OZONE Widget Framework

Quick Start Guide

November 26, 2013

Publication/Revision History

Release	Date
Revised – OWF 7.5.0	November 26, 2013
Revised – OWF 7.4.0	October 25, 2013
Revised – OWF 7.3	October 8, 2013
Revised – OWF 7.3	July 31, 2013
Revised – OWF 7.2	June 11, 2013
Revised – OWF 7	December 21, 2012
Revised – OWF 6	September 28, 2012
Revised – OWF 5	June 29, 2012
Initial Document – OWF 4.0	March 30, 2012

Contents

1 Introduction 1
1.1 Objectives 1
1.2 Document Scope 1
1.3 Related Documents 1
2 OWF Components 3
2.1 Toolbar 3
2.2 Drop-down User Menu 3
2.3 The Store 4
2.4 Applications 5
2.4.1 My Apps Menu 5
2.4.2 Opening an Application 6
2.5 Application (App) Components 6
2.5.1 App Component Menu 7
2.5.2 Adding an App Component to an Application 7
2.5.3 Intents (Launching App Components) 7
2.6 Themes 9
2.7 Sign Out 9
3 The OWF Bundle 10
3.1 Unzipping and Starting OWF 10
3.2 Authenticating to OWF 11
3.2.1 Using a User Certificate 11
3.2.2 Installing a User Certificate 11
3.2.3 Using the Central Authentication Service (CAS) 13
3.3 Accessing OWF 14
4 Allowing Remote Access to OWF 15
4.1 Identifying a Server Name 15
4.2 Generate a Server Certificate 15
4.2.1 Install the Server Certificate 15
4.3 Modify the Externalized Configuration Files 16

5 Adding the Store or Metrics Service to OWF 17

Appendix A Contact Information A-1

A.1 Discussion Group A-1

Tables

Table 1: Tested Browsers 1

Figures

Figure 1: Toolbar 3
Figure 2: Administrator Drop-down User Menu 3
Figure 3: Store Switcher 4
Figure 4: My Apps Menu 6
Figure 5: App Components Carousel 7
Figure 6: Instructional Window for Intent 8
Figure 7: Send information to a different App Component 9
Figure 8: Drop-down User Menu 9
Figure 9: Internet Explorer User-Certification Dialog 12
Figure 10: Firefox User-Certificate Dialog 13
Figure 11: CAS Sign-in Screen 13

1 Introduction

1.1 Objectives

The purpose of this guide is to explain how to use the Ozone Widget Framework (OWF). This is including, but not limited to, the use of application components, full applications and their configuration settings. This guide provides an introduction to the Ozone Widget Framework (OWF). OWF consists of an environment and a set of tools used for discovering, organizing and displaying Web applications in a single browser window. The guide explains how to use OWF, set up an OWF environment on a user's local machine and navigate OWF security.

1.2 Document Scope

This guide is not an exhaustive reference. It is intended for users, developers and administrators seeking a quick introduction on how to deploy, start and use OWF. For information about specific areas, see the relevant documentation included with the OWF bundle.

In this document, the term **"Store" refers both Marketplace and AppsMall**. Both applications share features described in this guide and both may be configured to OWF.

The OWF Bundle ships with Tomcat 7.0.21 which requires JDK 1.6 or higher. If running OWF with a web server other than Tomcat, please see that Web server's documentation for requirements. OWF supports Internet Explorer 7 and higher and Firefox 17 and higher. OWF is tested against the following browsers:

Table 1: Tested Browsers

Browsers	Versions
Internet Explorer	7 & 9
Firefox	17
Chrome	25

1.3 Related Documents

Document	Purpose
User's Guide	Understanding the **OWF user interface**; adding, deleting, modifying **application components** and using **intents**; accessing and using

Document	Purpose
	the **Store**; creating, deleting, adding, switching, modifying **applications**; defining accessibility features such as **high-contrast themes**
Administrator's Guide	Understanding **administrative tools**: adding, deleting, and editing application components, users, groups and applications; **creating default content** for users, groups and applications
Developer's Guide	**Creating application components and integrating existing applications into OWF**; application component upgrade instructions; walkthroughs for creating application components; adding the following elements to application components: intents, descriptor URLs, preference API; logging and launching API
Configuration Guide	Overview of **basic architecture and security**; OWF **installation** instructions; instructions for modifying default settings; **database** set up and logging guidance; **framework and theme** customization instructions; OWF **upgrade instructions**, directions for adding and deleting **help** content
Quick Start Guide	Walkthrough of **basic OWF functions** such as using application components and applications; instructions for **setting up a local instance of OWF**, unpacking the OWF Bundle and installing **security certificates**; Truststore/Keystore changes

2 OWF Components

This is an overview explaining how to start using the Ozone Widget Framework. Find detailed information in the OWF User's Guide.

2.1 Toolbar

Figure 1: Toolbar

My Apps: Used to find, start and manage applications.

App Components: Used to add application components to applications.

The Store: Search and add applications and app components from associated Stores. (If OWF is not configured for the Store, the icon will not appear.)

Online Help: Repository of OWF guides and tutorials.

2.2 Drop-down User Menu

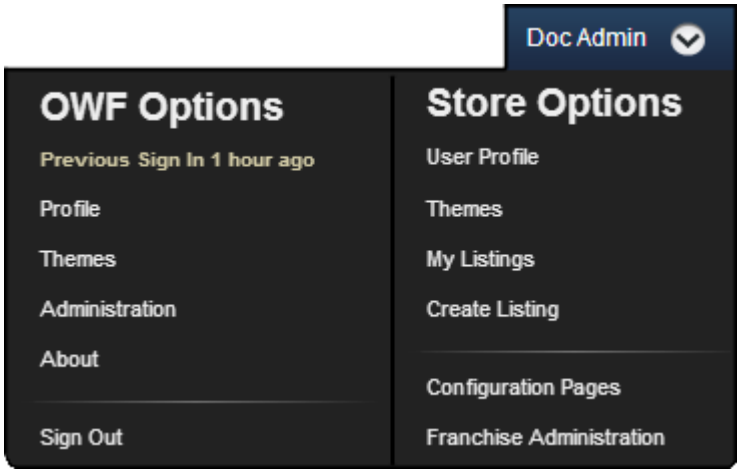

Figure 2: Administrator Drop-down User Menu

Metrics: Launch the Metrics Service. (If not configured for Metrics, the toolbar option will not appear.)

Administration – Use it to manage applications, application component, users, groups and system configurations. The Configuration tool allows administrators to change select OWF configuration settings from within the user interface. (Drop-down option will only appear to those with Administrator privileges.)

2.3 The Store

In this section, the term **"Store" refers both Marketplace and AppsMall**. Both share features described in this guide and can be configured to OWF.

The Store, similar to a commercial application store, operates as a thin-client registry of applications and services. The Store provides search and discovery functionality that enables OWF users to find, add and share useful tools including but not limited to applications, app components and Web Apps.

Provided OWF has been configured to recognize an instance (or instances) of the Store, clicking 

on the toolbar opens the Store Switcher and connects users to multiple Stores. Unlike application listed in the My Apps Menu, the Store is only accessible through the Store Switcher.

Figure 3: Store Switcher

Opening a single Store:

In the Store Switcher, a user must click on a Store to open it. If only one Store is available, it will open automatically when 

is clicked. To close the Store and return to the previous application, click 

on the toolbar.

2.4 Applications

In simple terms, an application is a screen where a user can dictate (for the most part) which application components to load, which layouts to use and the arrangement of the app components within the specified layouts.

Each time a saved application loads, the screen and app component layout maintain the same look and feel as the last time the application was accessed by the user. Users can receive applications by the following methods:

- Create their own
- Add from the Store
- Assign to individual user by an administrator
- Assign to a group by an administrator

Group assigned applications provide identical applications for each member of a group. Each group member can customize their instance of a pre-configured application. Applications that have not been created by the user can be restored to their default states.

2.4.1 My Apps Menu

The My Apps Menu lists all of the user's applications. Applications included here are OZONE Apps, either created in OWF or obtained from the Store, and Web apps. From the My Apps Menu, users can start an application, create new ones or search for applications in the Store. Applications can be shared, restored, edited or deleted through the manager buttons in the My Apps Menu.

To open the My Apps menu, click **My Apps** in the OZONE Toolbar.

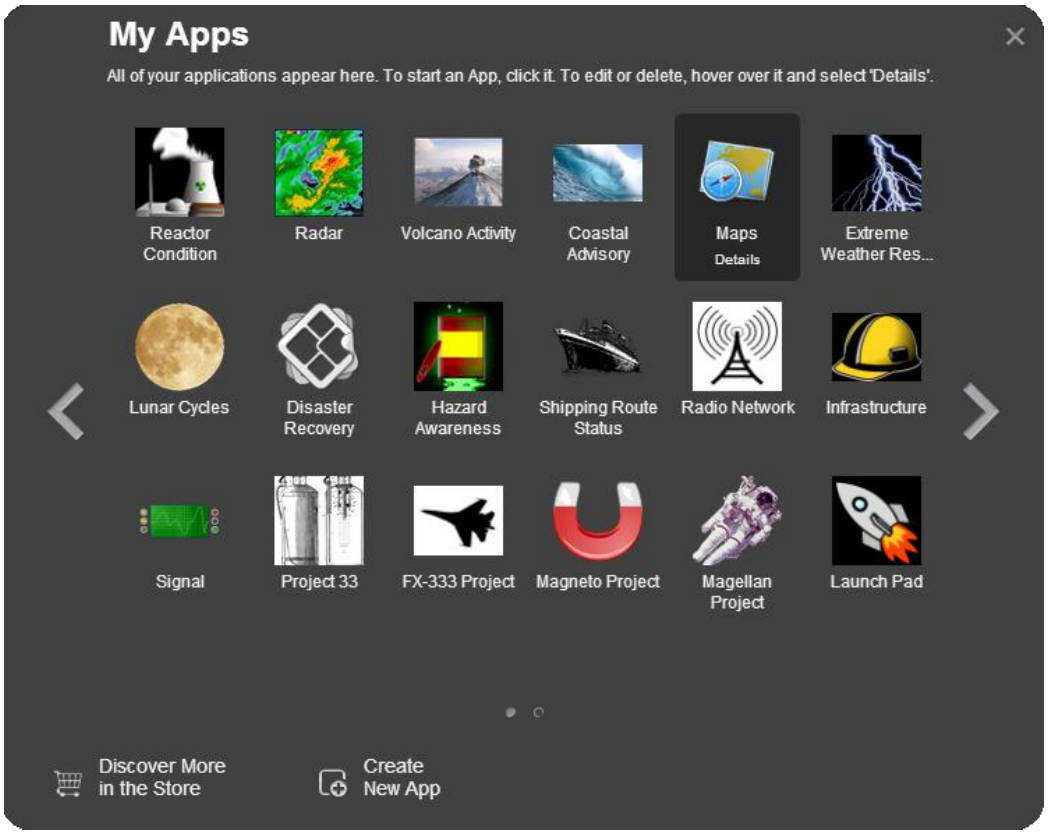

Figure 4: My Apps Menu

2.4.2 Opening an Application

Applications can contain one or more layouts called Pages. If an application has only one page, then clicking its icon in the My Apps Menu will start it.

To open an application:

1) Click the **My Apps** button in the toolbar to open the My Apps Menu.
2) Click an application, it will automatically open.

If the application has more than one page, then clicking the application's icon will open a carousel displaying all the application's pages.

2.5 Application (App) Components

An application (app) component is a lightweight, single-purpose application that offers a summary or limited view of a larger application. In OWF, an application component is a global description for a piece of Web content that can be configured by the user and displayed within an application.

2.5.1 App Component Menu

Users can access of their app components from the App Components Menu by clicking the App Components button in the toolbar. Once open, the App Components Menu appears as a carousel.

Figure 5: App Components Carousel

2.5.2 Adding an App Component to an Application

To start an app component in an application:

1) Open the App Components Menu by clicking the App Components button in the toolbar.
2) Start an app component using one of the following methods:
    a) Click the app component.
    b) Drag the app component from the App Components Menu into the current application.
3) The App Components Menu disappears revealing the current, open application. If the application is a Fit layout, click or drop the app component in the location where it should open. Otherwise, select the pane to place the app component; the app component will open in the highlighted pane.
4) The app component will automatically start in the selected pane and the App Components Menu will reappear.
5) Repeat this action to open another app component.
6) When finished, close the App Components Menu by clicking the X in the upper-right corner.

2.5.3 Intents (Launching App Components)

Intents are the instructions for carrying out an app component's intentions. One app component requests an action (think of actions as verbs like view, share, edit, etc.), then

another app component receives that request and performs the action. Intents build on OWF's publish/subscribe feature by allowing users to choose the app component(s) that will use data. This binding capability enables two app components to share data in a way that improves their function.

For example, the NYSE app component charts data about the stock exchange. Some users may want to view that data as a Web page. This is possible if the NYSE app component has an intent that tells it to send data to app components that display data in a Web format.

Note: App Component s may have multiple intents associated with them. Users cannot create app component intents. Administrators and developers (logged in as administrators) add app component intents through the OWF interface. Developers also add the intents through app component descriptor URLs. OWF follows standard Web Intent specifications documented at Webintents.org.

How to use intents:

When an app component sends an intent request, a pop up window appears displaying all of the open app components that can receive the requested intent action and data for an intended purpose (graphing, displaying, etc.).

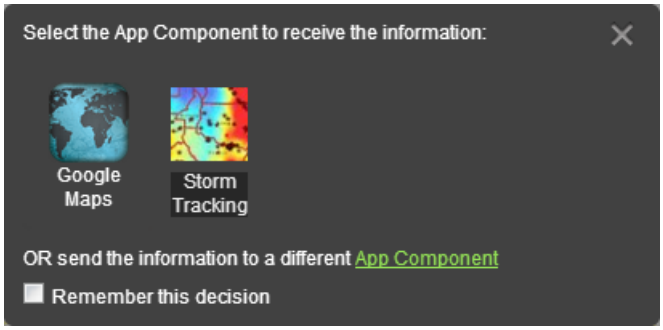

Figure 6: Instructional Window for Intent

Select an app component to accept the requested intent:

- Click one of the app components displayed on the window OR
- OR click the App Component link to send the information to an App Component that is not open on the screen:

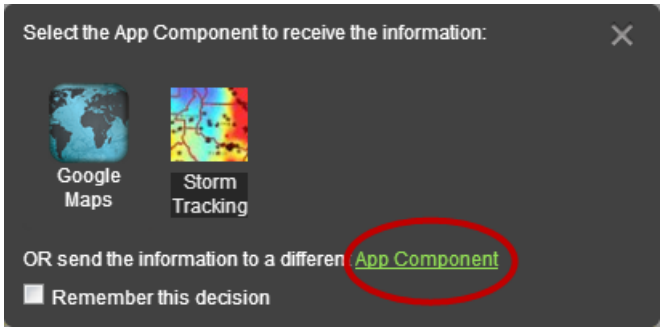

Figure 7: Send information to a different App Component

Note: Checking the "Remember" box will allow the selected app component to automatically open the requesting app component's data. This function will continue until the user breaks the connection by closing either the sending app component or the receiving app component.

After a user selects a receiving app component, the intent data is automatically sent to and processed by the receiving app component. To place the app component on the App, click or drag it from the menu into the application. If every pane in the application is occupied by other app components, then selecting a pane to place this app component will replace the app component currently there. Once the receiving app component is in place and open in the application, it will receive the sending app components intent request.

2.6 Themes

OWF provides a default theme and two high contrast themes for accessibility. To select a theme:

1) Open the drop-down User Menu in the toolbar, and then click the Themes button.

   The Theme Settings window will open. Theme options will appear in the left column. The current theme will display in the right column.

2) Select a theme in the left column, then click OK. The theme will automatically change to the new selection.

2.7 Sign Out

To sign out of OWF: Mouse over the drop-down User Menu in the toolbar and select Sign Out.

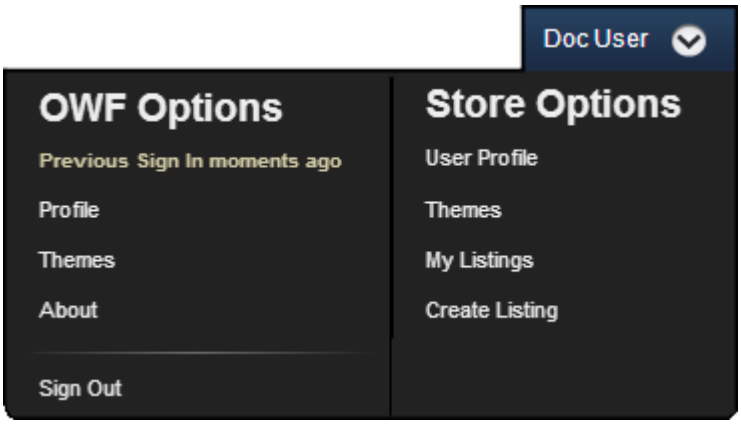

Figure 8: Drop-down User Menu

3 The OWF Bundle

OWF is normally distributed as **OWF-bundle-7-GA.zip**. This bundle contains the three main components needed to deploy, run and develop widgets for the framework. They are as follows:

- A Tomcat Web server which runs the supplied **WAR** files.
- An **owf.war** file which contains both **HTML** and **JavaScript** files. These files make up the OWF user interface and the server backend which handles persistence of the user, user preferences, widget definitions and dashboards.
- An optional file **cas.war** implements the Central Authentication Service (CAS). If CAS security is not required, **cas.war** can be removed. To run OWF in a local (development) environment, execute the following steps:

Note: The following is a summary. Expanded details begin in the section, ***Unzipping and Starting OWF****.*

1. Unzip **OWF-bundle-7-GA.zip**.
2. From a command-line, run start.sh or start.bat from within the **apache-tomcat-7.0.21\ directory**.
3. In a supported browser, navigate to **https://localhost:8443/owf**.
4. Authenticate access to OWF by entering username “testAdmin1” and password “password.” Alternatively, install the testAdmin1.p12 (see Installing a User Certificate for details) certificate from **apache-tomcat-7.0.21\certs** into the browser as a certificate.

3.1 Unzipping and Starting OWF

A user can start the OWF server by unpacking the **OWF-bundle-7-GA.zip** into the directory from where it will be run, and activating the start script. This requires the use of a ZIP utility. Detailed steps for each operation are shown below.

The following is an example of how to copy, unzip and launch OWF in *nix

```
mkdir /opt/OWF
cp OWF-bundle-7-GA.zip/opt/OWF
cd /opt/OWF
unzip OWF-bundle-7-GA.zip
cd apache-tomcat-7.0.21
./start.sh
```

The following example shows how an administrator might unpack OWF from the bundle on Windows operating systems:

- Right-click on the **OWF-bundle-7-GA.zip**. Using a **ZIP** utility, select "Extract to [folder name]." If the **ZIP** utility does not support this feature, manually create a sub-directory and extract the **ZIP** contents into that directory.

The use of the bundled deployment archive provides all of the necessary mechanisms to deploy and run the Tomcat Web container on any JDK 1.6 enabled system.

3.2 Authenticating to OWF

Users must be authenticated by the system before they can access OWF services. This can be achieved by installing a certificate into the user's Web browser or via authentication with the Central Authentication Service (CAS).

3.2.1 Using a User Certificate

In order to take advantage of the X.509 PKI user authentication mechanism, the **testUser1** or **testAdmin1** certificate for localhost must be installed in a user's browser. These certificates can be found under the **apache-tomcat-7.0.21\certs** directory where **OWF-bundle-7-GA.zip** was deployed. See screenshots on the pages that follow for general instructions on importing certificates.

*Note: The password for the user certificate is "**password**."*

3.2.2 Installing a User Certificate

When using Internet Explorer, navigate to Tools → Internet Options → Content → Certificates → Personal. Click the Import button and navigate to the **apache-tomcat-7.0.21\certs** directory where **OWF-bundle-7-GA.zip** was deployed. Select the testUser1 certificate and click OK. Click Next and enter "password" as the password when prompted. Click Finish. A dialog box should display, stating that the import was successful.

Note: The default dialog filter for a particular browser may be set for CER or CRT files. If this is the case, the drop-down for file type must be set for .p12. Once .p12, is selected, the certificate will show up as being available for importing.

Note: Some Intranets may require additional customization.

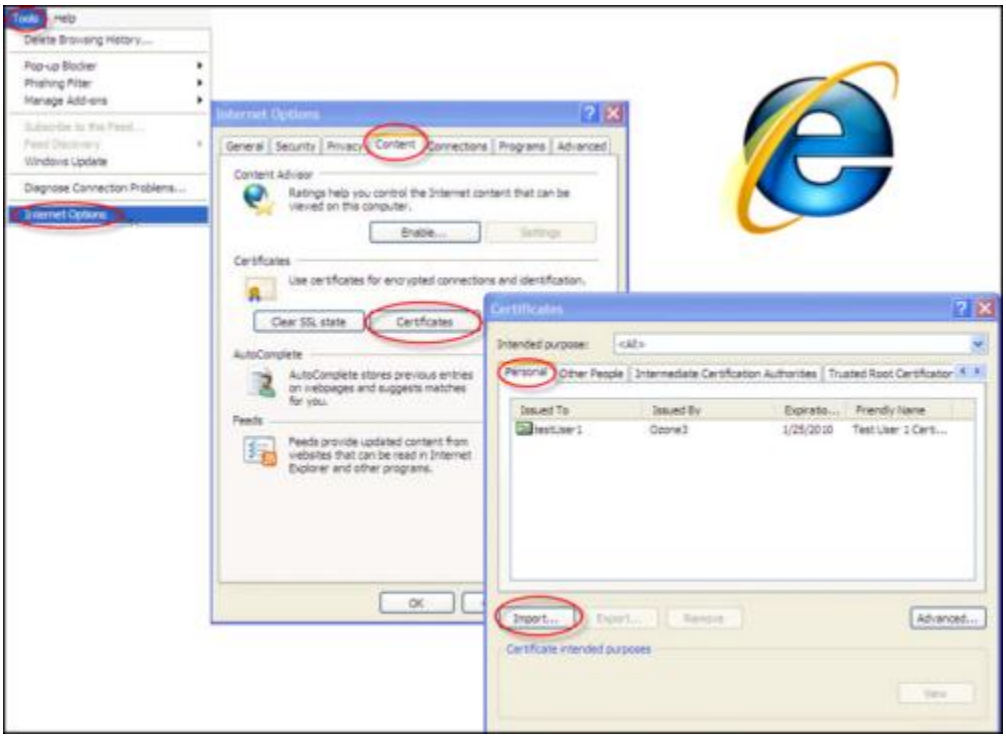

Figure 9: Internet Explorer User-Certification Dialog

Note: In certain versions of Internet Explorer, certificate/connection failures are shown, despite a successful import of the testUser1 certificate. If this is the case, follow the directions above and select the X.509 from the dropdown and import the certificate titled ca.crt.

When using Firefox, navigate to Tools->Options->Advanced->Encryption->View Certificate->Your Certificates. Click the import button and navigate to the **apache-tomcat-7.0.21\certs** directory where **OWF-bundle-7-GA.zip** was deployed. Select the **testUser1** certificate, click OK and enter password as the password when prompted. Click Finish. A dialog box should display, stating that the import was successful.

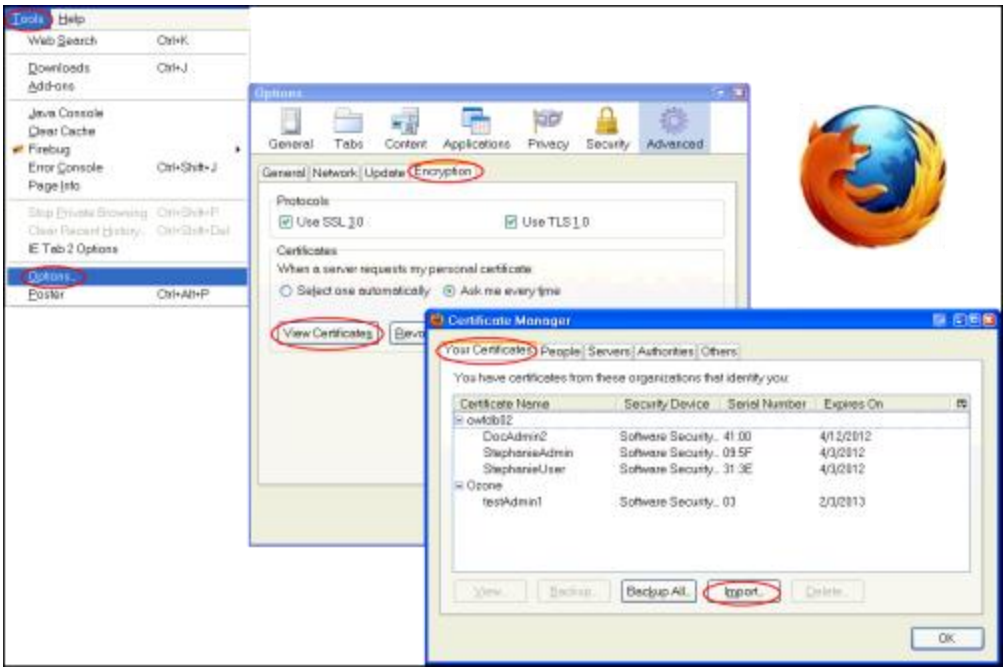

Figure 10: Firefox User-Certificate Dialog

3.2.3 Using the Central Authentication Service (CAS)

The CAS authentication server is located at **https://localhost:8443/cas**. It can be accessed by preventing the Web browser from transmitting a user certificate to the OWF server. (This can be achieved by clicking the cancel button when asked for certificate authentication.)

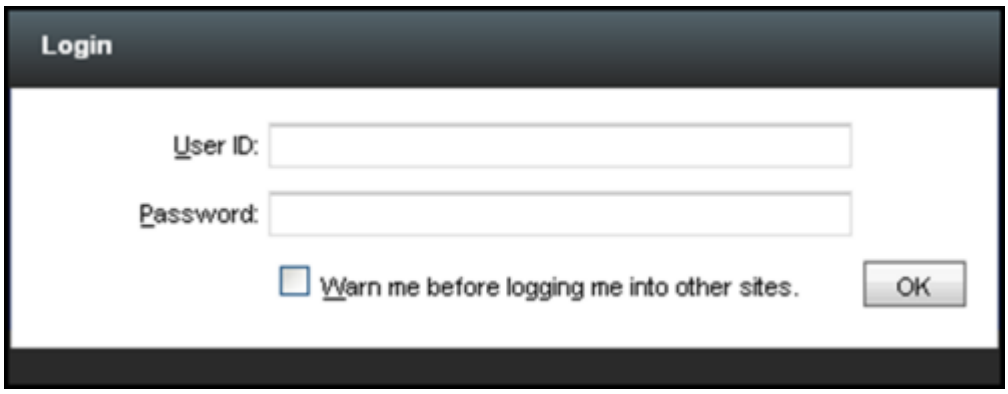

Figure 11: CAS Sign-in Screen

By default, the bundle is configured so that CAS will recognize the username **testUser1** or **testAdmin1** with the password of "password." Once logged in via CAS, the browser will redirect the user to OWF.

3.3 Accessing OWF

The Tomcat instance, located in the directory where **OWF-bundle-7-GA.zip** was deployed, must be started (**start.bat** or **start.bat**) before OWF will become accessible. When the Tomcat server finishes initializing, OWF can be launched by navigating to **https://servername:port/owf/**. In the default OWF installation, the URL is **https://localhost:8443/owf**. To modify the default location, see section **4 Allowing Remote Access to OWF**.

Note: OWF is configured to redirect to a secure connection using https. Should a non-secure connection be attempted via http, the user may need to grant a browser level security exception depending on the browser's security settings.

The first time that **testUser1** accesses OWF using the aforementioned methods, they will be presented with a dashboard containing a few sample widgets. The user can then immediately configure and organize widgets or change to a different dashboard.

4 Allowing Remote Access to OWF

To run OWF remotely, and NOT from a localhost environment, execute the following steps:

1. Identify a server name.
2. Generate a server certificate.
3. Install the server certificate.
4. Modify configuration files.

4.1 Identifying a Server Name

The server name can be chosen arbitrarily and entered into the users' HOST files, or it can be obtained from DNS. This quick start guide will refer to the selected server name as **servername** and to OWF as **https://servername:port/owf/**.

4.2 Generate a Server Certificate

The certificates that ship with OWF are configured with a domain (**servername**) of **localhost**. If the domain name is changed, new certificates are required. The server certificate must reflect the **servername**.

Navigate to the **\etc\tools** folder and execute **create-certificates.bat** or **.sh**, depending on the operating system in use. Once this is done, the default user p12 certificates (**testUser1** and **testAdmin1**) will no longer be compatible. To correct this, create new user certificates using **create-certificates.bat** (or **.sh**).

Follow the prompts on screen and create the necessary certificates for the installation.

4.2.1 Install the Server Certificate

The OWF start script, located at **apache-tomcat-7.0.21\bin\setenv.bat** (**apache-tomcat-7.0.21\bin\setenv.sh** on ***nix** systems) must be edited to point to the new keystore (defined while answering the prompts discussed in section **2 Generate a Server Certificate**) file found in **setenv.bat/setenv.sh**. Edit the `servername` domain (found in lines 1 and 2 in the code below) to reflect the certificate.

```
    1 set CATALINA_OPTS=-
Djavax.net.ssl.trustStore="%CATALINA_HOME%\certs\servername.jks" -
    2 Djavax.net.ssl.keyStore="%CATALINA_HOME%\certs\servername.jks"
-
    3 Djavax.net.ssl.keyStorePassword=changeit -
Djavax.net.ssl.trustStorePassword=changeit server -
```

```
    4 Xmx1024m -Xms512m -XX:PermSize=128m -XX:MaxPermSize=256m
%JAVA_OPTS%
```

The Tomcat configuration file, located at **apache-tomcat-7.0.21\conf\server.xml**, must also be edited to point to the new Keystore file. This section can be found below the "Define a SSL…" section of the **XML** file:

```
      <Connector port="8443" protocol="HTTP/1.1" SSLEnabled="true"
                  maxThreads="150" scheme="https" secure="true"
                  keystoreFile="certs/servername.jks"
keystorePass="changeit"
        truststoreFile="certs/servername.jks"
truststorePass="changeit"
                  clientAuth="want" sslProtocol="TLS" />
```

4.3 Modify the Externalized Configuration Files

In order to access OWF from remote computers, externalized configuration files must point to the correct location. This is done by changing a properties file that is referenced by the following two configuration files:

• apache-tomcat-7.0.21\lib\OwfConfig.groovy • etc\override\CASSpringOverrideConfig.xml

1. By default, the configuration files allow access from localhost but not from other locations. To access other locations:
2. Copy **CASSpringOverrideConfig.xml** from the **etc\override to apache-tomcat-7.0.21\lib**. By default, **OwfConfig.groovy** is located on the classpath. Therefore, it does not need to be moved.
3. In the **apache-tomcat-7.0.21\lib\OzoneConfig.properties** file, replace **localhost** with **servername** for the **ozone.host** property.
4. Restart the server.

5 Adding the Store or Metrics Service to OWF

OWF is flexible and scalable, which allows the Store or the Metrics Service to be included in OWF's deployment. This allows a user to develop with the products working together, without having to activate multiple ports via configuration. To include the Store or Metrics Service in the OWF Bundle, do the following:

1. Unpack the zipped bundles containing the applications to be included. Navigate to **apache-tomcat-7.0.21/webapps** in each unpacked bundle.
2. Copy the appropriate **WAR** files into the **apache-tomcat-7.0.21/webapps** directory where OWF was deployed.
3. Restart the OWF server.

Note: If using a Marketplace release earlier than 5.0, see the OZONE Store Configuration Guide regarding additional setup requirements.

Appendix A Contact Information

A.1 Discussion Group

For information about OZONE or access to its resources, please open a ticket regarding the AppsMall Service at http://www.intelink.ic.gov/ticket/secure/CreateIssue!default.jspa and then, email the team at AppsMall@intelink.gov.