EUROPEAN COMMISSION

DIGIT
Connecting Europe Facility

Quick Start Guide

for the

Business Document Metadata Service Location (BDMSL)

Version [1.10]

Status [Final]

© European Union, 2018

Reuse of this document is authorised provided the source is acknowledged. The Commission's reuse policy is implemented by Commission Decision 2011/833/EU of 12 December 2011 on the reuse of Commission documents.

Date: 02/05/2018

Document Approver(s):

Approver Name	Role
Joao RODRIGUES	CEF eDelivery

Document Reviewers:

Reviewer Name	Role
Adrien FERIAL	Product Owner

Summary of Changes:

Version	Date	Created by	Short Description of Changes
1.5	26/08/2016	Adrien FERIAL	Version
1.6	30/09/2016	Yves ADAM	Align to new template
1.7	19/07/2017	Flavio SANTOS	Add configuration parameters
1.8	16/01/2018	Flavio SANTOS	Add encryption configuration parameters
1.9	27/03/2018	CEF Support	Reuse policy notice added, e-SENS profile replaced by eDelivery profile
1.10	24/04/2018	Flavio SANTOS	Configuration files in detail

Table of Contents

1. INTRODUCTION 4

1.1. Purpose of the Quick Start Guide 4

1.2. Pre-requisites 4

2. PROCEDURE 5

3. CONFIGURATION 7

3.1. Initial parameters 7

3.2. How to generate a private key file 7

3.3. How to encrypt a password 8

3.4. Certificate to sign responses 10

3.5. Files to be copied under application server 11

4. CONTACT INFORMATION 12

1. INTRODUCTION

BDMSL stands for Business Document Metadata Service Location. BDMSL is the sample implementation of the SML maintained by DG DIGIT. The version of the BDMSL refered in this document is 3.1.2. This version implements the eDelivery BDXL profile (see https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eDelivery+BDXL)

1.1. Purpose of the Quick Start Guide

This document provides a brief description of the installation of the BDMSL component. This application uses Liquibase as a database management tool. Every war includes the SQL updates written as XML Liquibase files. For the installation of the component, to avoid a failed deployment, it is necessary to extract the XML Liquibase files, configure them and then execute the database updates. Then the application can be launched.

This guide illustrates the different steps for an installation on a Tomcat server with a MySQL database. If you use a different application server and/or a different database provider, please adapt the command lines.

1.2. Pre-requisites

- JDK 8
- Tomcat 8
    - Unzip Tomcat in the directory `tomcatDir`
- MySQL driver
    - Put this driver into the folder `tomcatDir/lib`
- Liquibase 3.4
- The user configured for the data source in Tomcat must have administrative rights on the database

2. PROCEDURE

For the example below we will assume the SML version 3.1.2-SNAPSHOT.

- Unzip `Liquibase` in a directory. We will refer to this directory as `liquibaseDir` in the rest of this document.
- Download `bdmsl-webapp-3.1.2-SNAPSHOT-tomcat-mysql.war` on CEF Digital Repository and put it at the root of `liquibaseDir`
- Open a command prompt in `liquibaseDir`
- Run the following command lines:

```
unzip -p bdmsl-webapp-3.1.2-SNAPSHOT-tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-master.xml >db.changelog-master.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-default-data-inserts.xml >db.changelog-default-data-inserts.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-create.xml >db.changelog-create.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-update-3.0.1.xml >db.changelog-update-3.0.1.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-update-3.1.RC1.xml >db.changelog-update-3.1.RC1.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-update-3.1.RC2.xml >db.changelog-update-3.1.RC2.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-update-3.1.2.xml >db.changelog-update-3.1.2.xml

unzip -p bdmsl-webapp-3.1.2-SNAPSHOT -tomcat-mysql.war WEB-INF/classes/liquibase/db.changelog-version.xml >db.changelog-version.xml
```

- Open the file "`db.changelog-default-data-inserts.xml`": it contains the configuration settings for the first launch. Change the values accordingly with your environment by uncommenting the last change set (under Environment Specific comment: id = "999999") and adding updates to the existing Data. For instance, you can change the default configuration path by modifying the `configurationDir` property like this:

```
<!-- Environment Specific  -->
<changeSet author="eDelivery" id="999999" logicalFilePath="path-independent">
    <update tableName="bdmsl_configuration">
        <column name="value" value='false' type="LONGTEXT"/>
        <where>property='dnsClient.SIG0Enabled'</where>
    </update>
    <update tableName="bdmsl_configuration">
        <column name="value" value='false' type="LONGTEXT"/>
        <where>property='signResponse'</where>
    </update>
    <update tableName="bdmsl_configuration">
        <column name="value" value='false' type="LONGTEXT"/>
        <where>property='dnsClient.enabled'</where>
    </update>
    <update tableName="bdmsl_configuration">
        <column name="value" value='E:/WorkingDirectory/__BDSML/_Environment/liquibase-3.4.1/confDir' type="LONGTEXT"/>
        <where>property='configurationDir'</where>
    </update>
</changeSet>
```

Note: `configurationDir` is the path to all the needed files such as privateKeys for DNS, Encryption and Signature. Please make sure this property is configured in the database and all files are under this location before start Webserver.

- Create a new file `liquibase.properties` at the root of `liquibaseDir` and put the following content in this file. You need to adapt the values for your environment:

```
driver: com.mysql.jdbc.Driver
classpath: <tomcatDir>/lib/mysql-connector-java-5.1.34.jar
url: jdbc:mysql://localhost:3306/bdmsl
username: root
password: root
changeLogFile: db.changelog-master.xml
```

- Run the following command line in `liquibaseDir`:

```
liquibase update
```

- Copy `bdmsl-webapp-3.1.2-SNAPSHOT-tomcat-mysql.war` in the `tomcatDir/webapp` folder

- Create a new data source in Tomcat named `java:comp/env/jdbc/edelivery`.
  For that go to TOMCAT_HOME/conf/context.xml and add the block:

```
<Resource name="jdbc/edelivery" auth="Container" type="javax.sql.DataSource"
          maxTotal="100" maxIdle="30" maxWaitMillis="10000"
          username="root" password="root" driverClassName="com.mysql.jdbc.Driver"
          url="jdbc:mysql://localhost:3306/bdmsl"/>
```

- Start tomcat

3. CONFIGURATION

3.1. Initial parameters

After the execution of the liquibase script for the database apply the followings:

- Switch the new property `authentication.bluecoat.enabled` in the table `BDMSL_CONFIGURATION` to true if the SML must manage BlueCoat authentication.
- Create the following property for each existing subdomain from the table `BDMSL_SUBDOMAIN` like `subdomain.validation.smpLogicalAddressProtocolRestriction.sampleSubDomain` in the table `BDMSL_CONFIGURATION` where `sampleSubDomain` is the subdomain name. The value of the property can be `all`, `http` **or** `https`.
- Create the following property for each existing subdomain from the table `BDMSL_SUBDOMAIN` like `subdomain.validation.participantIdRegex.sampleSubDomain` in table `BDMSL_CONFIGURATION` where `sampleSubDomain` is the subdomain name. The value of this property must be populated with a regular expression to define the syntax of accepted subdomain names in addition to ISO 15459 constraints governing these identifiers. By default, the regular expression ***^.*$*** may be used. It accepts any sequence of characters and therefore adds no restriction.
- Configure properties `dataInconsistencyAnalyzer.senderEmail` and `dataInconsistencyAnalyzer.recipientEmail` for sending and receiving Data Inconsistency reports with appropriate email addresses.
- Create the following property for each existing subdomain from the table BDMSL_`SUBDOMAIN` like `dnsClient.domain.sampleSubDomain` in table `BDMSL_CONFIGURATION` where `sampleSubDomain` is the domain name. The value of the property must be the `domain`.
- Create the following property for each existing subdomain from the table `BDMSL_SUBDOMAIN` like `dnsClient.recordTypes.sampleSubDomain` in table `BDMSL_CONFIGURATION` where **sampleSubDomain** is the domain name. The value of the property must be `all`, `cname` or `naptr`.
- If value of the property `useProxy` is true, please follow step 3.2 and 3.3. Afterwards copy the encoded and encrypted password to the value of the property `httpProxyPassword` in the table `BDMSL_CONFIGURATION`.
- Application passwords stored in configuration table have to be encrypted by administrator. This action needs to be done only if "Signing response" or "proxy" features are used. More information at item 3.2 and 3.3 .

3.2. How to generate a private key file

SML provides a tool to create a private key to encrypt proxy and signing keystore passwords. In order to create a private key, please follow the steps below:

- Download one of the latest BDMSL war files (eg: bdmsl-webapp-3.1.2-weblogic-oracle.war) from the repository https://ec.europa.eu/cefdigital/artifact/content/repositories/eDelivery/eu/europa/ec/bdmsl/bdmsl-webapp/3.1.2/

- Extract the war file using any extracting tool

- Run the following commands to create a private key

    1. cd bdmsl-webapp-3.1.2-weblogic-oracle

    2. java -cp "WEB-INF/lib/*" eu.europa.ec.bdmsl.common.util.PrivateKeyGenerator c:\temp\encriptionPrivateKey.private

    Required parameter = Full directory path where the private key will be created

Example:

Printed result:

Private key created at c:\temp\encriptionPrivateKey.private

Once the private key is generated, please copy the private key file name to the value of the property `encriptionPrivateKey` in the table BDMSL_Configuration, and copy the private file to the path configured in the property `configurationDir`.

3.3. How to encrypt a password

After generating a private key at item "§3.2- How to generate a private key file" please configure the proxy or keystore (used to sign response) password if needed as follows:

- Inside the folder already extracted from BDMSL .war file, please run the command below:

  java -cp "WEB-INF/lib/*" eu.europa.ec.bdmsl.common.util.EncryptPassword c:\temp\privateKey.private Password123

  1st parameter = private key location

  2nd parameter = plain text password

- To configure the proxy password, please copy the printed encrypted and base64 encoded password to the value of the `httpProxyPassword` property in the table `BDSML_CONFIGURATION`.

  Example:

  httpProxyPassword = vXA7JjCy0iDQmX1UEN1Qwg==

- To configure the keystore password, please copy the printed encrypted and base64 encoded password to the value of the `keystorePassword` property in the table BDSML_CONFIGURATION.

  Example:

  keystorePassword = vXA7JjCy0iDQmX1UEN1Qwg==

3.4. Certificate to sign responses

If the flag `signResponse=true` in the table `BDMSL_CONFIGURATION`, a keystore file name, its alias and password must be provided in the same table.

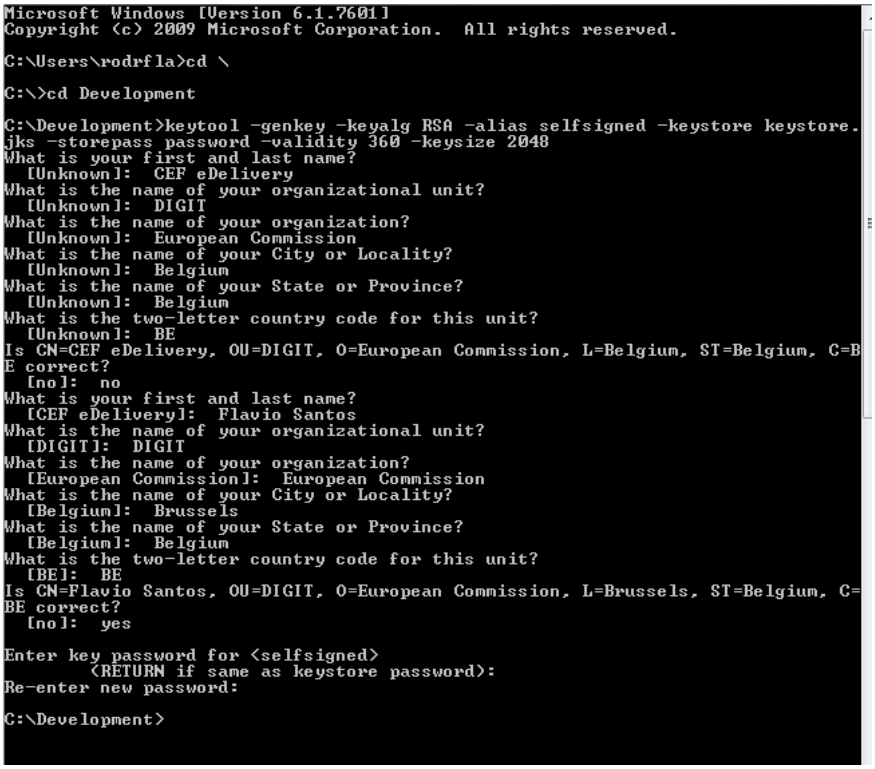
For testing purposes only, it is possible to create a self-signed keystore as follows:

Open the command console on whatever operating system you are using and navigate to the directory where keytool.exe is located (usually where the JRE is located, e.g. c:\Program Files\Java\jre8\bin on Windows machines).

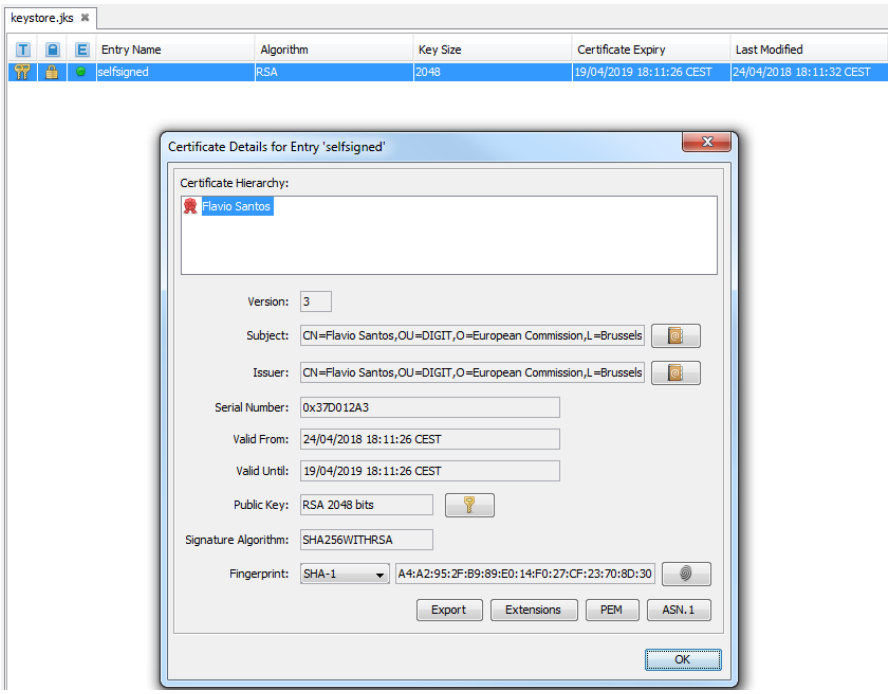
Run the following command (where validity is the number of days before the certificate will expire):

```
keytool -genkey -keyalg RSA -alias selfsigned -keystore keystore.jks -storepass password -validity 360 -keysize 2048
```

Fill in the prompts for your organization information as below:



This will create a keystore.jks file containing a private key and your sparklingly fresh self signed certificate. Now you just need to configure your Java application to use the .jks file.



3.5. Files to be copied under application server

In the configuration directory that you specified in the `configurationDir` property, you need to put the following files:

- `keystore.jks` (the name can be changed in the property `keystoreFileName`): this keystore must contain your private key with the alias and password defined in the `keystoreAlias` and `keystorePassword` properties.
- `sig0.private` (the name can be changed in the property `dnsClient.SIG0KeyFileName`): this file is only required if you use DNSSEC (i.e. property `dnsClient.SIG0Enabled` set to true).
- `encriptionPrivateKey.private` (the name can be changed in the property `encriptionPrivateKey`): this private key file is only required if you use Proxy or Sign Response.

Once the needed files have been copied, restart the server(s).

4. CONTACT INFORMATION

CEF Support Team

By email: CEF-EDELIVERY-SUPPORT@ec.europa.eu

By phone: +32 2 299 09 09

- Standard Service: 8am to 6pm (Normal EC working Days)
- Standby Service*: 6pm to 8am (Commission and Public Holidays, Weekends)

* *Only for critical and urgent incidents and only by phone*