

Oracle
Primavera
Portfolio Management System Administration Guide

19
July 2020

Contents

Overview	5
Configuring SMTP	5
Enabling LDAP	9
What is LDAP?	9
Properties That are Being Synchronized.....	11
Using the Primavera Portfolio Management Active Directory Synchronization Tool	12
Configuring Primavera Portfolio Management for Automatic, Scheduled Synchronization	13
Enabling Single Sign-on	15
Integrated Windows Authentication Windows	15
Integration with Third-Party Single Sign-On Products.....	16
Configuring Single Sign-On with Oracle Access Manager	18
Prerequisites for Configuring Single Sign-On.....	18
Installing Oracle HTTP Server	18
Configuring the Proxy Plugin Module for Oracle HTTP Server.....	19
Installing the LDAP Directory Server.....	19
Installing and Configuring Oracle Access Manager	19
Configuring Oracle Access Manager and the Oracle HTTP Server	19
Registering an Identity Store	19
Creating an Authentication Module.....	19
Configuring a Host Identifier	20
Configuring an Authentication Scheme.....	20
Protecting Your Resources.....	21
Configuring Protected Resources under an Application Domain	22
Mapping Your Authentication Scheme to Your Authentication Policy	22
Configuring Primavera Portfolio Management for Single Sign-On	23
Testing Your Single Sign-On Implementation	24
Configuring IIS for Single Sign-On	24
Enabling SAML Authentication	25
Prerequisites for Configuring Identity Federation Using SAML 2.0	26
Configuring Oracle Access Manager for Federated Identity Using SAML 2.0.....	26
Enabling Identity Federation.....	26
Creating an Identity Store for Account Linking.....	26
Enabling Automatic User Provisioning for the Local Identity Store used by Service Providers.....	27
Creating an Identity Provider Partner.....	28
Exporting SAML 2.0 Service Provider Metadata	29
Creating a SAML Authentication Policy.....	29
Assigning an Authentication Policy to Application Resources.....	30

Enabling SSL.....	31
Request a Certificate.....	31
Receive and Install the Certificate on your IIS server	31
Testing SSL and Establishing a Trust Relationship between the Certificate Authority and the Browser	31
Packing Debug Information	33
What Information is Collected?	33
Usage	33
Copyright.....	35

Overview

After installing or upgrading Primavera Portfolio Management, use this guide to enable supporting technologies, before users begin to work with the application.

Configuring SMTP

You can configure Primavera Portfolio Management to work properly with an organizations' SMTP server under a variety of circumstances.

For more information about these parameters, their use and effects, see the Microsoft documentation at <https://technet.microsoft.com/en-us/library/dn592151%28v=exchg.150%29.aspx>.

Notes:

- The URL-like strings shown below in the explanation of each registry value are not actual URLs. They are schema identifiers used by Microsoft. To find more information about a particular schema identifier, search the Microsoft web site for the particular schema identifier.
 - In a distributed installation of PPM ("Scale Out"), the below described settings must be made on all Primavera Portfolio Management (front and back-end) application servers.
 - All registry values discussed below should be inserted into the Registry under the key:
HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\Primavera Portfolio Management\ Portfolios\Server\UI.
-

The following registry values control all aspects of integration with SMTP:

Registry Value "SMTP Mode" (DWORD)

Data: mode of operation of SMTP Service.

Default: 1 (CdoSendUsing.cdoSendUsingPickup)

Other values: 2 (CdoSendUsing.cdoSendUsingPort)

Explanation: This value corresponds to the *sendusing* schema field.

Registry Value "SMTP Server Name" (String)

Data: full name of the organization's main SMTP Server.

Default: "localhost" Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort). This value corresponds to the *smtpserver* schema field.

Registry Value "SMTP Server Port" (DWORD)

Data: port to use to connect to the SMTP Server.

Default: 25 Decimal. Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort).

This value corresponds to the *smtpserverport* schema field.

Registry Value "SMTP Authentication" (DWORD)

Data: authentication mode to be used with the SMTP Server. Default: 0 (CdoProtocolsAuthentication.cdoAnonymous)

Other values: 1 (CdoProtocolsAuthentication.cdoBasic) and 2 (CdoProtocolsAuthentication.cdoNTLM)

Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort).

This value corresponds to the *smtpauthenticate* schema field.

Registry Value "SMTP Username" (String)

Data: username for authentication with the SMTP Server, only for basic authentication. Default: ""

Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort), and the value for "SMTP Authentication" is 1 (CdoProtocolsAuthentication.cdoBasic). This value corresponds to the *sendusername* schema field.

Registry Value "SMTP Password" (String)

Data: password for authentication with the SMTP Server, only for basic authentication. Default: ""

Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort), and the value for "SMTP Authentication" is 1 (CdoProtocolsAuthentication.cdoBasic). This value corresponds to the *sendpassword* schema field.

Registry Value "SMTP Server Pickup Directory" (String)

Data: if mode is cdoSendUsingPickup, the pickup directory for the SMTP Service. May be a full path or a path without drive letter. Note that the PROSIGHT SYSTEM user must have full permissions to the directory specified.

Default: "\\NetPub\\mailroot\\pickup"

Explanation: This value is relevant only if the value for "SMTP Mode" is 1 (CdoSendUsing.cdoSendUsingPickup). This value corresponds to the *smtpserverpickupdirectory* schema field.

Registry Value "SMTP Connection Timeout" (DWORD)

Data: when connecting to the SMTP Server, the timeout value for the connection in seconds. Default: 60 Decimal

Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort). This value corresponds to the *smtpconnectiontimeout* schema field.

Registry Value "SMTP Use SSL" (DWORD)

Data: when connecting to the SMTP Server, whether to use SSL. Note that SSL can be used only if an appropriate certificate is installed on the application server, and the mail server configured using the value for "SMTP Server Name" is configured to authenticate and accept SSL connections from clients with such certificates.

Default: 0 (do not use SSL)

Other values: 1 (use SSL)

Explanation: This value is relevant only if the value for "SMTP Mode" is 2 (CdoSendUsing.cdoSendUsingPort). This value corresponds to the *smtpusessl* schema field.

For more details, see [https://msdn.microsoft.com/en-us/library/ms527294\(v=exchg.10\).aspx](https://msdn.microsoft.com/en-us/library/ms527294(v=exchg.10).aspx)

Enabling LDAP

You can integrate Primavera Portfolio Management with the Lightweight Directory Access Protocol (LDAP). This integration enables organizations to manage users and user groups to be used within PPM, in one central location.

In This Section

What is LDAP?	9
Properties That are Being Synchronized	11
Using the Primavera Portfolio Management Active Directory Synchronization Tool. 12	
Configuring Primavera Portfolio Management for Automatic, Scheduled Synchronization.....	13

What is LDAP?

Lightweight Directory Access Protocol (LDAP) is a system and protocol that enables various applications to exchange information with an organization's central directory of information. Several vendors offer implementations of LDAP; Microsoft's LDAP implementation is called Active Directory. While the Primavera Portfolio Management Active Directory Synchronization Tool should work properly with any compliant LDAP system, it has been tested only with Microsoft Active Directory.

LDAP implementations such as Microsoft Active Directory allow organizations to define, organize, modify and manage objects such as domains, organizational units, servers and computers, groups of users and users, and others, in one central location. LDAP-aware applications can connect to the central LDAP directory and manage objects that are relevant to them there, or they can periodically connect to the central LDAP directory and synchronize their local lists of relevant objects with the objects found in LDAP.

Note that the functionality of Single Sign-On is not handled through LDAP. LDAP is simply a repository of objects, including users. Allowing those users to login once and then have access to all of their (authorized) applications is a different issue. For more information about the support for Single Sign-On in PPM, see “**Enabling Single Sign-on** (on page 15)”.

LDAP systems arrange the objects of an entire enterprise in tree-like structures. Usually it is not appropriate for specific applications to access or synchronize with the entire content of the LDAP system. A given application typically looks at a specific sub tree within the LDAP system.

Synchronization typically takes place based on a strict “parent-child” pattern: the LDAP system is the “parent”, and the application is the “child”.

PPM follows this same pattern and hence can be configured to synchronize with a sub tree residing under a specific container in the LDAP system. Any object within the LDAP system can be considered a container for this purpose; therefore, PPM can be made to synchronize with a sub tree residing under an Organizational Unit, a group, a folder, or any other object. When synchronizing, all users and user groups residing under this container will be created within PPM (if they did not exist) or will be updated to have the same properties as their counterparts in LDAP (if they already existed). Users in PPM that do not exist under the specified container in LDAP will be disabled in PPM, whereas PPM groups that do not exist under the specified LDAP container will be deleted in PPM (except for specific users and groups that the synchronization process was instructed to ignore).

Note that for the purposes of LDAP synchronization, users that are members of a group which is going to be synchronized according to the rules outlined above, will be synchronized, even if the users themselves really reside elsewhere in the LDAP system (i.e. reside under a different Organizational Unit). In other words, users belonging to a group are considered to “reside” under that group, even though, strictly speaking, the user objects reside elsewhere and the group only contains a reference to these objects.

The above is not true for user groups: user groups that are members of another user group which is going to be synchronized according to the rules outlined above, will not be synchronized unless they themselves also reside under the container that PPM is going to synchronize with. In other words, user groups belonging to a user group will be synchronized only if these groups actually reside under the specified LDAP container.

For example, let's assume the following hierarchy:

PPM Root Container

User One

User Two

User Three

Group 1 Residing under PPM's Root Container

Group 2 Residing under PPM's Root Container

- ▶ “PPM Root Container” is a container created somewhere in the LDAP hierarchy;
- ▶ “User One”, “User Two” and “User Three” are users that reside under the “PPM Root Container”;
- ▶ “User Four” and “User Five” (not shown above) are users residing elsewhere in the LDAP hierarchy;
- ▶ “Group 1 Residing under PPM's Root Container” and “Group 2 Residing under PPM's Root Container” are user groups that reside under the PPM's LDAP Root Container”;
- ▶ “Group 3 Not Residing Under PPM's Root Container” (not shown above) is a group that resides elsewhere in the LDAP hierarchy;
- ▶ “Group 1 Residing Under PPM's Root Container” has as members: “Group 2 Residing Under PPM's Root Container”, “Group 3 Not Residing Under PPM's Root Container”, “User One” and “User Four”;
- ▶ “Group 2 Residing Under PPM's Root Container” has as member: “User Two”;
- ▶ “Group 3 Not Residing Under PPM's Root Container” has as members: “User Three” and “User Five”;

In this example, the following users will be created/updated in PPM:

“User One”, “User Two”, “User Three” and “User Four”.

The following user groups will be created/updated in PPM:

“Group 1 Residing Under PPM's Root Container” and “Group 2 Residing Under PPM's Root Container”.

Note that “User Five” will not be created in PPM, and if this user already existed it will be deleted.

Also note that “Group 3 Not Residing Under PPM's Root Container” will not be created in PPM, and if it already existed it will be deleted. Within PPM, “Group 1 Residing Under PPM's Root Container” will contain “Group 2 Residing Under PPM's Root Container” and “User One” and “User Four”. “Group 2 Residing Under PPM's Root Container” will contain “User Two”. “User Three” will not be residing in any user group in PPM because the group it is a member of (“Group 3 Not Residing Under Primavera Portfolio Management's Root Container”) does not reside under the PPM's LDAP Root Container”.

Synchronization is one-way only: users created in PPM that do not exist under the specified container in LDAP will be disabled upon the next synchronization. Properties of users and/or groups that were updated in PPM will be overwritten by the values of these same properties in LDAP upon the next synchronization.

Properties That are Being Synchronized

The following table shows which properties of Primavera Portfolio Management users are being synchronized with which LDAP properties:

PPM group property	LDAP user property
Login*	sAMAccountName
Whether the user is a user or a contact	ObjectType.Contact
First Name	givenName
Last Name	sn
Title	title
Department	department
Company	company
Telephone number	telephoneNumber
Fax number	facsimileTelephoneNumber
Address	concatenated from LDAP properties "streetAddress", "postOfficeBox", "l", "st", "postalCode", and "co"
Email address	mail

Note: a PPM user and a LDAP user are considered the same user (and

will be synchronized) if the PPM user's login name matches the LDAP user's SAMAccountName property.

An existing PPM user that is not present under the LDAP container which is being synchronized with, or which is present but disabled in LDAP, will be disabled, unless listed as a user that should be ignored during LDAP synchronization.

An existing PPM user that is present under the LDAP container which is being synchronized with and is enabled in LDAP, will be enabled.

Users that exist and are enabled in LDAP but do not exist in PPM will be created in PPM, up to the licensed number of named users.

The following table shows which properties of PPM groups are being synchronized with which LDAP properties:

PPM group property	LDAP group property
Name*	name
Description	description

*Note: a PPM group and a LDAP group are considered the same group (and will be synchronized) if the PPM group's name matches the LDAP group's name property.

An existing PPM group that is not present under the LDAP container which is being synchronized with, will be deleted in PPM, unless listed as a group that should be ignored during LDAP synchronization.

Groups that exist in LDAP but do not exist in Primavera Portfolio Management will be created in PPM. All members of the group in LDAP will be made members of the group in PPM. Members of the group in PPM that are not members of the group in LDAP will be removed from the group in PPM.

Groups in LDAP which have the property isCriticalSystemObject set to TRUE will not be synchronized with PPM. This includes most built-in groups such as Domain Admins, Domain Users, etc.

Using the Primavera Portfolio Management Active Directory Synchronization Tool

Note: In a distributed installation of Primavera Portfolio Management ("Scale Out"), you can use the Active Directory Synchronization Tool from any one of the PPM application servers. This tool is used for one-time, user-driven synchronization with LDAP.

- 1) Double-click the file **psActiveDirectorySync.exe** located in the "bin" subfolder of the PPM installation directory. By default, this is located at C:\Program Files\Oracle\Primavera Portfolio Management\Portfolios\bin.
- 2) In the **Active Directory** pane, enter the following:
Login: The Windows login name of a user with access to the LDAP server
Password: The password of the user with access to the LDAP server.

Server Name: The name of the LDAP server.

LDAP Root Container: The container within LDAP that contains the sub tree of users and user groups relevant to PPM. The container is specified using LDAP syntax such as `cn=container,dc=domain,dc=com`. Any object can be specified in this field using this syntax.

- 3) In the **Primavera Portfolio Management** pane, enter the following:

Login: The PPM login name of a user with administrative rights to PPM.

Password: The PPM password of this user.

Ignore PPM Users: A list of PPM login names, separated by semicolons, that will not be synchronized with LDAP. By default, the login name of the PPM System Administrator, "admin", is listed here, to avoid causing this user to become disabled upon running a LDAP synchronization.

Ignore PPM Groups: A list of PPM group names, separated by semicolons, that will not be synchronized with LDAP. By default, the group name "Administrators" is listed here to avoid causing this group to either get deleted or contain the names of (Windows) Administrators as defined in LDAP.

License Type: The license type that will be assigned to each new user synchronized with LDAP. The options are:

FULL – new users will be assigned full licenses.

READ – new users will be assigned read-only licenses.

FORM – new users will be assigned forms-only licenses.

- 4) When the required information in all fields has been entered, click **Synchronize** to start the LDAP synchronization process. Click **Cancel** to exit the tool without performing synchronization.

Note that the information entered in these fields is not saved to the database and does not affect any other, scheduled, LDAP synchronization.

Configuring Primavera Portfolio Management for Automatic, Scheduled Synchronization

Note: Use the Schedule Tasks tool in a distributed installation of Primavera Portfolio Management ("Scale Out") from any one of the PPM application servers, to configure for automatic scheduled LDAP synchronization.

- 1) Double-click the file **psScheduleTasks.exe** located in the "bin" subfolder of the PPM installation directory. By default, this is located at `C:\Program Files\Oracle\Primavera Portfolio Management\Portfolios\bin`.
- 2) Log in with a PPM user name and password.
- 3) On the **Schedule Task** window:
 - a. In the list of tasks, select **10. Active Directory Sync**.
 - b. In the **Next Run** field, select the date for the next synchronization.
 - c. In the **First run starts at** field, select the time of day at which the synchronizations should start running.

- d. In the **Frequency** field, select the frequency with which the synchronization should take place.
 - e. Click **Additional Parameters**.
- 4) On the Task Parameters window:
- a. Select **Enable Active Directory Sync**. To temporarily disable this task, it is possible to clear this checkbox without the need to clean out the other fields.
 - b. In the **Active Directory Login** field, enter the Windows login name of a user with access to the LDAP server.
 - c. In the **Active Directory Password** field,, enter the password of the user with access to the LDAP server.
 - d. In the **Server Name** field, enter the name of the LDAP server with which the synchronization should take place.
 - e. In the **LDAP Root Container** field, enter the container within LDAP that contains the sub tree of users and user groups relevant to PPM. The container is specified using LDAP syntax such as cn=container,dc=domain,dc=com. Any object can be specified in this field using this syntax.
 - f. In the **Login** field, enter the PPM login name of a user with administrative rights to PPM.
 - g. In the **Password** field,enter the PPM password of this user.
 - h. In the **Ignore PPM Users** field,enter a list of PPM login names, separated by semicolons, that will not be synchronized with LDAP. By default, the login name of the PPM System Administrator, "admin", is listed here, to avoid causing this user to become disabled upon running a LDAP synchronization.
 - i. In the **Ignore PPM Groups** field,enter a list of PPM group names, separated by semicolons, that will not be synchronized with LDAP. By default, the group name "Administrators" is listed here to avoid causing this group to either get deleted or contain the names of (Windows) Administrators as defined in LDAP.
 - j. In the **License Class** field, enter the type of license that should be assigned to new users.
 - k. When all fields have been defined as required, click **OK** to accept the entered values and close the Task Parameters window.
- 5) On the **Schedule Tasks** window, click **OK** to accept all entered values and schedule the LDAP synchronization task as specified.
- 6) Restart PPM for the changes to take effect.

Enabling Single Sign-on

Login is the action the user takes to authenticate and gain access to a desired application. Single Sign-On (SSO) is a process that gives users the ability to access multiple protected resources (web pages and applications) with a single authentication, and eliminates the need for additional or different logins to access other applications at the same (or lower) authentication level during the same session.

You can enable Primavera Portfolio Management to work with three different types of SSO:

- 1) Integrated Windows Authentication
- 2) Oracle Access manager
- 3) Integration with 3rd party Single Sign-On products

Note: In a distributed installation of PPM (“Scale Out”), the below described settings must be made on all PPM application servers.

In This Section

Integrated Windows Authentication Windows	15
Integration with Third-Party Single Sign-On Products	16
Configuring Single Sign-On with Oracle Access Manager.....	18

Integrated Windows Authentication Windows

The Primavera Portfolio Management application can be integrated with Microsoft Windows domain authentication, such that a user, who has been authenticated by a Microsoft Windows domain controller, is automatically authenticated with PPM also. By enabling this functionality your users will not be prompted for their user names and passwords by PPM. They will be automatically logged into PPM.

Note: This feature works only for users who have logged into a Microsoft Windows domain via Windows authentication and whose Windows user name is identical to their login name in PPM. If these do not match exactly, when accessing PPM the user will be presented with the regular PPM login dialog screen, together with the message “invalid username/password”.

This topic outlines the process of enabling Integrated Windows Authentication for PPM.

- 1) Open Internet Information Services (IIS) Manager by clicking **Start, Programs, Administrative Tools**, and **Internet Information Services (IIS) Manager**.
- 2) Navigate to the PPM virtual directory and select it.
- 3) Select **Authentication**.
- 4) Under **Actions**, select **Open Feature**.

- 5) On the **Authentication** screen:
 - a. Select **Windows Authentication**.
 - b. Under **Actions**, click **Enable**.
 - c. Select each of the other authentication types.
 - d. Under **Actions**, click **Disable**.
- 6) Exit the Internet Information Services (IIS) Manager.

Integration with Third-Party Single Sign-On Products

The Primavera Portfolio Management application can be integrated with 3rd party Single Sign-On (SSO) products, such that a user, who has been authenticated by a 3rd party SSO product, will automatically be authenticated with PPM as well. By enabling this functionality users will not be prompted for their usernames and passwords by PPM, but will be automatically logged into the PPM application without the need to use the login dialog screen.

Note: This feature works only for users who have logged in using the 3rd party SSO product and whose 3rd party SSO product user name is identical to their login name in PPM. If these do not match exactly, when accessing PPM the user will be presented with the regular PPM login dialog screen, together with the message "invalid username/password".

This chapter outlines possibilities for enabling integration with 3rd party SSO products for PPM. For the exact procedure to follow, please refer to the 3rd party SSO product manuals.

Third Party SSO Product Requirements

In order to be able to integrate with PPM, the 3rd party SSO product must be able to fulfill the following requirements:

- 1) Ability to intercept access to PPM web server through your browser.
- 2) Ability to set a HTTP header variable to a fixed value
- 3) Ability to set another HTTP header variable to the name of the authenticated user

Note: PPM does not accept "cookies" as an authentication method.

PPM Configuration for Integration with Third-Party SSO Product

PPM can be configured to accept any HTTP header variables. The following registry values control the names and values of the HTTP header variables used for integration:

Note that all registry values discussed below may be inserted into the Registry under the key: HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\Primavera Portfolio Management\Portfolios\Server\UI

Registry Value "SSO Authentication Type HTTP Header Name" (String). Data: name of the HTTP header variable containing the authentication type.

Default: "AUTH_TYPE"

Explanation: The type of SSO Authentication is carried in the HTTP header variable called "AUTH_TYPE". If an SSO product is unable to use this particular header variable, then this registry entry can be used to cause PPM to look at a different HTTP header variable.

Note: PPM uses the "raw" HTTP header interface. However, it is recommended to also configure PPM with an "HTTP_" prefix.

Registry Value "SSO Authentication Type HTTP Header Value" (String). **Data:** value to be assigned to the "SSO Authentication Type HTTP Header", which indicates that the SSO product will perform user authentication.

Default: "Negotiate"

Explanation: The "Negotiate" value in the HTTP header variable "AUTH_TYPE" is interpreted by PPM to mean that a SSO product is responsible for user authentication. If an SSO product assigns a different value to the HTTP header variable, then this registry entry can be used to cause PPM to accept the value which the SSO product assigns.

Registry Value "SSO Authenticated User HTTP Header Name" (String). **Data:** name of the HTTP header variable containing the user name of the user authenticated by the SSO product.

Default: "LOGON_USER"

Explanation: The name of the user authenticated by the SSO product. It should be placed in a HTTP header variable called "LOGON_USER". If an SSO product is unable to use this header variable, then this registry entry can be used to cause PPM to accept the name of the authenticated user in a different HTTP header variable.

Note: PPM uses the "raw" HTTP header interface. However, it is recommended to also configure PPM with an "HTTP_" prefix.

Registry Value "SSO Logout URL" (String). **Data:** The value to be assigned to the "SSO Logout URL", which indicates that the PPM should be redirected to the SSO Logout screen.

Default: "Logout URL of SSO"

Explanation: The value of SSO Logout URL in the HTTP header variable is interpreted by PPM to redirect it to the Logout screen.

Example

SSO products such as OAM and Netegrity SiteMinder can be configured to set up custom HTTP headers. In Netegrity SiteMinder see the SiteMinder log file. These "custom HTTP headers", when seen by PPM, are prefixed by "HTTP_". Therefore, a typical Netegrity SiteMinder setup is as follows:

- ▶ Configure Netegrity SiteMinder to create a custom HTTP header called "AUTH_TYPE", whose value is set to "Negotiate". Also configure Netegrity SiteMinder to create a "Response Attribute" custom HTTP header called "AUTH_USER", and set its value to the login name (id) of the authenticated user.
- ▶ Configure the appropriate Netegrity SiteMinder policy to send.
- ▶ Configure PPM accordingly by creating the following registry string values:

- ▶ **SSO Authentication Type HTTP Header Name** should have a value of HTTP_AUTH_TYPE.
- ▶ **SSO Authentication Type HTTP Header Value** should have a value of Negotiate.
- ▶ **SSO Authenticated User HTTP Header Name** should have a value of HTTP_AUTH_USER.
- ▶ **SSO Logout URL** should have a value of SSO Logout URL.

Configuring Single Sign-On with Oracle Access Manager

There are several supported authentication schemes that you can use to enable SSO for PPM, such as: Form (LDAP), X509 (Certificate), WNA (Windows Native Authentication); however, this section covers the necessary procedures for form based authentication. If you prefer to use one of the other authentication schemes, you should review *Managing Access Manager SSO, Policies, and Testing* in the *Fusion Middleware Administrator's Guide for Oracle Access Management* guide.

Consider the following workflow when configuring Oracle Access Manager for PPM SSO:

- 1) Ensure that you have completed the prerequisites for configuring SSO. For more information about the prerequisites, see ***Prerequisites for Configuring Single Sign-On*** (on page 18) and the *Tested Configurations* documents for each product that you plan to configure with SSO or SAML.
- 2) Implement SSO in Oracle Access Manager using the procedures that are documented in ***Configuring Oracle Access Manager and the Oracle HTTP Server*** (on page 19).
- 3) Configure PPM for SSO using the steps in ***Configuring Primavera Portfolio Management for Single Sign-On*** (on page 23).

Prerequisites for Configuring Single Sign-On

The following prerequisites must be completed for all of the Primavera applications.

Installing Oracle HTTP Server

To learn more about installing Oracle HTTP Server 12c, see:

Installing and Configuring Oracle HTTP Server, which can be found on Oracle Technical Network at <http://docs.oracle.com/middleware/1213/core/WTINS/toc.htm>.

Note: After you navigate to *Installing and Configuring Oracle HTTP Server*, see the following chapters:

- Chapter 1: *Planning Your Oracle HTTP Server Installation*
 - Chapter 2: *Installing the Oracle HTTP Server Software*
-

The following documents on My Oracle Support:

How To Install Oracle HTTP Server(OHS)12c In Standalone And Colocated (Managed through WebLogic Server) Domains (Doc ID: 1575618.1)

How To Install Oracle HTTP Server(OHS)12c In Colocated (Managed through WebLogic Server) Domains (Doc ID: 1606339.1)

Note: Oracle Access Manager 12c is bundled with the Oracle HTTP Server 12c download. When you install Oracle HTTP Server 12c, you can install Oracle Access Manager 12c at the same time.

Configuring the Proxy Plugin Module for Oracle HTTP Server

To learn more about configuring OHS as a proxy, see:

Using Oracle WebLogic Server Proxy Plug-Ins 12.1.3, which can be found on Oracle Technical Network at <http://docs.oracle.com/middleware/1213/webtier/PLGWL/oracle.htm#PLGWL4330>.

Note: After you navigate to *Using Oracle WebLogic Server Proxy Plug-Ins 12.1.3*, see the following sections:

- Section 2.1: *Prerequisites for Configuring the WebLogic Proxy Plug-In*
 - Section 2.4: *Configuring the WebLogic Proxy Plug-In Manually*
-

Installing the LDAP Directory Server

See the *Tested Configurations* document for the supported versions of LDAP.

Installing and Configuring Oracle Access Manager

To install and configure Oracle Access Manager, see the *Installing and Configuring Oracle Identity and Access Management* chapter of the *Oracle Fusion Middleware Installation Guide for Oracle Identity and Access Management* guide.

Configuring Oracle Access Manager and the Oracle HTTP Server

Complete the following tasks in Oracle Access Manager to configure SSO for PPM:

Registering an Identity Store

Oracle Access Manager needs to be configured with a data source that will hold a connection to your LDAP directory server.

For more information about managing data sources for Oracle Access Manager, see the *Managing Data Sources* and *Introduction to Managing Common Data Sources* sections in the *Fusion Middleware Administrator's Guide for Oracle Access Management* guide.

To configure a data source in Oracle Access Manager to connect to an LDAP server, follow the instructions in *Managing User Identity Stores* in the *Fusion Middleware Administrator's Guide for Oracle Access Management* guide.

Creating an Authentication Module

After you have your directory store registered in Oracle Access Manager, you need to create an authentication module that links to it. The authentication module needs to be linked to an authentication scheme.

To create an authentication module:

- 1) Log in to the Oracle Access Manager Administration Console.
- 2) Navigate to the **Application Security** tab.
- 3) Under **Plugins**, click on **Authentication Modules**.
- 4) Click **+Create** and then select **Create LDAP Authentication Module**.
- 5) In the Create LDAP Authentication Module dialog box, do the following:
 - a. In the **Name** field, enter a name for the authentication module that you want to create.
 - b. From the **User Identity Store** list, select the link that matches the LDAP data source that you created.
 - c. Click **Apply** to save the changes.

Configuring a Host Identifier

Oracle Access Manager needs to be configured with a host identifier. To create a new host identifier, follow the instructions in *Managing Host Identifiers* section of the *Fusion Middleware Administrator's Guide for Oracle Access Management*.

To *confirm* that you have a configured Host Identifier:

- 1) Log in to the Oracle Access Manager Administration Console.
- 2) Navigate to the **Application Security** tab.
- 3) Under **Access Manager**, click on **Host Identifier**.
- 4) Click **Search**.
- 5) Select the link for your Host Identifier.
- 6) In the **Host Identifier** dialog box, do the following:

In the Host Name Validation list, ensure that the name of your host identifier under Host Name matches the host identifier that you setup when you registered your WebGate with Oracle Access Manager.

Note: The host identifier field is a value that replaces *hostname:port* in requests from the web server to the Oracle Access Manager.

Configuring an Authentication Scheme

Once you have a data source that stores a connection to your LDAP server, you have to create an authentication scheme for PPM. An authentication scheme is a named component that defines the challenge mechanism that is required to authenticate a user. For example, the authentication scheme determines if you will use form based authentication, basic authentication, Windows Native Authentication, and so on.

To create a new authentication scheme, follow the instructions in the *Managing Authentication Schemes* section of the *Fusion Middleware Administrator's Guide for Oracle Access Management*.

If you already have an authentication scheme, you can use it as a template to provide form based authentication for your applications.

To duplicate an authentication scheme:

- 1) Log in to the Oracle Access Manager Administration Console.
- 2) Navigate to the **Application Security** tab.
- 3) Under **Access Manager**, click on **Authentication Schemes**.
- 4) Click on **LDAP Scheme**.
- 5) Click on the  **Duplicate** icon.
- 6) In the **Authentication Schemes** dialog box, complete the following:

Note: When you duplicate an existing authentication scheme and are use it as a template for your Primavera applications, many of the fields in the Authentication Scheme dialog box will be prepopulated. You do not need to alter the following fields:

- Description
 - Authentication Level
 - Default
 - Challenge Method
 - Challenge Redirect URL
 - Challenge URL
 - Context Type
 - Context Value
 - Challenge Parameters
-

- a. In the **Name** field, enter a name for your authentication scheme.
- b. In the **Authentication Module** field, select the authentication module that you created for your LDAP data source.
- c. Click **Apply** to create the new authentication scheme.

Protecting Your Resources

After you have Oracle Access Manager configured with a connection to your LDAP server, a host identifier that links to your Oracle HTTP Server WebGate for Oracle Access Manager, and an authentication scheme, you need to create an application domain so that you can setup policies to protect your resources and to configure a policy that points to the authentication scheme that you want to use.

For more information about resource policies, refer to the *Managing Policies to Protect Resources and Enable SSO* section of the *Fusion Middleware Administrator's Guide for Oracle Access Management*, which can be found at the following URL. For the steps to protect your resources, refer to **Configuring Protected Resources under an Application Domain** (on page 22).

Oracle recommends that you protect your context roots with the following conventions:

► */context*

For example, the connection `http://<host_name>:<port>/<context>` will be recognized as a protected resource.

▶ */context/*

For example, the connection `http://<host_name>:<port>/<context>/` will be recognized as a protected resource.

▶ */context/** or /context/.../***

For example, the connection `http://<host_name>:<port>/<context>/<additional_context_roots>` will be recognized as a protected resource.

Use the following context root for PPM:

`/proSight/**`

`"/ProSight/**"`

Configuring Protected Resources under an Application Domain

To protect the context root of PPM:

- 1) Log in to the Oracle Access Manager Administration Console.
- 2) Navigate to the **Application Security** tab.
- 3) Under **Access Manager**, click on **Application Domains**.
- 4) Click **Search** and then select the application domain.
- 5) Navigate to the **Resources** tab.
- 6) Click on the **+ Create** icon to create new resource as follows:
 - a. In the **Type** field, select *HTTP*.
 - b. In the **Host Identifier** field, select the name of the host identifier that you created.
 - c. In the **Resource URL** field, enter a protected context root. For example, see **Protecting Your Resources** (on page 21).
 - d. In the **Protection Level** field, select *Protected*.
 - e. In the **Authentication Policy** field, select *Protected Resource Policy*.
 - f. Click **Apply**.
- 7) In the **Resources** tab, highlight the entire field to the right of the Resource Type column and then close it.
- 8) Click  **Close**.
- 9) Repeat this procedure for each protected resource.

Mapping Your Authentication Scheme to Your Authentication Policy

After you create your resources and tie them to the authentication policy that was created for you when the application domain was created (for example, protected resource policy), you need to map your authentication scheme to your authentication policy so that your resources will present the login form to users for authentication:

- 1) Log in to the Oracle Access Manager Administration Console.
- 2) Navigate to the **Application Security** tab.
- 3) Under **Access Manager**, click on **Application Domains**.
- 4) Click **Search** and then select the application domain.

- 5) Click on the application domain and then navigate to the **Authorization Policies** tab.
- 6) Select **Protected Resource Policy** and then navigate to the **Responses** tab
- 7) In the **Resources** tab, click **Add** to add the resources that you created in ***Configuring Protected Resources under an Application Domain*** (on page 22).
- 8) In the **Responses** tab, complete the following:
 - a. Deselect the **Identity Assertion** option.
 - b. Click **Add**.
 - c. In the **Type** field, enter **Header**.
 - d. In the **Name** field, enter the name of the HTTP header variable that contains the authentication type (for example, the name you entered in ***Configuring Primavera Portfolio Management for Single Sign-On*** (on page 23)).
 - e. In the **Value** field, enter the value assigned to the SSO Authentication Type HTTP Header that indicates that the Oracle Access Manager performs the authentication (for example, the name you entered in ***Configuring Primavera Portfolio Management for Single Sign-On*** (on page 23)).
- 9) Click **Apply**.

Configuring Primavera Portfolio Management for Single Sign-On

You need to configure PPM to accept headers from a successful Oracle Access Manager authentication in order to automatically log in to the PPM applications.

- **Notes:** For OAM and OHS version 12.2.1.4, make the following configuration changes to make it work with PPM application:
- In OHS, comment the below line in the "httpd.conf" file:
Header always set X-Content-Type-Options nosniff
File Location:
{OHS_HOME}/user_projects/domains/{ohsinstance}_domain/config/fmwconfig/components/OHS/{ohsinstance}/
- In OAM, add the following line in the "User Defined Parameters" in OAM webgate through OAM console:
X-Content-Type-Options=false
- The above OHS and OAM settings are however not applicable for PPM 19.0.1.0 and later versions.

PPM can be configured to accept any HTTP header variables. Configure the following registry values to control the names and values of the HTTP header variables used for integration:

Note: all registry values discussed below may be inserted into the Registry under the key:
HKEY_LOCAL_MACHINE\SOFTWARE\Oracle\Primavera Portfolio Management\Portfolios\Server\UI

Registry Value: SSO Authentication Type HTTP Header Name (String)

- ▶ **Data:** The name of the HTTP header variable that contains the authentication type.

- ▶ **Default:** OAM_AUTH_TYPE

Registry Value: SSO Authentication Type HTTP Header Value (String)

- ▶ **Data:** The value assigned to SSO Authentication Type HTTP Header and indicates that the service provider authenticate users.
- ▶ **Default:** Negotiate

Registry Value: SSO Authenticated User HTTP Header Name (String)

- ▶ **Data:** The name of the HTTP header variable that contains the name of the user authenticated by SSO.
- ▶ **Default:** OAM_REMOTE_USER

Registry Value: Registry Value "SSO Logout URL" (String)

- ▶ **Data:** The value of SSO Logout URL that redirects user to the SSO Logout screen.
- ▶ **Default:** SSO Logout URL

Note: PPM uses the "raw" HTTP header interface. As a result, the name of the HTTP headers within PPM registry entries should include the "HTTP_" prefix.

Testing Your Single Sign-On Implementation

After you complete all of the tasks for SSO for Oracle Access Manager and PPM, you should test your implementation.

To test your SSO implementation:

- 1) Close all of your open browsers.
- 2) Open a new browser.
- 3) Enter the URL of PPM.
You are redirected to a SSO form for authentication.
- 4) Enter the required information.

After you have been successfully authenticated, you will be redirected to the PPM application landing page. This confirms the successful setup SSO.

Configuring IIS for Single Sign-On

To configure IIS for SSO:

- 1) Navigate to the default web site in IIS Manager.
- 2) Click **ProSight** and select **Anonymous Authentication**.

Note: Do not select any other options.

Enabling SAML Authentication

Security Assertions Markup Language (SAML) associates a principal with additional identity information that can be used to determine the principal's access rights within a specific domain.

SAML is a standard that provides a means for exchanging security information across security domains. In a typical exchange between SAML messages between two domains, one party acts as a relying party while the other acts as an asserting party. The asserting party asserts information, such as whether a user has been authenticated, authorized to perform a certain action, and so forth. The relying party uses information provided by the asserting party to make security-related decisions (for example, what types of access to a specific resource the user should be granted).

When a user signs into a SAML-compliant service of a relying party, the service sends a "request for authentication assertion" to the issuing authority. The issuing authority returns an "authentication assertion" reference stating that the user was authenticated by a particular method at a specific time. The service then passes this assertion reference to other relying parties to validate the user's credentials. When the user accesses another SAML-compliant site that requires authentication, that site uses the reference to request the "authentication assertion" from the issuing authority, which states that the user has already been authenticated. At the issuing authority, an assertion layer handles request and response messages using SAML, which can bind to various communication and transport protocols (for example, HTTP, SOAP, and so on).

While the user who requests an assertion always consumes assertions, the issuing authority can act as producer and consumer since it can both create and validate assertions.

Identity Federation

Federated identity is the mapping of user credentials across security domains (identity providers and service providers) to allow access to hosted computing resources and services. In a federated environment, business that utilize federated identity can obtain identity information about an individual or other entity from the user's home organization or security domain. This provides twin benefits:

- ▶ End users do not need to enter login information to access each entity, or site, where business is conducted. This eliminates the need for users to remember and manage multiple passwords. Users will still need accounts for each site so that the accounts can be mapped.
- ▶ Enterprises do not need to create additional accounts to manage the identities of users who are already known to a partner organization.

In This Section

Prerequisites for Configuring Identity Federation Using SAML 2.0.....	26
Configuring Oracle Access Manager for Federated Identity Using SAML 2.0	26

Prerequisites for Configuring Identity Federation Using SAML 2.0

Prior to configuring your Primavera applications for SAML authentication and identity federation, ensure that you have completed the following prerequisites:

- ▶ Configure PPM for SSO using Oracle Access Manager.
For information to configure SSO, see ***Configuring Single Sign-On with Oracle Access Manager*** (on page 18).
- ▶ Obtain A SAML 2.0 metadata file that was exported from an IdP
For more information about exporting SAML 2.0 metadata, see your IdP documentation.

Configuring Oracle Access Manager for Federated Identity Using SAML 2.0

The procedures in this section have been described from the perspective of the service provider. Refer to your IdP documentation for instructions on configuring your IdP for federated identity. After your IdP has been enabled for identity federation, complete the tasks from the in the order that they appear:

- 1) Oracle Access Manager Administration Console as an administrator.
- 2) Enabling Identity Federation
- 3) Creating an Identity Store for Account Linking
- 4) (Optional) Enabling Automatic User Provisioning for the Local Identity Store used by Service Providers
- 5) Creating an Identity Provider Partner
- 6) Exporting SAML 2.0 Service Provider Metadata
- 7) Creating a SAML Authentication Policy
- 8) Assigning an Authentication Policy to Application Resources

Enabling Identity Federation

Enabling **Identity Federation** establishes trust between services by exchanging the following:

- ▶ X.509 certificates used for sign/verify and encrypt/decrypt the Federated messages
- ▶ Locations of the federated services
- ▶ SAML 2.0 metadata

To enable identity federation:

- 1) In the Launch **Pad** tab, under **Configuration**, select **Available Services**.
- 2) In the **Available Services** tab, click **Enable Service** for **Identity Federation**.

Creating an Identity Store for Account Linking

When defining an identity provider partner record, the service provider requires local user accounts to be mapped for imposing its access control model. The process of mapping SAML user accounts from the IdP to the local user accounts at the service provider is known as account linking. In this case, external user accounts that are authenticated by the identity provider need to be mapped to generic local user accounts with permission to access resources.

To create an identity store for account linking:

- 1) In the **Launch Pad** tab, under **Federation**, click **Service Provider Management**.
- 2) In the **Service Provider Administration** tab, click **Create Identity Provider Partner** and then complete the following:
 - a. In the **Name** field, enter a name (for example, *FederationStore*) for the identity provider partner.
 - b. In the **User Identity Store** menu, under **User Mapping**, select the name of the identity store that you used to configure SSO.
 - c. In the relevant fields, enter the information that you recorded from the identity store earlier.
 - d. Click **Apply**.
- 3) (Optional) Enable automatic user provisioning for the local identity store used by service providers by completing the tasks in **Enabling Automatic User Provisioning for the Local Identity Store used by Service Providers** (on page 27).

Enabling Automatic User Provisioning for the Local Identity Store used by Service Providers

When creating a local identity store mapping for SAML users, it is recommended that you ensure a corresponding user account for an identity provider user ahead of time. For example, if a user does not exist in the local store, the SAML assertion map to that user in the local identity store will fail. To handle an identity mapping failure, Oracle Access Manager Identity Federation features a plug-in that you can enable to automatically provision a missing identity to the local identity store during a federated SSO operation which enables the federated SSO to proceed.

Note: This is an optional task. If you do not enable automatic user provisioning and a user does not exist in this generic LDAP server, then the authentication / SAML assertion can fail.

To enable automatic user provisioning for the local identity store used by service providers:

- 1) Navigate to <Oracle_Access_Manager_Middleware_Home>/common/bin and then complete the following based on your operating system to open the WebLogic Scripting Tool:
 - ▶ If using Linux, run **wlst.sh**.
 - ▶ If using Windows, run **wlst.cmd**.
- 2) Connect to the WLS admin server by running the following:
`connect()`
- 3) Navigate to the domain runtime branch by running the following:
`domainRuntime()`
- 4) Enable automatic user provisioning by running the following:
`putBooleanProperty("/fedserverconfig/userprovisioningenabled",
"true")`
- 5) Exit the WebLogic Scripting Tool environment by running the following:
`exit()`

Creating an Identity Provider Partner

An identity provider is responsible for managing, authenticating, and asserting a set of user identities for its service provider partners. In order for the identity federation service to perform SSO with external identity providers, they must be defined as trusted partners.

To create an Identity Provider Partner:

- 1) In the **Launch Pad** tab, under **Identity Federation**, click **Service Provider Administration**.
- 2) In the **Service Provider Administration** tab, click **Create Identity Provider Partner**.
- 3) In the **Create Identity Provider Partner** tab, under **General**, complete the following:
 - a. In the **Name** field, enter a unique name for the identity provider partner.
For example, *FederatedProviderPartner*
 - b. In the **Description** field, enter a unique description for the identity provider partner.
 - c. Select the **Enable Partner** check box.
 - d. Deselect the **Default Identity Provider Partner** check box.
- 4) In the **Create Identity Provider Partner** tab, under **Service Information**, complete the following:
 - a. In the **Protocol** list, select **SAML 2.0**.
 - b. For **Service Details**, select **Load from provider metadata**.
 - c. For **Metadata File**, click **Browse** and then select a metadata file.

Note: The XML metadata file should be provided by an IdP.

- 5) In the **Create Identity Provider Partner** tab, under **User Mapping**, complete the following:
 - a. In the **User Identity Store** list, select the identity store that you created in **Creating an Identity Store for Account Linking** (on page 26).
For example, *FederationStore*
 - b. Select the **Map assertion Name ID to User ID Store attribute** option.
 - c. In the **Map assertion Name ID to User ID Store attribute** field, enter the LDAP attribute which identifies the unique login ID for your users. This should match the defined value in **Creating an Identity Store for Account Linking** (on page 26).
 - d. Click **Save**.
- 6) In the identity provider partner tab, complete the following:

Notes:

- This tab opens automatically after you save the identity provider partner that you create.
 - The name of tab has the name of the identity provider partner that you entered.
-

- a. Click **Create Authentication Scheme and Module**.

Note: The name of the authentication scheme and module is a combination of the name of the identity provider that you created with either `FederationScheme` or `FederationModule` appended to it.

For example, *FederatedProviderPartnerFederationScheme* or *FederatedProviderPartnerFederationModule*

- b. In the **Advanced** pane, complete the following:
 - Select **Enable global logout**.
 - Select **HTTP POST SSO Response Binding**.
 - In the **Authentication Request NameID Format** list, select **None**.
- c. Click **Save**.

Exporting SAML 2.0 Service Provider Metadata

Establishing trust between federation partners is a pre-requisite to perform any federation SSO operation between federation servers. Establishing trust involves exchanging certificate information. If a protocol relies on PKI X.509 certificates to secure message exchanges, as well as the locations and URLs of the services that implement the federation protocol, you can create a service provider SAML 2.0 metadata file in XML format for use by IdP containing information about profiles that the service provider supports. Sites acting as identity providers can import this metadata file to establish a relationship with the service provider.

To export SAML 2.0 service provider metadata:

- 1) In the **Launch Pad** tab, under **Configuration**, click **Federation Settings**.
- 2) In the **Federation Settings** tab, under **General**, click **Export SAML 2.0 Metadata...**
- 3) For later use, record the location to which you export the SAML 2.0 metadata.
- 4) Provide the metadata file to the IdP when establishing a service provider partner.

Creating a SAML Authentication Policy

When the IdP partner is created, an authentication module and scheme were also created to impose an access control model to protect Primavera application resources. The authentication scheme and module must then be mapped to an authentication policy in the application domain that is created to protect Primavera application resources.

To create an authentication policy and map the federated identity authentication scheme:

- 1) In the **Launch Pad** tab, under **Access Manager**, click **Application Domains**.
- 2) In the **Application Domain** tab, complete the following:
 - a. Click **Search**.
 - b. Click the name of an application domain.
- 3) In the application domain tab, open the **Authentication Policies** tab.

Note: The name of the tab is the name of the application domain that you clicked.

- 4) In the **Authentication Policies** tab, click **Create Authentication Policy**.
- 5) In the Create Authentication Policy tab, complete the following:
 - a. In the **Name** field, enter a name for the authentication policy.
For example,
 - b. (Optional) In the **Description** field, enter a description of the authentication policy.
 - c. In the **Authentication Scheme** list, select the authentication scheme that you created in **Creating an Identity Provider Partner** (on page 28).
For example, *FederatedProviderPartnerFederationScheme*
 - d. Click **Apply**.

Assigning an Authentication Policy to Application Resources

To assign an authentication policy to application resources:

- 1) In the **Launch Pad** tab, under **Access Manager**, click Application Domains.
- 2) In the **Application Domain** tab, complete the following:
 - a. Click **Search**.
 - b. Click the name of an application domain.
- 3) In the application domain tab, open the **Resources** tab.

Note: The name of the tab is the name of the application domain that you clicked.

- 4) In the **Resources** tab, complete the following:
 - a. Select a resource.

Note: You can only select one resource at a time.

- b. In the **Search Results** toolbar, click **Edit**.
- 5) In the resource tab, complete the following:

Note: The name of the tab is the name of the resource that you clicked.

- a. In the Authentication Policy list, under Protection, select the authentication policy that you created using **Creating a SAML Authentication Policy** (on page 29).
 - b. Click **Apply**.
- 6) Repeat this procedure for each resource in every application domain that is associated with a Primavera application.

Enabling SSL

Primavera Portfolio Management requires Secure Socket Layer (SSL), which allows for secure communications between web browser and server, using Secure HTTP (HTTPS).

In This Section

Request a Certificate	31
Receive and Install the Certificate on your IIS server	31
Testing SSL and Establishing a Trust Relationship between the Certificate Authority and the Browser	31

Request a Certificate

To request a certificate, follow the instructions here:

<https://technet.microsoft.com/en-us/library/hh831637.aspx#RequestCertificate>

You can also follow instructions provided by your certificate authority (CA).

Receive and Install the Certificate on your IIS server

After you send the file to your Certificate Authority, you will receive an email containing the new certificate. Save this file in a known location on your disk.

To install the certificate, follow the instructions here:

https://technet.microsoft.com/en-us/library/hh831637.aspx#Import_Certificate

You can also follow instructions provided by you certificate authority (CA).

Testing SSL and Establishing a Trust Relationship between the Certificate Authority and the Browser

Access PPM using a secure URL starting with "https://", such as "https://myservername/PPM". You will not be able to access the application using a non-secure URL such as "http://myservername/PPM", and if you attempt to do so, you will receive a failure message.

When first accessing (pages in) PPM using secure URLs, you may receive one or two "Security Alert" messages from your browser.

The first message will appear only if your web browser is not set up to trust certificates from the certificate authority from which you received your server certificate.

To add a certificate to your browser certificate store, follow instructions form your certificate authority.

Packing Debug Information

Often, when you have a support issue, you will need to collect information (such as Primavera Portfolio Management log files) to help Customer Support investigate the issue. You can use the psPackDebugInfo.exe utility to help you collect the right information. This utility collects all of the required information to a specific directory, which can then be zipped and uploaded to Customer Support.

Note: in a distributed installation of PPM (“Scale Out”), you would use the utility on all PPM application servers.

In This Section

What Information is Collected?	33
Usage	33

What Information is Collected?

The following information is collected:

- ▶ Windows events (from the Event Viewer) – will be converted into 3 csv files:
 - ▶ Application.csv – for Application events.
 - ▶ Security.csv – for Security events
 - ▶ System.csv – for System events.
- ▶ Primavera Portfolio Management log files – will be collected to a sub folder called: “ProSight Log”.
- ▶ IIS log files – will be collected to a sub folder called: “IIS Log”.
- ▶ Machine information (in information.txt file)
 - ▶ Machine Name
 - ▶ CPU specification
 - ▶ Amount of memory.
- ▶ Versions of software that is required by PPM (in information.txt file)
 - ▶ (Same as psVersionsExtractor information)
- ▶ List of the software that is installed on this machine (in information.txt)
- ▶ Registry export of the Key: HKLM/Software/Oracle/Primavera Portfolio Management (in portfolios.reg file)

Usage

- 1) On the Primavera Portfolio Management server, click “Start” -> “Programs” -> “Primavera Portfolio Management” -> “Utilities” -> “Debug Information”.
- 2) In the **Pack Primavera Portfolio Management debug** information window, click **Ok**.

By selecting “Ok”, this utility will collect information that will help support to determine and resolve the customer issue. The debug information is collected to a specific directory (specified by the “Directory” edit box - default: c:\psdebug). You can change the destination directory by editing the “Directory” edit box, or by using the “Browse” button.

Note: if the directory does not exist it will be created. If it exists, it will be removed and recreated.

- 3) In the **Finished successfully** window, click **OK**.
- 4) Zip this folder (c:\psdebug for example) and upload the zip to Customer Support.

Copyright

Oracle Primavera Portfolio Management System Administration Guide

Copyright © 1999, 2020, Oracle and/or its affiliates.

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 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, 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 failsafe, 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.

This software or hardware and documentation may provide access to or information on 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. 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.