

URQUI CAS Add On

1. Introduction.

- The URQUI CAS add on enhances the CAS single sign on software, by adding one time password (OTP) capability. OTP, greatly increases the security of the platforms CAS is being used to protect by eliminating the easily compromised static password operation.
- URQUI-CAS add-on is a new authentication handler that can be added to existing handlers or used on its own.
- The URQUI-CAS add-on is a free open-source software that can be modified to meet individual requirements. More information about URQUI, can be found at www.urqui.com.

2. Requirements specific for URQUI.

1. The URQUI-CAS add-on had been tested to CAS release 3.5.
2. Java policy files to enable unlimited strength encryption. These files and installation guidelines can be found at:
<http://www.oracle.com/technetwork/java/javase/downloads/ice-6-download-429243.html>
3. A JDBC database (MySQL, Oracle, etc.) and associated API that can be used to store user information. This can be an existing database or create a new one if required.
4. A unique URQUI host identifier and associated key that can be found at:
http://urqui.net/urqui_hostactivate.php.

3. Installation.

1. The URQUI-CAS add-on is available for download from GIT-Hub:
<https://github.com/urqui/cas>. This contains the jar file as well as associated source.
2. Place the jar file in the {JAVA-SERVER-CAS} lib directory. This would be the same directory where the JDBC API exists.
3. Create a new table, or use an existing table, and create at least two columns that will contain the user identifier (such as screenname, email address), and the users associated RQUI (the uses unique identifier to the URQUI system).

4. Retrieve the database access parameters, and user table credentials related to the table in 3. above. (dB name, table, login credentials). These need to be added (if not already present) to the deployerConfigContext.xml described below.
5. Modify the deployerConfigContext.xml to add the URQUI authentication handler. The handler can be added in sequence with other handlers or stand on its own. The following is a standard implementation of the URQUI handler. Modify the attributes in {BLUE} to your own requirements.

```
<bean class="org.kissi.urqui.urquicas.URQUIAuthenticationHandler">
  <property name="dataSource" ref="dataSource" />
  <property name="sql"
    <value="select {rqui field name} from {user table} where lower( {user
      identifier} ) = lower(?)" />
  <property name="url">
    <value> {URL for Validation } </value>
  </property>
  <property name="urquiid">
    <value> {URQUI Host Id } </value>
  </property>
  <property name="urquikey">
    <value> {URQUI Host Id Key} </value>
  </property>
</bean>
```