Chapter 9: *Hierarchy Management* in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- For the Classic Client, see Chapter 7: *Hierarchy Management* in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

ODBC/JDBC Driver

The RPAS ODBC/JDBC Driver provides a SQL interface to the Oracle RPAS Embedded Database (OREDB) which includes both domain data and workbook data.

This driver presents OREDB as a read-only relational database to ODBC and JDBC client applications for reporting or integration purposes.

The ODBC/JDBC Driver requires authentication by RPAS user name and password and supports the same position level security as the regular RPAS Server does. SSL can be configured to protect the network communications of the driver.

ODI

Oracle Data Integrator (ODI) provides a declarative design approach for defining data transformation and integration processes, resulting in faster and simpler development and maintenance. Based on its unique ELT (Extract, Load, and Transform) architecture (as opposed to the traditional ETL architecture), ODI guarantees the highest level of performance possible for the execution of data transformation and validation processes. ODI helps with the data integration and sharing among heterogeneous hardware platforms and software systems. Specifically, data integration among Relational Databases (such as Oracle DBMS) and RPAS-based applications, including data transfer between RDBMS and RPAS domains, and data transfer/sharing across multiple RPAS domains. ODI is built on several components all working together

around a centralized metadata repository. Among the components, there are graphical modules that ODI users directly interact with, and run time components (ODI Agents) that run on source and target systems.

General Considerations (applies to all integration)

- ODI logs into RPAS domains using a user name and password set up by RPAS's usermgr utility.
- ODI connects to an RPAS domain using a JDBC protocol through the ODBC data service provided by RPAS. Information about RPAS's ODBC data service can be found in the RPAS Administration Guides in section "RPAS ODBC/JDBC Driver".
- RPAS domains are read-only to ODI, ODI cannot modify domains except by running RPAS server's loadmeasure.
- When ODI reads data from an RPAS domain, it uses the domain's security, including user-level and dimension-level security settings.
- ODI keeps the following data in Oracle tables:
 - Configuration information, including domain paths and usernames/passwords.
 - Activity logs, including time of data transfer, names of measures, number of records transferred, and error messages.

RPAS application to RPAS application integration Considerations

- ODI creates an OVR or RPL file in the "input" folder of the receiving RPAS domain and then runs RPAS server's loadmeasure.
- Application-specific information (for RPAS-to-RPAS integration) is in the RPAS Apps ODI Implementation Guide.

RPAS domain to Oracle DBMS Considerations

- ODI logs into Oracle using an Oracle schema/password pair and uses the security settings corresponding to that schema.
- ODI can read from and write to the Oracle tables, although the current integration, MFP-to-RA, only writes to Oracle tables.
- More information about MFP-to-RA integration is in the Retail Analytics Installation Guide.

RETL

RETL is an Oracle program. The name is an acronym for Retail Extract Transform (and) Load. It is also called "rfx". It is used to transform the data from one system's format to the other. It needs two sets of XML schema files. One set describes the format of the incoming data; the other describes the format into which it will be transformed. A RETL data transform is typically invoked from within a shell script.

RETL requires both READ and WRITE on both the source and destination domain.

The integration between RMS and RPAS/RDF is accomplished by one system exporting data in flat files, transforming the exported file format to match the target system, and then loading the transformed data files into the target system. We currently use a program called RETL to transform the data from one system's format to the other.

The RMS to RDF integration is composed of the set of these scripts, the schema files, and the RETL program.

RPAS Hybrid Storage Architecture Option

This chapter contains information on securing the RPAS Hybrid Storage Architecture (HSA).

RPAS Data Mart Construction and Security Model

RPAS includes an optional component known as HSA. RPAS HSA allows a configurable subset of RPAS data and metadata to be stored in a group of Oracle Database tables, collectively known as an RPAS Data Mart (RDM). Here we present details on the security model of the RPAS Server to Oracle Database connection.

In order to enforce the "least privileges" model of access control, the RDM installation process will create, in its standard configuration, eight Oracle schemas (a schema is the equivalent of an Oracle Database "user," with a particular set of privileges). Only one of these schemas, the RPAS Data Mart schema, will own persistent data tables. The remaining schemas have defined access rights, as needed by particular RPAS Server processes. Oracle login details for these schema/users will be stored in an Oracle Wallet, with default permissions allowing access only by the Unix account used for RPAS Server administration.

The following table shows all schemas and their corresponding role and connection alias. While the schema names can be customized to the customer's naming standard, all the role names and connection aliases are constants.

Table 8–1 Schemas

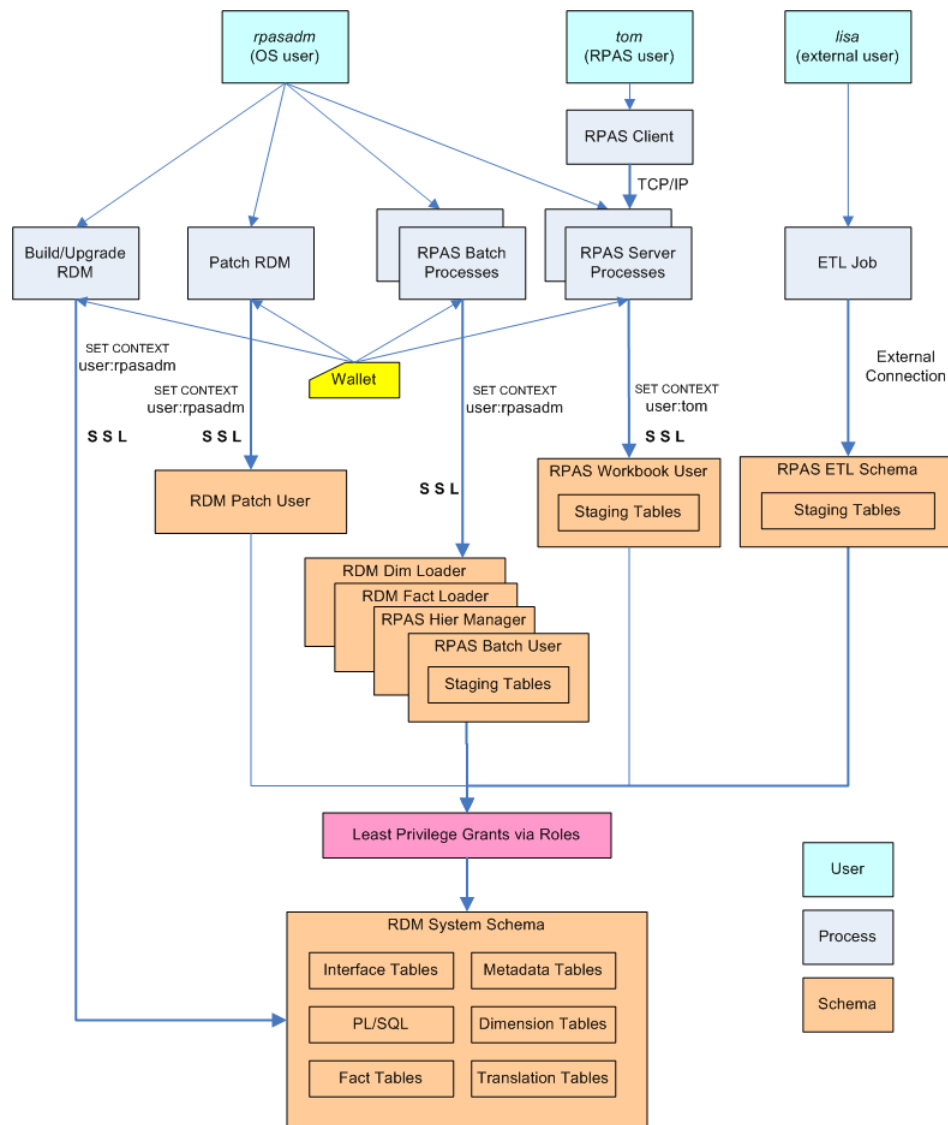
Schema (Default)	Role	DB Connection Alias
rpas_data_mart	NA	rpas_data_mart_conn
rpas_patch_user	rpas_patch_role	rpas_patch_conn
rpas_batch_user	rpas_batch_role	pas_batch_conn
rpas_dimload_user	rpas_dimload_role	rpas_dimload_conn
rpas_factload_user	rpas_factload_role	rpas_factload_conn
rpas_hiermgr_user	rpas_hiermgr_role	rpas_hiermgr_conn
rpas_wkbk_user	rpas_wkbk_role	rpas_wkbk_conn
rpas_etl_user	rpas_etl_role	rpas_etl_conn

The RPAS RDM creation process (a set of binary utilities and shell scripts) will create the required schemas and set their permissions. It will also construct the metadata and data tables required in the RPAS Data Mart schema. Options exist in the RDM creation process to allow a customer DBA to examine and customize the generated Oracle

DDL/DML scripts before the installation process proceeds, both for database layout and efficiency concerns, as well as to verify that the Database constructs meet their corporate security standards. It is important to note, however, that the permissions for the various schemas (roles) have been set as restrictively as possible, and if they are modified, it will likely break RPAS functionality.

The following diagram gives an overview of an HSA-enabled RPAS installation, with particular attention to the supported security model, as described above.

Figure 8–1 RDM Schema Security Model



External Integration APIs

In addition to RPAS Server utilities, data may also flow to and from RDM from other database applications. To facilitate this, we provide a set of External Integration APIs in the form of PL/SQL procedures. They allow a process from another RPAS application or any other customer application to load data into or extract data from an RDM in a controlled, secure manner. We provide specifications for staging tables (full details in the *Oracle Retail Predictive Application Server Administration Guide for the*

Fusion Client section on HSA features), and the external application may then call our External API procedures in PL/SQL, which will first validate the data and then load it into the appropriate RDM tables. External application code will not need (or be granted) access to the RDM data tables directly, but will only need to call our External API routines in PL/SQL.

SSL Connection

For best security practices in the connection between the RPAS Server utilities and the Oracle Database, we recommend enabling the optional SSL feature. The Oracle Database server natively supports SSL connections from clients, and the feature needs only to be enabled and configured for use with RPAS. Note that only one-way SSL is covered in this document. For two-way SSL setup, see the *Oracle Database Security Guide* at the following URL:

<http://docs.oracle.com/database/121/DBSEG/asossl.htm>

SSL Overview

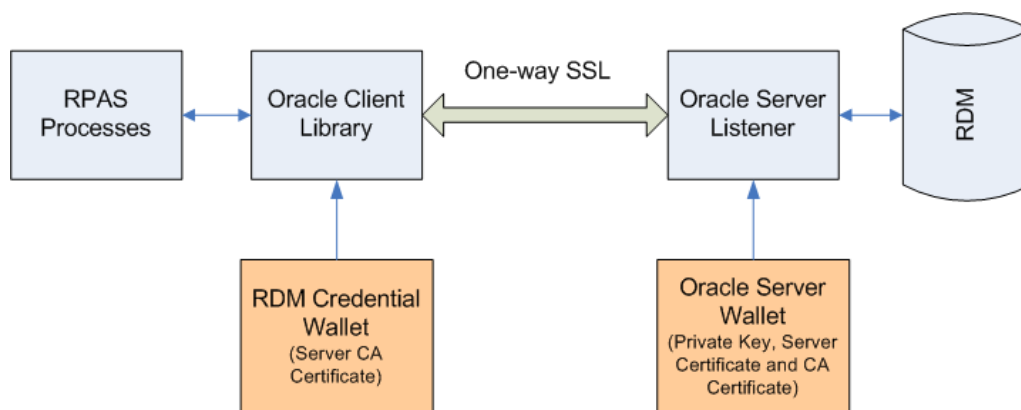
The following diagram shows the components of the SSL configuration for RDM. The client/server connection will be protected by one-way SSL, which only authenticates the server on the client side. As a result, only the Oracle server wallet is required to have a private key. The client wallet will only need to contain the server CA (Certificate Authority) certificate.

The CA certificate can be either self-signed or issued by a third-party CA. In the case of third-party CA, more than one certificate may need to be imported into the client wallet (usually called CA certificate chain).

The server side configuration is mostly done manually by the DBA. A script is provided to create the wallet if self-signed certificate is used.

The client side configuration is done through scripts by the RPAS admin.

Figure 8–2 SSL Configuration for RDM



Set Up SSL on Oracle Server

This section contains information on setting up SSL on the Oracle server side. This should be done by a DBA who has the permission to modify the configuration files of the Oracle server.

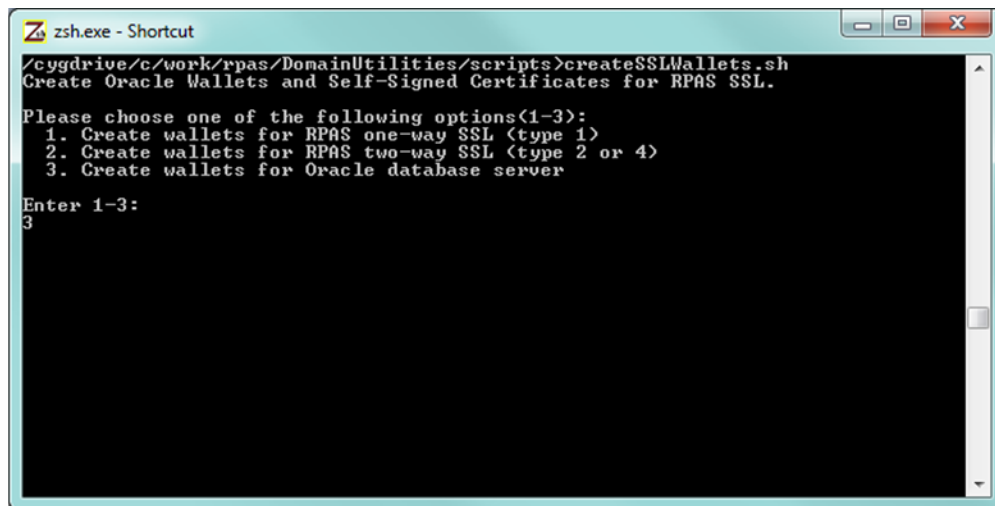
Create Oracle Server Wallet

The Oracle server wallet can be created the same way as is the RPAS server wallet for one-way SSL. The process has been documented in great detail in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client* "SSL" chapter. On UNIX, the wallet must be created by the user account which starts the Oracle database processes and for security reason the wallet is only accessible by its creator.

If self-signed CA certificate is used, the shell script "createSSLWallets.sh" provided by RPAS can be used to create the root and server wallets, private keys and all related certificates.

When running the script "createSSLWallets.sh," choose option 3 to create wallets for Oracle database server, as shown in the following figure:

Figure 8–3 Create Wallet Select Option



The script asks for the root directory where the wallets will be created, your organization name (any name such as "QA"), and passwords for the root and server wallets. In the end of a successful run, the wallet location and the path to the CA certificate file will be displayed to the user (see the following figure).

The CA certificate file can be copied to a common location and will need to be imported into the Oracle client wallet later on.

Figure 8–4 Create Wallet Complete

```

zsh.exe - Shortcut
Wallet(s) and certificate(s) have been created successfully.
The Oracle server wallet is: C:/work/rpas/DomainUtilities/scripts/mywallets/odbs
erver
The CA certificate is: C:/work/rpas/DomainUtilities/scripts/mywallets/root_chain
.txt
/cygdrive/c/work/rpas/DomainUtilities/scripts>

```

Update Oracle Server Network Configuration

The network configuration file "sqlnet.ora" normally resides in \$ORACLE_HOME/network/admin directory.

The following table lists the required settings for SSL.

Required SSL Settings in sqlnet.ora on Oracle server

```

SSL_CLIENT_AUTHENTICATION = FALSE
WALLET_LOCATION =
(SOURCE =
(METHOD = FILE)
(METHOD_DATA =
(DIRECTORY = wallet_location)
)
)

```

The SSL client authentication parameter must be set to FALSE for one-way SSL. The wallet_location is the absolute path to the Oracle server wallet as discussed in the previous section.

The following table lists the optional settings for SSL.

Optional SSL Settings in sqlnet.ora on Oracle server

```

SSL_CIPHER_SUITES= (SSL_cipher_suite1 [,SSL_cipher_suite2])
SSL_VERSION=3.0

```

For more information regarding these settings, refer to Oracle Database Security Guide at the following URL:

<http://docs.oracle.com/database/121/DBSEG/asossl.htm>

Update Oracle Listener Configuration

The listener configuration file "listener.ora" normally resides in \$ORACLE_HOME/network/admin directory. After modification, the Oracle listener must be restarted for the changes to take effect.

The following table shows the changes in bold.

Required SSL Settings in listener.ora on Oracle server

SSL_CLIENT_AUTHENTICATION = FALSE

WALLET_LOCATION =

(SOURCE =

(METHOD = FILE)

(METHOD_DATA =

(DIRECTORY = wallet_location)

)

)

LISTENER =

(DESCRIPTION_LIST =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

)

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = odbsvr-host)(PORT = 1521))

)

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCPS)(HOST = odbsvr-host)(PORT = 2484))

)

)

The wallet location and SSL settings are the same as that in the network configuration file (must be in both files). A new listener endpoint with protocol TCPS must be added to the LISTENER setting. Port number 2484 is the typical port for TCPS but other number can also be used.

Set up SSL on Oracle Client

This section contains information on setting up SSL on the Oracle client side for use by RPAS or RDM processes.

Update Schema Info Configuration The schema info configuration must have matching endpoint parameters to the Oracle listener. The protocol must be "tcps" and the port number must be the same as specified in the listener configuration.

TNS Parameters in schemaInfo.xml

```

<tns_parameters>
  <protocol>tcps</protocol>
  <host>odbserver-host</host>
  <port>2484</port>
  <server>dedicated</server>
  <service_name>myservice</service_name>
  <sid></sid>
</tns_parameters>

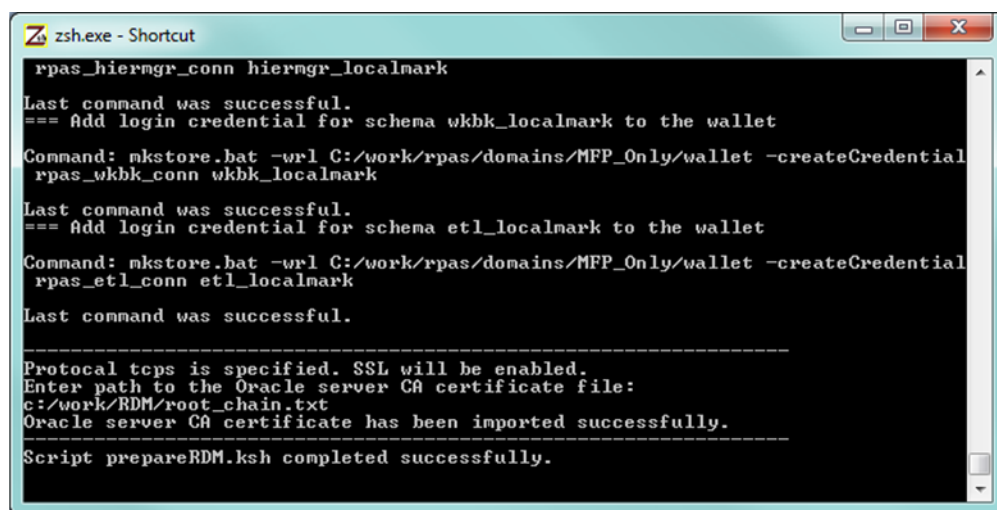
```

Import Server CA Certificate

To import the server CA certificate:

1. The prepareRDM.ksh script in the RDM build process already creates an Oracle wallet to hold the login credentials for RDM. The same wallet should be used to hold the Oracle server CA certificate. If protocol "tcps" is specified in the schema info, this same script will ask for the path to the CA certificate file and import it into the wallet automatically.
2. The following figure shows the importing of CA certificate by the prepareRDM.ksh script.

Figure 8–5 Import CA Certificate



If third-party CA is used and there is more than one file in the CA certificate chain, the user must specify the top certificate for the script to import and manually import the rest in the order of the chain using the following command. The client wallet location is the "wallet" subdirectory under the RDM repository.

```
orapki wallet add -wallet {client_wallet_directory} -trusted_cert -cert {ca_cert_chain_file} -pwd {client_wallet_password}
```

Example:

```
orapki wallet add -wallet C:/wallets/client -trusted_cert -cert C:/wallets/ca_chain2.txt -pwd clientpass1
```

Update Oracle Client Network Configuration

The network configuration file "sqlnet.ora" used by RPAS processes is created automatically by the RDM Manager during the RDM build process. This file resides under the "tns_admin" subdirectory of the RDM repository.

For your information, the following SSL settings are required.

Required SSL Settings in sqlnet.ora on Oracle Client

```
SSL_CLIENT_AUTHENTICATION = TRUE
```

```
WALLET_LOCATION =
```

```
(SOURCE =
```

```
(METHOD = FILE)
```

```
(METHOD_DATA =
```

```
(DIRECTORY = wallet_location)
```

```
)
```

```
)
```

The SSL client authentication parameter must be set to TRUE for one way SSL. The wallet_location is the "wallet" subdirectory of the RDM repository.

Update Oracle Net Service Names

The Oracle Net Service Names configuration file "tnsnames.ora" is created automatically by the RDM Manager during the RDM build process under the "tns_admin" subdirectory of the RDM repository. All entries will use the endpoint parameters specified by the schema info configuration.

Net Service Names using SSL in tnsnames.ora on Oracle Client

```
rpas_data_mart_conn =
```

```
(DESCRIPTION =
```

```
(ADDRESS_LIST =
```

```
(ADDRESS = (PROTOCOL = tcps)(HOST = odbserver-host)(PORT = 2484))
```

```
)
```

```
(CONNECT_DATA =
```

```
(SERVER = dedicated)
```

```
(SERVICE_NAME = myservice)
```

```
)
```

```
)
```

```
.
```

```
.
```

```
.
```

Test and Confirm SSL Connection

After SSL configuration is done on both Oracle server and client and the Oracle listener has been restarted, the user can use RDM Manager to verify the connection. (There are a small delay, sometimes a few minutes, between the listener is restarted and the SSL connection is up.)

```
rdmMgr -rdm rdmPath -testConnection
```

Extending and Customizing Products

The RPAS platform provides the ability for individual applications to extend the functionality provided by RPAS for application specific needs. This extension process can also be used by individual implementations to further customize the capabilities of RPAS to suit business needs.

Custom Libraries and Custom Template Libraries

There are two methods by which the functionality of the RPAS platform can be extended.

Custom Function Libraries

Custom function libraries allow the creation of new functions and expressions that extend the functions and expressions provided by RPAS to allow the expression of more complex relationships between various measures within the application.

Custom Template Libraries

Custom template libraries allow the creation of application specific workbook build processes. They allow the platform standard workbook build process to be modified or extended to handle application specific business practices.

Creating Custom Libraries and Custom Template Libraries

The process of creating a custom extension is documented in the RPAS Extension Writing Guide. For more information, see the *Oracle Retail Predictive Application Server: RPAS Extension Development Guide (Doc ID 1926977.1)* on My Oracle Support.

This document outlines the requirements and procedures for building extensions for the 13.2.1 - 14.1.x releases of RPAS through a process that conforms to Oracle Retail standards. In particular, this document describes the software tools required for each platform, including instructions for how to procure and build them when necessary. It also covers the commands, variables, and file structures of the RPAS build system. Finally, the document describes the contents of the RPAS distribution.

When creating or using custom extensions to RPAS, care should be taken to ensure that those extensions do not misuse their access to RPAS internal APIs to circumvent security measures. This includes avoiding the use of custom extensions to access or modify security information such as user account information and/or privileges.

When deploying custom extensions, the extension libraries should be granted the same file permissions as standard RPAS libraries. Consult the section on OS Level security for more information on proper permissions for RPAS libraries.

