CAST-AAD-AED - Encrypt login and password for datasource and Active Directory

On this page:

Target audience:

CAST Administrators

Summary: this page describes how to encrypt logins and passwords for 1) connecting to database servers and 2) when configuring Active Directory authentication.

Introduction

When configuring CAST Application Analytics / Engineering Dashboards or RestAPI connections to **RDBMS/CSS database servers** (i.e. Measurement or Dashboard Services) or to an **LDAP server for Active Directory login mode** (see Installing and configuring the CAST Application Analytics Dashboard, Installing and configuring the CAST Application Engineering Dashboard and Installing and configuring the CAST-RestAPI.war) logins and passwords are defined in the relevant configuration files in **clear text**. This therefore represents a potential security risk. If your organization requires these logins and passwords to be **encrypted**, you can use the following instructions to do so.

Note that this document already assumes that you have a working connection to your deployed dashboard or RestAPI.

Encrypting access to RDBMS/CSS database servers

To encrypt the login and password that are defined when configuring access to the RDBMS/CSS database server where your Measurement or Dashboard Services are located, please proceed as follows:

- Browse to the following **URL** to access the built in **login/password key generation** page:

```
AAD - http://<server>:[<port>]/CAST-AAD/static/key.html
AED - http://<server>:[<port>]/CAST-AED/static/key.html
RestAPI - http://<server>:[<port>]/CAST-RestAPI/static/key.html
```

- Login with a user (whether static list or Active Directory) that has the **ROLE_ADMIN** role - by default no users have this role in either static list mode or in Active Directory mode - see CAST-AAD - Configuring user authentication or CAST-AED - Configuring user authentication for more information.

- When successfully authenticated, you now need to enter the **credentials (login and password)** for your target RDBMS/CSS2 database server (that you would ordinarily enter into the **context.xml file** for configuring access to the Measurement or Dashboard Service) and that you wish to encrypt. In the example below, we have entered the default credentials for a CSS2 database server:

- Now click the **Encrypt** button - CAST will then generate a key that relates to the credentials you entered:

- You now need to copy this key to the clipboard or to a text file.
- To use the key in place of clear text database credentials, browse to the following file:

```
AAD - %CATALINA_HOME\webapps\CAST-AAD\META-INF\context.xml
AED - %CATALINA_HOME\webapps\CAST-AED\META-INF\context.xml
RestAPI - %CATALINA_HOME\webapps\CAST-RestAPI\META-INF\context.xml
```

- Open this file with a text editor and scroll down to the location of a database access resource you have previously configured, for example:

```
<Resource name="jdbc/domains/AAD" url="jdbc:postgresql://localhost:2280/postgres"
    initConnectionSqls="SET search_path TO CAST_MEASURE;"
    username="operator" password="CastAIP"

    auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

- Replace the line containing "username" and "password" with your generated key using the following syntax:

```
key="D228ED8B5E5690B3A75"
```

- Add a new line directly underneath the line containing the "key" as follows - take note of the line that is specific to your release of Tomcat:

```
Tomcat 7: factory="com.castsoftware.adg.webservice.security.BasicDataSourceFactory"
Tomcat 8/8.5: factory="com.castsoftware.adg.webservice.security.BasicDataSourceFactory2"
```

- Your database access resource should now look like this (this is an example for Tomcat 7):

```
<Resource name="jdbc/domains/AAD" url="jdbc:postgresql://localhost:2280/postgres"
    initConnectionSqls="SET search_path TO CAST_MEASURE;"
    key="D228ED8B5E5690B3A75"
    factory="com.castsoftware.adg.webservice.security.BasicDataSourceFactory"

        auth="Container" type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"
    validationQuery="select 1"
    initialSize="5" maxActive="20" maxIdle="10" maxWait="-1"/>
```

- Save the file.
- Reload the cache - see CAST-AAD-AED - Reload the cache to reset the connection.
- Now reload your CAST Application Analytics / Engineering Dashboard or CAST RestAPI and ensure you can login and view the data you need to.

> You may need to repeat the above for each database server resource you have configured in the **context.xml** file.

Encrypting access to an Active Directory LDAP server

When configuring access to an LDAP server for Active Directory authentication, an Active Directory **user** and **password** must be specified in the **web.xml** file in clear text as described in Installing and configuring the CAST Application Analytics Dashboard, Installing and configuring the CAST Application Engineering Dashboard and Installing and configuring the CAST-RestAPI.war):

```
    <context-param>
        <description>Active directory: user</description>
        <param-name>authentication.activedirectory.login</param-name>
        <param-value>[user@domaine.societe.com]</param-value>
    </context-param>

    <context-param>
        <description>Active directory: password</description>
        <param-name>authentication.activedirectory.password</param-name>
        <param-value>[password]</param-value>
    </context-param>
```

To avoid the need to do this, please proceed as follows:

- Browse to the following **URL** to access the built in **login/password key generation** page:

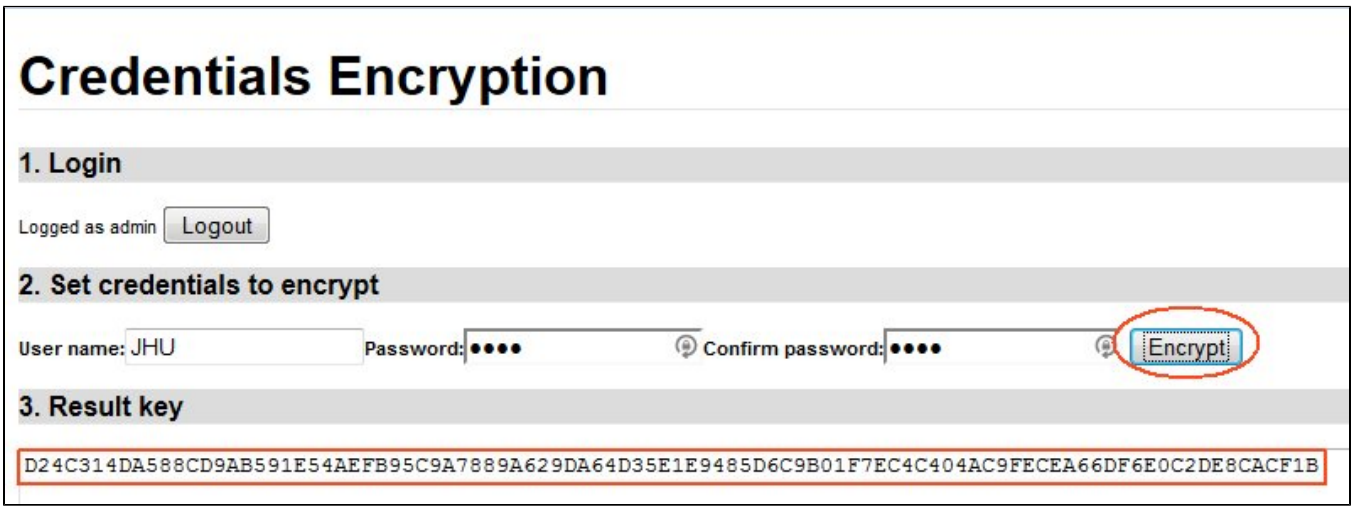
```
AAD - http://<server>:[<port>]/CAST-AAD/static/key.html
AED - http://<server>:[<port>]/CAST-AED/static/key.html
RestAPI - http://<server>:[<port>]/CAST-RestAPI/static/key.html
```

- Login with a user (whether static list or Active Directory) that has the **ROLE_ADMIN** role - by default no users have this role in either static list mode or in Active Directory mode - see CAST-AAD - Configuring user authentication or CAST-AED - Configuring user authentication for more information.

- When successfully authenticated, you now need to enter the **credentials (login and password)** for your Active Directory server (that you would ordinarily enter into the **web.xml file** for configuring Active Directory mode) and that you wish to encrypt. In the example below, we have entered the required Active Directory credentials:

Note that if you previously entered the username in the format **"username@domain.company.com"** (as oppose to **"username"** in the **web.xml**, you MUST also enter the username in the format **"username@domain.company.com"** here.

- Now click the **Encrypt** button - CAST will then generate a key that relates to the credentials you entered:



- You now need to copy this key to the clipboard or to a text file.
- Now open the following file with a text editor:

```
AAD - %CATALINA_HOME\webapps\CAST-AAD\WEB-INF\application-security-activedirectory.xml
AED - %CATALINA_HOME\webapps\CAST-AED\WEB-INF\application-security-activedirectory.xml
RestAPI - %CATALINA_HOME\webapps\CAST-RestAPI\WEB-INF\application-security-activedirectory.xml
```

- Locate the following configuration in the file:

```
<bean id="activeDirectoryServer" class="com.castsoftware.adg.webservice.security.
LdapSpringSecurityContextSource">
    <constructor-arg value="${authentication.activedirectory.ldapurl}"/>
    <property name="userDn" value="${authentication.activedirectory.login}"/>
    <property name="password" value="${authentication.activedirectory.password}"/>
    <!-- <property name="key" value="0AC811..."/>-->
    <property name="baseEnvironmentProperties">
        <map>
            <entry key="java.naming.referral" value="follow" />
        </map>
    </property>
</bean>
```

- First comment out the two lines containing **<property name="userDn"** and **<property name="password"**
- Then remove the comments around the line containing **<property name="key" value=** and paste in your encryption key as generated previously
- This should give you the following:

```
<bean id="activeDirectoryServer" class="com.castsoftware.adg.webservice.security.
LdapSpringSecurityContextSource">
    <constructor-arg value="${authentication.activedirectory.ldapurl}"/>
    <!--<property name="userDn" value="${authentication.activedirectory.login}"/> -->
    <!--<property name="password" value="${authentication.activedirectory.password}"/> -->
    <property name="key" value="0AC81167899"/>
    <property name="baseEnvironmentProperties">
        <map>
            <entry key="java.naming.referral" value="follow" />
        </map>
    </property>
</bean>
```

- Save the file.
- Reload the cache - see CAST-AAD-AED - Reload the cache to reset the connection.
- Now reload your CAST Application Analytics / Engineering Dashboard or CAST RestAPI and ensure you can login.

Optional

If you have **previously configured Active Directory mode without encryption** (i.e. username and password visible in the web.xml configuration file), browse to the following file to adjust the user and password used to access the Active Directory LDAP server:

```
AAD - %CATALINA_HOME\webapps\CAST-AAD\WEB-INF\web.xml
AED - %CATALINA_HOME\webapps\CAST-AED\WEB-INF\web.xml
RestAPI - %CATALINA_HOME\webapps\CAST-RestAPI\WEB-INF\web.xml
```

- Open this file with a text editor and scroll down to the location of the user and password you have previously configured for authentication to the Active Directory LDAP server, for example:

```
    <context-param>
        <description>Active directory: user</description>
        <param-name>authentication.activedirectory.login</param-name>
        <param-value>JHU@domaine.societe.com</param-value>
    </context-param>

    <context-param>
        <description>Active directory: password</description>
        <param-name>authentication.activedirectory.password</param-name>
        <param-value>some_password</param-value>
    </context-param>
```

- In place of the **user** and **password** values entered into both <param-value> parameters, enter [NOT USED – see application-security-activedirectory.xml].
- This will result in the following:

```
    <context-param>
        <description>Active directory: user</description>
        <param-name>authentication.activedirectory.login</param-name>
        <param-value>[NOT USED - see application-security-activedirectory.xml]</param-value>
    </context-param>

    <context-param>
        <description>Active directory: password</description>
        <param-name>authentication.activedirectory.password</param-name>
        <param-value>[NOT USED - see application-security-activedirectory.xml]</param-value>
    </context-param>
```

- Ensure you save the file.