

# Legal Notices

**Copyright © 2016, Oracle and/or its affiliates. All rights reserved.**

## **Trademark Notice**

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

## **License Restrictions, Warranty/Consequential Damages Disclaimer**

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.

## **Warranty Disclaimer**

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.

## **Restricted Rights Notice**

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.

## **Hazardous Applications Notice**

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

**Third Party Content, Products, and Services Disclaimer**

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.

## Overview

Security planning is a critical step to help protect your company's valuable data and ensure that information is not compromised. Established security policies and goals should guide the security plan your organization executes to secure its systems.

The Oracle Insurance Policy Administration (OIPA) system stores sensitive data and requires security measures to be taken. Security policies should align with those already established at your organization, or new ones should be established if they are not already defined.

This document provides guidelines for securing an OIPA installation, including the configuration and installation steps needed to meet security goals. Details on the types of security features and services that are available to detect and prevent a potential security breach are provided. These details encompass secure system deployment, protection of sensitive data, reliability and availability of the application, authentication and authorization mechanisms.

You may use this document to develop your organization's security policies and practices in the context of OIPA. It is critical that an organization set security standards and properly implement them. The development and review of security documentation, an evaluation of business requirements, and the configuration and validation of available security measures and services should all be performed.

## Customer Support

- » If you have any questions about the installation or use of our products, please visit the My Oracle Support website: <https://support.oracle.com>, or call (800) 223-1711.
- » 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.

## System Deployment

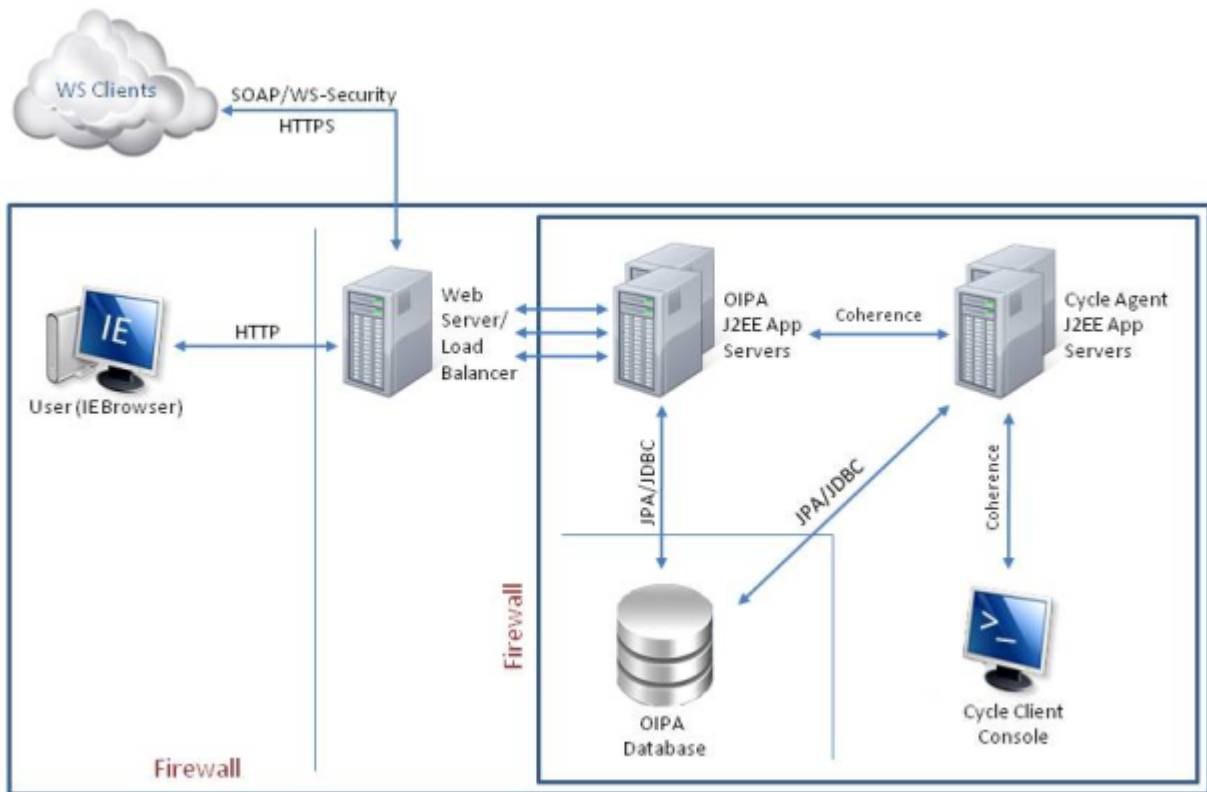
### Network Security in OIPA Environment

When deploying OIPA on a network there are many security issues to take into consideration, especially the use of firewall and VPN technologies. A firewall will permit or deny network permissions based on configured rules, to protect the internal network from unauthorized access while permitting legitimate communications. Firewalls perform the following functions in a typical OIPA environment:

- » Guard the company Intranet from unauthorized outside access.
- » Separate Intranet users accessing the OIPA system from internal sub-networks where critical corporate information and services reside.
- » Protect from IP spoofing and routing threats.
- » Prohibit unauthorized users from accessing protected networks and control access to restricted services.

The OIPA user interface is browser-based and allows home-office users to access the application services. It is recommended that the users access the application from within the company network, secured behind the outside firewall. Virtual Private Network (VPN) technology should be used to allow employees working remotely to access the OIPA application. A VPN tunnels outside traffic through the firewall, placing outside clients virtually inside the firewall.

It may be required to provide access to the OIPA web services for external clients that are not allowed inside the company firewall. In that case, the web services must only be accessed through HTTP secured with SSL. OIPA web services support WS-Security standards, enabling web service user authentication using OIPA user accounts. Please make sure that the firewalls used to secure an OIPA environment support the HTTP 1.1 protocol. This enables browser cookies and inline data compression for improved performance.



*Firewalls in the OIPA environment*

Typical OIPA environment usually has the following security zones:

- » Internet - External web service clients may come from outside of the company network.
- » Intranet - A company network separated by the external firewall that gives home users access to the OIPA user interface. This is also where OIPA web servers and load balancers may be placed. Alternatively, for additional protection, web and load balancing servers may be placed in a separate demilitarized zone (DMZ) where external and internal clients first interact with the OIPA environment.
- » OIPA application server and database zone - OIPA application servers, including Cycle Web servers, database servers and possibly authentication servers (for example, if a customer chooses to implement a single sign-on using LDAP servers) reside in this zone. Access to the database that holds critical client information must be secured, with access restricted to system and database administrators only.

**Database**

Appropriate database users for the OIPA application should be set up as per the instructions in the associated version of the OIPA Database Install Instructions, which is located on the Oracle Technology Network. OIPA expects a Read-Only database user to be set up so that additional restrictions can be enforced on certain operations.

## OIPA Use of Coherence

The OIPA application uses the Oracle Coherence distributed cache solution to minimize database traffic. In addition to using the cache, OIPA Cycle uses the Coherence Processing Pattern as a computing grid to allow task distribution among all OIPA Cycle Web. Batch processing on the grid is initiated through the Coherence communication protocol by the Cycle Client. Even though all parties involved in Coherence communications are located behind the firewall in the OIPA application server and database zone, it is important, nevertheless to secure Coherence according to the Coherence User Guide.

Oracle Coherence also provides workload management to distribute tasks across a computer cluster or other resources. This enables Cycle to achieve optimal resource utilization, maximize throughput, minimize response time and avoid overload, as well as avoid having a single point of failure for tasks processed in the grid. Along with the security provided by the firewalls, Coherence workload management provides these additional security features:

- » TCP port exposure is limited to a single port that allows easier port security and firewall configuration.
- » A virtual IP address hides actual physical IP addresses of the OIPA application servers.
- » The suspect protocol protects against Denial of Service (DoS) attacks by detecting and barring "rogue" clients that attempt to overuse server resources.

## Configuring SSL

The Secure Sockets Layer (SSL) protocol provides communication security by encrypting traffic across a network in a way designed to prevent eavesdropping and tampering. It uses asymmetric cryptography for privacy and a keyed message authentication code for message reliability. Setting up an SSL-secured connection requires a digital certificate issued by a trusted certificate authority. Self-signed digital certificates should only be used for internal testing.

Any entry points for OIPA web services that are consumed by external third party clients should be secured with SSL. Also, organization standards may require securing communication between browser-based clients and web servers in the demilitarized zone that host the front end of the OIPA system.

Setting up a web server to use SSL-secured HTTP protocol (HTTPS) instead of unsecure HTTP is server-specific. The information below should help locate information to navigate through the configuration process.

### SSL in WebLogic 12.2.1.1

WebLogic Application Server supports SSL 3.0 and Transport Layer Security (TLS) 1.0 specifications. TLS V1.1 is the default minimum protocol version configured in WebLogic Server. Oracle recommends the use of TLS V1.1 or later in a production environment. WebLogic does not support SSL version 2.0 and below.

For information on how to configure SSL in WebLogic please refer to the following websites or follow the steps below:

[https://docs.oracle.com/middleware/1221/wls/SECMG/ssl\\_overview.htm#SECMG718](https://docs.oracle.com/middleware/1221/wls/SECMG/ssl_overview.htm#SECMG718)

<http://download.oracle.com/javase/6/docs/technotes/guides/security/jsse/JSSERefGuide.html>

## Steps to Configure SSL/https:

1. Login to the WebLogic console.
2. In the Domain Structure box, expand **Environment** and click **Servers**.
3. Click on the server that you created. Example: **OIPA\_SERVER**.
4. Select the SSL Listen Port Enabled checkbox.

**Example:** 7002 is port number.

5. Click **Save**.
6. Restart the server.
7. Navigate to <https://machinename:7002/PASJava> in your browser to access the login page of OIPA.

[http://docs.oracle.com/middleware/1221/wls/SECMG/identity\\_trust.htm#SECMG720](http://docs.oracle.com/middleware/1221/wls/SECMG/identity_trust.htm#SECMG720)

## Steps to Configure Certificates:

The steps listed below are based on the default JDK certificate.

WEBLOGIC\_JAVA\_SECLIB = Specify the location of JDK 1.8.x. /jre/lib/security.

For Example: /opt/oracle/jdk1.8.0\_77/jre/lib/security

WEBLOGIC\_JAVA\_HOME = Specify the location of JDK 1.8.x.

For Example: /opt/oracle/jdk1.8.0\_77/

**Note:** If JDK is not installed on your machine, then download and install latest update of Oracle 1.8 JDK

1. Install the Oracle WebLogic 12.2.1.1 application server.
2. Go to WEBLOGIC\_JAVA\_HOME\bin and run the commands listed below.
  - » keytool -genkey -keystore jre/lib/security/wsse.keystore -keyalg RSA -keysize 1024 -validity 1000 -alias localhost -dname "CN=localhost"
  - » keytool -export -keystore jre/lib/security/wsse.keystore -alias localhost -file server/default/conf/localhost.cer
  - » keytool -import -keystore jre/lib/security/wsse.truststore -trustcacerts -alias localhost -file jre/lib/security/localhost.cer
3. The above step will create two files within WEBLOGIC\_JAVA\_SECLIB.
  - » wsse.keystore
  - » wsse.truststore
4. Move wsse.keystore and wsse.truststore to the conf folder where all properties files reside. Example: C:\OIPA\conf.
5. Log in to the Oracle Weblogic console and go to Environment > Server > OIPA > Server Start and add the details listed below to Arguments.
  - » -Duser.language=en -Duser.region=US -Djava.net.preferIPv4Stack=true -Djava.net.preferIPv6Addresses=false -javaagent:C:\OIPA\lib\spring-instrument-4.2.3.RELEASE.jar -Dtangosol.coherence.override=C:\OIPA\conf\coherence-config.xml -Dtangosol.coherence.cacheconfig=C:\OIPA\conf\coherence-cache-config.xml -Dtangosol.pof.config=com-adminserver-pas-web-pof-config.xml -Djavax.net.ssl.trustStore=C:\OIPA\conf\wsse.truststore -Djavax.net.ssl.trustStorePassword=Djavax.net.ssl.keyStore=C:\OIPA\conf\wsse.keystore -Djavax.net.ssl.keyStorePassword=jbossws
6. Go to WEBLOGIC\_JAVA\_SECLIB and create a back-up of the cacerts file.

7. Create a new certification (cacerts) file by following the steps below.
  - » Copy InstallCert.class and InstallCert\$SavingTrustManager.class in WEBLOGIC\_JAVA\_HOME\bin.
  - » From WEBLOGIC\_JAVA\_HOME\bin, run InstallCert through a command prompt like java InstallCert localhost:7002. The KeyStore jssecacerts will load and a connection will be opened. Messages will then be presented regarding the certificates.
  - » When the process is complete, the following message will appear: Enter certificate to add to trusted key-store or 'q' to quit. Type 1 to continue.
  - » When the process is complete, another message will appear: Added certificate to keystore 'jssecacerts' using 'jssecacerts' using alias 'localhost-1'. Run java InstallCert localhost:7002 one more time, then enter q to exit. This will create a new jssecacerts keystore file in WEBLOGIC\_JAVA\_SECLIB and rename it to cacerts.

**Note:** Repeat step 7 to enable SSL for different port numbers.

8. Stop the WebLogic application server (JVM, Node, Manager).
9. Restart the machine.
10. Start the WebLogic application server (JVM, Node, Manager).
11. Enter https://machinename:7002/PASJava in your browser to access the login page of OIPA.

## SSL in WebSphere 8.5.5.9

Version 8 of WebSphere Application Server, everything is done from the admin console, which includes a complete overview of the SSL management capabilities.

For more information about managing SSL in WebSphere please refer to the following website or follow the steps listed below.

<http://pic.dhe.ibm.com/infocenter/wasinfo/v8r0/index.jsp>

**Note:** Search for Overview and new features: Securing under Network Deployment

## Steps to Configure SSL/https

1. Login to the WebSphere console.
2. Expand **Server Types** and click **WebSphere Application Servers**.
3. Click on the server that you created. Example: OIPA\_WILDCAT\_10.0.0.0
4. Expand Port and copy WC\_defaulthost\_secure=port number. This will be pasted in step 7.
5. From the left side menu expand **Environment** and click **Virtual Host**.
6. Click **default\_host** and click **Host Aliases**.
7. Click **New** and copy the port number from step 4, then click **OK**.
8. Restart the server/JVM.
9. Navigate to **https://machinename:9444/PASJava** in your browser to access the login page of OIPA.

## Steps to Configure Certificates

### 32 bit WebSphere Application Server

IBM\_JAVA\_SECLIB = C:\Program Files (x86)\WebSphere\AppServer\java\jre\lib\security

IBM\_JAVA\_HOME = C:\Program Files (x86)\IBM\WebSphere\AppServer\java



## 64 bit WebSphere Application Server

IBM\_JAVA\_SECLIB = C:\Program Files\WebSphere\AppServer\java\jre\lib\security

IBM\_JAVA\_HOME = C:\Program Files\IBM\WebSphere\AppServer\java

1. Download and install IBM JDK, if WebSphere is not installed on the machine.
  - » URL to download <http://www.ibm.com/developerworks/java/jdk/>
2. Start the WebSphere application server
3. Enable SSL in WebSphere.
  - » Log in to the WebSphere console.
  - » Expand Server Types and click WebSphere Application Servers.
  - » Click on the server that you created. Example: OIPA\_WILDCAT\_10.0.0.0
  - » Expand Port and copy WC\_defaulthost\_secure=port number. This will be copied later in the process.
  - » From the left menu, expand Environment and click Virtual Host.
  - » Click default\_host and then click Host\_Aliases.
  - » Click New and copy the port number then click OK.
  - » Go to IBM\_JAVA\_SECLIB\security and comment the details below in the java.security file.

**Note:** Make sure to uncomment Default JSSE socket factories and comment WebSphere socket factories (in cryptosf.jar).

### Default JSSE socket factories

```
ssl.SocketFactory.provider=com.ibm.jsse2.SSLSocketFactoryImpl
```

```
ssl.ServerSocketFactory.provider=com.ibm.jsse2.SSLServerSocketFactoryImpl
```

### WebSphere socket factories (in cryptosf.jar)

```
#ssl.SocketFactory.provider=com.ibm.websphere.ssl.protocol.SSLSocketFactory
```

```
#ssl.ServerSocketFactory.provider=com.ibm.websphere.ssl.protocol.SSLServerSocketFactory
```

4. Stop the server, Node Agent and Deployment Manager.
5. Start the Deployment Manager, Node Agent and server.
6. Navigate to <https://localhost:9445/PASJava> in your browser to make sure SSL works as expected.
7. Log in to the application. If this action is successful, then SSL is set up correctly from the server side.
8. Go to IBM\_JAVA\_HOME\bin and run the commands listed below.
  - » `keytool -export -keystore ../lib/security/wsse.keystore -alias localhost -file ../lib/security/localhost.cer`
  - » `keytool -import -keystore ../lib/security/wsse.truststore -trustcacerts -alias localhost -file ../lib/security/localhost.cer`
9. The step above will create two files within IBM\_JAVA\_SECLIB.
  - » wsse.keystore
  - » wsse.truststore
10. Move wsse.keystore and wsse.truststore to the conf folder where all properties files reside. For example:  
C:\OIPA\conf
11. Login to the WebSphere console, and go to Application servers > OIPA > Process definition > Java Virtual Machine. Add the arguments listed below to JVM.
  - » `-Duser.language=en --Duser.region=US -Djava.net.preferIPv4Stack=true -Djava.net.preferIPv6Addresses=false -javaagent:C:\OIPA\lib\spring-instrument-4.2.3.RELEASE.jar -Dtangosol.coherence.override=C:\OIPA\conf\coherence-config.xml -Dtangosol.coherence.cacheconfig=C:\OIPA\conf\coherence-cache-config.xml -Dtangosol.pof.config=com-adminserver-pas-web-pof-config.xml -`

```
Djavax.net.ssl.trustStore=C:\OIPA\conf\wsse.truststore -Djavax.net.ssl.trustStorePass-
word=Djavax.net.ssl.keyStore=C:\OIPA\conf\wsse.keystore -Djavax.net.ssl.keyStorePass-
word=jbossws
```

12. Go to IBM\_JAVA\_SECLIB and take a backup of the **cacerts** file.
13. Create a new certification (cacerts) file by following the steps listed below.
  - » Copy InstallCert.class and InstallCert\$SavingTrustManager.class in IBM\_JAVA\_HOME\bin.
  - » From IBM\_JAVA\_HOME\bin, run InstallCert through a command prompt like java **InstallCert localhost:9445**. The KeyStore jssecacerts will load and a connection will be opened. Then messages will be presented regarding the certificates.
  - » When the process is complete, the following message will appear: **Enter certificate to add to trusted keystore or 'q' to quit**. Type **1** to continue.

When the process is complete, another message will appear: Added certificate to keystore 'jssecacerts' using 'jssecacerts' using alias 'localhost-1'. Run java InstallCert localhost:9445 one more time, then enter q to exit. This will create a new jssecacerts keystore.

**Note:** Repeat step 7 to enable SSL for different port numbers.

14. Stop the WebSphere application server (JVM, Node Agent, Deployment Manager).
15. Restart the machine.
16. Start the WebSphere application server (JVM, Node Agent, Deployment Manager).
17. Navigate to **https://machinename: 9445/PASJava** in your browser to access the login page of OIPA.

When OIPA is running in HTTPS, Server shall be configured to set HSTS (HTTP Strict Transport Security) policy. This policy lets a web site inform the browser that it should never load the site using HTTP and should automatically convert all attempts to access the site using HTTP to HTTPS requests instead.

For security reasons, any cross-origin requests, i.e., requests coming from different domains to OIPA Rest Services will be restricted by browsers. To enable cross-origin requests, the server shall be configured to add a header 'Access-Control-Allow-Origin' in the response. A list of allowed domains shall be set to this header. Setting Access-Control-Allow-Origin:\*, allows all domains. OIPA also supports this configuration by setting the property application.AccessControlAllowOrigin with list of allowed domains in System Properties document.

By default, OIPA does not support loading pages in an iframe when loaded from other domains. To allow iframes from other domains, the property application.X-Frame-Options shall be set with list of allowed domains. See the System Properties document for further details.

## JMS

JMS set-up is optional. It is only required if the Data Intake feature is being utilized. See Data Intake document for additional details.

Data Intake is the process of receiving files from Group Customers for the purpose of importing data into the Oracle Insurance Policy Administration system (OIPA). The data in the files may result in many changes, including but not limited to the following:

OIPA - Security Guide, Release 11.0.0.0

- » Adding a new member to the system
- » Changing an Employee's elected coverage
- » Adding a dependent to a coverage
- » Enrolling a member and dependents
- » Auto-cancelling coverage for a member
- » Updating member information in the system

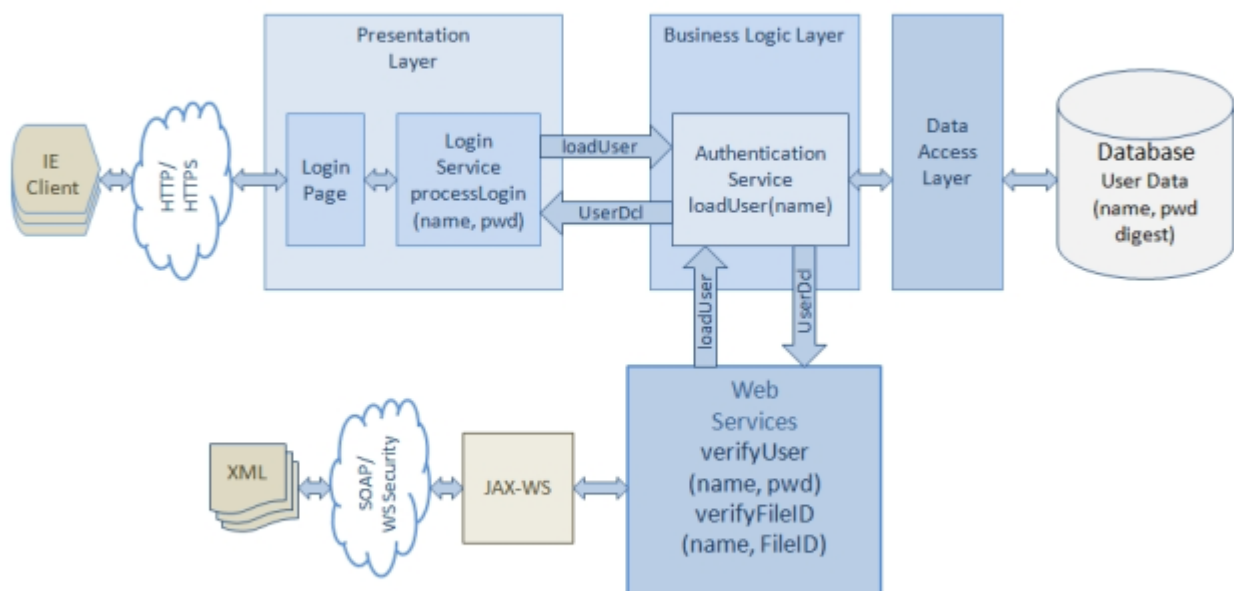
The received files are parsed and information about the data in them is put on a JMS queue. OIPA listens for messages on the queue and updates the business data based on pre-configured rules.

## User Authentication

The OIPA application provides an out-of-the box user authentication mechanism as well as an ability to implement alternative authentication models like a Single Sign-On (SSO) authentication through the OIPA extensions. If the system is implemented with SSO, additional measures need to be taken to properly secure the authentication infrastructure. Depending on the implementation chosen, either an authentication server should be placed within the OIPA application server and database zone, or the call to an authentication service needs to be made via a secure connection.

Out-of-the box OIPA user authentication is performed for interactive users using web browsers to access the system, and for incoming web service calls. Interactive users are prompted on the application's login page to provide a username and password to authenticate to the server. Web services are protected with WS-Security, which requires incoming web service calls (which must be transmitted on a secure SSL connection to carry a security header with a user name and password).

Both web service and interactive user authentication are implemented through the same authentication service provided by the business logic tier of the OIPA application. The authentication service retrieves a matching user record from the OIPA database that contains basic user information and a secure digest of a password. The password digest is then compared to the digest of the incoming password and an authentication decision is made based on the result of the comparison. For certain web services, apart from user authentication, additional functional security is also enforced to control whether those services can be executed



The encrypted password digest is created by the Rules Palette when a user is created. When a new OIPA environment is created using the Rules Palette's Web Application Utility, the process allows for the configuration of the encryption parameters to be used by the encryption algorithm. The settings include the particular encryption algorithm (from the list of the supported algorithms below), and the number of iterations of the algorithm.

- » SHA-256
- » SHA-384
- » SHA-512

The number of encryption iterations is a value between 1000 and 9999. A higher number of iterations makes the password more secure, but also requires more computation to encrypt. For more information, please refer to the associated version of the Rules Palette Help System that is located on the Oracle Technology Network.

## User Management

### User Registration

A user must have an existing OIPA user account identified by username and password to log into the OIPA application. An OIPA administrator uses the Rules Palette to create a new OIPA user account. The OIPA administrator's Rules Palette credentials must be associated with a security group that allows for the management of security. With the proper security rights, the administrator may use the Rules Palette to add, edit and delete OIPA user accounts. The administrator can also add and edit Security Groups that determine what features and authorizations are available to the users that belong to each Security Group. When creating a new user account, an administrator enters or selects the following information:

- » User's login name and password
- » Basic information about user – first and last name, email, gender, etc.
- » User's primary company
- » Locale
- » Security groups to which the user belongs

This information is persisted in the OIPA database, with the encrypted password digest stored as discussed in the User Authentication section of this document.

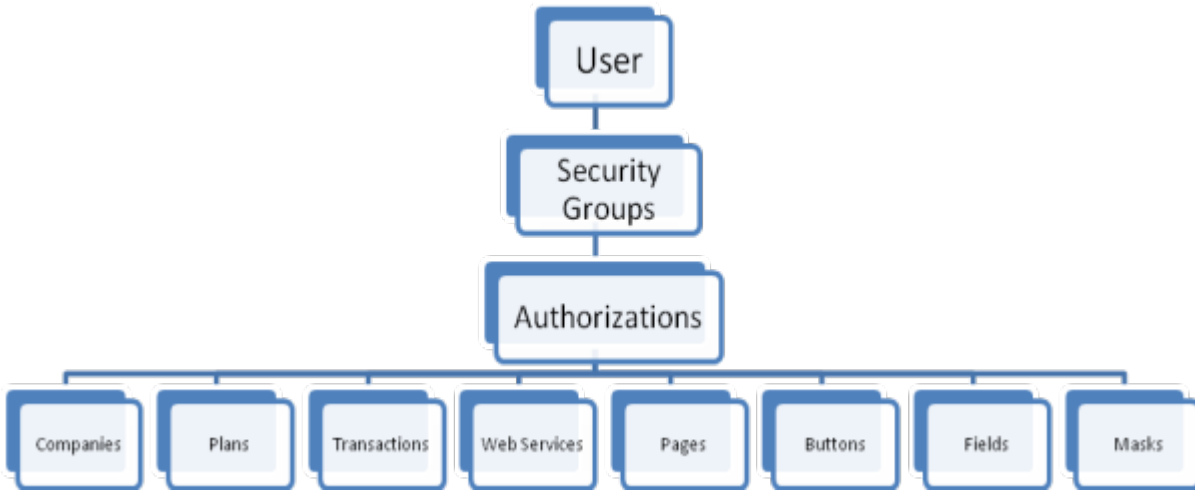
There are no pre-existing or default user accounts or security groups in the OIPA application that need to be disabled after the system is deployed. The OIPA application user interface may be accessed only after at least one user account is created through the Rules Palette.

### User Privileges and Group-Based Access Control

The OIPA user privileges and access restrictions implementation is based on the role-based access control (RBAC) model. According to the model, user permissions are assigned to specific groups or roles that are created for various job functions. A user who is assigned to a particular group gains permissions through those groups to perform particular system functions. If a user is assigned to multiple groups, the user will have access to all resources authorized for all of those groups.

For example, users that are assigned to the CSR group (or role) may not be able to execute such activities as issuing a policy or paying a death benefit. By contrast, a user in an Underwriter group should be able to issue a policy. A user in an administrator group is usually allowed access to all resources.

The following figure shows what application resources are protected by OIPA security.



**Figure 3. Hierarchy of User Authorizations**

By default, a newly created user account does not have authorizations to access any of the application restricted resources. Authorizations have to be explicitly granted by an OIPA security administrator. When setting up the user groups, an administrator needs to be careful to include only the minimum set of permissions that allow users of a particular group to perform their job functions.

For more information on how to create security groups and manage user accounts please refer to the Rules Palette Help.

## Web Services Security

OIPA uses JAS-WS for implementing Web Services. For securing web services, WS-Security standards are used to perform authentication and authorization against OIPA user accounts. The SOAP header contains the appropriate security credentials. The password can be sent as a digest or as a text.

The SOAP header with WS-Security would look like the following when a password digest is used:

```
<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
<soapenv:Header>
<wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
<wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
<wsse:Username>username</wsse:Username>
<wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordDigest">passwordencrypted</wsse:Password>
<wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">kC5eI6iq8x17/qA3mzs6/g==</wsse:Nonce>
<wsu:Created>2010-03-22T14:12:34.223Z</wsu:Created>
</wsse:UsernameToken>
</wsse:Security>
</soapenv:Header>
```

By default, starting with the 10.2.2.0 release, OIPA will not allow web services to be invoked with hashed passwords. To continue using PasswordDigest, the property `webservice.allowHashedPassword` should be set to Yes. See the System Properties document in OTN for further details.

For more information on the WS Security standard please refer to the website:

<http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>

## Using Cookies in the OIPA application

The OIPA application is accessed by users through a Web Browser. Because OIPA uses session cookies to manage user sessions, cookies must be enabled in the browser. To allow the use of cookies in Internet Explorer, open the Privacy tab of the Internet Options dialog, then choose the Sites popup dialog and add the OIPA server address to the list of Allowed sites.

OIPA uses 'jSessionIDHeader' in the HTTP request header to manage data associated with the user's session. When user first logs into the OIPA application successfully, system creates a unique sessionID and passes to the browser. Browser sets this to jSessionIDHeader and retains this for the duration of session. When user logs out or session expires, browser deletes this sessionID. Requests that do not contain valid session IDs are not processed by the server.



## Additional Sources of Security Information

In addition to securing the OIPA application, all infrastructure resources – Linux/Windows servers, J2EE application and database servers – that compose an OIPA environment must be secured. The following list of links should be helpful while planning how to secure an OIPA environment.

### Coherence 12.2.1.1 User Guide

[http://docs.oracle.com/cd/E24290\\_01/index.htm](http://docs.oracle.com/cd/E24290_01/index.htm)

### Securing Oracle Coherence 12.2.1.1

<https://docs.oracle.com/middleware/12211/coherence/secure/introduction-oracle-coherence-security.htm#COHSG5492>

### Oracle 12c Database

[http://docs.oracle.com/cd/E16655\\_01/network.121/e17607.pdf](http://docs.oracle.com/cd/E16655_01/network.121/e17607.pdf)

[http://docs.oracle.com/cd/E16655\\_01/network.121/e17729/toc.htm](http://docs.oracle.com/cd/E16655_01/network.121/e17729/toc.htm)

[http://docs.oracle.com/cd/E16655\\_01/network.121/e17731.pdf](http://docs.oracle.com/cd/E16655_01/network.121/e17731.pdf)

### Microsoft SQL Server 2008 Database

<http://www.microsoft.com/sqlserver/2008/en/us/Security.aspx>

### IBM DB2 10.5 Database

[http://www-01.ibm.com/support/knowledgecenter/SSEPGG\\_10.5.0/com.ibm.db2.luw.kc.doc/welcome.html](http://www-01.ibm.com/support/knowledgecenter/SSEPGG_10.5.0/com.ibm.db2.luw.kc.doc/welcome.html)

Note: Note: Search for DB2 Security model or Security.

### Microsoft Windows 2008 Server

<http://www.microsoft.com/download/en/details.aspx?id=17606>

### Oracle WebLogic 12c J2EE Application Server

[http://download.oracle.com/docs/cd/E12840\\_01/wls/docs103/security.html](http://download.oracle.com/docs/cd/E12840_01/wls/docs103/security.html)

