Agile
Version e6.1

Oracle® Agile

Engineering Data Management

Security Guide for Agile e6.1.2.2

Part No. E29053-01

April 2012

Copyright and Trademarks

CONTENTS

Preface

The Oracle documentation set includes Adobe® Acrobat™ PDF files. The Oracle Technology Network (OTN) Web site (http://www.oracle.com/technology/documentation/agile.html) contains the latest versions of the Oracle Agile e6 PDF files. You can view or download these manuals from the Web site, or you can ask your Agile administrator if there is an Oracle Documentation folder available on your network from which you can access the documentation (PDF) files.

Note To read the PDF files, you must use the free Adobe Acrobat Reader™ version 7.0 or later. This program can be downloaded from the Adobe Web site (http://www.adobe.com).

Note Before calling Agile Support about a problem with an Oracle Agile e6 manual, please have the full part number ready, which is located on the title page.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Readme

Any last-minute information about Oracle Agile e6 can be found in the Release Notes file on the Oracle Technology Network (OTN) Web site (http://www.oracle.com/technology/documentation/agile_eseries.html)

Agile Training Aids

Go to the Oracle University Web page (http://www.oracle.com/education/chooser/selectcountry_new.html) for more information on Agile Training offerings.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Chapter 1

Overview of Agile e6 System

Agile Engineering Data Management (Agile EDM) is a Product Lifecycle Management solution that enables the engineering industry to manage its complete lifecycle of product development activities in a secure and collaborative application environment.

This document provides an overview of the Agile e6 system and discusses the security objectives and security architecture of Agile e6 modules. It also explains how to install and use the Agile e6, release e6.1.2.2 system securely. It includes specific information on how to enable security features, such as SSL, as well as more open ended discussions of the security implications of configuration choices.

Overview of Agile e6 System Architecture

The Agile e6 system is built using a 3-Tier architecture, which contains the following parts:

- **Client** - The client is responsible for the presentation logic.
- **Application Server** - The application server process is responsible for the business logic.
- **Database** - The database server takes care of the physical storage of all data.

Three types of clients are used for serving the differing needs of casual and power users:

1. Java Client
2. Web Client
3. Windows Client

The Agile e6 server processes provide the core PLM functionalities, like Item Management, BOM Management, Document Management, Change Management, etc. For each client-side process, Agile e6 server processes run in parallel.

User data is stored in the Oracle database and is accessed by the application server process. This ensures that the clients do not require a direct database connection. The connection from the application server process to the database is established through SQL*Net.

Agile e6 System Services

Some responsibilities of the application server process have been assigned to dedicated services, being able to service several client processes in parallel. These are:

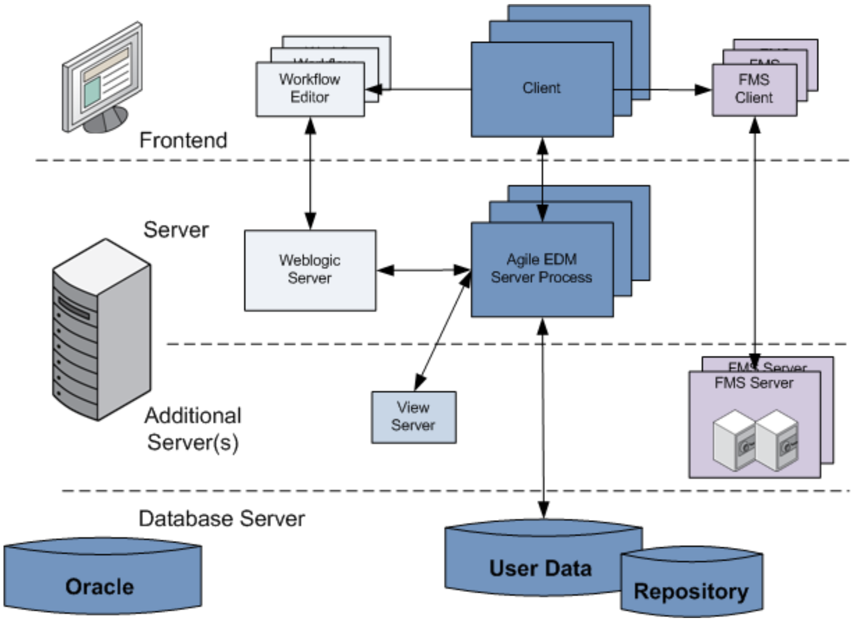
1. File Management Services
2. Business Services
3. Technical Services

The File Management Services manages the files and attachments transaction and storage services, thus facilitating the *check-in* and *check-out* functionality provided by the Document Management System in the Agile e6 system.

The Business Services provide PLM functionalities for Workflow Management, Product Configurator and Permission Manager.

Technical Services encompass Java Client WebStart deployment, Java Client HTTPS support, Web Presentation Service, Web-Fileservice and Administration Client.

The Business Services as well as the Technical Services run on top of the Oracle WebLogic Application Server.



Agile e6 System Components

Server-Side Components

The Agile e6 application server components can reside on the same server where the Agile e6 server processes are executed (this is recommended for Business Services) or can reside on any other computer in the network (especially for the File Management Service FMS).

The entire communication is based on TCP/IP. Only the unprivileged ports (above 1024) are used. Once a TCP/IP connection is established the port will not be changed dynamically.

- File Management Services
- ViewServer – (external) component of AutoVue Viewer used view and redline documents (Office documents, 2D/3D-CAD Models)
- LDAP Server – (external) component (e.g. Oracle Identity Management Suite) to provide centralized store for managing user/password
- Batch Client – component to run PLM batch processes
- Java Daemon
- FMS Daemon
- PLM-API Proxy

Client Side Components

- Workflow Editor - to model and view workflows
- Office Suite – to check-in/check-out documents from/to Microsoft Office

Security Objectives of Agile e6 System

- Providing Basic Security Services
- Supporting Standards
- Deployment and Configuration Flexibility
- Scalability and Predictability

System wide Advice

Some advice applies to the entire system and the infrastructure in which it operates.

- Keep the software up to date
- Restrict network access to critical services
- Follow the principle of least privilege
- Monitor system activity
- Keep up to date on latest security information

Chapter 2

Agile e6 System Security Architecture

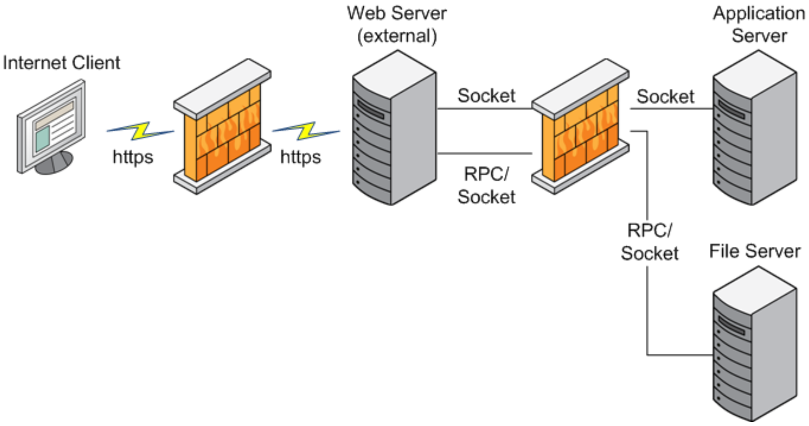
The Internet access to the Agile e6 system fulfills high security requirements. The usage of the HTTPS protocol is only the first step. The system administrators install firewalls to protect the internal network. These firewalls block most ports so that only the port identified to communicate through the firewall is available. At best, only one port is required.

The Agile e6 system provides a solution to access files in a secure environment. The files are stored in a vault, which is managed by a file server. This file server provides the files for every application, which has access to the Metadata. The Metadata is stored in the Agile e6 system.

In the web environment, the Web Client has access to the Agile e6 system via an Enterprise Communication Interface (ECI) connection. ECI is the API of Agile e6 application. The Web Client provides the access to the file access operations, which are executed by the Web File Service. The Metadata is only available within the web server; an access from the Internet is possible, but the system uses encrypted tickets to grant the access to the metadata.

Medium Secure Environment

The medium secure environment has one or more firewalls and could have a standard proxy in front of the external web server. The first firewall grants access for one port only. The second firewall blocks most of the ports, but lets pass some of the dedicated ports.



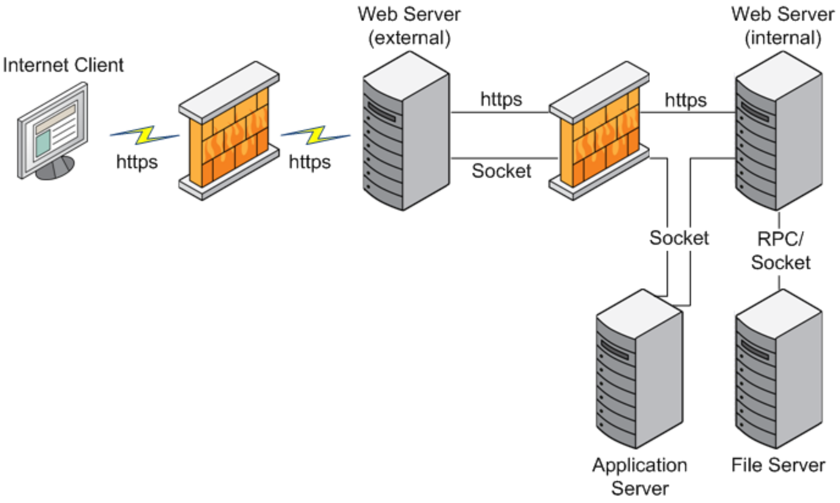
The first firewall lets pass the HTTP or HTTPS port only to access the web server, all the other ports are blocked. The second firewall lets pass the ports needed by the ECI, the RPC port and the file server ports, which are necessary to exchange the file data.

The following table shows the connections used. Refer to the Agile e6 Administrator Guide for more information.

Module	Port Configurable	Description
ECI	yes	PLM Java Daemon
ECI	yes	ECI communication socket
FMS	no	RPC Portmapper
FMS	no	FMS communication socket, the port range depends on the maximum number of concurrent users.

High-End Security Environment

The high-end security environment has two or more firewalls and could have a standard proxy in front of the external web server. The firewall, which protects the external web server, lets pass only one port. This port is a HTTP or HTTPS port to access the web server.



Most customers have an Intranet web server, which provides a web access to the employees. The high-end solution uses this internal web server to provide the access to the files for Internet users. The Internet Web File Service works as a proxy, which sends the request to the internal Web File Service. The internal Web File Service requests the file from the file server and sends the file data to the external Web File Service. The external Web File Service sends the file to the Internet user. Any Web File Service does not cache the file data, the incoming file data is sent as response to the user or calling Web File Service.

Note This environment supports only the viewing of files.

Chapter 3

Secure Environment - HTTP(S) Support

Note Local FMS is not working in Secure Environment. Only Web File Service can be used in Secure Environment.

Prerequisites

Before starting to setup a secure environment make sure your standard installation as described in the installation documentation works without issues.

In the secure environment we will just make some modifications to a standard environment. To configure the secure environment you need a certificate from a trusted certificate authority.

In this example scenario a self signed certificate is used. The example scenario uses windows operating system. Self signed certificates require adding security exceptions to browsers and java virtual machines to accept these certificates. Certificates from a trusted certificate authority do not need these security exceptions.

Warning We DO NOT recommend the use of Self Signed Certificates in the production environments.

Setup of a production secure environment needs expertise in network security setup. The following setup is only a simple example which shows the Agile e6 requirements. This must be extended for production use by customer experts with their networking security infrastructure requirements.

Secure External Communication

In Agile e6 system, it is possible to setup external access to the Agile e6 environment with the Java Client or a Web Browser using the Web Client. It is possible to setup the complete external communication over internet using the HTTPS protocol. Internal intranet communication will still be over HTTP and RPC calls.

Note For Agile e6, the Java Client is the main client. The Dataview uses RPC connection which is not secure.

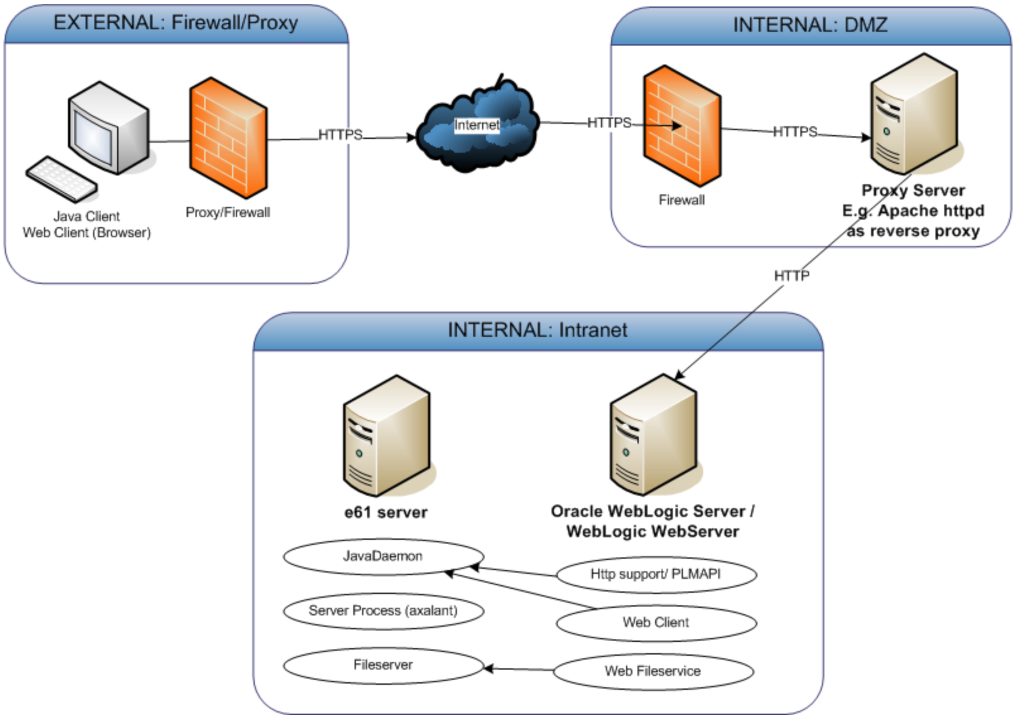
This section describes how to setup a secure environment for this use case.

Secure External communication means the communication in the internet. From a Java/Web

Client over the internet to a server in the DMZ which acts as the *End Point* for the Java and Web Client. In case of Agile e6, the *End Point* is apache HTTPD, configured as a Reverse Proxy in the DMZ.

Certificate needs to be installed for apache in the DMZ in this scenario.

Following picture illustrates an example communication.



As shown in the diagram above, the Java and Web Client connect to the Proxy Server over HTTPS.

The proxy server is configured to pass the incoming requests from the internet to the end points on the Oracle WebLogic Server in the intranet.

The end points in the intranet are:

- PLMAPI for Java Client HTTP support
- Java Client Web Start for Java Client download
- Web Presentation Service for the Web Client
- Web Fileservice for file transfer over http
- AutoVue proxy for AutoVue Applet

- VueLink Servlet for Applet download

For this scenario, you have to adapt the Java Client to use HTTPS protocol and configure the proxy server to pass the incoming requests and accept HTTPS connections.

Note Setting up the firewalls or other DMZ infrastructure is not part of this document.

Setup Apache HTTPD as SSL Reverse Proxy

This document describes only the minimal apache configuration needed, which is not sufficient for securing production environments. Refer to the applicable Agile e6 Administration and Installation documents for the configurations that are required for securing the production environments.

This scenario uses Apache HTTP Server (HTTPD) 2.2.x, including OpenSSL (the version used is: httpd-2.2.16-win32-x86-openssl-0.9.8o.msi).

Next steps require installed apache, which need not be fully configured.

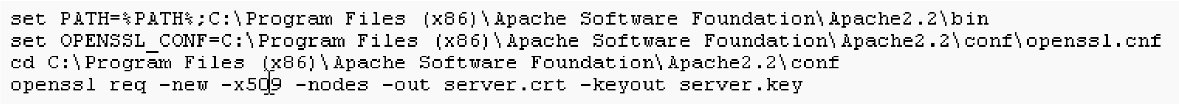
In this example we're using a self signed certificate for apache. The certificate identifies the apache server to the client.

This first configuration is not a full SSL reverse proxy, because intranet communication from proxy to internal WebLogic server still goes over HTTP. Full SSL reverse proxy is configured in the *Secure Internal Communication* section in this document.

Create Self Signed Certificate

Creating self signed certificates is only recommended for testing. For production environments certificate from a trusted certificate authority is recommended.

Example:



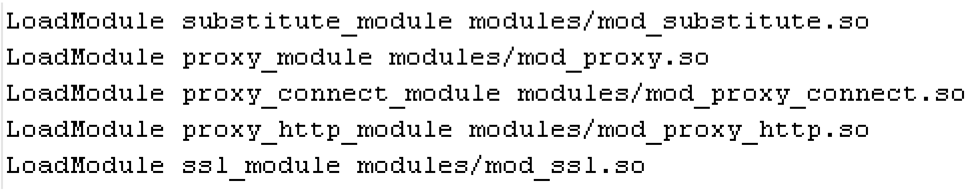
```
set PATH=%PATH%;C:\Program Files (x86)\Apache Software Foundation\Apache2.2\bin
set OPENSSL_CONF=C:\Program Files (x86)\Apache Software Foundation\Apache2.2\conf\openssl.cnf
cd C:\Program Files (x86)\Apache Software Foundation\Apache2.2\conf
openssl req -new -x509 -nodes -out server.crt -keyout server.key
```

This will create the certificate that will be used automatically by apache HTTPD.

The self-signed certificate can be created in a command (cmd) shell.

Changes in <apache_home>\conf\httpd.conf

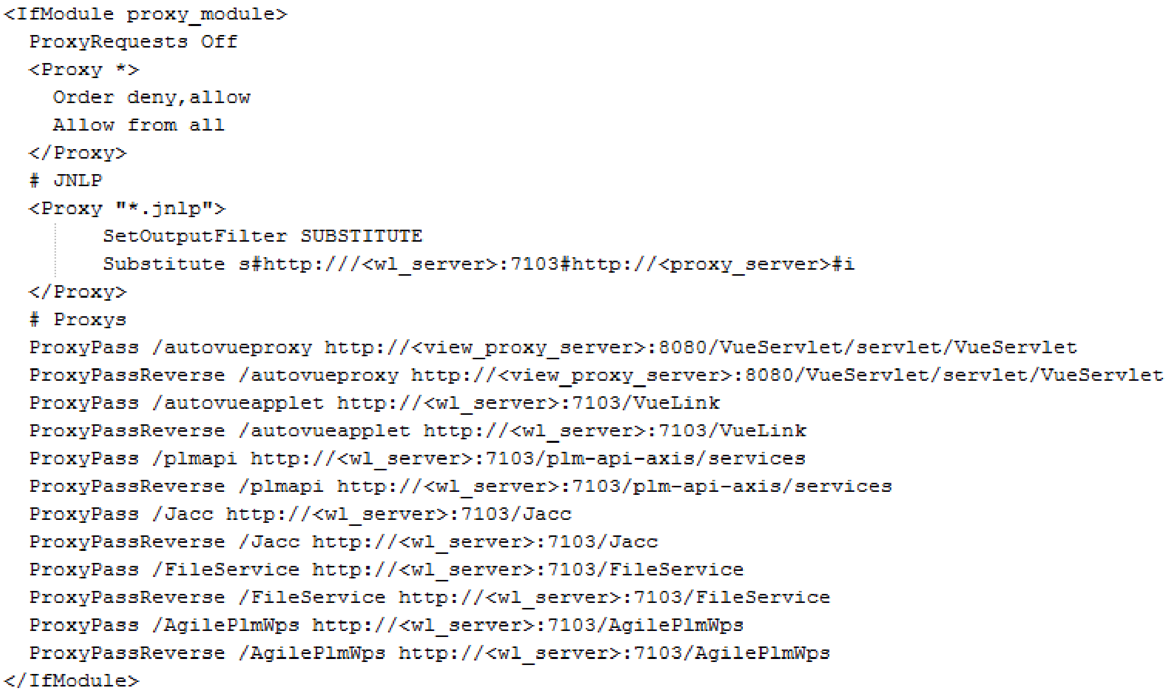
Activate Modules:



```
LoadModule substitute_module modules/mod_substitute.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule ssl_module modules/mod_ssl.so
```

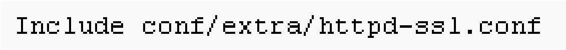
This will also activate the needed proxy modules.

Add a proxy section:



```
<IfModule proxy_module>
  ProxyRequests Off
  <Proxy *>
    Order deny,allow
    Allow from all
  </Proxy>
  # JNLP
  <Proxy "*.jnlp">
        SetOutputFilter SUBSTITUTE
        Substitute s#http:///<wl_server>:7103#http://<proxy_server>#i
  </Proxy>
  # Proxys
  ProxyPass /autovueproxy http://<view_proxy_server>:8080/VueServlet/servlet/VueServlet
  ProxyPassReverse /autovueproxy http://<view_proxy_server>:8080/VueServlet/servlet/VueServlet
  ProxyPass /autovueapplet http://<wl_server>:7103/VueLink
  ProxyPassReverse /autovueapplet http://<wl_server>:7103/VueLink
  ProxyPass /plmapi http://<wl_server>:7103/plm-api-axis/services
  ProxyPassReverse /plmapi http://<wl_server>:7103/plm-api-axis/services
  ProxyPass /Jacc http://<wl_server>:7103/Jacc
  ProxyPassReverse /Jacc http://<wl_server>:7103/Jacc
  ProxyPass /FileService http://<wl_server>:7103/FileService
  ProxyPassReverse /FileService http://<wl_server>:7103/FileService
  ProxyPass /AgilePlmWps http://<wl_server>:7103/AgilePlmWps
  ProxyPassReverse /AgilePlmWps http://<wl_server>:7103/AgilePlmWps
</IfModule>
```

Activate SSL:



```
Include conf/extra/httpd-ssl.conf
```

Restart apache HTTPD.

Windows Server 2008 or Windows 7 special adaption in conf/extra/httpd-ssl.conf:

The adaptation of the “SSLSessionCache” is only necessary on Windows Server 2008 or Windows 7. If httpd is installed in directory like: C:\Program Files (x86)\Apache Software Foundation\Apache2.2, the directory “C:\app\apache\sslcache\” should be manually created and configured in conf/extra/httpd-ssl.conf like below:

```
SSLSessionCache         "shmcb:C:\app\apache\sslcache\ssl_scache(512000)"
```

Restart apache HTTPD.

Check Your Configuration

If the reverse proxy is accessible from internal:

- https://<proxy_server>

If you’re using a self signed certificate you are required to add a security exception in your browser for this proxy server. The default apache web page, which normally just shows **It works !,** must appear.

Following web links must be accessible without error:

- https://<proxy_server>/fms
- https://<proxy_server>/plmapi
- https://<proxy_server>/Jacc
- https://<proxy_server>/AgilePlmWps

Additionally, if you use AutoVue:

- https://<proxy_server>/autovueproxy
- https://<proxy_server>/autovueapplet/jvue.jar

Setup the Java Client/Java Virtual Machine

To enable HTTPS support in the Java Client you have to activate/change the HTTP/S support setting.

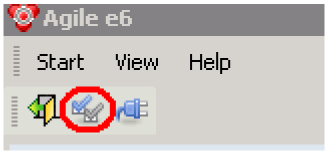
1. Download Java Client Web Start.
2. Open *https://<proxy_server>/Jacc* and select the download link.

Note If you’re using self signed certificate, a message appears that the certificate can not be validated. Accept it if you trust the certificate.

3. After the download of the application next warning appears that the digital signature of the application can not be verified. If you trust the publisher, select *Always trust content of this publisher* and run the application.

   The Java Client starts.

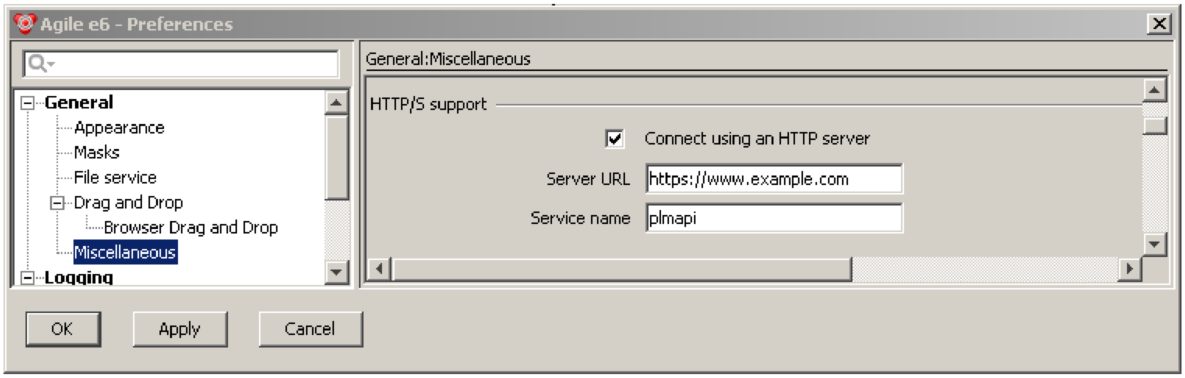
4. Open the Java Client Preferences mask.



5. Enable the HTTP/S support and change/add the Server URL and Service name to:

```
Server URL: https://<fully qualified proxy_server name>
Service name: plmapi
```

   Following picture shows an example configuration:



General:Miscellaneous

HTTP/S support

☑ Connect using an HTTP server

Server URL: https://www.example.com

Service name: plmapi

6. If you're using self signed certificate import certificate into Java Virtual Machine keystore.

   Connecting with the Java Client using *plmapi* over https requires Java to trust the end point of the communication. The end point in our example is the apache HTTPD reverse proxy server.

7. Perform the following steps:

   1. Transfer the file **server.crt** created on the proxy server in *<apache_home>/conf* to your Java Client machine.

   2. Determine JAVA_HOME of the Java Virtual Machine where your Java Client Web Start is running (for example, in Windows with process explorer).

   3. Copy the Public Certificate (server.crt, see above) to %JAVA_HOME%\lib\security.

   4. Open an Administrative Command shell and change to directory %JAVA_HOME%\lib\security.

5. Execute:

```
%JAVA_HOME%\bin\keytool -import -keystore cacerts -storepass
changeit -trustcacerts -alias mytrustcert -file server.crt
```

8. Close Java Client, open *https://<proxy_server>/Jacc* and select the download link again.

Java Client with Proxy

Proxy in this case does not mean not the reverse proxy, but the proxy for the outgoing connections from the client to the internet.

By default the Java Client uses the proxy configuration of your java environment. This is configured on the client side in the Java Control Panel.

To configure the proxy settings:

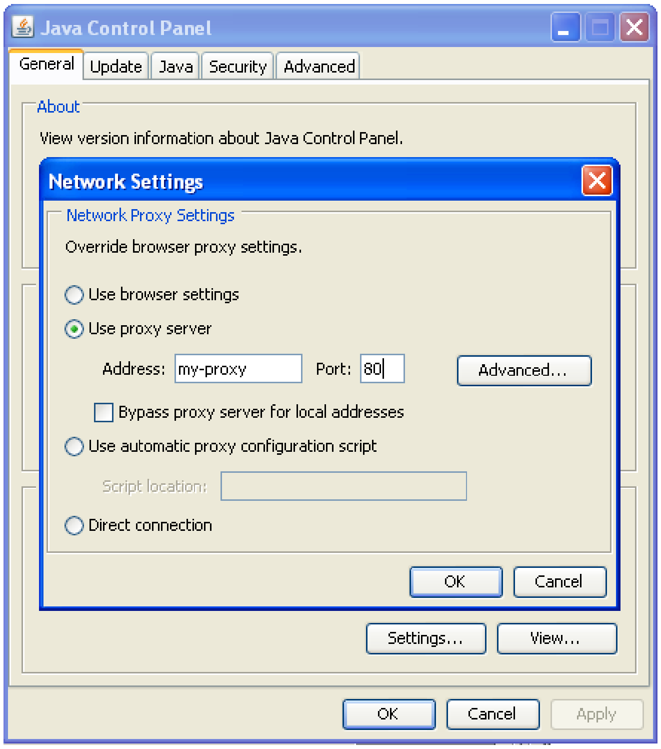
1. Open MS Windows Control Panel.

2. Execute Java.

   The Java Control Panel opens.

3. In the **General** tab, open the **Network Settings…**.

Note Depending on your proxy configuration, one of the proxy settings has to be selected.



View version information about Java Control Panel.

Override browser proxy settings.

○ Use browser settings

● Use proxy server

Address: my-proxy Port: 80

☐ Bypass proxy server for local addresses

○ Use automatic proxy configuration script

○ Direct connection

Setup the Web Fileservice

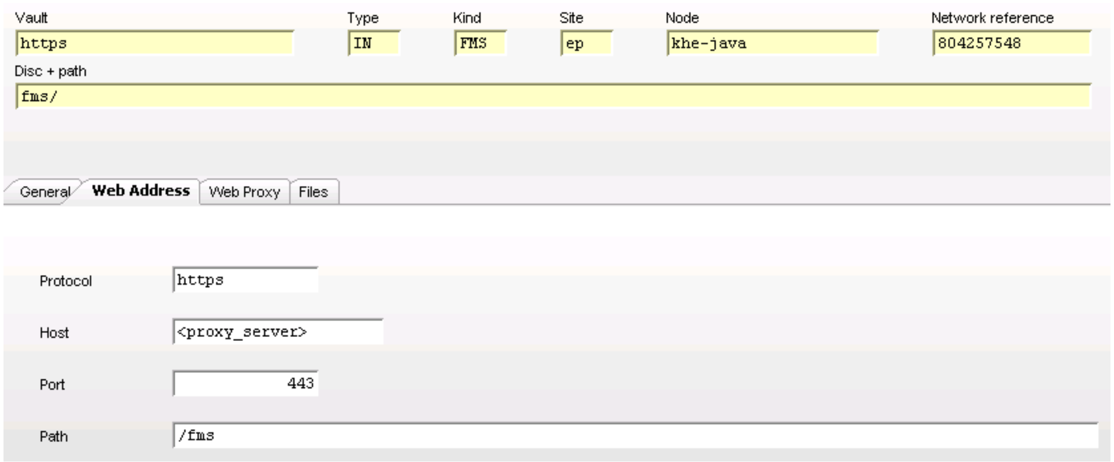
To enable HTTPS support for the Web Fileservice the Web address in the vault configuration has to be changed.

Limitation: As you can see in the following screenshot, you can only define one “Web Address” per Vault. This means the “Web Address” is same for internal and external Java/Web Client. If you want to use this vault for internal and external Java/Web Clients the <proxy_server> must be reached under his name from external and also from internal.

Start the Java Client with a manager user and select Manager > File Management > Vaults.

```
Protocol: https
Host: <proxy server>
Port: 443
Path: /fms
```

The following picture shows the configuration:



Vault	Type	Kind	Site	Node	Network reference
https	IN	FMS	ep	khe-java	804257548

Disc + path: fms/

Protocol: https
Host: <proxy_server>
Port: 443
Path: /fms

Setting Up AutoVue

General setup of AutoVue is described in the “*Installation and Administration Manual for AutoVue Integration on Windows for Agile e6.1.2.2*”.

Perform these steps only if you are using AutoVue.

1. Setup the AutoVue tunnelling servlet that it can be reached over https. For example, tomcat is configured with https.

2. Start a Java Client with a manager user and select “System > AutoVue > Configuration”.
3. To enable secure communication the following values must be changed:
    - EDB-PVM-AV-PROXY:

      Value (for example): https://<proxy_server>/autovueproxy

      Description: The URL where you can reach the AutoVue tunneling servlet over https.
    - EDB-PVM-AV-USE-PROXY:

      Value: true

      Description: To use HTTPS communication, set this to true.
    - EDB-PVM-AV-DMS:

      Value (for example): http://<weblogic_server>:<weblogic port>/VueLink/Vuelink

      Description: The Oracle Agile DMS Servlet address, where the AutoVue Server can reach the DMS Servlet.

      This must NOT be configured with https. Use http.
    - EDB-PVM-APPLET-BASE-URL

      Value (for example): https://<proxy_server>/autovueapplet

Description: The base https URL for the AutoVue viewer applet download.

Setup the Web Client

No configuration changes have to be done if you want to use the Web Client with HTTPS.

Use only the HTTPS protocol and port in your browser.

The Web Fileservice adaptations in the dump also have to be done (see above) to use file checkin/out in the Web Client with https. Proxy configuration will be used from the browser.

Change Lightweight Report URL

The **Report_Service_URL** attribute in all application configuration files in *<ep_root>/init/<application>.xml* must be changed.

1. Search the following text:

```
<PLMPresentationServices Report_Service_URL="…"/>
```

2. Adapt the *Report_Service_URL* to:

   `https://<proxy_server>/AgilePlmWps/reporter/report`

   If you change this line, it is valid for the complete application. An internal Client will also use this URL.

Secure Internal Communication

Secure internal communication is only possible by also enabling PLMAPI for the Java Client like for the external secure communication; otherwise the communication from the Java Client to the axalant server process is not secure.

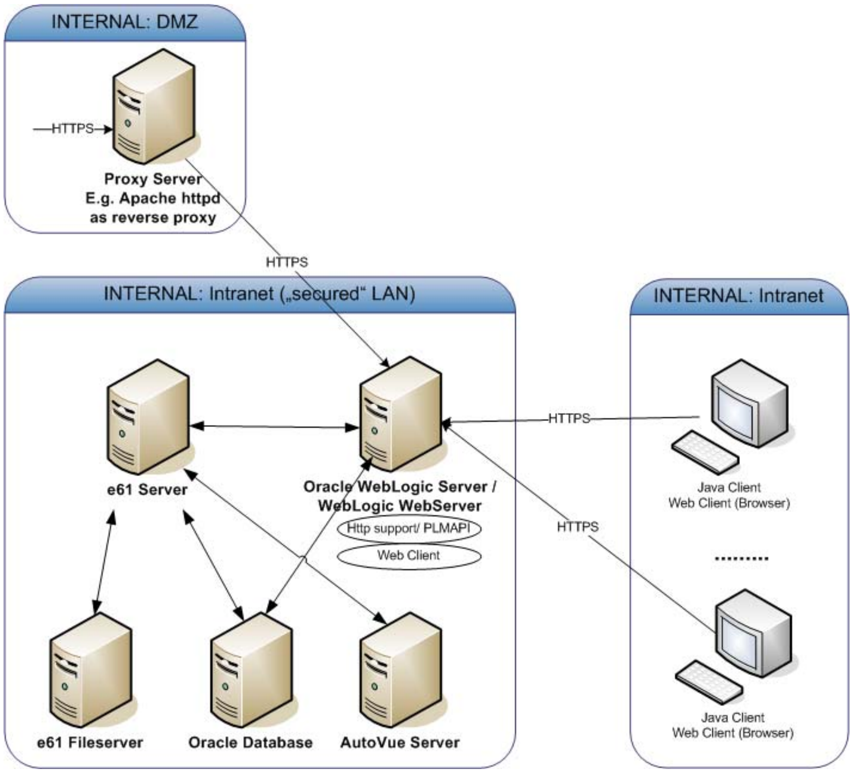
The Windows Client only uses unsecure communication to the axalant server process. Secure connection from the Windows Client to the axalant server process is not possible. Using the Windows Client in a secure environment is strongly discouraged.

Using Web Client with HTTPS and Web Fileservice with HTTPS protocol is also secure.

To use secure internal communication together with secure external communication, the proxy configuration has to be modified to a full SSL reverse proxy.

The communication between the servers in the “secured” LAN is unencrypted. Customers need to take care that network traffic in this area cannot be read by unauthorized persons.

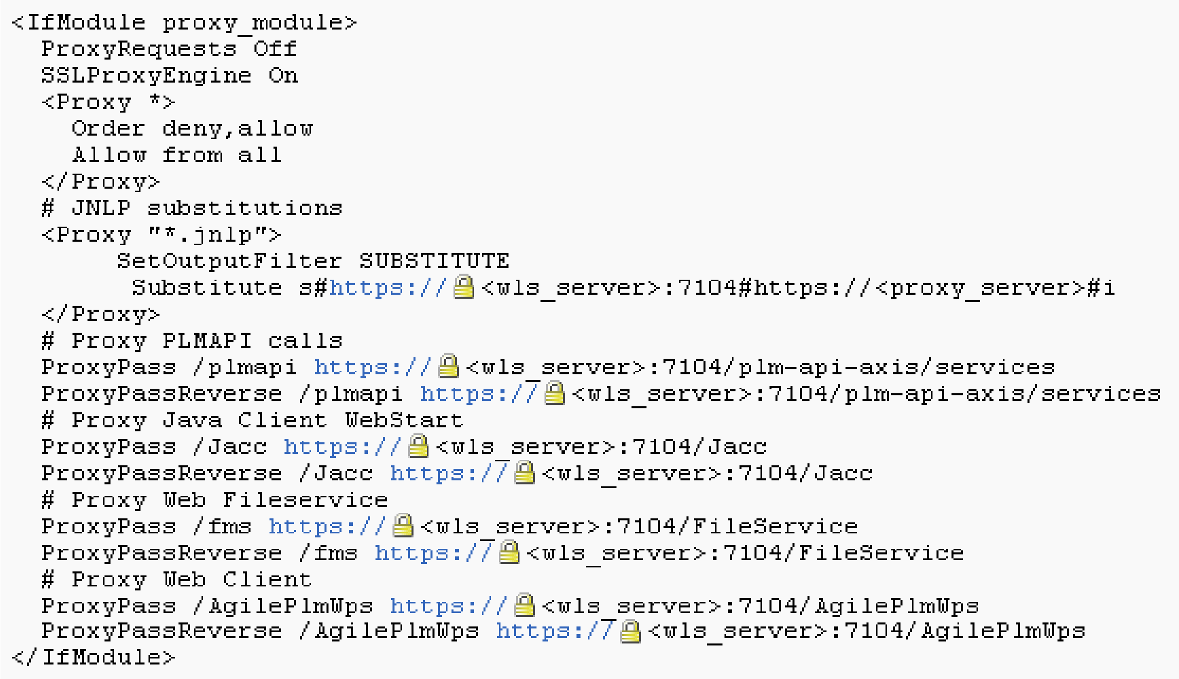
Following picture shows the secure internal communication.



Modify apache HTTPD to Full SSL Reverse Proxy

This proxy is still used only for external communication.

Make the following adaptations to the current proxy section in the *httpd.conf* file and restart apache HTTPD:



```
<IfModule proxy_module>
  ProxyRequests Off
  SSLProxyEngine On
  <Proxy *>
    Order deny,allow
    Allow from all
  </Proxy>
  # JNLP substitutions
  <Proxy "*.jnlp">
        SetOutputFilter SUBSTITUTE
         Substitute s#https://🔒<wls_server>:7104#https://<proxy_server>#i
  </Proxy>
  # Proxy PLMAPI calls
  ProxyPass /plmapi https://🔒<wls_server>:7104/plm-api-axis/services
  ProxyPassReverse /plmapi https://🔒<wls_server>:7104/plm-api-axis/services
  # Proxy Java Client WebStart
  ProxyPass /Jacc https://🔒<wls_server>:7104/Jacc
  ProxyPassReverse /Jacc https://🔒<wls_server>:7104/Jacc
  # Proxy Web Fileservice
  ProxyPass /fms https://🔒<wls_server>:7104/FileService
  ProxyPassReverse /fms https://🔒<wls_server>:7104/FileService
  # Proxy Web Client
  ProxyPass /AgilePlmWps https://🔒<wls_server>:7104/AgilePlmWps
  ProxyPassReverse /AgilePlmWps https://🔒<wls_server>:7104/AgilePlmWps
</IfModule>
```

This proxy is still used only for external communication. Internal communications use the direct connect to WebLogic HTTPS port and tomcat HTTPS port.

Restart apache HTTPD.

Setup HTTPS on the Oracle WebLogic Servers

The demonstration of digital certificates, private keys, and the trusted CA certificates used in this description should NOT be used in a production environment.

They are provided by default during WebLogic installation/domain setup.

For a production environment follow the steps in the WebLogic documentation to use non Demo digital certificates, private keys, and trusted CA certificates instead.

To activate SSL:

1. Start the Administration Console (e.g.: http://<>:7101/console)
2. If you have not already done so, in the Change Centre of the Administration Console, click Lock & Edit.
3. In the left pane of the Console, expand “Environment” and select “Servers”.
4. Click the name of the server for which you want to activate SSL Port.

5. At the tab “Configuration” -> “General”, configure "SSL Listen Port Enabled" and "SSL Listen Port" If you want to disable the non-SSL Port uncheck the “Listen Port Enabled” checkbox.

   If you disable non-SSL Port of the “AdminServer”, Agile e6 batch deployment will not work.

6. Select an SSL port which is not in use. Using a port number 1 above the "Listen Port" should be fine in most cases.

7. Save and release your configuration.

8. Activate the SSL ports for all servers (e.g: AdminServer, eSeries-01) in all Domains (e.g. eSeriesDomain, eSeriesDomain_plmref).

Setup the Java Client/Java Virtual Machine

For Internal usage of HTTP/S, , a different setup is required, because from internal Java Client the connection to the WebLogic server normally does not use any proxy.

To enable https support in the Java Client you have to activate/change the HTTP/S support setting.

1. Download Java Client Web Start

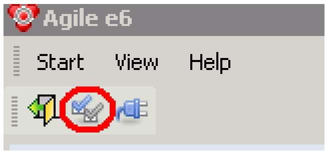
   Open: *https://<weblogic_server>:7104/Jacc* and select the download link.

   If you’re using WebLogic demo certificate a message occurs that the certificate cannot be validated. Accept it if you trust the certificate. This adds exception to your browser. Additionally the Java Web Start again warns that the certificate cannot be validated. Accept if you trust the certificate, and select “Always trust content of this publisher”.

   After the download of the application, an additional warning – “the digital signature of the application cannot be verified” appears. If you trust the publisher, select “Always trust content of this publisher” and run the application.

   The Java Client starts.

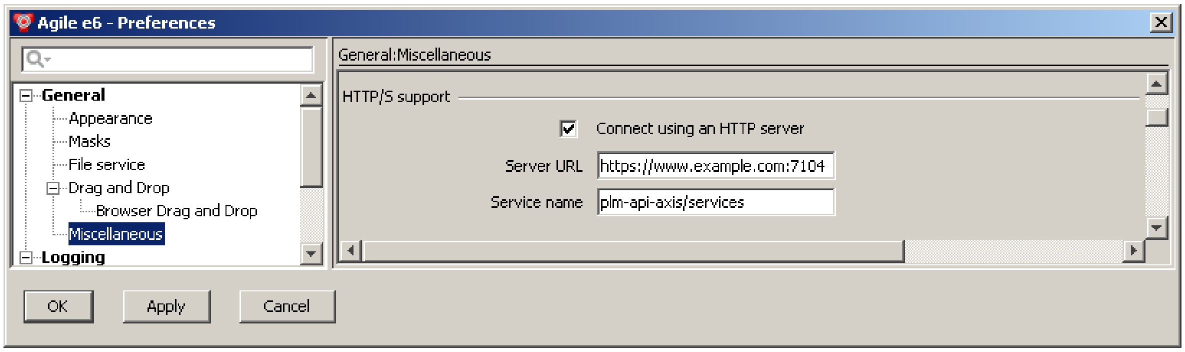
2. Open the Java Client Preferences mask.



3. Enable the HTTP/S support and change/add the Server URL and Service name to:

```
Server URL: https://<fully qualified weblogic_server
name>:<https_port>
Service name: plm-api-axis/services
```

Following picture shows an example configuration:



☑ Connect using an HTTP server

Server URL: https://www.example.com:7104

Service name: plm-api-axis/services

4. If you're using WebLogic demo certificate import certificate into Java Virtual Machine keystore.

   Connecting with the Java Client using plmapi over HTTPS requires Java to trust the end point of the communication. The end point in this example is the WebLogic server.

5. Perform the following steps:

   1. Convert the WebLogic demo certificate to "pem" format:

      1. Login to your weblogic server.

      2. Open a cmd shell and execute:

         ```
         cd /D <wlhome>/server/bin
         setWLSenv.cmd
         cd <wlhome>/server/lib
         java utils.der2pem CertGenCA.der
         ```

   2. Transfer the file "<wlhome>/server/lib/CertGenCA.pem" to your Java Client machine

   3. Determine JAVA_HOME of the Java Virtual Machine where your Java Client Web Start is running (e.g. in windows with process explorer)

   4. Copy the Public Certificate (CertGenCA.pem, see above) to %JAVA_HOME%\lib\security

   5. Open an Administrative cmd shell and change to directory %JAVA_HOME%\lib\security

   6. Execute:

      ```
      %JAVA_HOME%\bin\keytool -import -keystore cacerts -storepass
      changeit -trustcacerts -alias mytrustcert -file CertGenCA.pem
      ```

6. Close Java Client, open *`https://<weblogic_server>:7104/Jacc`* and select the download link again.

Setup Web Fileservice

For Internal usage of HTTP/S supports a different setup as for external usage is required, because from internal Java Client connection to the WebLogic server normally does not use a proxy.

To enable https support for the Web Fileservice the Web address in the vault configuration has to be changed.

Limitation see also external configuration.

Start the Java Client with a manager user and select Manager > File Management > Vaults.

```
Protocol: https
Host: <weblogic_server>
Port: 7104
Path: /FileService
```

Setup AutoVue

For Internal usage of HTTP/S supports a different setup as for external usage is required, because from internal the connection to the AutoView Proxy uses normally no proxy from dmz.

Start a Java Client with a manager user and select “System > AutoVue > Configuration”.

To enable secure internal communication the following values are different to the secure external communication:

- EDB-PVM-AV-PROXY:

  Value e.g.: `https://<view_proxy_server>:8443/ VueServlet/servlet/VueServlet`

  Description: The URL where you can reach the AutoVue tunneling servlet over https.

- EDB-PVM-APPLET-BASE-URL

  Value e.g.: https://<weblogic_server>:7104/VueLink

  Description: The base https URL for the AutoVue viewer applet download.

Change Lightweight Report URL

The “Report_Service_URL” attribute in all application configuration files in <ep_root>/init/<application>.xml must be changed.

Search the line:

<PLMPresentationServices Report_Service_URL="…"/>

and adapt “Report_Service_URL” to:

https://<weblogic_server>:7104/AgilePlmWps/reporter/report

If you change this line it is valid for the complete application. An internal/external Client will also use this URL.

Chapter 4

Authentication

LDAP Support

LDAP (Lightweight Directory Access Protocol) is an application protocol for querying and modifying directory services running over TCP/IP.

With Agile e6.1.2.2, LDAP-based authentication is supported. While a PLM user is logging on to the PLM system (through any supported client), the password of that PLM user is checked against an LDAP repository instead against the password which is normally stored in the PLM database.

The communication between the PLM server and the LDAP repository has to be set up. Every PLM user, as configured in the PLM database, has to exist in the LDAP repository in order to be authenticated upon login.

Note Although LDAP support helps in needing only ONE password for many different systems, this should not be mistaken with automatic Single-Sign-On (SSO) support, which would allow a user to log on automatically without even being asked to provide user login and password!

Prerequisites

- LDAP server (Oracle Internet Directory / MS Active Directory (does not work with encryption) / other LDAP server).
- Oracle LDAP client (part of the Oracle client installation).
- A site must have been defined in the DDM Site Vaults (see module “General Replication” of the online help).
- A PLM user name and a LDAP user name must have been created.

Note Please note that the PLM user name and the LDAP user name don’t need to be identical. But it is required that the LDAP user name is configured in the PLM system.

User Authentication via LDAP

A LDAP directory is often used to manage users and organization units in a central environment. Products like the Oracle Internet Directory are able to manage users, groups and organization units in a standard LDAP environment and are compatible with the most other LDAP servers which are based on the LDAP standards.

Note For more information on password policy for Oracle Internet Directory, refer to OID documentation on http://www.oracle.com/technetwork/documentation/oid-089101.html.

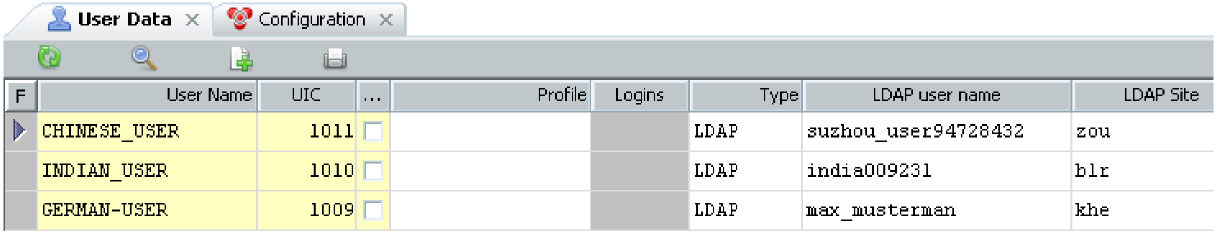
With Agile e6.1.2.2 the LDAP authentication mechanism supports the authentication of a PLM user password against a LDAP repository.

The LDAP for Agile e6.1.2.2 uses the Base-DN for a direct access path to authenticate the user. LDAP integration does not support relative search paths.

Note Most of these are configured on the LDAP server. Agile e6 uses its internal password security mechanism. Besides it can also authenticate against LDAP. Agile e6 uses its own policy for definition of internal password functionality.

Setup an LDAP User

To change the authentication mechanism of a user, select the “LDAP” entry as “Type” in the user list of Agile e6.1.2.2.



F	User Name	UIC	...	Profile	Logins	Type	LDAP user name	LDAP Site
▷	CHINESE_USER	1011	☐			LDAP	suzhou_user94728432	zou
	INDIAN_USER	1010	☐			LDAP	india009231	blr
	GERMAN-USER	1009	☐			LDAP	max_musterman	khe

Typically, an Agile e6 user has a different user name in the LDAP repository, therefore a LDAP user name field is supported to map the user names. To support LDAP multi-domains, the administrator can link each user to the site specific LDAP configuration.

The LDAP system takes care of the password policies (expiration and format).

Note The enhanced security module and the possibility to change the password within PLM are deactivated for LDAP users.

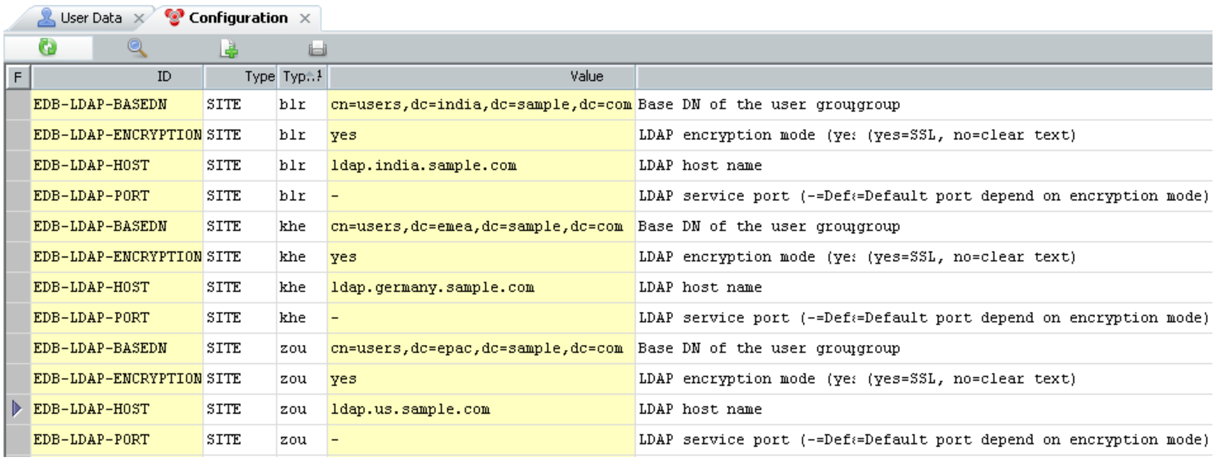
Configuration

The LDAP configuration used by the PLM system is stored in the database as configuration parameters (T_CFG_DAT).

Name	Default Value	Description
EDB-LDAP-HOST	<LDAP-host>	LDAP host name
EDB-LDAP-PORT	-	LDAP service port (=default port depends on encryption mode)
EDB-LDAP-BASEDN	cn=users, dc=agile, dc=com	Base DN of the user group
EDB-LDAP-ENCRYPTION	Yes	LDAP encryption mode (yes=SSL, no=clear text)

The configuration entries are site specific to support multi-domains. For each site different LDAP settings can be configured.

Example:



F	ID	Type	Typ...	Value	
	EDB-LDAP-BASEDN	SITE	blr	cn=users,dc=india,dc=sample,dc=com	Base DN of the user grougroup
	EDB-LDAP-ENCRYPTION	SITE	blr	yes	LDAP encryption mode (ye: (yes=SSL, no=clear text)
	EDB-LDAP-HOST	SITE	blr	ldap.india.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	blr	-	LDAP service port (-=Defa=Default port depend on encryption mode)
	EDB-LDAP-BASEDN	SITE	khe	cn=users,dc=emea,dc=sample,dc=com	Base DN of the user grougroup
	EDB-LDAP-ENCRYPTION	SITE	khe	yes	LDAP encryption mode (ye: (yes=SSL, no=clear text)
	EDB-LDAP-HOST	SITE	khe	ldap.germany.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	khe	-	LDAP service port (-=Defa=Default port depend on encryption mode)
	EDB-LDAP-BASEDN	SITE	zou	cn=users,dc=epac,dc=sample,dc=com	Base DN of the user grougroup
	EDB-LDAP-ENCRYPTION	SITE	zou	yes	LDAP encryption mode (ye: (yes=SSL, no=clear text)
▶	EDB-LDAP-HOST	SITE	zou	ldap.us.sample.com	LDAP host name
	EDB-LDAP-PORT	SITE	zou	-	LDAP service port (-=Defa=Default port depend on encryption mode)

Note Please note that a language specific configuration – though selectable from the Type column – is not relevant for the LDAP support!

Chapter 5

Securing Ports

The *Internet Assigned Numbers Authority* (IANA) administrates the port numbers in the range of 0 to 65535.

When it comes to assigning port numbers for services that are not registered, only port numbers of the so-called dynamic (private) range of 49152 to 65535 should be assigned in order to meet minimum security requirements. However, conflicts with already installed applications can occur.

But in practice already numbers from 1024 on are used for only the range of 0-1023 is protected. For example, operating systems should only allow the processes with appropriate privileges to open the server ports that are within the given range.

Prior to any installation, you should contact your system administrator in order to evaluate the ports that are already in use by the system and applications. Thus conflicts can be avoided when assigning ports that are used by Agile e6. Additionally, an existing firewall needs to be configured accordingly.

In case a system administrator is not available, a list of currently used TCP- and UDP –ports can be created with the command `netstat –a.` The RPC ports that are available through PortMapper can be determined using the command `rpcinfo –p.`

Range of Ports

The port numbers are divided into three ranges:

1. The well-known ports

   The well known ports are those from 0 through 1023. DCCP well known ports should **not** be used without IANA registration. The registration procedure is defined in (RFC4340), section 19.9.

2. The registered ports

   The registered ports are those from 1024 through 49151. DCCP registered ports should **not** be used without IANA registration. The registration procedure is defined in (RFC4340), section 19.9

3. The dynamic and/or private ports

   The dynamic and/or private ports are those from 49152 through 65535.

Note Unassigned port number should not be used. The IANA will assign the number for the port after your application has been approved.

Note Assignment of a port number does not in any way imply an endorsement of an application or product, and the fact that network traffic is flowing to or from a registered port does not mean that it is “good” traffic. Firewall and system administrators should choose how to configure their systems based on their knowledge of the traffic in question, not whether there is a port number registered or not.

Well-Known Port Numbers

The well known ports are assigned by the IANA and on most systems can only be used by system (or root) processes or by programs executed by privileged users.

Ports are used in the TCP [RFC793] to name the ends of logical connections which carry long term conversations. For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port. The contact port is sometimes called the "well-known port".

To the extent possible, these same port assignments are used with the UDP [RFC768]. The range for assigned ports managed by the IANA is 0-1023.

Registered Port Numbers

The registered ports are listed by the IANA and on most systems can be used by ordinary user processes or programs executed by ordinary users.

Ports are used in the TCP [RFC793] to name the ends of logical connections which carry long term conversations. For the purpose of providing services to unknown callers, a service contact port is defined. This list specifies the port used by the server process as its contact port.

The IANA registers uses of these ports as a convenience to the community. To the extent possible, these same port assignments are used with the UDP [RFC768]. The Registered Ports are in the range 1024-49151.

Dynamic and/or Private Ports

The Dynamic and/or Private Ports are those from 49152 through 65535.

Range of Values and Dependencies

Service	Ports (default value)	Dependencies
Sun Portmapper (RPC)	111	Under Unix always present, under Windows a component of the

		Agile e6 delivery
Admin Server	▫ HTTP (8027) ▫ HTTPS (8028)	
DataView Daemon	▫ RPC Port (552000566) ▫ one free port per application server	Sun Portmapper
Java Daemon	▫ StandardPort (16077) ▫ AdminPort (16078), only local ▫ RegistrationPort (16079), only local ▫ One free port from the port range per application server	
FileServer	▫ RPC port (804257548) ▫ Web File Service (8088) ▫ one free port per client connection	Sun Portmapper Web Presentation Service
e6 Server	Per session one port assigned from the daemon.	Sun Portmapper Business Service FileServer
Web Presentation Service	Ajp12 Port (7077), integration TomCat in Apache Shutdown Port (8005) Web Client (8088) Web Report Service (8088) ViewCafé (2099)	Java Daemon
Business Service	JMX (12808) ECI Port (19997) one free port per connection to the Agile e6 server	Java Daemon Agile e6 Server SMTP port (25).
Java Client	ECI Topic (4444) Needs to be distinct for every client call und can be set with the start.	Java Daemon Agile e6 Server
DataView Client	ECI Topic (DDE: eci_dde_loop) Needs to be distinct for every client call und can be set with the start. Instead of DDE also TCP/IP is possible.	Sun Portmapper DataViewDaemon Agile e6 Server
Workflow Editor		Business Service (ECI Port)
Office Suite	DDE/OLE/COM	DataView Client Agile e6 Server
EIP	Queue Port (9001)	Java Daemon

	Admin Port (9876) Log Port (4445) Web Server (8080) synchronous: ECI Server Port (19997) Note Here there's a conflict with the standard ECI port of the Business Service.	Agile e6 Server