STC ONE

VOMS

Installation Guide

v. March 2018

Support Services

For general support on this product, contact your system administrator or help desk. For up-to-date documentation, visit the STC Documentation Portal at https://documentation.stchome.com/.

Connect with Us on Social Media

Copyrights and Trademarks

This documentation describes the following: VOMS (v. March 2018) installation

This document, along with any embedded media, is provided for informational purposes only. Scientific Technologies Corporation (STC) provides this material "as is" and does not assert that this document is error free. The information contained herein may not necessarily accurately represent the current or planned functions of the application, and may be subject to significant and frequent modification. Scientific Technologies Corporation (STC) may change this document, any embedded content, and the product described herein at any time. Any changes will be incorporated in new versions of this document.

Scientific Technologies Corporation (STC) recognizes the rights of the holders of all trademarks used in its publications.

This document may provide hyperlinks to third-party websites or access to third-party content. Links and access to third-party sites are provided for your convenience only. Scientific Technologies Corporation does not control, endorse, or guarantee third-party content and is not responsible for any content, associated links, resources or services associated with a third-party site. Scientific Technologies Corporation shall also not be liable for any loss or damage associated with your use of any third-party content. (20180508)

Table of Contents

VOMS Installation Scenarios

The following provides an overview of the process for installing the VOMS application.

Components

VOMS is composed of several server components. The following includes their functions and brief technical descriptions.

JasperReports Server (JRS)

JRS is a web application that provides the reports for VOMS. These reports are imported into JRS via the JRS Management Console. See VOMS JasperReports Deployment - Linux or Windows for installation instructions.

VOMS Application

The UI and application server logic are provided via the VOMS application. This is a Node.js application and uses Keycloak for user authentication.

Keycloak

User authentication is provided by Keycloak. It is used by all STC Suite applications for Single Sign-On (SSO). Each application requires a client within Keycloak. The client contains references (URLs, ports, etc.) that are specific to each individual VOMS application.

Release Distribution Files

The release distribution files are located as follows:

- **Linux** - `/Application Components/VOMS/voms-2.17.5-linux-x64.zip`
- **Windows** - `\Application Components\VOMS\voms-2.17.5-windows.zip`

New Application vs. Upgrade Installations

The following refers to new application installations (whether installing VOMS into an existing STC Suite, or installing both VOMS and the rest of the STC Suite for the first time) and upgrades as defined in the *New or Upgrade* column. The instructions differ between Linux and Windows installations as well. If you are building a new Linux Server, see the Build a Linux (CentOS) Server instructions.

NOTE: The VOMS components should be installed in the order listed below.

Component	New or Upgrade	Server	Linux	Windows
Keycloak Installation (if not already installed)	New	Service (SVC)	See either *Keycloak Installation Guide for STC-Hosted Clients* or *Keycloak Installation Guide for Self-Hosted Clients*, located on the STC Documentation Portal.	
JasperReports Server (if not already installed)	New	Application (APP)	JasperReports Server Installation - Linux	JasperReports Server Installation - Windows
VOMS Reports	New	Application (APP)	VOMS JasperReports Deployment - Linux	VOMS JasperReports Deployment - Windows
Node.js Server (if not already installed)	New	Application (APP)	Node.js Installation - Linux	Node.js Installation - Windows
VOMS Application Installation	New	Application (APP)	VOMS Application Installation Instructions - Linux	VOMS Application Installation Instructions - Windows
Keycloak Configuration for VOMS	New, Upgrade	Service (SVC)	Keycloak Configuration for VOMS	

Linux Instructions

The following instructions are for the Linux operating system.

Build a Linux (CentOs) Server

The instructions below explain how to install CentOS 7 Minimal and configure it. This is a base configuration suitable for reuse for all servers. An image or snapshot should be taken, if possible (such as in a cloud or AWS environment), so that these steps do not need to be performed again. Initially, the server is usually set up with the following:

- CentOS 7 Minimal (64-bit)
- 64-bit CPU
- 2GB RAM minimum
- 50GB HD minimum

These requirements are only for the initial setup. CPU, RAM, and storage requirements vary based on server use. Check the STC Suite specifications document prior to installing additional components on the server.

The instructions below explain how to build a base server from scratch. After following these steps, an image should be created so that these steps do not need to be performed again.

Install CentOS 7 Minimal and Configure

CentOS 7 Minimal is installed in order to be able to start from a lean Linux installation. Packages and software components are added as needed.

1. Disable the SSH login as root - Run the remainder of this steps as root or use sudo.
2. Edit `sshd_config`. Vi is used here, but any editor can be used.

   ```
   sudo vi /etc/ssh/sshd_config
   ```

3. On the line that says *#PermitRootLogin yes*, update it to the following (remember to remove the #):

   ```
   PermitRootLogin no
   ```

4. Restart the sshd service:

   ```
   sudo service sshd restart
   ```

5. By default, SELinux is set to be enforced. Change this by running:

```
sudo setenforce 0
```

6. Edit the following file so *permissive* persists across reboots:

```
sudo vi /etc/selinux/config
```

7. Change the line *SELINUX=enforcing* to the following and save the changes:

```
SELINUX=permissive
```

8. Make sure the software is completely up to date:

```
sudo yum update
```

9. Finally, install some basic packages:

```
sudo yum -y install net-tools gcc-c++ openssl-devel make git unzip bzip2
wget vim-enhanced ntp epel-release tmux dos2unix patch
```

JasperReports Server Installation - Linux

The following instructions explain how to install the JasperReports Server.

Prerequisite

Download all of the JasperReports Server contents from the distribution site. The JRS folder contains the following files:

- applicationContext-externalAuth-oAuth
- applicationContext-security
- applicationContext-security-web
- InstallCert$SavingTrustManager.class
- InstallCert.class
- jasperreports-server-6.2.1-linux-x64-installer.exe
- jasperserver.license
- LatoFont
- stc-jaspersoft-oauth-0.0.1-SNAPSHOT

Installation Steps

1. Download the JasperReports Server installer from the release distribution:

```
/Server Components/JRS/jasperreports-server-6.2.1-linux-x64-
installer.run
```

2. Place the JasperReports Server installer into the target directory:

```
/opt/jasperreports-server-6.2.1-linux-x64-installer.run
```

3. Navigate to the target directory:

```
$ cd /opt
```

4. Set the execute permissions for the JasperReports Server installer:

```
$ sudo chmod u+x ./jasperreports-server-6.2.1-linux-x64-installer.run
```

5. Run the installer:

```
$ sudo ./jasperreports-server-6.2.1-linux-x64-installer.run
```

6. Follow the prompts to accept the license agreement.
7. Follow the prompts to select an install option. Select option 1 - **Install All Components and Samples**.
8. Follow the prompts to specify the installation folder. Specify the following directory:

```
/opt/jasperreports-server-6.2.1
```

9. When the installer completes, start the JasperReports Server (see Start/Stop the JasperReports Server below).
10. Before continuing, you need to update a configuration file on the JasperReports Server. Navigate to:

```
cd /opt/jasperreports-server-6.2.1/apache-tomcat/webapps/jasperserver-
pro/WEB-INF/classes/
```

11. Open `jasperreports.properties` for editing and add the following under the section marked *# Highcharts static export properties*:

```
com.jaspersoft.jasperreports.highcharts.function.properties.allowed=true
```

It should look something like this:

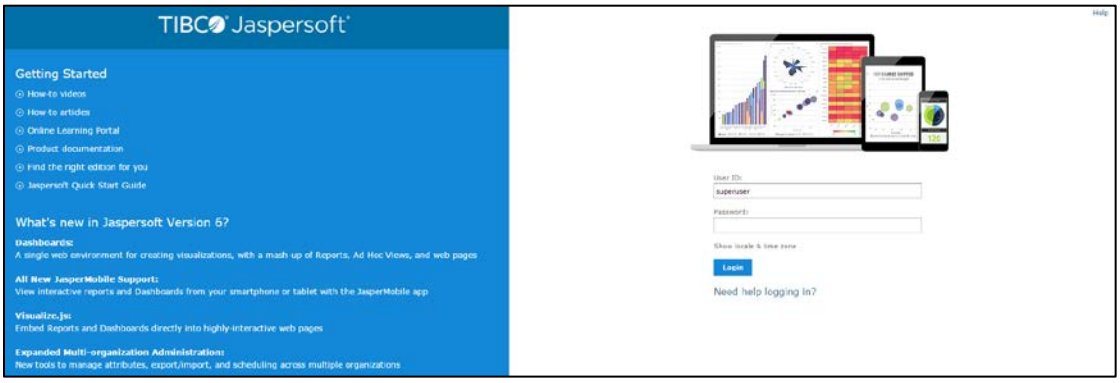
```
# Highcharts static export properties
com.jaspersoft.jasperreports.highcharts.render.require.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/require/require-2.1.6.src.js
com.jaspersoft.jasperreports.highcharts.jquery.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/jquery-1.10.2.min.js
com.jaspersoft.jasperreports.highcharts.highcharts.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.highcharts.more.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-more-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.highcharts.heatmap.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-heatmap-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.highcharts.treemap.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-treemap-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.highcharts.3d.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-3d-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.highcharts.solid.gauge.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/render/scripts/highcharts-solid-gauge-4.2.1.src.js
com.jaspersoft.jasperreports.highcharts.data.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/data.service.js
com.jaspersoft.jasperreports.highcharts.default.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/default.service.js
com.jaspersoft.jasperreports.highcharts.item.hyperlink.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/item.hyperlink.service.js
com.jaspersoft.jasperreports.highcharts.y.axis.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/y.axis.service.js
com.jaspersoft.jasperreports.highcharts.dual.pie.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/dual.pie.service.js
com.jaspersoft.jasperreports.highcharts.treemap.service.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/services/require/treemap.service.js
com.jaspersoft.ji.adhoc.highcharts.setting.service.js$context.url=reportresource?resource=com/jaspersoft/ji/adhoc/jr/require/adhocHighchartsSettingService.js
com.jaspersoft.jasperreports.highcharts.highcharts.chart.producer.js$context.url=reportresource?resource=com/jaspersoft/jasperreports/highcharts/charts/resources/highcharts.chart.producer.js
com.jaspersoft.jasperreports.highcharts.function.properties.allowed=true

# highcharts: phantomjs's executable absolute path
#com.jaspersoft.jasperreports.highcharts.phantomjs.executable.path=path/to/phantomjs
```

12. Save the file and restart the JasperReports Server (see Start/Stop the JasperReports Server below).
13. Validate the installation by logging in to the JasperReports Server console. Use your browser to access the server. Replace <hostname> below with the name or IP address of the server:

```
http://<hostname>:8080/jasperserver-pro
```

14. Log in with the user ID of *superuser* and the default password of *superuser*.



Update the License File

Follow these steps to update the license file for the JasperReports server:

1. Copy the license file (`jasperserver.license`) from the distribution site and move it to your home directory using FTP or WinSCP.
2. Replace the license file in the JasperReports installation directory (`/opt/jasperreports-server-6.2.1/`) with the license file from the above step, overwriting the existing `jasperserver.license` file.
3. Restart the JasperReports server (see Start/Stop the JasperReports Server below).

Increase the JasperReports Session Timeout Period

By default, the JasperReports timeout period is set to 20 minutes. However, the timeout period should be reset to 32 minutes in order to match with Keycloak.

1. Open `web.xml` from the path `/opt/jasperreports-server-6.2.1/apache-tomcat/webapps/jasperserver-pro/WEB-INF/web.xml`
2. Search for the property `<session-timeout>` under `<session-config>` and update the value to **32**. After this, the property and value should be similar to the image below:

3. Restart the JasperReports server (see Start/Stop the JasperReports Server below).

Start / Stop the JasperReports Server

Use these commands to start and stop the JasperReports server.

Start

```
$ cd /opt/jasperreports-server-6.2.1
$ ./ctlscript.sh start
```

Stop

```
$ cd /opt/jasperreports-server-6.2.1
$ ./ctlscript.sh stop
```

JasperReports Configuration with Keycloak

A few configuration changes need to be made in order for JasperReports to be used with Keycloak.

Prerequisite

Keycloak must be installed and set up, if the SSO-enabled version of IWeb and VOMS is being used.

JasperReports Server Modifications

1. Edit the following values in the `applicationContext-externalAuth-oAuth.xml` file:

Key	XML Tag Name(s) to Update	Value
[KEYCLOAK_URL]	<authorization_location> <token_location> <userdetails_location> <logoutUrl>	URL of the Keycloak server. Example: `https://sso-dev.stchome.com/`
[KEYCLOAK_REALM]	<authorization_location> <token_location> <userdetails_location> <logoutUrl>	Keycloak realm that the client belongs to. This value needs to be changed, along with <keycloak_URL>. In this example, the realm name is *Dev-Integration*: `https://sso-dev.stchome.com/auth/realms/`**`Dev-Integration`**
[JASPERSERVER_URL]	<property name="redirecturl"> <property name="logoutUrl">	URL of the Jasper server. Example: `http://<server_name>:8080/jasperserver-pro/oauth`
[CLIENT_ID]	<property name="client_id">	Client ID as set on Keycloak. Example: <property name="client_id"> <value>jasper-aws-dev</value>
[CLIENT_SECRET]	<property name="clientsecret"> <property name="userdetails_secret">	Credentials secret key generated by Keycloak.
State_ID	<property name="state">	This is the global setting used to populate the STATE_ID attribute in Jaspersoft. Example: <bean id="oAuthUserDetailsService" class="com.jaspersoft.jasperserver.ps.OAuth.OAuthUserDetailsServiceImpl"> <property name="state"> <value>WA</value> </property> </bean>
Jaspersoft	<bean	All the Jaspersoft Roles are configured

Key	XML Tag Name(s) to Update	Value
Roles	id="mtExternalUserSetupProcessor" class="com.jaspersoft.jasperserver.multipleTenancy.security.externalAuth.processors.MTExternalUserSetupProcessor" parent="abstractExternalProcessor"> <property name="organizationRoleMap"> <map> <!-- Example of mapping customer roles to JRS roles --> <entry> <key> <value>ROLE_ADMIN_EXTERNAL_ORGANIZATION</value> </key> <value>ROLE_ADMINISTRATOR</value> </entry> </map>	inside this tag. If any new Jaspersoft Roles are needed, they must be configured here. In the <entry> tag, the key value should match with the client role defined at the Keycloak end. Currently, the following roles are configured: 1. Administrator 2. Superuser 3. User Role

2. Copy the `applicationContext-externalAuth-oAuth.xml` file (edited in Step 1) to `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF`.
3. Copy stc-jaspersoft-oauth-1.0.0.jar (available in the JRS distribution folder) to `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/lib`.
4. Create and save a backup of the `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml` file.
5. In `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml`, replace the access value for the method `com.jaspersoft.jasperserver.api.metadata.user.service.ProfileAttributeService.putProfileAttribute` with **ROLE_USER**.
6. Save a backup copy of the `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml` file again and replace the constructor-arg value for org.springframework.security.web.authentication.LoginUrlAuthenticationEntryPoint with **/oauth**.
7. Add the SSL certificate used by Keycloak to the Java installation that Jaspersoft is using. (By default, Java only knows about a small number of root certificates.) To do this, follow these steps:
   a. Copy `InstallCert$SavingTrustManager.class` and `InstallCert.class` (both available in the `STC Suite Distribution/Server Components/JRS` directory) to the `<JASPERSERVER_DIR>/java/bin` directory.

b. NOTE for sso-dev.stchome.com: This is only for Dev SSO SSL certificate installation. This URL changes based on your server hostname of the Keycloak/SSO server.

c. Go to the <JASPERSERVER_DIR>/java/bin directory and execute these commands:

```
sudo ./java InstallCert sso-dev.stchome.com
sudo ./keytool -exportcert -alias sso-dev.stchome.com-1 -keystore
jssecacerts -storepass changeit -file sso-dev.stchome.com.cer
sudo ./keytool -importcert -alias sso-dev.stchome.com -keystore
../lib/security/cacerts -storepass changeit -file sso-
dev.stchome.com.cer
```

8. Make database changes with the following commands. Jaspersoft stores all of its metadata in the Postgres database. All of the user-based profile attributes are stored in the *jiprofileattribute* table in Jaspersoft. By default, user attributes support a character size of 200. In Step E below, the data type size is changed from 200 to 1,000.

   a. Navigate to the <Jasper Installation Directory>/postgres directory.

   b. Run this command: `sudo bin/psql -U postgres -W` (the password is *postgres*).

   c. Now you should see at the prompt: `postgres=#`

   d. Enter this command to connect to the Jasperserver database in Postgres: `\c jasperserver;` (the password is *postgres*).

   e. Execute this SQL alter script: `ALTER TABLE jiprofileattribute ALTER COLUMN attrvalue TYPE character varying(100000);`

9. Restart the Jasper server and navigate to JasperUrl/oauth (see example). You should now be redirected to Keycloak to log in. Example: `http://52.10.228.158:8080/jasperserver-pro/oauth`

Steps to Debug Issues Related to Keycloak/JasperReports Server Integration

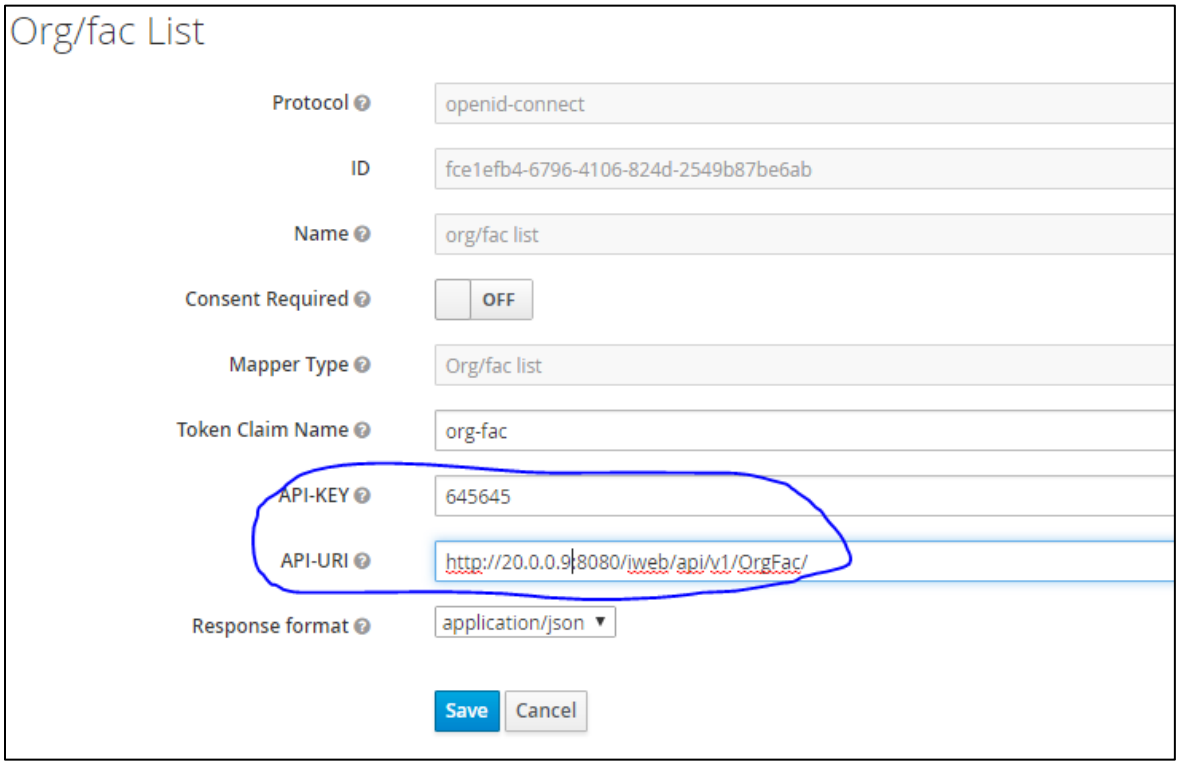
If there are any issues with integrating Keycloak with Jasper, follow these suggestions below.

Check the applicationContext-externalAuth-oAuth.xml File

Make sure that `applicationContext-externalAuth-oAuth.xml` is configured with the correct Keycloak and Jasper server parameters and with the correct client_secret.

Keycloak Mappers

Make sure that the Jasper client has the below mappers mapped correctly in Keycloak, and check if the Org/fac List mapper contains the correct parameters. (Note that your API-KEY and API-URI will differ from the image below.



Name	Category	Type
org/fac list	Token mapper	Org/fac list
email	Token mapper	User Property
groups	Token mapper	Group Membership
username	Token mapper	User Property
realm roles	Token mapper	Realm Role List
roles	Token mapper	Client Role List
given name	Token mapper	User Property
family name	Token mapper	User Property
full name	Token mapper	User's full name

Validate the IWeb Web Service

Make sure that the IWeb web service URL is accessible. The output of this should return the org/fac list. If there is an error, it needs to be fixed.

Check the Whitelist Rules

If Jaspersoft is installed in an AWS environment, the outbound connection on port 8080 should be enabled for the IWeb server.

If Keycloak is installed in an AWS environment, the outbound connection on port 8080 should also be enabled for the IWeb server.

Check the Firewall Rules

If there are any special firewall rules preventing the Jasper and/or Keycloak servers from accessing IWeb through port 8080, the need to be fixed so that the port is enabled.

VOMS JasperReports Deployment - Linux

The following outlines how to import (migrate) Jaspersoft Reports for VOMS on Linux.

Prerequisites

Confirm that the release distribution file has been downloaded and unzipped.

Installation

1. Log on to the JasperReports Server Console:

   ```
   http://myjasperserver:8080/jasperserver-pro
   ```

2. Perform an export (backup) of the Jaspersoft repository. The zipped file is downloaded to your local file system.

   ```
   Manage > Server Settings > Export (Export Everything)
   ```

3. Perform an import of the VOMS reports zipped file:

   ```
   Manage > Server Settings > Import > Choose File (Application
   Components/VOMS/JRS/VOMS_Reports.zip) > Import
   ```

4. Navigate to the data source connection properties:

   ```
   View > Repository > root > VOMS > Data Sources > VOMS Connection > Edit
   ```

5. Set the data source connection properties. The data source is to the IWeb Oracle database. See your IWeb administrator for database connection information.

Property	Comment
Host	IWeb hostname or IP address
Port	Oracle port number (usually 1521)
SID	Oracle site identifier (usually SIIS)
User Name	H33ASIIS
Password	See your IWeb administrator

6. Validate the connection via the **Test Connection** button.
7. Click **Save** to save the connection information.

Node.js Installation - Linux

Follow these steps to install Node.js on Linux.

1. Create the user environment: A user needs to be created to run Node.js applications. However, it is not recommended that Node.js applications be run as a privileged user.

```
sudo adduser node
```

2. Become the node user:

```
sudo su - node
```

3. To install Node.js for the previously created node user, Node Version Manager is recommended. This allows for more flexible control over the environment and an easier future upgrade path.

```
curl -o-
https://raw.githubusercontent.com/creationix/nvm/v0.32.1/install.sh |
bash
source ~/.bashrc
```

4. Install the required version of Node.js:

```
nvm install 6.9.1
nvm use 6.9.1
nvm alias default 6.9.1
```

5. Install the PM2 node process manager:

```
npm install -g pm2
```

VOMS Application Installation Instructions - Linux

The following instructions explain how to install the application, the Oracle Instant Client, and Redis; how to configure the application dashboard and quick links; general configuration options; and information on how to run the application and start the application upon bootup. These are the common installation requirements for the SSO dashboard, VOMS, and iQ for CentOS 7.

Prerequisites

The following should already be installed and configured:

- Linux CentOS 7
- PostgreSQL server (DBB)
- Mongo server (DBB)
- Oracle server (COR)
- Node.js (APP)
- PM2 (APP)

Installation and Deployment of iQ, SMaRT AFIX and VOMS

If you are installing VOMS alongside the SMaRT AFIX and/or iQ applications, you only need to install the Oracle Instant Client and Redis with the first application installation. Once these are installed, for subsequent applications that are running on the same server, you can skip down to the Install the Deployment Package section and continue from there instead.

Install the Oracle Instant Client

1. Download and install the Oracle Instant Client (both basic and SDK). The `.rpm` files can be found at http://www.oracle.com/technetwork/topics/linuxx86-64soft-092277.html. Install them with the RPM command.
2. Add the following to the users: `.bashrc`, `.cshrc`, and `.zshrc`, or use the default per-interactive-shell startup file of the default shell of the user for which you want to run the application. Optionally, instead of adding them on a per-user basis, the following can be added to `/etc/profile` or `/etc/profile.d/oracle.sh`:

```
export OCI_LIB_DIR=/usr/lib/oracle/12.1/client64/lib
export OCI_INC_DIR=/usr/include/oracle/12.1/client64
```

3. Create a file in `/etc/ld.so.conf.d/oracle.conf` with the path to the Oracle Instant Client libraries, which by default are as follows:

```
/usr/lib/oracle/12.1/client64/lib
```

Install Redis

1. Ensure epel repo is enabled and run the following with sudo/root:

```
yum install redis
```

2. Edit the `/etc/redis.conf` file and make sure the following variable line is set as follows (the default is usually set to an empty string ""):

```
notify-keyspace-events EKx
```

3. Set redis to autostart across system boots and start the service:

```
systemctl enable redis
systemctl start redis
```

Install phantomjs

VOMS 2.0 requires phantomjs to be installed at the root of the node user:

```
npm install phantomjs
```

Install the Deployment Package

Copy the provided zip/tar file to the target server. Extract the archive and change into the parent of the directory where it was extracted.

Configure the Application Dashboard and Quick Links

If you are installing VOMS alongside the SMaRT AFIX and/or iQ applications, the steps below only need to be completed the first time an application is installed. This file should be saved for all subsequent installations.

1. Download the `apps.json` file from the release distribution application components shared folder.
2. Edit this file with a text editor and replace the placeholders DASHBOARD_URL, IWEB_URL, AFIX_URL, PHC-HUB_URL, IQ_URL, and VOMS_URL (if applicable for your installation) with their respective URLs.
3. Save this file and keep it handy as you will need to use it in the installation of SMaRT AFIX, iQ, and the SSO dashboard.

An example of one such configuration object in the JSON file is shown below. In most cases, only the URL value needs to change (note that this example is for iQ, not VOMS):

```
{
"icon": "/static/public/img/iQ_dark.svg",
"name": "iQ",
"url": "http://iq.stchome.com/",
"description": "An application that allows for an efficient assessment of HL7
Data quality, and provides tools and metrics to help onboard new providers
and improve data quality.",
"accessRole": "Access_interop"
}
```

Copy this file to the following location:

```
<application_install_dir>/src/shared/helpers/appActions/
```

Populate the Location Lookup Table (VOMS Only)

1. Navigate to `scripts/voms/locationLookup`.
2. Ensure that the Oracle configuration in `updateLocationMapping.es6` is correct.
3. From the command line, run the following command:

```
node updateLocationMapping.js <StateCode>
```

Configure the Reverse Proxy

A reverse proxy is recommended to provide access to the various products. This is not explicitly required unless the products are running with the ENABLE_SSL option set to *true*. However, this will generally be the case when the products are being run in conjunction with IWeb on the same server. See the example Apache configurations at `package/docs/apache`.

General Product Configuration Options

Once the package containing the products has been installed, there are some configuration options that can be set depending on the environment. The majority of these lie in `package/process.json`, which is simply a JSON configuration file for pm2. The options listed in bold below need to be changed to match your environment. Other options not in bold might not need to change, and in most cases can be the default setting. Some options in the `process.json` file are not listed at all below; those options should be left unchanged.

The options are as follows:

Option	Description
script	Location of the startup script application. The path to the script

Option	Description
	may need to be customized.
cwd	The path to the application. May need to be customized.
PRODUCT	The name of the product.
HOST	The host name for the product.
APIHOST	The host name of the product's respective API server (generally *localhost*).
PORT	The port for the product.
APIPORT	The port for the product's API server.
NODE_ENV	This should always be **production**.
SAML_ISSUER	The SAML issuer (client) from Keycloak.
SAML_ENTRY_POINT	The SAML entry point from Keycloak.
JASPER_SERVER	The domain name of the JasperReports server.
REDIS_URL	The Redis URL used for storing client sessions.
REVERSE_PROXY	When configuring VOMS with a reverse proxy, this should be set to *true*.
ENABLE_SSL	When set to *true*, this enables HTTPS support. If REVERSE_PROXY is set to *true*, this should be set to *false* even when the environment is configured for HTTPS.
SECRET_SESSION	The secret key for session storage. **IMPORTANT NOTE:** This value must be exactly the same for every application, in both the server and the API environment variables.

Run the Application

Note: Run the following under the context of the node user.

Use pm2 to run the deployment package as shown below. In many cases, environment-specific configuration options must be set prior to running an application's package. See the configuration option sections above for more details.

```
pm2 start package-parent/process.json
```

Start the Products upon Bootup

To have the products automatically start on every boot, do the following once it has been verified that they are running correctly:

```
pm2 save
pm2 startup
```

Windows Instructions

The following instructions are for the Windows operating system.

JasperReports Server Installation - Windows

The following instructions explain how to install the JasperReports Server on Windows.

Prerequisite

Download all of the JasperReports Server contents from the distribution site. The JRS folder contains the following files:

- applicationContext-externalAuth-oAuth
- applicationContext-security
- applicationContext-security-web
- InstallCert$SavingTrustManager.class
- InstallCert.class
- jasperreports-server-6.2.1-windows-64bit-installer.exe
- jasperserver.license
- LatoFont
- stc-jaspersoft-oauth-0.0.1-SNAPSHOT

Installation Steps

1. Download the JasperReports Server for Windows and the license file from the release distribution:

```
Server Components\JRS\jasperreports-server-6.2.1-windows-64bit-
installer.exe
Server Components\JRS\jasperserver.license
```

2. Run the JasperReports Server Installer by right-clicking on the file and selecting **Run as administrator:**

```
jasperreports-server-6.2.1-windows-64bit-installer.exe
```

3. Follow the prompts to accept the license agreement.

4. Select the **Install All Components and Samples** option.
5. Follow the prompts to specify the installation folder. Specify the following directory:

```
C:\Jaspersoft\jasperreports-server-6.2
```

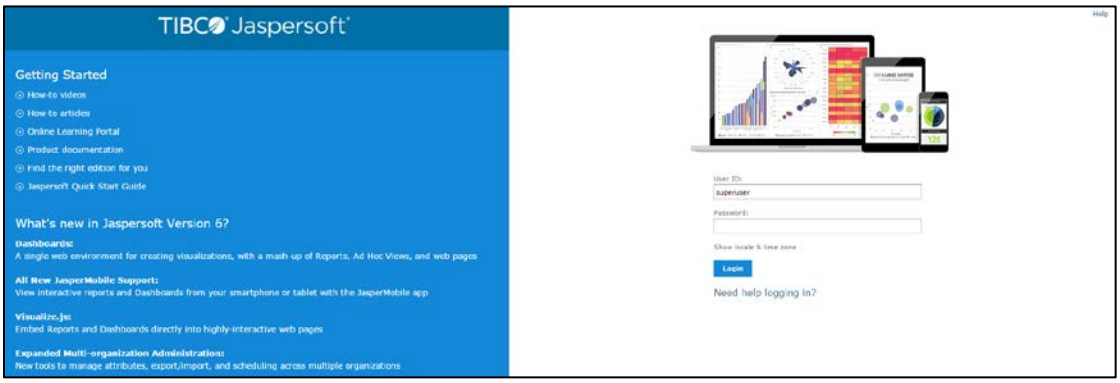
6. On completion of the installation, start the JasperReports server by selecting the following:

```
Launch JasperReports Server Now (for bundled Tomcat and PostgreSQL only)
If you're installing on Linux, don't close the terminal window running
the start script.
```

7. Validate the installation by logging in to the JasperReports Server console. Use your browser to access the server. Replace <hostname> below with the name or IP address of the server:

```
http://<hostname>:8080/jasperserver-pro
```

8. Log in with the user ID of *superuser* and the default password of *superuser*.



Update the License File

Follow these steps to update the license file for the JasperReports server:

1. Copy the license file (`jasperserver.license`) from the distribution site and move it to your home directory/desktop.
2. Replace the license file in the JasperReports installation directory (`C:\Jaspersoft\jasperreports-server-6.2\`) with the license file from the above step, overwriting the existing `jasperserver.license` file.
3. Restart the JasperReports server (see Start/Stop the JasperReports Server below).

Increase the JasperReports Session Timeout Period

By default, the JasperReports timeout period is set to 20 minutes. However, the timeout period should be reset to 32 minutes in order to match with Keycloak.

1. Open `web.xml` from the path `C:\Jaspersoft\jasperreports-server-6.2\apache-tomcat\webapps\jasperserver-pro\WEB-INF\web.xml`.
2. Search for the property `<session-timeout>` under `<session-config>` and update the value to **32**. After this, the property and value should be similar to the image below:

3. Restart the JasperReports server (see Start/Stop the JasperReports Server below).

Start / Stop the JasperReports Server

To start or stop the JasperReports server, click the Start button, go to **All Programs > JasperReports Server > Start or Stop Services**, and then select either **Start Service** or **Stop Service**.

JasperReports Configuration with Keycloak

A few configuration changes need to be made in order for JasperReports to be used with Keycloak.

Prerequisite

Keycloak must be installed and set up, if the SSO-enabled version of IWeb and VOMS is being used.

JasperReports Server Modifications

1. Edit the following values in the `applicationContext-externalAuth-oAuth.xml` file:

Key	XML Tag Name(s) to Update	Value
[KEYCLOAK_URL]	<authorization_location> <token_location> <userdetails_location> <logoutUrl>	URL of the Keycloak server. Example: `https://sso-dev.stchome.com/`
[KEYCLOAK_REALM]	<authorization_location> <token_location> <userdetails_location> <logoutUrl>	Keycloak realm that the client belongs to. This value needs to be changed, along with <keycloak_URL>. In this example, the realm name is *Dev-Integration*: `https://sso-dev.stchome.com/auth/realms/`**`Dev-Integration`**
[JASPERSERVER_URL]	<property name="redirecturl"> <property name="logoutUrl">	URL of the Jasper server. Example: `http://<server_name>:8080/jasperserver-pro/oauth`
[CLIENT_ID]	<property name="client_id">	Client ID as set on Keycloak. Example: <property name="client_id"> <value>jasper-aws-dev</value>
[CLIENT_SECRET]	<property name="clientsecret"> <property name="userdetails_secret">	Credentials secret key generated by Keycloak.
State_ID	<property name="state">	This is the global setting used to populate the STATE_ID attribute in Jaspersoft. Example: <bean id="oAuthUserDetailsService" class="com.jaspersoft.jasperserver.ps.OAuth.OAuthUserDetailsServiceImpl"> <property name="state"> <value>WA</value> </property> </bean>
Jaspersoft Roles	<bean id="mtExternalUserSetupProcessor" class="com.jaspersoft.jasperserver.multipleTenancy.security.externalAuth.processors.MTExternalU	All the Jaspersoft Roles are configured inside this tag. If any new Jaspersoft Roles are needed, they must be configured here. In the <entry> tag, the key value should match with the client role defined at the Keycloak end. Currently, the following roles are configured:

Key	XML Tag Name(s) to Update	Value
	serSetupProcessor" parent="abstractExternalProcessor"> <property name="organizationRoleMap"> <map> <!-- Example of mapping customer roles to JRS roles --> <entry> <key> <value>ROLE_ADMIN_EXTERNAL_ORGANIZATION</value> </key> <value>ROLE_ADMINISTRATOR</value> </entry> </map>	1. Administrator 2. Superuser 3. User Role

2. Copy the `applicationContext-externalAuth-oAuth.xml` file (edited in Step 1) to `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF`.
3. Copy stc-jaspersoft-oauth-1.0.0.jar (available in the JRS distribution folder) to `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/lib`.
4. Create and save a backup of the `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml` file.
5. In `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml`, replace the access value for the method `com.jaspersoft.jasperserver.api.metadata.user.service.ProfileAttributeService.putProfileAttribute` with **ROLE_USER**.
6. Save a backup copy of the `<JASPERSERVER_DIR>/apache-tomcat/webapps/jasperserver-pro/WEB-INF/applicationContext-security.xml` file again and replace the constructor-arg value for org.springframework.security.web.authentication.LoginUrlAuthenticationEntryPoint with **/oauth**.
7. Add the SSL certificate used by Keycloak to the Java installation that Jaspersoft is using. (By default, Java only knows about a small number of root certificates.) To do this, follow these steps:
   a. Copy `InstallCert$SavingTrustManager.class` and `InstallCert.class` (both available in the `STC Suite Distribution/Server Components/JRS` directory) to the `<JASPERSERVER_DIR>/java/bin` directory.

b. NOTE for sso-dev.stchome.com: This is only for Dev SSO SSL certificate installation. This URL changes based on your server hostname of the Keycloak/SSO server.

c. Go to the <JASPERSERVER_DIR>/java/bin directory and execute these commands. The [sso_link] differs based on the environment; an example might be something like sso-dev.stchome.com.

```
java -cp .\ InstallCert [sso_link]
keytool -exportcert -alias [sso_link]-1 -keystore jssecacerts -
storepass changeit -file [sso_link].cer
keytool -importcert -alias [sso_link] -keystore
../lib/security/cacerts -storepass changeit -file [sso_link].cer
```

8. Make database changes with the following commands. Jaspersoft stores all of its metadata in the Postgres database. All of the user-based profile attributes are stored in the *jiprofileattribute* table in Jaspersoft. By default, user attributes support a character size of 200. In Step E below, the data type size is changed from 200 to 1,000.

   a. Navigate to the <Jasper Installation Directory>/postgres directory.

   b. Run this command: `psql -U postgres -W` (the password is *postgres*).

   c. Now you should see at the prompt: `postgres=#`

   d. Enter this command to connect to the Jasperserver database in Postgres: `\c jasperserver;` (the password is *postgres*).

   e. Execute this SQL alter script: `ALTER TABLE jiprofileattribute ALTER COLUMN attrvalue TYPE character varying(100000);`

9. Restart the Jasper server and navigate to JasperUrl/oauth (see example). You should now be redirected to Keycloak to log in. Example: `http://52.10.228.158:8080/jasperserver-pro/oauth`

Steps to Debug Issues Related to Keycloak/JasperReports Server Integration

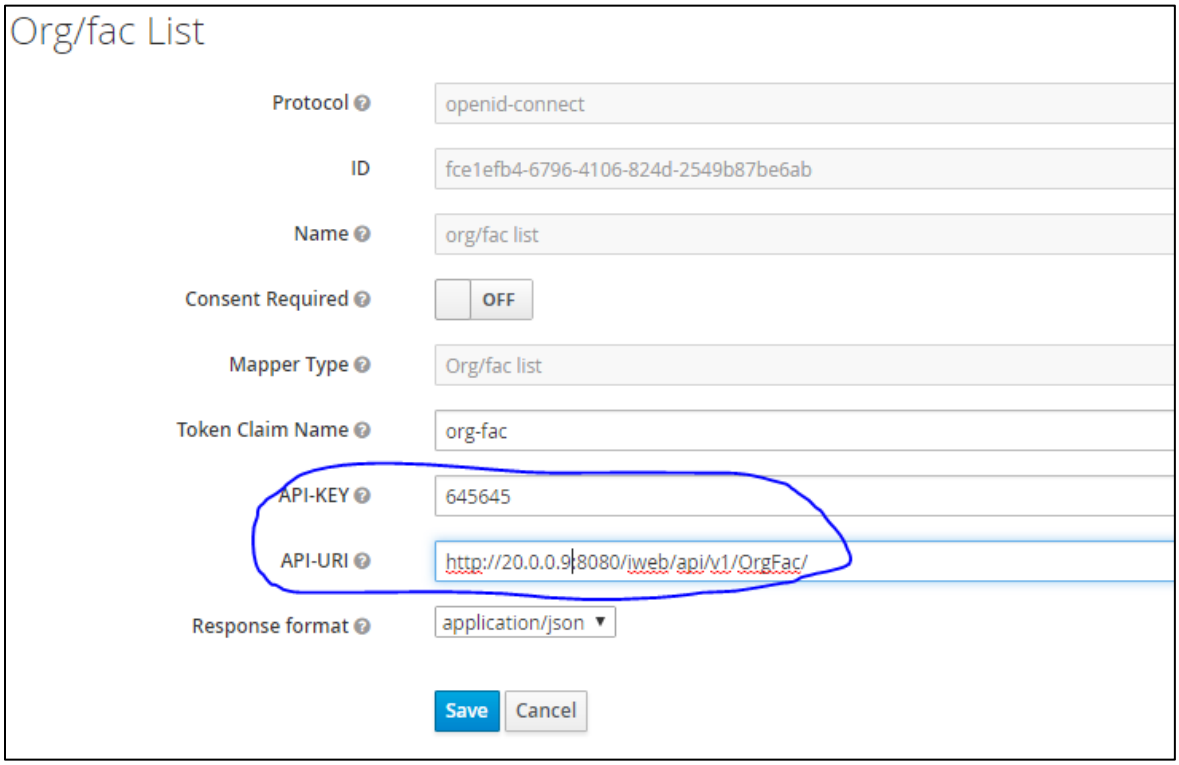
If there are any issues with integrating Keycloak with Jasper, follow these suggestions below.

Check the applicationContext-externalAuth-oAuth.xml File

Make sure that applicationContext-externalAuth-oAuth.xml is configured with the correct Keycloak and Jasper server parameters and with the correct client_secret.

Keycloak Mappers

Make sure that the Jasper client has the below mappers mapped correctly in Keycloak, and check if the Org/fac List mapper contains the correct parameters. (Note that your API-KEY and API-URI will differ from the image below.



Name	Category	Type
org/fac list	Token mapper	Org/fac list
email	Token mapper	User Property
groups	Token mapper	Group Membership
username	Token mapper	User Property
realm roles	Token mapper	Realm Role List
roles	Token mapper	Client Role List
given name	Token mapper	User Property
family name	Token mapper	User Property
full name	Token mapper	User's full name

Validate the IWeb Web Service

Make sure that the IWeb web service URL is accessible. The output of this should return the org/fac list. If there is an error, it needs to be fixed.

Check the Whitelist Rules

If Jaspersoft is installed in an AWS environment, the outbound connection on port 8080 should be enabled for the IWeb server.

If Keycloak is installed in an AWS environment, the outbound connection on port 8080 should also be enabled for the IWeb server.

Check the Firewall Rules

If there are any special firewall rules preventing the Jasper and/or Keycloak servers from accessing IWeb through port 8080, the need to be fixed so that the port is enabled.

VOMS JasperReports Deployment - Windows

The following outlines how to import (migrate) Jaspersoft Reports for VOMS on Windows.

Prerequisites

Confirm that the release distribution file has been downloaded and unzipped.

Installation

1. Log on to the JasperReports Server Console:

   ```
   http://myjasperserver:8080/jasperserver-pro
   ```

2. Perform an export (backup) of the Jaspersoft repository. The zipped file is downloaded to your local file system.

   ```
   Manage > Server Settings > Export (Export Everything)
   ```

3. Perform an import of the VOMS reports zipped file:

   ```
   Manage > Server Settings > Import > Choose File (Application
   Components/VOMS/JRS/VOMS_Reports.zip) > Import
   ```

4. Navigate to the data source connection properties:

```
View > Repository > root > VOMS > Data Sources > VOMS Connection > Edit
```

5. Set the data source connection properties. The data source is to the IWeb Oracle database. See your IWeb administrator for database connection information.

Property	Comment
Host	IWeb hostname or IP address
Port	Oracle port number (usually 1521)
SID	Oracle site identifier (usually SIIS)
User Name	H33ASIIS
Password	See your IWeb administrator

6. Validate the connection via the **Test Connection** button.
7. Click **Save** to save the connection information.

Node.js Installation - Windows

Follow these steps to install Node.js on Windows:

1. Go to https://nodejs.org and download the recommended version of Node.js.
2. Run the installer you downloaded.
3. Follow the prompts in the installer.
4. Restart your computer.
5. Open a command window and type: `npm install pm2 -g`
6. After the above, type: `npm i pm2-windows-service -g`

VOMS Application Installation Instructions - Windows

The following instructions explain how to install the application, the Oracle Instant Client, Visual Studio 2013 Community Edition, Windows SDK v8.1, and Redis; how to configure the application dashboard and quick links; specific configuration options, and information on standalone versus multiple application deployment.

If you are upgrading from a previously installed version of VOMS, you can skip directly to the Install the Deployment Package section and continue from there.

Prerequisites

The following should already be installed and configured:

- Windows Server 2012
- PostgreSQL server (DBB)
- Oracle server (COR)
- Node.js (APP)
- PM2 (APP)

Installation and Deployment of SMaRT AFIX, iQ, and VOMS

If you are installing VOMS alongside the iQ and/or SMaRT AFIX applications, you only need to install the Oracle Instant Client, Visual Studio, Windows SDK, and Redis with the first application installation. Once these are installed, for subsequent applications that are running on the same server, you can skip down to the Install the Deployment Package section and continue from there instead.

Install the Oracle Instant Client

1. Download Oracle Instant Client Basic and SDK for Windows. Extract Instant Client Basic somewhere on the disk (i.e., `C:\oracle\instantclient_12_1`), then extract Instant Client SDK into the same folder.
2. Add the path you extracted Oracle Instanct Client into to the PATH environment variable. This can be accessed via the Advanced System Settings window.
3. Add the following system variables. (See https://github.com/oracle/node-oracledb/blob/master/INSTALL.md#instwin and https://community.oracle.com/docs/DOC-931127 for more information.)
    - OCI_LIB_DIR to `C:\oracle\instantclient_12_1\sdk\lib\msvc`
    - OCI_INC_DIR to `C:\oracle\instantclient_12_1\sdk\include`

Install Visual Studio 2013 Community Edition

See https://www.visualstudio.com/en-us/news/releasenotes/vs2013-community-vs for download and installation instructions.

Install Windows SDK v8.1

See https://developer.microsoft.com/en-us/windows/downloads/windows-8-1-sdk for download and installation instructions.

Install Redis

See https://github.com/MicrosoftArchive/redis for more information. Download the release at https://github.com/MicrosoftArchive/redis/releases/tag/win-3.2.100.

1. Use this command to install Redis as a Windows Service:

```
msiexec /i Redis-x64-3.2.100.msi
```

2. Edit the `redis.widnows-service.conf` file which is located (in default installations) in `C:\Program Files\Redis`. Update the following variable line as follows (the default is usually set to an empty string ""):

```
notify-keyspace-events EKx
```

3. Restart the Redis Windows Service for the above change to take effect.

Install the Deployment Package

Copy the provided zip/tar file to the target server. Extract the archive and change into the parent of the directory where it was extracted.

Configure the Application Dashboard and Quick Links

If you are installing VOMS alongside the iQ and/or SMaRT AFIX applications, the steps below only need to be completed the first time an application is installed. This file should be saved for all subsequent installations.

1. Download the `apps.json` file from the release distribution application components shared folder.
2. Edit this file with a text editor and replace the placeholders DASHBOARD_URL, IWEB_URL, AFIX_URL, PHC-HUB_URL, IQ_URL, and VOMS_URL (if applicable for your installation) with their respective URLs.
3. Save this file and keep it handy as you will need to use it in the installation of iQ, SMaRT AFIX, and the SSO dashboard.

An example of one such configuration object in the JSON file is shown below. In most cases, only the URL value needs to change (note that this example is for SMaRT AFIX, not VOMS):

```
{
"icon": "/static/public/img/STC - SmartAFIX.svg",
"id": "AFIX",
"name": "SMaRT AFIX",
"url": "https://afix.stchome.com ",
"description": "Utilized to make AFIX assessments efficient, standardized and
meaningful.",
"hideIfUnavailable": true,
"accessRole": "Access_afix"
}
```

Copy this file to the following location:

```
<application_install_dir>/src/shared/helpers/appActions/
```

Populate the Location Lookup Table (VOMS Only)

1. Navigate to `scripts/voms/locationLookup`.
2. Ensure that the Oracle configuration in `updateLocationMapping.es6` is correct.
3. From the command line, run the following command:

```
node updateLocationMapping.js <StateCode>
```

Configure the Reverse Proxy

A reverse proxy is recommended to provide access to the various products. This is not explicitly required unless the products are running with the ENABLE_SSL option set to *true*. However, this will generally be the case when the products are being run in conjunction with IWeb on the same server. See the example Apache configurations at `package/docs/apache`.

General Product Configuration Options

Once the package containing the products has been installed, there are some configuration options that can be set depending on the environment. The majority of these lie in `package/process.json`, which is simply a JSON configuration file for pm2. The options listed in bold below need to be changed to match your environment. Other options not in bold might not need to change, and in most cases can be the default setting. Some options in the `process.json` file are not listed at all below; those options should be left unchanged.

The options are as follows:

Option	Description
script	Location of the startup script application. The path to the script may need to be customized.
cwd	The path to the application. May need to be customized.
PRODUCT	The name of the product.
HOST	The host name for the product.
APIHOST	The host name of the product's respective API server (generally *localhost*).
PORT	The port for the product.
APIPORT	The port for the product's API server.
NODE_ENV	This should always be **production**.
SAML_ISSUER	The SAML issuer (client) from Keycloak.
SAML_ENTRY_POINT	The SAML entry point from Keycloak.

Option	Description
JASPER_SERVER	The domain name of the JasperReports server.
REDIS_URL	The Redis URL used for storing client sessions.
REVERSE_PROXY	When configuring VOMS with a reverse proxy, this should be set to *true*.
ENABLE_SSL	When set to *true*, this enables HTTPS support. If REVERSE_PROXY is set to *true*, this should be set to *false* even when the environment is configured for HTTPS.
SECRET_SESSION	The secret key for session storage. **IMPORTANT NOTE:** This value must be exactly the same for every application, in both the server and the API environment variables.

Run the Application

Use pm2 to run the deployment package as shown below. In many cases, environment-specific configuration options must be set prior to running an application's package. See the configuration option sections above for more details.

```
pm2 start package-parent/process.json
```

Start the Products upon Bootup

To have the products automatically start on every boot, do the following once it has been verified that they are running correctly:

```
npm install -g pm2-windows-service
pm2-service-install [-n service-name]
pm2 save
```

Keycloak Configuration for VOMS

This task is to be performed during SSO installation of the application. The table below lists each application and component, along with the JSON file name, client name, and URL placeholder name for each. Make sure to use the correct file and other information corresponding to the specific application/component installation.

Application Name	JSON File Name	Client Name	URL Placeholder Name
IWeb	iweb.json	iweb	IWEB-URL
PHC-Hub	phchub.json	phc-hub	PHC-HUB-URL
Jasper Report Server	jasper.json	jasper	JASPER-URL
iQ	interop.json	interop	INTEROP-URL
VOMS	voms.json	voms	VOMS-URL
Logviewer	interop-logviewer.json	interop-logviewer	LOGVIEWER-URL
Dashboard	dashboard.json	dashboard	DASHBOARD-URL

Import the Client and Roles

These steps explain how to import the client and configure the roles.

Prerequisites

The location of the JSON file for the application being installed.

Import Steps

1. Log in to Keycloak.
2. Select the desired realm for the application suite in the top left drop-down list.
3. Click **Import** on the left side menu.
4. Click the **Select File** button and browse to the VOMS release `Application Components/VOMS/SSO/JSON` directory.
5. Select the `voms.json` file and click **Open**.
6. **Important:** In the *If a resource exists* drop-down list, select **Skip**.

Configure the Client

1. On the left side menu, click **Clients**.
2. Locate the client named *voms* you just imported (see above) and click **Edit**.
3. On the Settings tab, enter the URL to the application everywhere there is a URL placeholder.

4. If the Client Protocol is SAML, expand the Fine Grain SAML Endpoint Configuration section at the bottom of the page and make sure the URL placeholders are replaced with the application URL.
5. Click the **Save** button at the bottom.
6. Click on the Mappers tab.
7. Locate the *user-type* mapper and click the **Edit** button.
8. In the API-URI box, replace the URL placeholder with the URL to your IWeb installation.
9. Verify the API-KEY is set to the API key of your IWeb instance.
10. Click the **Save** button at the bottom.