OWF Quick Start Guide

DOD GOSS

Exported on Oct 16, 2018

Table of Contents

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 terms Store and Marketplace are used interchangeably. Both applications share features described in this guide and both may be configured to OWF.

The OWF Bundle ships with Tomcat v8.5.23 which requires JDK 1.7 or higher. If running OWF with a web server other than Tomcat, please see that Web server's documentation for requirements.

1.3 Supported Browsers

OWF is tested against the following browsers:

Table 1: Tested Browsers

Browser	Versions
Internet Explorer	11
Firefox	57
Chrome	43
Edge	38

1.4 Related Documents

Document	**Purpose**
User's Guide	Understanding the **OWF user interface**; adding, deleting, modifying **app components** and using **intents**; accessing and using the **Store**; creating, deleting, adding, switching, modifying **app pages**; using **applications**; defining accessibility features such as **high-contrast themes**

Document	Purpose
Administrator's Guide	Understanding **administrative tools**: adding, deleting, and editing app components, users, groups, applications; **creating default content** for users, groups and group dashboards
Developer's Guide	**Creating app components and descriptor files; integrating app components into OWF**; app component upgrade instructions; walkthroughs for creating app components; adding intents, descriptor URLs, preference API to app components; 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 applications; instructions for **setting up a local instance of OWF**, unpacking the OWF Bundle and installing **security certificates**; Truststore/Keystor

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

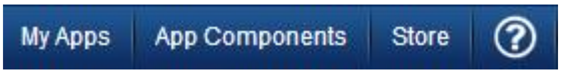


Figure 1: Toolbar

My Apps	**My Apps**: Used to find, start and manage applications.
App Components	**App Components**: Used to add application components to applications.
Store	**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

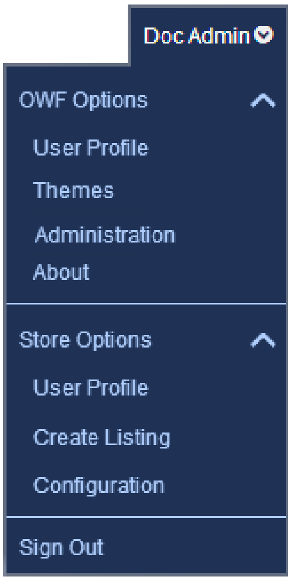


Figure 2: Administrator Drop-down User Menu

Metrics: Launch the Metrics Service. (If not configured for Metrics, the drop-down 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 and if OWF is running independent of the Store. If OWF is running with a Store, the Administrative functions will appear an App. Click My Apps to open it.)

2.3 The Store

In this section, the term Store and Marketplace are used interchangeably. 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 Store on the toolbar opens the Store Switcher and connects users to multiple Stores. Unlike applications listed in the My Apps Menu, the Store is only accessible through the Store Switcher.

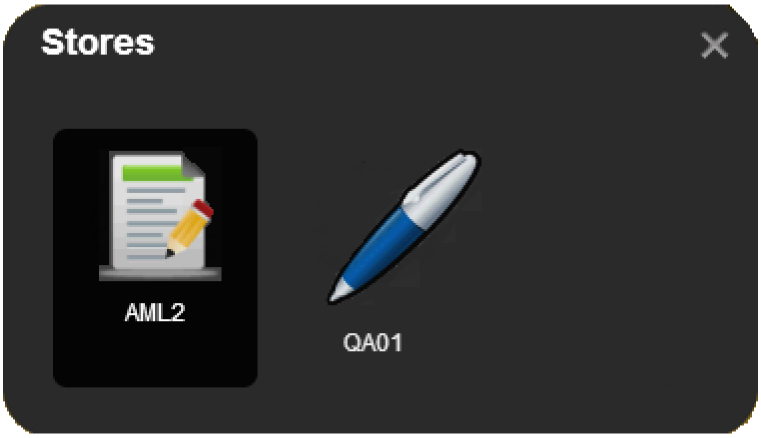


Figure 3: Store Switcher

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

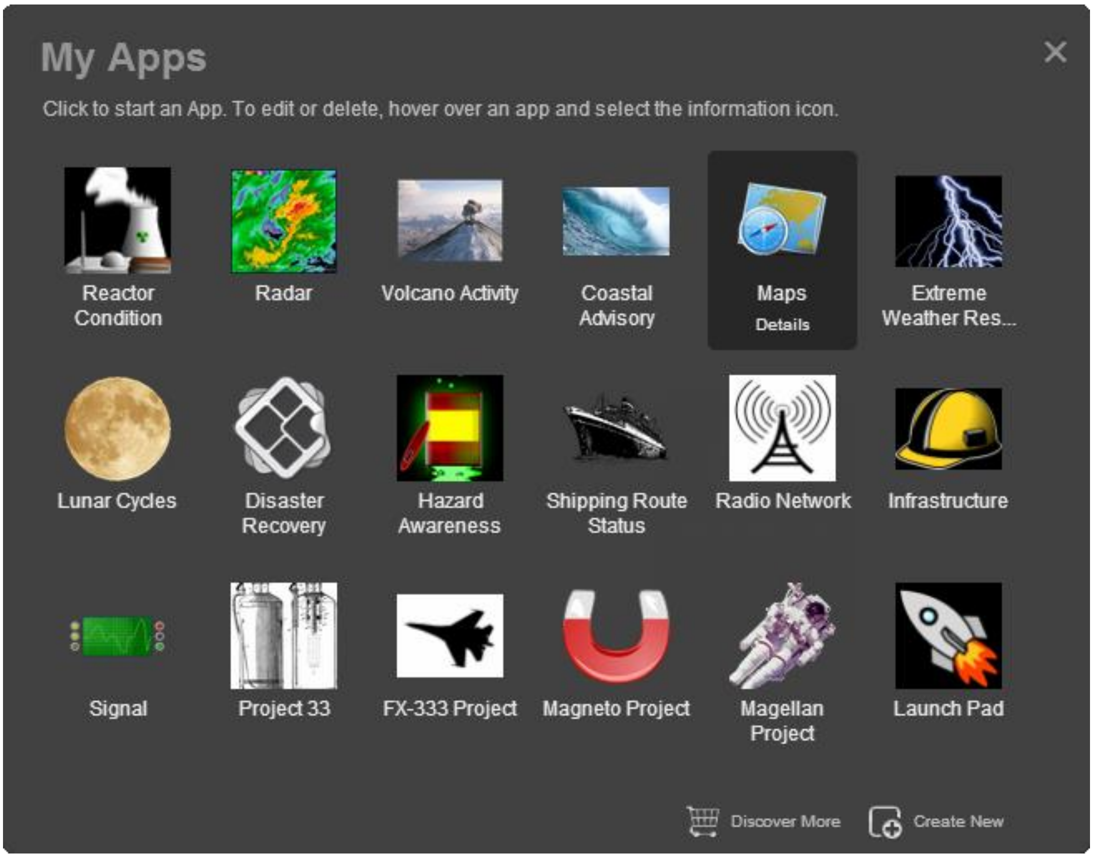
- Create their own
- Add from the Store
- Assign to individual user by an administrator
- Assign to a group by an administrator

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.



My Apps

Click to start an App. To edit or delete, hover over an app and select the information icon.

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 their app components from the App Components Menu by clicking the App Components button in the toolbar. Once open, the App Components Menu appears at the top of the screen.

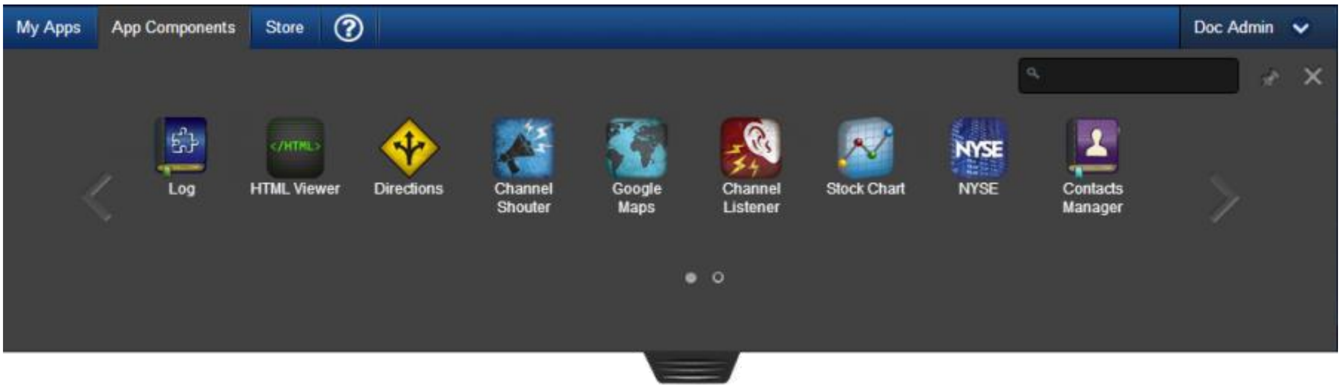


Figure 5: App Components Menu

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.

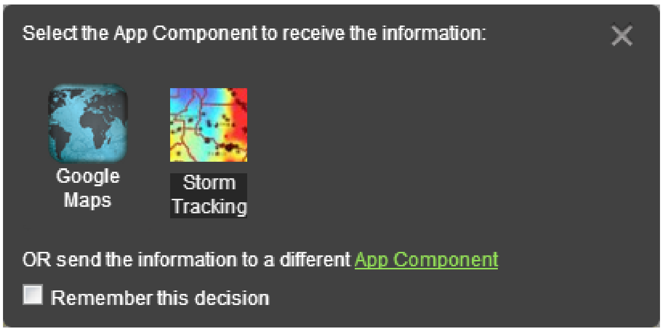
This binding capability enables two app components to share data in a way that improves their function.

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.).


Select the App Component to receive the information:

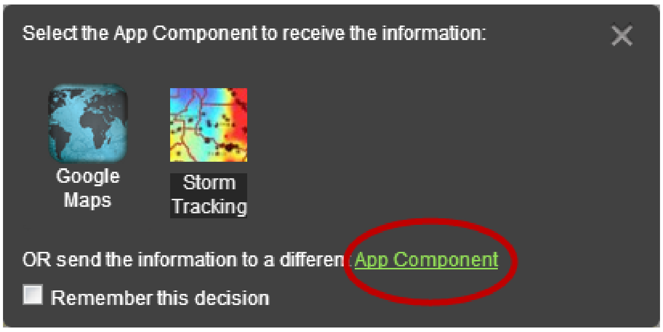
OR send the information to a different App Component

□ Remember this decision

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:


Select the App Component to receive the information:

OR send the information to a differen App Component

□ Remember this decision

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:

1. Click the drop-down User Menu in the toolbar and select Sign Out.

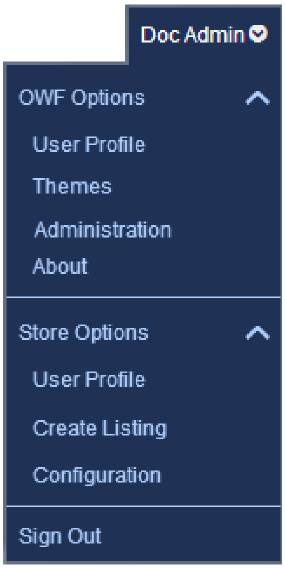


Figure 8: Drop-down User Menu

3 The OWF Bundle

3.1 Introduction

OWF is normally distributed as a zipped bundle. This bundle contains the two 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 file.
- An owf.war file which contains the OWF application. It includes the back-end server, as well as the front-end HTML and JavaScript files.

3.2 Prerequisites

3.2.1 Java Runtime Environment (JRE)

Running the OWF bundle requires that the Java Runtime Environment (JRE) be installed on the host machine.

OWF has been developed and fully tested using Java 8. Versions prior to or later than Java 8 are not currently supported.

The full instructions for the installation of the JRE is beyond the scope of this document. However, installation media and instructions may be obtained from the Oracle Java download page at the following location:

- http://www.oracle.com/technetwork/java/javase/downloads/index.html

3.3 Instructions

3.3.1 Overview

Note: The following is a summary. Please refer to the sections below for extended details.

1. Unzip the zipped OWF bundle.
2. From a command-line, run start.sh or start.bat from within the `tomcat` 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 the `tomcat\certs` folder into the browser as a certificate.

3.3.2 Unzipping the Bundle

A user can start the OWF server by unpacking the OWF bundle 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.

Linux example:

```
$ mkdir /opt/owf-framework
$ cp ozone-framework-7.17.2.0.zip /opt/owf
$ cd /opt/owf
$ unzip ozone-framework-7.17.2.0.zip
```

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

- Right-click on the OWF bundle. 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.

3.3.3 Install Patch

IMPORTANT:The patch MUST be applied using a pre-existing OWF v7.17.1.0 .WAR file.

1. Download and extract the v7.17.1 bundle. https://github.com/ozoneplatform/owf-framework/releases/download/v7.17.1/OWF-bundle-7.17.1.zip
2. Copy the v7.17.1 .WAR file into the new bundle root directory:

   ```
   C:\> copy C:\owf-bundle-7.17.1\apache-tomcat\webapps\owf.war
   C:\owf-bundle
   ```

3. From the root directory of the v7.17.2.0 bundle, run the patch to do an in-place update of the new `owf.war` file.

   ```
   C:\owf-bundle> java -jar owf-patch.jar owf.war
   tomcat\webapps\owf.war
   ```

4. Verify that the patch has been successfully applied.
   The Tomcat `webapps` folder (example: `C:\owf-bundle\tomcat\webapps`) should contain the following files:
   1. `owf.war` – the patched WAR file
   2. `owf.war.orig` – a copy of the original unpatched WAR file

3.3.4 Starting the Bundle

Scripts to start the server are included in the bundle in the `tomcat` directory.

The server may be started in **development mode** to utilize the embedded, in-memory H2 database. The initial data is generated by the application.

To run the server in **production mode**, the database must be pre-populated with the initial data using the supplied database scripts. Please refer to the **OWF Configuration Guide** for instructions on configuring the database and running the initial data scripts.

3.3.4.1 Windows start script

Example:

```
C:\owf-bundle\tomcat> start.bat /dev
```

Usage:

```
start.bat [/dev] [/db database]
  /dev       Start in DEVELOPMENT mode
  /db        Use the selected database configuration
  database     h2     - Embedded H2 file-based database (default)
               pg     - PostgreSQL
               mysql  - MySQL
               oracle - Oracle RDBMS
               mssql  - Microsoft SQL Server
```

3.3.4.2 Linux start script

Example:

```
/opt/owf-bundle/tomcat$ ./start.sh --dev
```

Usage:

```
./start.sh [--dev] [--db database]
  --dev      Start in DEVELOPMENT mode
  --db       Use the selected database configuration
  database      h2     - Embedded H2 file-based database (default)
                pg     - PostgreSQL
                mysql  - MySQL
                oracle - Oracle RDBMS
                mssql  - Microsoft SQL Server
```

3.3.5 Accessing OWF

After the Tomcat server has finished the initialization process, OWF can be accessed by opening a web browser and navigating to the following URL:

- https://localhost:8443/owf/

To modify this default location, see the section: **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.

3.3.6 Authenticating to OWF

In **development mode**, no login is required and a default profile will be used.

In **production mode**, users must be authenticated by the system before they can access OWF services.

In the default installation, the user may authenticate by using the default login form or by installing a certificate into their Web browser.

3.3.6.1 Using the Login Form

To login as the default administrator, use the following credentials at the login form:

- username: `testAdmin1` / password: `password`

To login as a default unprivileged user, use the following credentials at the login form:

- username: `testUser1` / password: `password`

3.3.6.2 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 `/tomcat/certs` directory where the OWF bundle was deployed. See screenshots on the pages that follow for general instructions on importing certificates.

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

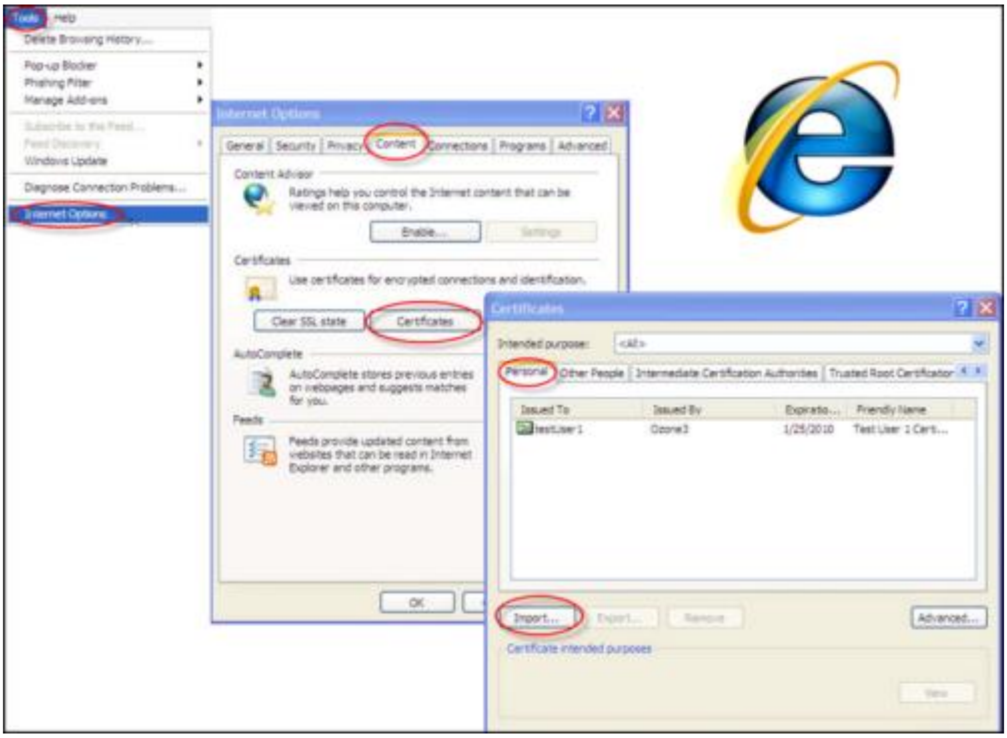
3.3.6.2.1 Installing a User Certificate using Internet Explorer (IE)

1. Navigate to Tools (Some versions use a gear icon ⚙) → Internet Options → Content → Certificates → Personal.
2. Click the Import button and navigate to the `/tomcat/certs` directory where the OWF bundle was deployed.
3. Select the **testUser1** certificate and click OK.
4. Click Next and enter `password` as the password when prompted.
5. Select a folder to house the certificate.
6. Click Finish.
7. A dialog box should display, stating that the import was successful.

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 drop-down and import the certificate titled ca.crt.

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.



Ratings help you control the Internet content that can be viewed on this computer.

Use certificates for encrypted connections and identification.

AutoComplete stores previous entries on webpages and suggests matches for you.

Feeds provide updated content from websites that can be read in Internet Explorer and other programs.

Intended purpose: <All>

Issued To	Issued By	Expiratio...	Friendly Name
testUser1	Ozone3	1/25/2010	Test User 1 Cert...

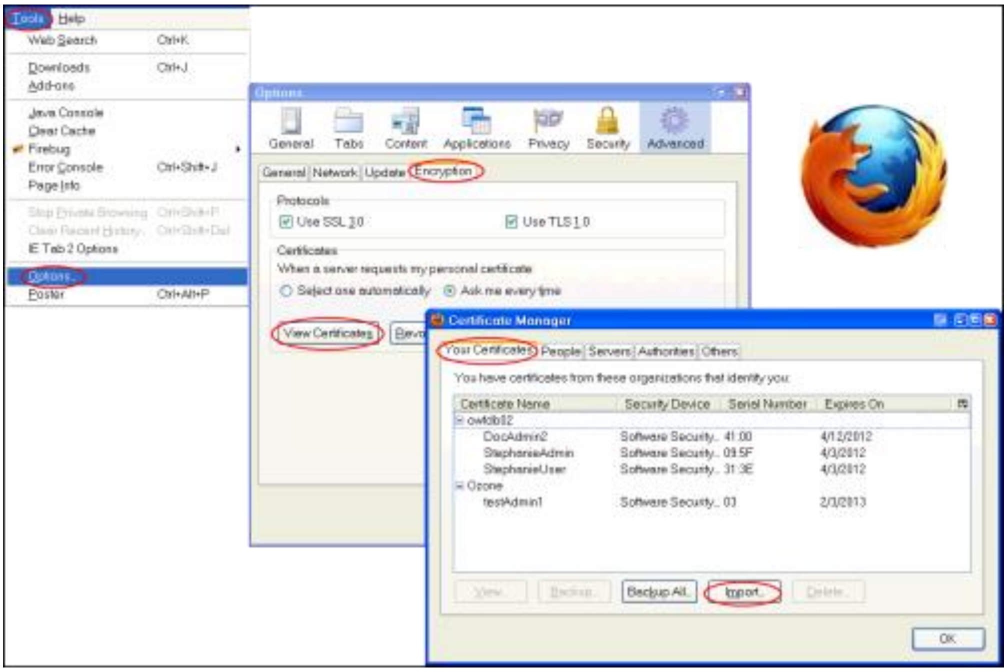
Figure 9: Internet Explorer User-Certification Dialog

3.3.6.2.2 Installing a User Certificate using Firefox

1. Navigate to Tools → Options → Advanced → Encryption → View Certificate → Your Certificates.
   (Newer versions:

   ☰

   → Options → Advanced → Certificates → View Certifications → Your Certificates.)
2. Click the import button and navigate to the `/tomcat/certs` directory where the OWF bundle was deployed.
3. Select the **testUser1** certificate, click OK.
4. Enter `password` as the password when prompted.
5. Click Finish.
6. A dialog box should display, stating that the import was successful.



You have certificates from these organizations that identify you:

Certificate Name	Security Device	Serial Number	Expires On
⊟ owfdb02			
DocAdmin2	Software Security...	41:00	4/12/2012
StephanieAdmin	Software Security...	09:5F	4/3/2012
StephanieUser	Software Security...	31:3E	4/3/2012
⊟ Ozone			
testAdmin1	Software Security...	03	2/3/2013

Figure 10: Firefox User-Certificate Dialog

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 **create-certificates.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 **tomcat\bin\setenv.bat** (**tomcat\bin\setenv.sh** on ***nix** systems) must be edited to point to the new keystore (defined while answering the prompts discussed in the section: **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 `tomcat\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"
           scheme="https"
           secure="true"
           SSLEnabled="true"
           maxThreads="150"
           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 configuration file **tomcat\lib\ozone\framework\application.properties**. By default, the configuration files allow access from localhost but not from other locations.

To access other locations:

1. In the **tomcat\lib\ozone\framework\application.properties** file, replace **localhost** with **servername** for the **ozone.host** property.
2. Restart the server.