

# **Agile Product Lifecycle Management**

Security Guide

Release 9.3.3

**E39280-02**

December 2013

Agile Product Lifecycle Management Security Guide, Release 9.3.3

E39280-02

Copyright © 2010, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

Contributing Author: Edlyn Sammanasu

Contributor:

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 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 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.

---

---

# Contents

<b>Preface</b> .....	v
Audience .....	v
Documentation Accessibility .....	v
Related Documents .....	v
Conventions .....	v
 <b>1 Document Scope</b>	
Documentation Audience .....	1-1
Guide to this Document .....	1-1
 <b>2 Agile PLM Overview</b>	
 <b>3 Overview of Security Fundamentals</b>	
Basic Security Considerations .....	3-1
Keep Software Up-To-Date .....	3-1
Restrict Network Access to Critical Services .....	3-1
Follow the Principle of Least Privilege .....	3-1
Monitor System Activity .....	3-2
Keep Up To Date on Latest Security Information .....	3-2
 <b>4 How to Perform a Secure Agile PLM Installation</b>	
Understanding the Agile PLM Environment .....	4-1
Recommended Deployment Topologies .....	4-1
Installation - Prerequisites .....	4-3
Installing the Oracle Database Server .....	4-3
Installing Oracle WebLogic Server .....	4-4
Installing Agile PLM and Database .....	4-4
Optional Component Configuration .....	4-5
Configuring AutoVue (Optional) .....	4-5
Configuring MCAD Connectors (Optional) .....	4-5
 <b>5 How to Securely Configure Agile PLM</b>	
Password Policy .....	5-1
Configuring and Using Authentication .....	5-1

LDAP-based Authentication .....	5-1
SSO-based Authentication .....	5-2
Database-based Authentication .....	5-3
<b>Configuring and Using Access Control .....</b>	<b>5-3</b>
<b>Configuring and Using Security Audit.....</b>	<b>5-3</b>
User Monitor .....	5-3
History Tab .....	5-4
Log Files .....	5-4

## 6 Security Considerations for Developers

Extensions Using Web Services .....	6-1
Extensions Using SDK .....	6-1

## A Secure Deploymnet Checklist

## B SSL Configurations

Basic SSL Configuration .....	B-1
<b>Configuring SSL on the WebLogic Server .....</b>	<b>B-3</b>
Configuring the Keystore on the WebLogic Server .....	B-3
Configure the Identity of the Server.....	B-4
Enable SSL and Assign Port .....	B-5
Import client.p12 to IE .....	B-5
Agile PLM Application SSL Configuration.....	B-5
<i>HTTPOnly and SecureFlag Flags in agile.properties</i> .....	B-6
<b>Configuring SSL on the File Manager .....</b>	<b>B-6</b>
<b>Configuring AutoVue 20.2 Securely .....</b>	<b>B-7</b>
<i>Configuring SSL Between the AutoVue Client and the VueServlet.....</i>	B-7
<b>Configuring SSL for SDK.....</b>	<b>B-7</b>
<b>Configuring SSL for Web Services .....</b>	<b>B-8</b>

---

---

# Preface

Agile PLM is a comprehensive enterprise PLM solution for managing your product value chain.

## Audience

This document is intended for administrators and users of the Agile PLM products.

## 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 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

Oracle's Agile PLM documentation set includes Adobe® Acrobat PDF files. The Oracle Technology Network (OTN) Web site <http://www.oracle.com/technetwork/documentation/agile-085940.html> contains the latest versions of the Agile PLM PDF files. You can view or download these manuals from the Web site, or you can ask your Agile administrator if there is an Agile PLM Documentation folder available on your network from which you can access the Agile PLM documentation (PDF) files.

## Conventions

The following text conventions are used in this document:

Convention	Meaning
<b>boldface</b>	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.

Convention	Meaning
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

---

# Document Scope

This document provides IT users and Agile PLM administrators with the information needed to securely set up and deploy Agile PLM.

## Documentation Audience

This document is written for IT users and Agile PLM administrators who will be setting up Agile PLM. It is assumed that those reading this documentation have a solid understanding of security concepts. The audience should also have basic knowledge of roles and privileges in Agile PLM.

## Guide to this Document

This guide provides information needed to help you to securely set up and configure Agile PLM.

The guide is organized as follows:

- ["Agile PLM Overview"](#) on page 2-1 gives an overview of Agile PLM and its modules.
- ["Overview of Security Fundamentals"](#) on page 3-1 provides an overview of basic security principles which should be considered while setting up Agile PLM.
- ["How to Perform a Secure Agile PLM Installation"](#) on page 4-1 provides guidance on how to securely install the Oracle Database Server, Oracle WebLogic Server, Agile PLM, and the Agile PLM Database.
- ["How to Securely Configure Agile PLM"](#) on page 5-1 provides information on how to use Agile PLM's security features to securely configure your deployment. User authentication and authorization is discussed in this chapter. Additionally, application-level configuration properties used to secure the application are discussed here.
- ["Security Considerations for Developers"](#) on page 6-1 provides information needed for developers to extend the Agile PLM application or produce applications using Agile PLM as a platform.





---

## Agile PLM Overview

The Agile PLM suite of solutions covers five primary areas of product lifecycle management:

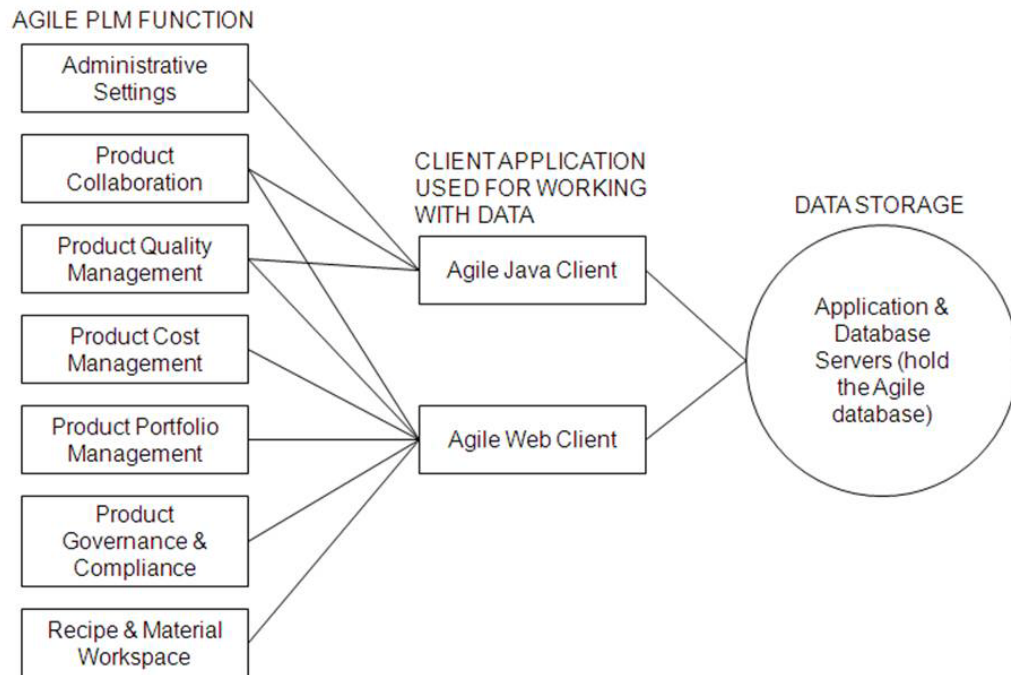
- **Agile Product Collaboration (PC)** - Management and collaboration of product record information throughout the product lifecycle, across internal organizations and the extended supply chain. Accessed through Web Client and Java Client.
- **Agile Product Governance & Compliance (PG&C)** - Management and tracking of all substances and materials contained by any item or manufacturer part, allowing companies to meet substance restrictions and reporting requirements, design recyclable products, minimize compliance costs, and eliminate noncompliance on future products. Accessed through Web Client.
- **Agile Product Portfolio Management (PPM, formerly Program Execution)** - Integration of program and product information, streamlining business processes across the product lifecycle and across a portfolio of programs. Accessed through Web Client.
- **Agile Product Quality Management (PQM, formerly Product Service & Improvement)** - Integration of customer, product, quality, and regulatory information with a closed-loop corrective action system. Accessed through Web Client and Java Client.
- **Agile Product Cost Management (PCM)** - Management of product costs across the product lifecycle and synchronization of product cost data and processes. Accessed through Web Client.
- **Agile Recipe & Material Workspace (RMW)** - Management of biotechnological and pharmaceutical products, and improvement of business productivity, visibility, scientific outcomes, and proactive compliance during the product development lifecycle. Accessed through Web Client. For more information, see *Agile PLM Getting Started with Recipe & Material Workspace*.

Agile administrators use Agile Java Client to set up and maintain settings for these solutions.

The Agile Application Server, the foundation of the Agile suite, manages data stored in the Agile database. All Agile data is contained or organized in business objects that are set up by the administrator, and specified and used by the enterprise's Agile users. For instance, the administrator configures the Parts class of objects, and users create and deploy specific instances of the kinds of Parts made available to them. Business objects is a general term that implies objects created from the classes available to the enterprise, but other entities in Agile are also objects, such as workflows, searches, reports, and so forth.

The following figure shows relationships between the Agile functional components, the primary client applications used to manipulate the data (Agile Web Client and Java Client), and the Agile Application and Database Servers (the database where the data is stored).

**Figure 2–1 Relationships Between Components in an Agile Setup**



---

## Overview of Security Fundamentals

This section describes some fundamental security principles and considerations.

### Basic Security Considerations

The following principles are fundamental to using any application securely.

#### Keep Software Up-To-Date

One principle for good security practice is to keep all software versions and patches up-to-date. To ensure that you have the most current and updated Agile PLM software for the latest version, regularly check the updates page.

#### Restrict Network Access to Critical Services

Keep both the Agile PLM application and the database behind a firewall. In addition, place a firewall between the middle-tier and the database. The firewall provides assurance that access to these systems is restricted to a known network route, which can be monitored and restricted, if necessary. As an alternative, a firewall router substitutes for multiple, independent firewalls.

If you cannot use firewalls, then configure the TNS Listener Valid Node Checking feature (it restricts access based upon IP address). Restricting database access by IP address often causes application client/server programs to fail for DHCP clients.

To solve this problem, use any of the following:

- static IP addresses
- software VPN
- hardware VPN
- software VPN and hardware VPN
- Windows Terminal Services or its equivalent

#### Follow the Principle of Least Privilege

The principle of least privilege states that users should be given the least amount of privilege to perform their jobs. Over-ambitious granting of responsibilities, roles, grants, and so on, especially early on in an organization's life cycle when people are few and work must be done quickly, often leaves a system wide open for abuse. User privileges should be reviewed periodically to determine relevance to current job responsibilities.

## **Monitor System Activity**

System security stands on three legs: good security protocols, proper system configuration and system monitoring. Auditing and reviewing audit records address this third requirement. Each component within a system has some degree of monitoring capability. Follow audit advice in this document and regularly monitor audit records.

## **Keep Up To Date on Latest Security Information**

Oracle continually improves its software and documentation. Check this note yearly for revisions.

---

# How to Perform a Secure Agile PLM Installation

This chapter describes a recommended deployment topology for your PLM system and then provides recommendations on how to securely install and configure the Agile PLM system.

## Understanding the Agile PLM Environment

When planning for a secure Agile PLM implementation, consider the following:

- **Which resources must be protected?**

- You must protect customer data, such as part numbers, file attachments, and so on
- You must protect internal data, such as proprietary source code.
- You must protect information in databases accessed by the Agile PLM server and the availability, performance, applications, and the integrity of the Web site.
- You must protect system components from being disabled by external attacks or intentional system overloads.

- **Who are you protecting data from?**

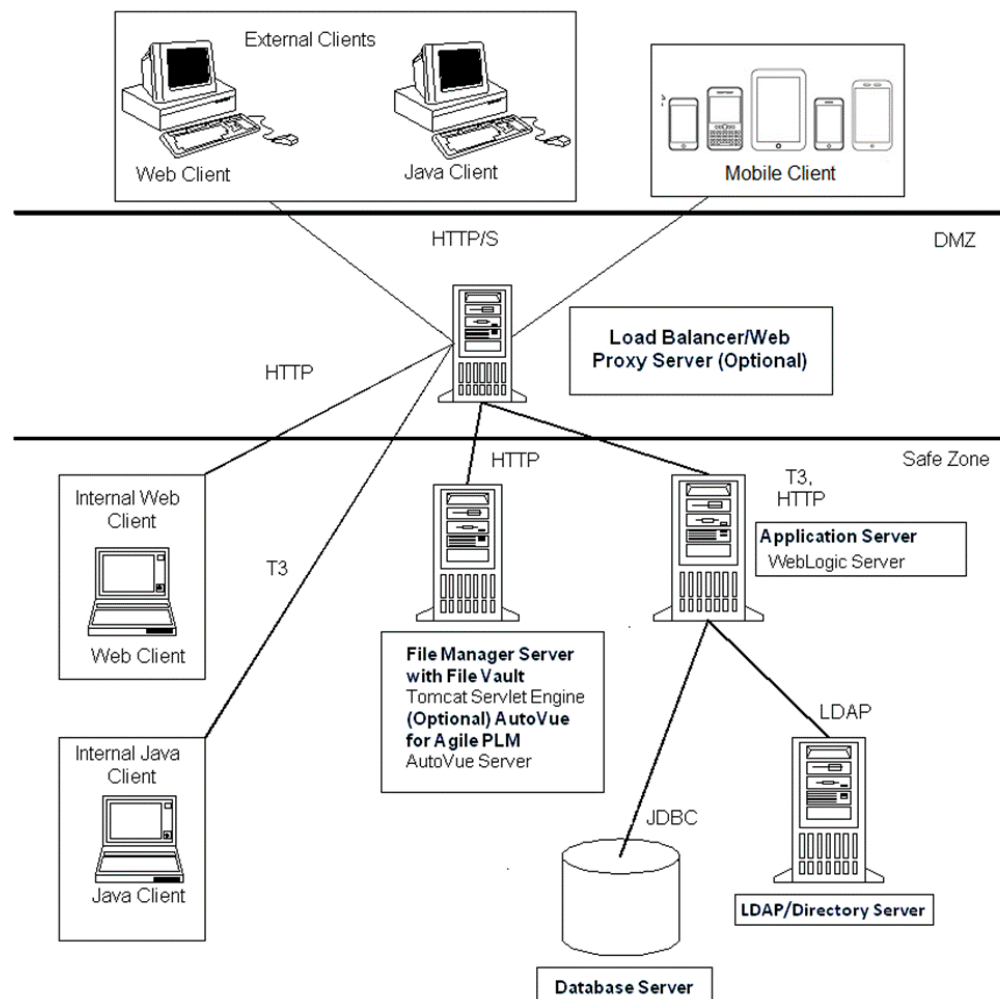
For example, you must protect your subscribers' data from other subscribers, but someone in your organization might need to access that data to manage it. You can analyze your workflows to determine who needs access to the data; for example, perhaps a system administrator can manage your system components without needing to access the system data.

- **What will happen if protections on a strategic resources fail?**

In some cases, a fault in your security scheme is nothing more than an inconvenience. In other cases, a fault might cause great damage to you or your customers. Understanding the security ramifications of each resource will help you protect it properly.

## Recommended Deployment Topologies

The following figure shows the general topology that is recommended for a secure Agile PLM installation.

**Figure 4–1 Recommended Deployment Topology**

The components included in the topology diagram are defined below:

- **Agile PLM Clients** - Agile PLM includes three clients: a Web client, a Java client, and a Mobile client. The Web client is a thin HTML client that uses firewall-friendly protocols (HTTP/S). The Mobile client is a mobile application that also uses firewall-friendly protocols (HTTP/S). The Java client is a Java-based client that can use application server-specific protocols, such as T3 for Oracle WebLogic, to connect to the server.
- **(optional) Proxy** - The hardware load balancer/proxy brokers client communications without compromising the security of your internal network. Clients communicate through the load balancer with the application server. There are no Agile software components running on the hardware load balancer. They are usually deployed in the Demilitarized Zone (DMZ) where it proxies requests from outside the corporate firewall to the application server in the Safe Zone.

Oracle recommends communication using HTTP over SSL (HTTPS) for the most secure deployment.

- For standalone application server deployments, both the load-balancer and web server components are optional.

- For deployments where the application server is clustered/redundant, a load-balancer is required and the web server is optional.

Refer to the documentation for your proxy server to determine the most secure configurations.

- **Agile PLM Application Server** - The Agile Application Server is the center of the Agile system, the base for the PLM platform, where all common services and business logic reside for the entire solution. The Agile Application Server runs on industry-leading J2EE application servers. As the System Configuration Overview figure illustrates, all client servers and users connect to the Application Server either directly or indirectly. The application server connects to the components in a persistence layer where product content is stored.

Oracle recommends communication using HTTP over SSL (HTTPS) for the most secure deployment.

- **Agile PLM Database Server** - The Agile Database Server persists or stores all product content and system settings. Agile's database server runs on Oracle 11g or 12c.
- **(optional) LDAP / Directory Server** - In an effort to better support the industry standard authentication schemes, Agile PLM supports Lightweight Directory Access Protocol (LDAP) based authentication. LDAP support enables you to integrate Agile with existing directory servers so user accounts can be managed in one place. Integrating with LDAP is optional. Users can be managed within Agile without a directory server. There are no Agile software components deployed on the Directory Server.

If using LDAP, Oracle recommends communication using LDAPS for the most secure deployment.

- **PLM File Manager / AutoVue Server** - The Agile PLM File Manager component provides file upload/download functionality for the Agile PLM application. Oracle recommends communication using HTTP over SSL (HTTPS) for the most secure deployment. The AutoVue Server component provides file viewing functionality for the Agile PLM application.
- **PLM File Vault** - The Agile PLM File Vault consists of one or more file system(s) on which the Agile PLM File Manager component stores and retrieves files uploaded/downloaded in the Agile PLM application.

---

**Note:** Oracle suggests that you create a similar Network Diagram to illustrate your deployment's specific network topology, including servers, routers, and firewalls. This document may be requested by Oracle Support should a network connectivity issue arise.

---

## Installation - Prerequisites

Before installing Agile PLM, you must install and configure Oracle Database Server and Oracle WebLogic Server. The following sections include recommendations on how to set these products up to ensure a secure configuration.

### Installing the Oracle Database Server

For the latest information on installing Oracle Database Server in a secure manner, refer to the *Oracle Database Security Guide* and make necessary configuration changes.

For additional information, refer to the "Installing Oracle Database Server" chapter in the *Agile Product Lifecycle Management Database Installation Guide*.

## Installing Oracle WebLogic Server

After installing the Oracle Database Server you should install the Oracle WebLogic Server.

For the latest information on how to install WebLogic Server, refer to the appropriate Oracle WebLogic Server documentation.

Additionally, Oracle recommends that you:

- Deploy WebLogic Server using SSL.
- After installation, change the WebLogic administrator username and password.
- Secure WebLogic Server by placing it behind a proxy server.

The following WebLogic Server documents contain information that is relevant to the WebLogic Security Service:

- *Understanding Security for Oracle WebLogic Server* - Summarizes the features of the WebLogic Security Service, including an overview of its architecture and capabilities. It is the starting point for understanding WebLogic security.
- *Securing Oracle WebLogic Server* - Explains how to configure security for WebLogic Server and how to use Compatibility security.
- *Securing a Production Environment for Oracle WebLogic Server* - This document highlights essential security measures for you to consider before you deploy WebLogic Server into a production environment.

## Installing Agile PLM and Database

This section describes best practices to be followed while using the Agile PLM and database installers.

For the latest information on installing Agile PLM, including the supported operating systems, refer to the *Installing Agile PLM for WebLogic* guide. The following users are created out-of-box for the application to start correctly and function as expected: admin, agileuser, etluser, ifsuser, propogation, superadmin.

---

---

**Note:** These OOB users should not be dropped or modified without consulting Oracle Support, as this will affect the functionality of the product.

---

---

For the latest information on installing the Agile PLM database schema, refer to the *Agile Database Installation Guide*.

Additionally, Oracle recommends that you:

- Use strong passwords.
- Deploy with SSL.
- Use the Agile PLM system for authentication.
- Use Oracle Platform Components such as OID or OAM for authentication requirements.



## Optional Component Configuration

To ensure a secure configuration, consider the following recommendations for optional components.

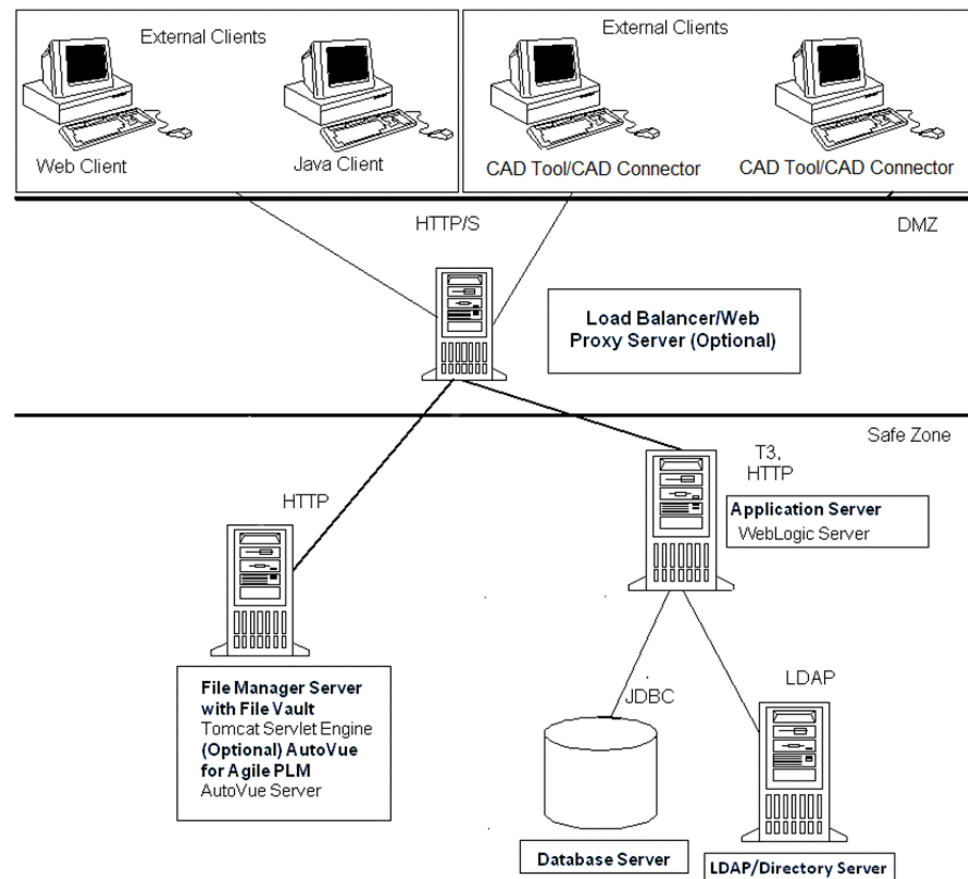
### Configuring AutoVue (Optional)

Refer to the *AutoVue Security Guide* for information about configuring AutoVue securely.

### Configuring MCAD Connectors (Optional)

The following diagram depicts how Oracle recommends that every CAD Tool/Connector be set up for optimal security.

**Figure 4–2 Recommended MCAD Configuration**



Oracle recommends that you configure the Engineering Collaboration Clients with HTTP(s). Refer to the "Configuring Engineering Collaboration Clients for HTTPS" section in the MCAD Connectors for Agile Engineering Collaboration Administration Guide for information about configuring MCAD Connectors securely.



---

## How to Securely Configure Agile PLM

This chapter describes how Agile PLM uses the following security features to provide data protection:

- **Authentication** - allows only permitted individuals to get access to the system and data.
- **Access Control (Authorization)** - provides authorized individuals access control to system privileges and data.
- **Audit** - allows Administrators to detect attempted breaches of authorization and attempted (or successful) breaches of access control.

### Password Policy

A password policy is a set of rules dictating how to use passwords. Some of the rules a password policy sets are:

- The maximum length of time a password is valid
- The minimum number of characters in a password

Password policies play an important role when attempting to access a directory. The directory server ensures that the entered password adheres to the password policy.

### Configuring and Using Authentication

Agile PLM supports the Lightweight Directory Access Protocol (LDAP), Single Sign-On (SSO), and database authentication configurations.

The three supported authentication configurations are discussed below.

#### LDAP-based Authentication

LDAP is an application protocol for querying and modifying directory services running over TCP/IP.

Agile PLM supports LDAP authentication through the Agile Directory Server Integration Module. You can integrate Agile with your existing directory server to manage your users in one place. This approach can be fully integrated into Agile PLM, for these supported directory servers:

- Oracle Internet Directory Server
- Microsoft Active Directory Server
- Sun Java System Directory Server

- Oracle Virtual Directory

If you chose to manage your user accounts through a directory server (instead of the database) during installation, then all new users are added, and certain user attributes are configured, only through the directory server. Users need to be synced from the LDAP system to the Agile PLM database.

For more information, refer to the "LDAP" chapter in the *Agile PLM Administrator Guide*.

## SSO-based Authentication

Agile PLM has the possibility of integrating aspects of your PLM system with Single Sign-On (SSO) capability. SSO is a Web-based solution that can be enabled only for Agile Web Client.

Single Sign-On integrates with the centralized security management system, other business and training applications, and improves user productivity in the Agile Web Client environment. With SSO configured and enabled for your PLM system, a user that has signed in to the system once (for instance, through the corporate portal) is not prompted again by a "login" dialog.

Agile PLM is certified on the following Single Sign-On solutions:

- Oracle Access Manager (OAM) - The secure configuration practices to configure Agile PLM with OAM Server can be found here:  
<http://fusionsecurity.blogspot.co.uk/2010/04/security-clarification-oam-identity.html>.
- NT LAN Manager (NTLM)

Note the following:

- Agile SDK code cannot connect to an Agile application URL protected by SSO.
- Users cannot develop Java Web Service client code and connect to an Agile Web Service protected by SSO.
- Webdav (AgileDrive) cannot connect to an Agile Application Server URL protected by SSO.
- Web Service clients or SDK code must connect directly to Agile server nodes with actual WebLogic ports or set up an alternate proxy that is not protected by SSO.

For more information, refer to the "Configuring Single Sign-On" chapter in the *Agile PLM Administrator Guide*. The chapter also includes a helpful diagram of the Agile SSO Plug-in Architecture.

### URL PX-based SSO

Customers use Process Extensions (PX) to extend Agile UI or business logic. Agile PLM has an SSO mechanism that allows the PX to access the Agile server without the user having to re-authenticate. Agile passes encrypted SSO tokens that the PX then submits back to the Agile server. This token is a one-time token. Additionally, the token is secure as it is stored in the Agile Database and not accessible. This token is used to ensure that the SSO mechanism will be valid only once after the UI PX has been clicked by the user. Once the validation has been successful, the token will be removed from the secure place before providing the Agile server access to the PX. Finally, the token itself expires after a certain interval. The expiration time is configurable and Oracle recommends that customers keep this interval to a very small value to prevent misuse of this token.

## Database-based Authentication

Customers can also use Agile Database authentication, instead of the LDAP or SSO authentication mechanisms. For more information, see the "Account Policy" section in the *Agile PLM Administrator's Guide*.

## Configuring and Using Access Control

Authorization primarily includes two processes:

1. Permitting only certain users to access, process, or alter data.
2. Applying varying limitations on user access or actions. The limitations placed on (or removed from) users can apply to objects, such as schemas, tables, or rows; or to resources, such as time (CPU, connect, or idle times).

Before creating a new Agile PLM user, make sure you answer the following questions:

- What does this user need to be able to do in Agile PLM? What default roles are required for this user?
- What should this user be prevented from doing in Agile PLM?
- Will this user need to have separate Login and Approval passwords?
- On which Agile PLM lists will the user's name appear?
- Which Agile PLM searches should the user be able to use?
- Is the user a Power User? A Power User can log in at any time and is not counted as a member of the concurrent user pool.

Do not assign too many users and designated escalation persons to user groups. Only assign users based on the requirements of each user group. Update user groups regularly.

For more information about access control using roles and privileges, see the *Agile PLM Administrator Guide*. Refer to the following relevant sections:

- *Overview of Roles and Privileges in Agile PLM*
- *Guidelines for Working with Roles*
- *Securing and Maintaining Roles and Privilege Masks*

## Configuring and Using Security Audit

Agile PLM enables you to audit your system by utilizing the User Monitor window, and through the data collected in an object's History Tab.

### User Monitor

The User Monitor window lists the users that are presently logged in to the Agile PLM system. It displays the following information about each logged-in user.

Table column	Description
User Name	The first and last name of the logged in user.
User ID	The login username of the user.
Host	Indicates the user's host.
Login Time	The time the user logged in.

For more information, see the "User Monitor" section in the *Agile PLM Administrator Guide*.

## History Tab

The History tab shows a summary of actions taken against an object. History is recorded for all objects in your Agile PLM system's database, and shows all actions by users and administrators. The History tab gets automatically populated.

The types of actions recorded for items are:

- Creation of the item
- Attachment actions: view, open, add, delete, get, check in, check out, cancel checkout, incorporate, unincorporate, and field modifications on the **Attachments** tab.
- Save As
- Send
- Print
- Modification of the subclass or any field of a released item
- Subscription modification and sharing

For more information see:

- *Getting Started with Agile PLM*
- *"History Tab" in the Agile Product Lifecycle Management Product Collaboration User Guide*

## Log Files

An additional source of audit information is log files. You can enable logging controls in Agile or in the WebLogic Server so that you can get more security-related information.

For more information about enabling logging, refer to the section "Logging Configuration" in the *Agile PLM Administrator Guide*.

For more information about enabling logging scripts in WebLogic, see "Application Logging and WebLogic Logging Services" in the WebLogic Server documentation.

---

## Security Considerations for Developers

This chapter discusses information useful to developers extending the application or producing applications using the product as a platform. Agile supports SDK if you prefer to use Java code for the extensions. Alternatively, Agile supports Web Services extensions so that you can use your preferred development language and platform.

### Extensions Using Web Services

The Agile PLM application includes web services as an extensibility point. The out-of-box Agile PLM web services can be leveraged to provide customized clients or integration modules. Agile PLM web services authenticate using basic authentication.

For optimal security protection, Oracle recommends configuring the web services using SSL. For more information about how to configure SSL for web services, refer to [Appendix B, "SSL Configurations"](#).

For more information about Agile web services, see *Agile Web Services User Guide*.

### Extensions Using SDK

Oracle's Agile Software Development Kit (SDK) is a collection of Java application programming interfaces (APIs), sample applications, and documentation that enable you to build custom applications to access, or extend the functionalities of the Agile Application Server. Using the SDK, you can create programs that extend the functionality of the Agile PLM and can perform tasks against the PLM system.

For optimal security protection, use SSL. For more information about how to configure SSL for SDK, refer to [Appendix B, "SSL Configurations"](#).

For more information about SDK, in general, see the *SDK Developer Guide - Developing PLM Extensions*.





---

## Secure Deployment Checklist

Follow the secure deployment checklist provided for the Oracle Database Server, as defined in the *Oracle Database Security Guide*. Similarly, follow guidelines for deploying your Oracle WebLogic Server, as defined in the Oracle WebLogic Server documentation.

The following security checklist includes guidelines that help secure your Agile PLM application:

1. Practice the principle of least privilege.
2. Enforce access controls effectively and authenticate clients stringently.
3. Restrict network access.
  - a. Use a firewall.
  - b. Never poke a hole through a firewall.
  - c. Monitor who accesses your systems.
  - d. Check network IP addresses.
  - e. Encrypt network traffic.
  - f. Harden the operating system.
4. Apply all security patches and workarounds.
5. Use strong passwords.
6. Deploy WebLogic Server using SSL.
7. Change the WebLogic administrator's username and password.
8. Set up a proxy server.
9. Contact the Oracle Security Support team if you come across any vulnerability in the Agile PLM application.



---

## SSL Configurations

You can set up SSL in your Agile PLM environment to work with the following:

- SDK
- Web Services
- WebLogic Server
- AutoVue Server
- Tomcat Server

Before you refer to the sections in this appendix to set up SSL for the any specific environment, follow the steps described in the following section, Basic SSL Configuration.

### Basic SSL Configuration

Follow these basic steps for setting up SSL. The steps provide general guidelines on how to generate the SSL certificates before you can apply for the Agile PLM Application server and other Agile PLM components like SDK and WebService samples.

1. Generate and procure the security certificate for your setup as per your company's policies.

Create a keystore file, for example, mykeystore.jks, which can be located on any disk path and contains a private key and a self-signed public key.

a. Go to the WLS domain directory and run setENV.cmd.

b. Run the following keytool command under the path of the keystore that you want to generate.

```
C:\CSR>keytool -genkey -keyalg RSA -alias mykey -keystore mykeystore.jks  
-keysize 2048
```

2. Generate a Certificate Signing Request (CSR) and send it to Certifying Authority using the command: C:\CSR>keytool -certreq -keystore mykeystore.jks

The result should look similar to this example:

**Figure B–1 Sending the CSR to Certifying Authority**

```

C:\CSR>keytool -certreq -keystore nykeystore.jks
Enter keystore password:
-----BEGIN NEW CERTIFICATE REQUEST-----
MIIB1TCCAT4CAQAugZQxCzAJBgNVBAYTAkNOMRAwDgYDUQEI EwdCZWlgaW5nMRAwDgYDUQHEwdC
ZWlgaW5nMRsuGQYDUQKExJPCmFjbGUGQ29ycG9yYXRpb24xIjAgBgNVBAstIGUzPUiBURUNUSU5H
IFB0U1BPUB0UIE9OTFkxIDAEBgNVBAMTF2JlajMwMTI4Mw5jb3I5bWVnFjBGUuY29tMIGFMA0GCSqG
SIb3DQEBAQUAA4GNADCBiQKBgQCm5PJ2emrfXE9U/t0jMFqYtXEPsLqZxiG0Kfe77SClW5tyL3s8
7PgBMXY7snzltlBco+xOnu.jANLAC5uyNz8g/oFyNkhirrRjdgQnxp+9dyBMB2GI17UwRFpaP7E8k
fgpgFxfvdxT8WhosngOw9HLaceBH5vhs3EiruKf pMLN0nQl DAQABoAAwDQYJKoZIhvcNAQEFBQAD
gYEAAJa2BGuyAK8PgbbFxEk8qKAverroC3/Nfu.jhnn4arEEM2iHm4uuDI jBxTNx8KJbHgGR74Pf y7
T6650N2D4U1TCXF99+190DsZPOI5wAAJYui/EzTacs0ZF1P32gUGed+9HnKr/Yz4us5/U1M/JdWQ
8dvPnJsp0L12MG/3u.jxSBzo=
-----END NEW CERTIFICATE REQUEST-----
C:\CSR>_

```

3. The CA returns with the certificate reply, RootCA, and sometimes an intermediateCA certificate. Installing the newly issued certificate normally involves installing it along with its certificate trust chain, which basically means installing (or verifying prior installation of) the certificates of (a) VeriSign's public primary root CA (the trust anchor CA) and (b) of the issuing (intermediate) SSL CA before (c) your newly issued SSL certificate is installed.

In CA replying mails, we get the OraclePKI-SSLCA.zip file, newly issued SSL certificate

Assuming you get the certificates as a zip file, for example

<Company>PKI-SSLCA.zip, it could contain the following set of certificates:

- a. root CA certificate: VTN-PCA-3G3.pem (included in <Company>PKI-SSLCA.zip)
  - b. intermediate SSL CA certificate: <Company>\_SSL\_CA.pem (included in <Company>PKI-SSL CA.zip)
  - c. newly issued SSL certificate: at the bottom of the mail "Your Standard SSL Certificate Is Ready", copy"-----BEGIN CERTIFICATE----- ... ..-----END CERTIFICATE-----" into a text file, and save as a .pem file myCert.pem
4. Import the certificates into the keystore.
- This can be done in two ways, either by importing the certificates in an order of RootCA, intermediateCA and then Certificate reply or by creating a certificate chain, clubbing them in order into a .pem file. In the following example, we create a certificate chain file CertChain.pem and import it into the identity keystore overriding the private key alias, which is mykey in this example.

---

**Note:** It is important to remember that the certificate can only be installed on the server that already has its corresponding private key present in its keystore. Additionally, it must be the same private key that was created during the CSR generation process described above).

---

5. Open a text editor and paste the contents of each certificate. Paste the certificate C:myCert.pem, B:<Company>\_SSL\_CA.pem, A:VTN-PCA-3G3.pem from top to bottom and save as CertChain.pem.

```
-----BEGIN CERTIFICATE-----
```

```
C:<myCert.pem>
```

```
-----END CERTIFICATE-----
```

```
-----BEGIN CERTIFICATE-----
```

```
B:<<Company>_SSL_CA.pem>
```

```
-----END CERTIFICATE-----
```

```
-----BEGIN CERTIFICATE-----
```

```
A:<VTN-PCA-3G3.pem>
```

```
-----END CERTIFICATE-----
```

The server should be presenting the whole chain of certificates to its clients during the SSL handshake. Certificates are typically presented in the leaf to root order (C-B-A).

Import the certificates chain to the keystore by running following command:

```
C:\CSR>keytool -import -file CertChain.pem -alias mykey -keystore  
mykeystore.jks -storepass <password>
```

6. Create a trust keystore. Do this by importing your Root CA certificate into another keystore that constitutes the trust:

```
C:\CSR>keytool -import -file VTN-PCA-3G3.pem -alias rootca -keystore  
trust.jks -storepass <password>
```

## Configuring SSL on the WebLogic Server

Once you have completed the steps in ["Basic SSL Configuration"](#) on page B-1, continue with the following procedures to configure SSL on the WebLogic Server that hosts the Agile PLM Application.

### Configuring the Keystore on the WebLogic Server

To configure the Keystore,

1. Access  
`http://<AgileApplicationServerName>:7001/console/login/LoginForm.jsp`.
2. Log in to the Admin Console.
3. Select the server on which you want to configure the SSL certificate.  
Server -> Click on the Keystore tab. By default it points to the Demo Certificates.
4. From the dropdown list, select the "Custom Identity and Custom Trust" option.  
Enter the identity and trust keystore details.

**Figure B–2 Identity and Trust Keystore Details**

Settings for BE3301283-AgileServer

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes

General Cluster Services **Keystores** SSL Federation Services Deployment Migration Tuning Overload Health Monitoring Server Start Web Services

Save

Keystores ensure the secure storage and management of private keys and trusted certificate authorities (CAs). This page lets you view and define various keystore configurations. These 1 of message transmissions.

Keystores: Custom Identity and Custom Trust [Change](#) Which configuration rules should be used for find keystores? [More Info...](#)

— Identity —

Custom Identity Keystore: C:\CSR\mykeystore.jks The path and file name of the identity keystore.

Custom Identity Keystore Type: JKS The type of the keystore. Generally, this is JKS.

Custom Identity Keystore Passphrase: \*\*\*\*\* The encrypted custom identity keystore's passph will be opened without a passphrase. [More Info...](#)

Confirm Custom Identity Keystore Passphrase: \*\*\*\*\*

— Trust —

Custom Trust Keystore: C:\CSR\trust.jks The path and file name of the custom trust keyst

Custom Trust Keystore Type: JKS The type of the keystore. Generally, this is JKS.

Custom Trust Keystore Passphrase: \*\*\*\*\* The custom trust keystore's passphrase. If empt without a passphrase. [More Info...](#)

Confirm Custom Trust Keystore Passphrase: \*\*\*\*\*

Save

## Configure the Identity of the Server

Click on the SSL tab and enter the alias of the private key, for example mykey, and the key passphrase, for example "Agile123".

**Figure B–3 Entering the Private Key and Key Passphrase**

Configuration Protocols Logging Debug Monitoring Control Deployments Services Security Notes

General Cluster Services Keystores **SSL** Federation Services Deployment Migration Tuning Overload Health Monitoring Server Start Web Services

Save

This page lets you view and define various Secure Sockets Layer (SSL) settings for this server instance. These settings help you to manage the security of message transmissions.

Identity and Trust Locations: Keystores [Change](#) Indicates where SSL should find the server's i the server's trust (trusted CAs). [More Info...](#)

— Identity —

Private Key Location: from Custom Identity Keystore The keystore attribute that defines the locati

Private Key Alias: mykey The keystore attribute that defines the string private key. [More Info...](#)

Private Key Passphrase: \*\*\*\*\* The keystore attribute that defines the pass key. [More Info...](#)

Confirm Private Key Passphrase: \*\*\*\*\*

Certificate Location: from Custom Identity Keystore The keystore attribute that defines the locati

— Trust —

Trusted Certificate Authorities: from Custom Trust Keystore The keystore attribute that defines the locati

## Enable SSL and Assign Port

**Figure B–4 Enabling SSL and Assigning the Port**

The screenshot shows the 'Configuration' tab for the 'General' settings of the server instance 'BEJ301283-AgileServer'. The 'Listen Port Enabled' checkbox is checked, and the 'Listen Port' is set to 7001. The 'SSL Listen Port Enabled' checkbox is also checked, and the 'SSL Listen Port' is set to 7002. A red box highlights the 'SSL Listen Port Enabled' checkbox and the 'SSL Listen Port' field.

<b>Name:</b>	BEJ301283-AgileServer	An alphanumeric name for this server instance
<b>Machine:</b>	(None)	The WebLogic Server host computer (machine). <a href="#">Info...</a>
<b>Cluster:</b>	(Standalone)	The cluster, or group of WebLogic Server instances. <a href="#">Info...</a>
<b>Listen Address:</b>		The IP address or DNS name this server uses to listen. <a href="#">Info...</a>
<input checked="" type="checkbox"/> <b>Listen Port Enabled</b>		Specifies whether this server can be reached through the listen port. <a href="#">More Info...</a>
<b>Listen Port:</b>	7001	The default TCP port that this server uses to listen for connections. <a href="#">More Info...</a>
<input checked="" type="checkbox"/> <b>SSL Listen Port Enabled</b>		Indicates whether the server can be reached through the SSL listen port. <a href="#">Info...</a>
<b>SSL Listen Port:</b>	7002	The TCP/IP port at which this server listens for SSL connections.

## Import client.p12 to IE

Import client.p12 to Internet Explorer.

## Agile PLM Application SSL Configuration

Modify the following configuration files:

### 1. jndiurl.properties

```
Path: <AgileHomePath>\agileDomain\servers\<AgileServer>\tmp\_WL_
user\AgilePLM\<RandomName>\APP-INF\classes
server1=t3s://<AgileApplicationServerName>:7002
```

### 2. agile.properties

```
Path:<AgileHomePath>\agileDomain\servers\<AgileServer>\tmp\_WL_
user\AgilePLM\<RandomName>\APP-INF\classes
<AgileHomePath>\agileDomain\config
##### Common Web Security Settings #####
# Specify whether to use the Secure flag to protect sensitive cookies
WebSecurity.ForceSecureCookies = true
```

### 3. ext.jnlp

```
Path: <AgileHomePath>\agileDomain\servers\<AgileServer>\tmp\_WL_
user\AgilePLM\<RandomName>\JavaClient.war\wls
<jnlp spec="1.0+"
codebase="https://<AgileApplicationServerName>:7002/JavaClient">
```

#### 4. `pcclient.jnlp`

*Path:* <AgileHomePath>\agileDomain\servers\<AgileServer>\tmp\\_WL\_user\AgilePLM\<RandomName>\JavaClient.war\

```
<jnlp spec="1.0+"  
codebase="https://<AgileApplicationServerName>:7002/JavaClient">  
<argument>serverURL=t3s://<AgileApplicationServerName>:7002/</argument>
```

Once you have completed modifying the configuration files, restart the application server to make the settings effective.

### ***HTTPOnly and SecureFlag Flags in agile.properties***

Whenever user-sensitive cookies are generated in Agile PLM, the HTTPOnly flag is also included in the Set-Cookie HTTP Response Header. This helps mitigate the risk of a client-side script accessing the protected cookie (if the browser supports it). You can change the value to 'false' to retain legacy behavior. From a secure system perspective, Oracle recommends that customers keep this set to 'true'.

Additionally, Agile PLM does not mandate use of SSL, so setting the Secure flag will prevent non-SSL enabled customers from using Agile. The solution is to introduce a setting for secure mode and if enabled, then set the Secure Flag on all the sensitive cookies. This ensures that sensitive cookies are available in another application only via HTTPS. These cookies will not be available via HTTP, even if both the Agile PLM Application and the external application are deployed in the same domain. You can change the value to 'false' to retain legacy behavior. From a secure system perspective, Oracle recommends that customers keep this set to 'true'.

### **Configuring SSL on the File Manager**

Once you have completed the steps in "[Basic SSL Configuration](#)" on page B-1, continue with the following procedures.

1. Add the following component to  
<AgileHomePath>\FileManager\conf\server.xml:

```
<Connector protocol="org.apache.coyote.http11.Http11Protocol"  
port="8443" maxThreads="200"  
scheme="https" secure="true" SSLEnabled="true"  
keystoreFile="<KeystorePath>" keystorePass="<KeystorePassword>"  
clientAuth="false" sslProtocol="TLS"/>
```

2. Configure the Java Client File Manager node as follows:

Name: iFS

API Name: IFS

File Manager URL: https://<FileServerName>:8443/Filemgr/AttachmentServlet

Enabled: Yes

Primary file server: Yes

File Manager Internal Locator:

https://<FileServerName>:8443/Filemgr/services/FileServer

Viewer Server URL: https://<FileServerName>:8443/Filemgr/VueServlet



Viewer Proxy URL: https://<FileServerName>:8443/Filemgr/VueLink

Viewer Content URL: https://<FileServerName>:8443/Filemgr/jVue

3. Configure <AgileHomePath>\agileDomain\config\server.conf

app.server.url

=http://<AgileApplicationServerName>:7001/Agile/services/FSHelper

file.server.url

=https://<FileServerName>:8443/Filemgr/services/FileServer

dms.server.url =http://

<AgileApplicationServerName>:7001/Agile/services/DmsService

4. Restart file manager server.

5. Access https://<FileServerName>:8443/Filemgr/Configuration to check the File Manager configuration.

## Configuring AutoVue 20.2 Securely

---

**Note:** You must restrict IP access between the VueLink/VueServlet and the AutoVue machine. For details, refer to the Security Features section "Configuring and Using Restrict IP Access" in the *Oracle AutoVue Integration Software Development Toolkit Security Guide*.

---

### Configuring SSL Between the AutoVue Client and the VueServlet

This section describes how to configure SSL between the AutoVue Client and the VueServlet. We do not provide steps for configuring SSL between the File Server and Auto Vue Server, because this communication takes place behind the firewall and SSL is not commonly used.

1. Connect to the application sever via HTTPS protocol in order get the application server's certificate, for example, https://<AgileApplicationServerName>.
2. Import the certificate into Internet Explorer.
3. Export the certificate from Internet Explorer as a base-64 encoded format and save the certificate onto the local disk, for example, C:\certs.cer.
4. Import the certificate into the AutoVue server's JRE using Java's keytool command:
 

```
<Java Install Directory>\bin>keytool -import -alias <servername> -file
c:\certs.cer -trustcacerts -v -keystore
C:\Oracle\AutoVue\jre\lib\security\cacerts
```
5. Restart the AutoVue server.
6. Configure the web page that embeds the AutoVue applet to point to the https:// URL for the VueServlet.

## Configuring SSL for SDK

To configure SSL for SDK, do the following:

1. Get the certification key, for example mykeystore.jks, that is generated using the steps in Appendix B, and keep the mykeystore.jks file in a folder located on the machine where you want to run SDK, such as "C:\SDKSSL".

2. Follow these steps to run SDK sample code with an SSL environment:
  - a. Download SDK sample files from OTN.
  - b. Go to "..\SDK\_AIS\_Samples\sdk\samples\api\Login".
  - c. Update URL, USERNAME and PASSWORD with SSL server information in Login.java.

Set URL as https://hostname:port/Virtualpath

- d. Update the file run.bat:

Set JAVA\_HOME & SDK\_HOME

Update Java Command:

```
java -classpath .;c:\SSL\SDK\AgileAPI.jar  
-Djavax.net.ssl.trustStore=C:\SDKSSL\mykeystore.jks  
-Djavax.net.ssl.trustStorePassword=Agile123 Login
```

- e. Execute run.bat.

## Configuring SSL for Web Services

To configure SSL for Web Services, do the following:

1. Get the certification key, for example mykeystore.jks, that is generated using the steps in Appendix B, and keep the file in a folder located on the machine where you want to run Web Services, for example "C:\SSL".
2. Follow these steps to run Web Services sample code in an SSL environment:
  - a. Download the Web Services sample files from OTN.
  - b. Copy AgileAPI.jar file into "C:\SSL\WS\".
  - c. Copy one of the sample Java files for creating an object, CreateObject.java in a location such as "C:\SSL\WS".

- d. Compile the file as follows:

- C:\SSL\WS>javac -classpath .;C:\SSL\WS\AgileAPI.jar  
CreateObject.java

- e. Run the sample as follows:

- C:\SSL\WS>java -classpath .;C:\SSL\WS\AgileAPI.jar  
-Djavax.net.ssl.trustStore=C:\SSL\mykeystore.jks  
-Djavax.net.ssl.trustStorePassword=Agile123 CreateObject

---

**Note:** The steps in this section do not make use of an Integrated Development Environment (IDE). Alternatively, you can set up SSL for Web Services using an IDE.

---