EUROPEAN COMMISSION

DIGIT
Connecting Europe Facility

Service Metadata Publisher

Administration Guide

SMP 4.X

Version [2.0]

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: 01/10/2018

Document Approver(s):

Approver Name	Role
Adrien FERIAL	

Document Reviewers:

Reviewer Name	Role
Pawel GUTOWSKI	Developer
Caroline AEBY	CEF Support

Summary of Changes:

Version	Date	Created by	Short Description of Changes
V0.1	22/02/2016	DHENEIN Christophe	Initial Document Creation
V1.0	15/06/2017	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	Document restructuring and updating.
V1.1	16/06/2016	Chaouki BERRAH (CEFS), Christophe DHENEIN (CEFS)	3.0.0-RC2 changed to 3.X Screens updated.
V1.2	12/02/2018	Chaouki BERRAH	Version 4.0 RC1 changes added
V1.3	16/02/2018	Chaouki BERRAH	Update after test installation
V1.4	21/02/2018	Chaouki BERRAH	Pawel changes included.
V1.5	26/02/2018	Chaouki BERRAH	Links added
V1.6	28/02/2018	Chaouki BERRAH Caroline AEBY	Updates
V1.7	20/03/2018	CEF Support	Reuse notice added
V1.8	11/06/2018	Chaouki BERRAH	Document update
V1.9	28/06/2018	CEF Support	References to nexus updated.
V2.0	01/10/2018	Caroline AEBY	No more standby service

Table of Contents

1. INTRODUCTION

This Administration Guide is intended for Administrators who are in charge of installing, managing and troubleshooting an eDelivery SMP (Service Metadata Publisher).

1.1. Purpose

The purpose of this guide is to provide detailed information on how to deploy and configure an SMP 4.X on either a WebLogic or Tomcat Application Server with either MySQL or Oracle database.

It also provides detailed descriptions of the related Security Configurations (Certificates).

There is also a section on the use of Soap UI to create, update and delete SMP Service Groups and Metadata.

Another section describes an alternative method to perform the creation, update and deletions using Swagger UI.

2. CONVENTION

The Commands and Configuration files listed in this document usually contain a mix of reserved words (commands, instructions and system-related special words) and user-defined words (chosen by the user) as well as comments and preferred values for certain variables. The conventions used in this document, to distinguish between them, are the following:

- **Bold** is used for "reserved" words and commands.
- *Normal italic* together with a short description of the argument, is used for user-defined names (chosen by yourself to designate items like users, passwords, database etc.). It normally contains at least 2 words separated by "_".
- ***Bold and italic*** is used for advisable values which can be changed by the user depending on their infrastructure.
- Comments are sometimes added to describe the purpose of the commands, usually enclosed in brackets ().
- By default, non-OS specific paths will be described using Linux patterns.

Example 1: Sample Oracle Statement:

create user *smp_user* **identified by** *smp_password*;

grant all privileges to *smp_user*;

(Where *smp_user* and *smp_password* are names chosen by the user)

Example 2: Sample Configuration File:

jdbc.driver = com.mysql.jdbc.Driver

jdbc.url = jdbc:mysql://localhost:3306/smp_database

jdbc.user = smp_user

jdbc.password = *smp_password*

target-database = MySQL

(Where: *smp_user, smp_database* and *smp_password* are names chosen by the user.

localhost:3306 represents hostname:port parameters of the MySQL database.)

3. PREREQUISITES

Please install the following software on the target system. For further information and installation details, please refer to the software owner's documentation.

- Java runtime environment is now (JRE) 8 **only**: http://www.oracle.com/technetwork/java/javase/downloads/index.html
- **One** of the supported Database Management Systems :
  - MySQL 5,6 or above
  - Oracle 10g+
- **One** of the supported Application Servers:
  - WebLogic 12c
  - Tomcat 8

3.1. Binaries repository

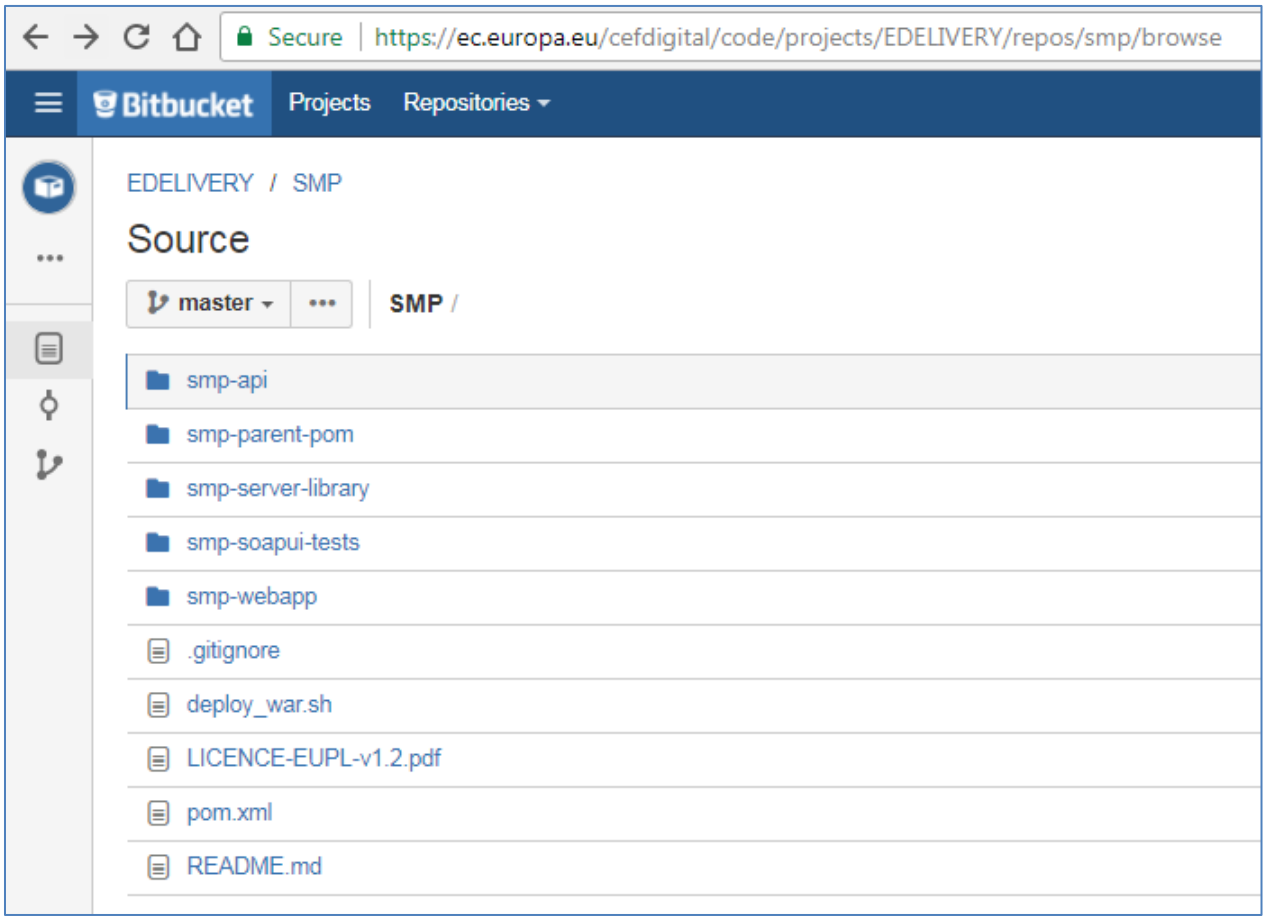
All the CEF SMP artefacts can be directly downloaded from the CEF Digital site[1].

3.2. Source Code Repository

The source code of CEF SMP is available in the **GIT** repository at the following location:

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp/browse

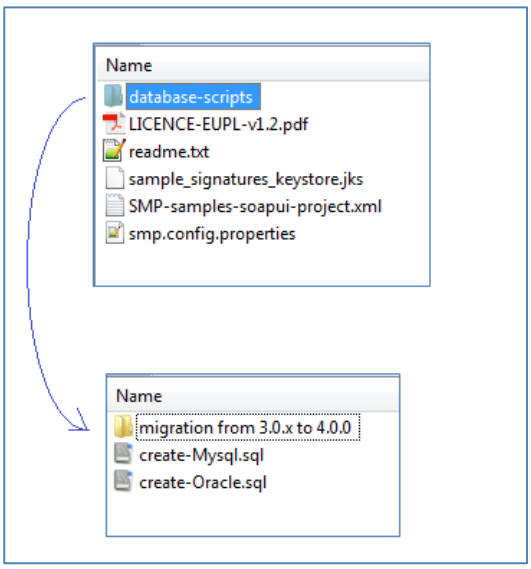
[1] https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/SMP



3.3. Database Scripts

The scripts to create (or migrate) the Oracle or MySQL databases are included in the following downloadable zip file:

https://ec.europa.eu/cefdigital/artifact/content/repositories/public/eu/europa/ec/edelivery/smp/4.0.0/smp-4.0.0-setup.zip



4. DEPLOYMENT

4.1. Deployment overview

As mentioned in the prerequisites, the deployment of the CEF SMP is only supported on Tomcat and WebLogic application servers.

The deployment of the CEF SMP on both platforms is almost identical and only minor platform specific changes will be documented in a dedicated section of this manual.

The deployment of the CEF SMP can be summarized in the following mandatory steps:

- Database Configuration
- Application Server Preparation
- SMP Initial Configuration
- SMP .WAR file Deployment

Remark:

*The environment variable, **cef_edelivery_path**, refers to the name of the folder where the SMP package is installed and will be used in the remainder of this document.*

*For Tomcat, it refers to **CATALINA_HOME**.*

*For Oracle WebLogic, it refers to **DOMAIN_HOME**.*

5. DATABASE CREATION

This section describes the steps necessary to create the database, tables and the SMP database user (**dbuser** used for database connection purpose).

It also includes the creation of an initial SMP user account that will be used by REST clients to connect to the SMP.

The SMP uses a direct connection to the database, which removes the need to configure a data source within WebLogic.

For this step you need to use the script included in the zip file downloaded in section §3.3.

https://ec.europa.eu/cefdigital/artifact/content/repositories/public/eu/europa/ec/edelivery/smp/4.0.0/smp-4.0.0-setup.zip

5.1. MySQL

1. Download and copy the create-Mysql.sql script to cef_edelivery_path/sql-scripts
2. Open a command prompt and navigate to the cef_edelivery_path/sql-scripts folder
3. Execute the following MySQL commands:

```
mysql -h localhost -u root_user --password=root_password -e "drop schema if
exists smp_schema;create schema smp_schema;alter database smp_schema
charset=utf8; create user smp_dbuser@localhost identified by
'smp_password';grant all on smp_schema.* to smp_dbuser@localhost;"
```

This creates a *smp_schema* and an *smp_dbuser* with (all) privileges to the smp_schema**.**

Execute the following command to create the required objects (tables, etc.) in the database:

```
mysql -h localhost -u root_user -proot_password smp_schema < create-Mysql.sql
```

5.2. Oracle Database

1. Download and copy the **create-Oracle.sql script** to *cef_edelivery_path***/sql-scripts**
2. Navigate to *cef_edelivery_path***/sql-scripts** directory
3. Execute the following commands :

```
sqlplus sys as sysdba   (password should be the one assigned during the Oracle
installation )

================================================================================
Once logged in Oracle:
create user smp_dbuser identified by smp_dbpassword;

grant all privileges to smp_dbuser;
```

```
connect smp_dbuser
show user;  (should return : smp_dbuser)
@create-Oracle.sql (run the scripts with the @ sign from the location of the scripts)
exit
==========================================================================
```

6. ORACLE WEBLOGIC CONFIGURATION

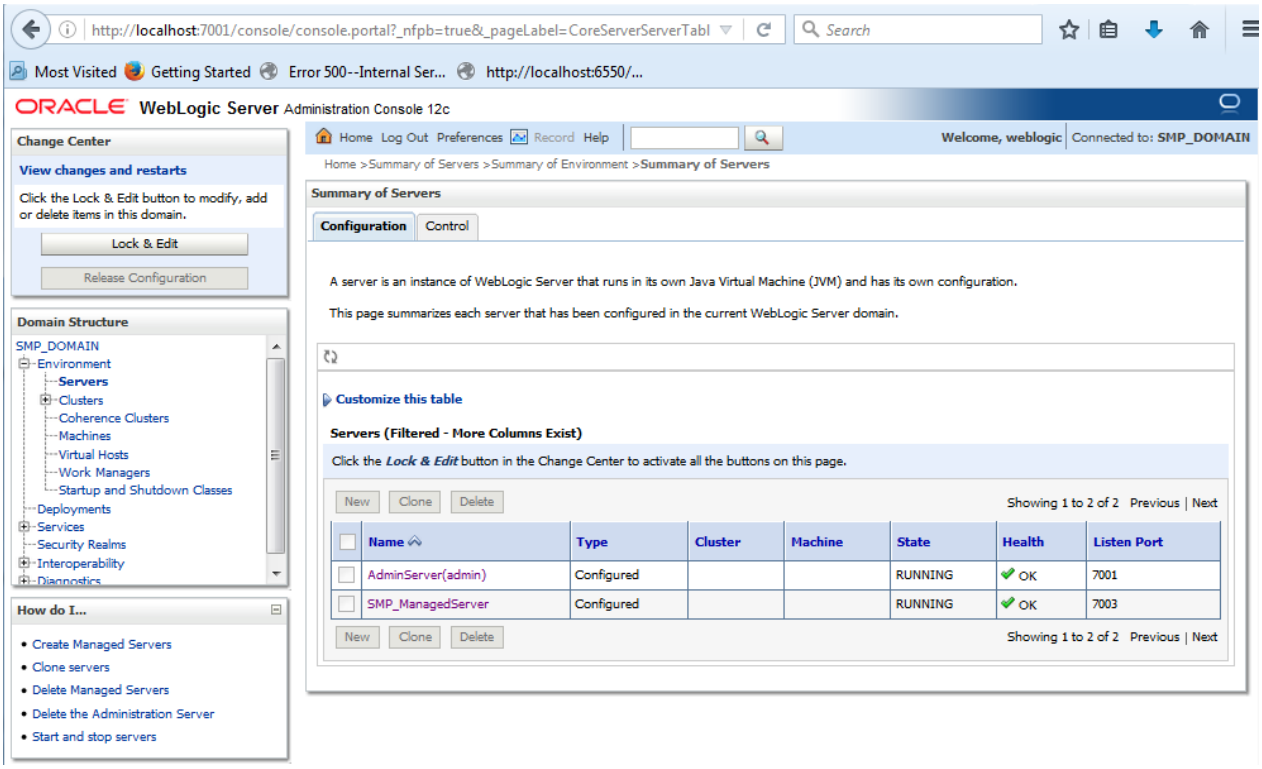
This section does not include the installation of a WebLogic application server. It is assumed that the WebLogic Server is installed and a WebLogic domain is created with an administration server and a managed server on which the CEF SMP will be deployed.

Hereafter the domain location will be referred as *DOMAIN_HOME* (user-defined name).

In the examples below, we will use the following Domain and Server names:

- Domain Name : SMPDOMAIN
- Administration Server : AdminServer
- SMP Managed Server : SMP_ManagedServer

As shown below:



In order to deploy the SMP on the WebLogic Application Server platform, two preliminary steps need to be completed:

- Disabling the Authentication on the Weblogic Server,
- Configuring the Extra CLASSPATH for WebLogic

This is described in the following 2 sections.

6.1. Disabling the Authentication on the WebLogic

The CEF SMP has its own authentication mechanism which makes the WebLogic authentication redundant. Therefore the WebLogic Authentication has to be disabled to stop it from interfering with the SMP authentication.

To do so, edit the config.xml file by adding the following tag before the `</security-configuration>` closing tag:

```
<enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credentials>
```

Here is an example:

```
../
<enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credentials>
</security-configuration>
/..
```

6.2. Configuring the Extra CLASSPATH for WebLogic

In this Oracle WebLogic example, a directory called **smp** will be created in the root path of the WebLogic installation (DOMAIN_HOME) and the CLASSPATH modified to include this new directory.

Create an smp directory in the DOMAIN_HOME directory.

Within the cef_edelivery_path/smp directory, create the following sub-directories:

- conf
- keystores
- temp
- logs

Edit the WebLogic DOMAIN_HOME/bin/setDomainEnv.sh.

For Linux:

Add the **EXPORT CLASSPATH=${CLASSPATH}:${DOMAIN_HOME}/smp/conf** statement at the end of the CLASSPATH definition as shown below:

```
../
if [ "${PRE_CLASSPATH}" != "" ] ; then
        CLASSPATH="${PRE_CLASSPATH}${CLASSPATHSEP}${CLASSPATH}"
        export CLASSPATH
```

```
fi
        CLASSPATH=${CLASSPATH}:${DOMAIN_HOME}/smp/conf
        export CLASSPATH
/..
```

For Windows:

```
../
If NOT "%PRE_CLASSPATH%"=="" (
      set CLASSPATH=%PRE_CLASSPATH%;%CLASSPATH%
)
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\smp\conf
/..
```

7. TOMCAT CONFIGURATION

In order to deploy the SMP on Tomcat, the steps below need to be completed.

7.1. Configuring the Extra CLASSPATH for Tomcat

In this Tomcat example, a directory called **smp** will be created in the root path of the Tomcat installation (**CATALINA_HOME**) and the **CLASSPATH** modified to include this new directory using an existing Tomcat batch file (CATALINA_HOME/bin/setenv.[sh|bat]).

Create a **smp** directory in the **CATALINA_HOME** directory.

Within the **cef_edelivery_path/smp** directory, create two new sub-directories:

- conf
- keystores

For Linux:

Edit the CATALINA_HOME/bin/setenv.sh file

```
#!/bin/sh
# Set CLASSPATH to include $CATALINA_HOME/smp/conf
# where the smp ‘smp.config.properties’ is located
export CATALINA_HOME=/cef_edelivery_path/smp
export JAVA_OPTS="$JAVA_OPTS -Dsmp.config.properties=$CATALINA_HOME/conf"
export CLASSPATH=$CATALINA_HOME/conf
```

For Windows:

Edit the %CATALINA_HOME%/bin/setenv.bat file

```
REM Set CLASSPATH to include $CATALINA_HOME/smp/conf
REM where the ‘smp.config.properties’ is located
set CATALINA_HOME=\cef_edelivery_path\smp
set JAVA_OPTS=%JAVA_OPTS% -Dsmp.config.properties=%CATALINA_HOME%\conf
set classpath=%classpath%;%catalina_home%\conf
```

7.2. JDBC Driver

The JDBC driver needs to be downloaded from the manufacturer website.

- For Oracle Database : http://www.oracle.com/technetwork/apps-tech/jdbc-112010-090769.html
- For Mysql : https://www.mysql.com/products/connector/

The JDBC driver (.jar file) must be copied to the following directory: cef_edelivery_path/lib.

8. SMP CONFIGURATION

For this step, use the following resources delivered within the zip file downloaded in section §3.3:

- An smp config file named **smp.config.properties** located in the CLASSPATH
- A keystore file called **sample_signatures_keystore.jks** used for XMLDSIG

The **smp.config.properties** file must be copied to the ***cef_edelivery/smp/conf/*** folder and referenced by the **CLASSPATH** variable as explained in the next section.

The **sample_signatures_keystore.jks** file must be copied to the **keystores** folder under ***cef_edelivery/smp/conf/*** created during the Server preparation (see $7.1).

8.1. Database configuration

The CEF SMP database back-end configuration is performed within the CEF SMP configuration file (**smp.config.properties** file).

Depending on the selected database back-end, modify the **smp.config.properties** files as indicated below.

8.1.1. *Oracle Database:*

```
../
## JDBC configuration for DB Oracle
jdbc.driver = oracle.jdbc.OracleDriver
jdbc.url = jdbc:oracle:thin:@dbhost:dbport:smp_database
jdbc.user = smp_user
jdbc.password = smp_user_pwd
target-database = Oracle
jdbc.read-connections.max = 10
/..
```

8.1.2. *MySQL:*

```
../
## JDBC configuration for DB MySQL
# JDBC configuration for DB MySQL
jdbc.driver = com.mysql.jdbc.Driver
jdbc.url = jdbc:mysql://dbhost:dbport/smp_database
jdbc.user = smp_user
jdbc.password = smp_user_pwd
target-database = MySQL
```

```
jdbc.read-connections.max = 10
/..
```

8.2. SMP Keystores

CEF SMP uses two distinct keystores for different purposes:

- One **mandatory** keystore is used for signing the responses to **GET** requests (XMLDSIG response signing)
- One **optional** (SML) keystore is used to authenticate SMP using 2-way-SSL when it is calling SML via HTTPS

8.2.1. *XMLDSIG response signing Keystore*

This keystore is **mandatory** and will prevent the CEF SMP from starting if not deployed and/or properly configured.

A sample keystore is included in the zip file downloaded in section §3.3.

Details of the sample keystore:

- xmldsig.keystore.classpath: /{path}/**sample_signatures_keystore.jks**
- *xmldsig.keystore.password*: **secure123.**

Remark:

*The value of the **xmldsig.keystore.password** (=**secure123**) also acts as the password for the **bdmslClientCertAlias** field in the **smp_domain** table described later.*

This keystore can be copied to ./smp/keystores/ or any other chosen directory of the SMP server, and then configured in the **smp.config.properties** file as shown in the following example:

```
../
## XMLDSIG response signing:
xmldsig.keystore.classpath    = ./smp/keystores/sample_signature_keys.jks
xmldsig.keystore.password     = secure123
/..
```

8.2.2. *SML Keystore*

The SML keystore is **optional** and only needs to be deployed and configured if the SMP will be accessing an SML using HTTPS.

The SML keystore should be deployed in a server location and configured in the **smp.config.properties** using a relative or an absolute location.

```
../
```

```
# BDMSL URL
bdmsl.integration.url=http://localhost:8080/manageparticipantidentifier


## Keystore JKS file path (optional). Used for X509 client 2-way-SSL
authentication in SML integration
## If there is more than one domain configured, alias of each key must be set up
in SMP_DOMAIN table
## Password must be equal for every key within the keystore and for keystore
itself
bdmsl.integration.keystore.path=
bdmsl.integration.keystore.password=
```

9. SMP .WAR FILE DEPLOYMENT

The CEF SMP is deployed following the steps listed below.

9.1.1. *Tomcat*

Copy the cef_edelivery/smp/temp/ smp-4.X.war file to the Tomcat **webapps** directory (cef_edelivery/webapps).

9.1.2. *Oracle WebLogic*

Deploy the **.war** file within WebLogic using the Oracle Weblogic deployer or using the Weblogic Administration Console.

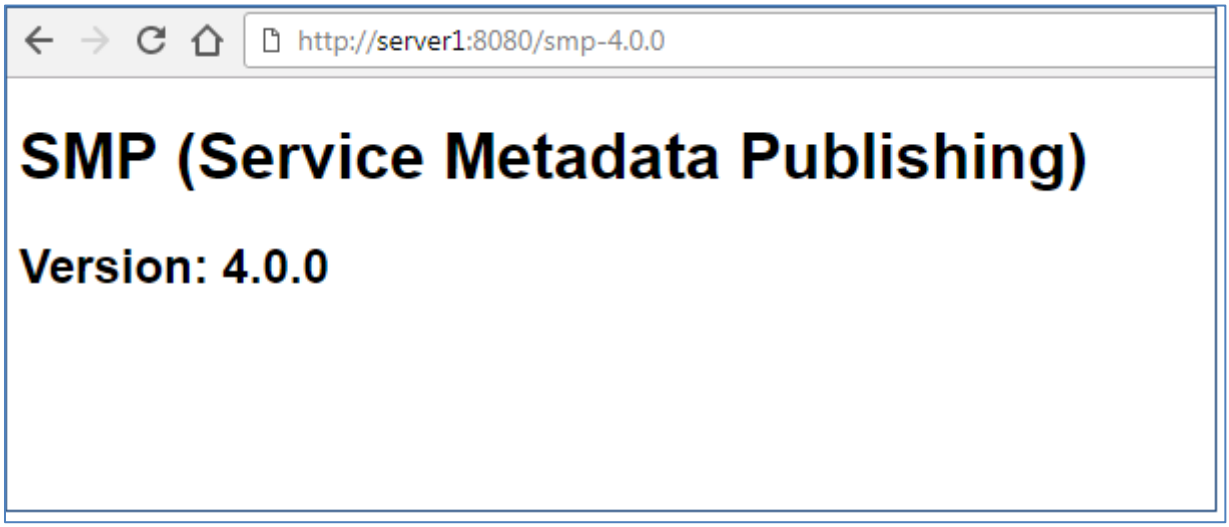
An example of using the Oracle the **weblogicc.deployer** is provided below:

```
java weblogic.Deployer -adminurl
t3://${WebLogicAdminServerListenAddress}:${WebLogicAdminServerPort} \
-username ${WebLogicAdminUserName} \
-password ${WebLogicAdminUserPassword} \
-deploy -name smp-4.X.war \
-targets ${SMP_ManagedServer} \
-source $TEMP_DIR/ smp-4.X.war
```

9.1.3. *Verification of the Installation*

Verify the installation by navigating with your browser to the following address: **http://*localhost*:*7003*/ smp-4.X**

The following page is displayed only if the deployment is successful:



10. CONFIGURING THE CEF SMP FOR USE WITH AN SML

The SMP can be registered to an SML using two identification mechanisms:

- Using HTTP and plain text with metadata embedded into the header of the REST request,
- Using HTTPS/TLS and a keystore containing a certificate.

10.1. Defining the SMP ID

The name of the SMP instance can be freely chosen but it is not currently possible to know whether the name chosen is free for registration or not.

The name of the SMP must be in the following format:

SMP-XXX-YYY or **XXX-SMP-YYY**

Where:

- XXX and YYY can be any user-defined word.
- No underscore character is allowed.
- It is also RECOMMENDED but not mandatory to use only uppercase characters.

The configuration of the SMP ID is achieved by the inserting the SMP ID into the **bdmslSmpId** field of the **smp_domain** table.

10.2. Configuring the BDMSL Integration

The first step is to configure the SMP so that it could be used with an SML.

This is achieved by setting the **bdmsl.integration.enabled** parameter to **true**, default is **false**.

```
## BDMSL (SML) integration ON/OFF switch
bdmsl.integration.enabled=false
```

10.3. Configuring the SML URL

The configuration of the SML URL end point is achieved by configuring the **bdmsl.integration.url** property in the **smp.config.property** file as follows:

```
## BDMSL URL
bdmsl.integration.url=http://localhost:8080/manageparticipantidentifier
```

10.4. SMP authentication to an SML

Once registered in an SML, the SMP needs to authenticate against the SML during normal operation.

This can be achieved by using plain text HTTP or HTTPS/TLS.

10.4.1. *Plain Text HTTP*

bdmslClientCertHeader contains the SMP's certificate needed if accessing BDMSL directly through HTTP. The configured "Client-Cert" HTTP header will be added to each BDMSL request bypassing SSL certificate verification made normally by SSL terminators.

bdmslClientCertHeader is the configuration attribute (column) in the **smp_domain** table described in section §16.3.

10.4.2. *HTTPS/TLS*

When using HTTPS/TLS, as mentioned in the **bdmsl.integration.url** covered earlier, all information related to the Keystore containing the SMP certificate must be configured under in the **smp.config.property** file:

bdmsl.integration.keystore.path=
bdmsl.integration.keystore.password=

REMARK

- **The Keystore, by default, must contain one key-pair that will be used for authentication.**
    - The Keystore JKS file path (optional): used for X509 client 2-way-SSL authentication in SML integration.
    - Note that in this case, the **bdmslClientCertHeader**, described earlier, must be set to **Blank**.
- **For more complex deployments (multiple domains), a link to the relevant section must be provided.**
    - *The **alias** of each key must be set up in **SMP_DOMAIN table.***
    - *The **Password** must be the same for every key within the Keystore and for the Keystore itself.*

11. SMP User Management

Only **Admin SMP** and Admin **ServiceGroup** users, who connect to the CEF SMP, need to be created in the SMP database.

Anonymous users or public users can access the SMP to retrieve only. They are not registered and therefore not added to the database.

There are no restrictions on the number of users that can be created to access the CEF SMP.

The (MYSQL or ORACLE) Database script provided creates an **smp_admin** user with password=**changeit**, by default. This password must be immediately changed for the purpose of using the **SMP** in **Acceptance** or **Production**.

11.1. User Roles

The CEF SMP users can be of three types as briefly described below:

Actor	UC	Short description	Oper.	Data
Admin SMP	Create or Update Service Group	Create a new ServiceGroup for a new receiver participant. This service stores the Service Group and links it to the specified duplet participantIdentifier + participantIndentifierScheme. Information is stored into ServiceGroup table. This same service is used to create and update a ServiceGroup.	PUT	ServiceGroup
Admin SMP	Erase Service Group	Erases the service group definition AND the list of services for the specified receiver participant.	DELETE	ServiceGroup
Admin Service Group	Create or Update Service Metadata	Publish detailed information about one specific document service (multiple processes and endpoints). This same service is used to create and update ServiceMetaData.	PUT	ServiceMetadata
Admin Service Group	Erase Service Metadata	Remove all information about one specific service (i.e. all related processes and endpoints definitions).	DELETE	ServiceMetadata
User	Retrieve Service Group	Obtain the list of services provided by a specific receiver participant (collection of references to the ServiceMetaData's). This service provides the information related to the Service Group according to the input duplet participantIdentifier + participantIndentifierScheme. Returns information	GET	ServiceGroup

Actor	UC	Short description	Oper.	Data
		from the ServiceMetadata table only (references to actual MetaData).		
User	Retrieve Service Metadata	Obtain detailed definition about one specific service of a specific participant for all supported transports. This service retrieves the SignedServiceMetadata according to the input quadruplet participantIdentifier+participantIndentifierScheme+ documentIdentifier+documentIdentifierScheme. Returns information from the Endpoint table.	GET	SignedServiceMetadata

Note: For a complete description of the SMP user management, please consult the SMP Interface Control Document (ICD) document available at: https://ec.europa.eu/cefdigital/wiki/x/0wvNAg.

11.2. BCRYPT password generation

The SMP v4.X uses the BCRYPT algorithm to hash users' passwords. A BCRYPT-hashing tool is bundled into the SMP WAR file. To get the hashing code, follow the steps below.

Place a copy of the **smp-4.X.war** file into a temporary directory of your choice.

Extract the war file using the **jar** command:

```
jar -xvf smp-4.X.war
```

Obtain one or multiple hashes at once, using the following command:

```
java -cp "WEB-INF/lib/*" eu.europa.ec.edelivery.smp.BCryptPasswordHash
password_to_be_hashed
```

The result will be a BCRYPT hash of the specified password (listed below in italic):

```
java -cp "WEB-INF/lib/*" eu.europa.ec.edelivery.smp.BCryptPasswordHash
password_to_be_hashed
$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
```

The next command shows the hashing of several passwords at once, separated by a space in the command.

```
java -cp "WEB-INF/lib/*" eu.europa.ec.edelivery.smp.BCryptPasswordHash
password_to_be_hashed_1 password_to_be_hashed_2
$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa
$2a$107zNzSeZpxiHeqY2BRKkHE.HknfIe3aiu6XzU.qHHnnPbUHKtfcmDG
```

11.3. SMP Database User Creation

Adding an SMP user is performed by adding a new entry in the SMP database (**SMP_USER** table).

User role is determined by setting the **isadmin** field in the SMP_USER table as follows:

User Role	**isadmin value**
Admin SMP	1
Admin Service Group	0
AnonymousUser (Not defined in the SMP User database)	N/A

In the following two examples, an **Admin SMP** and **Admin ServiceGroup** users are created.

11.3.1. *Admin SMP User creation*

Example:

Username : smp_admin

Password (Hashed) : $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa

IsAdmin : 1:

Execute the following database command using the database user/password created in the Database Configuration section of this guide.

```
SQL> INSERT into SMP_USER (USERNAME, PASSWORD, ISADMIN) values ('smp_user',
'$2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa', 1);
```

11.3.2. *Admin ServiceGroup User Creation*

Username : smp_user1

Password (Hashed) : $2a$10$6nYTSUSh2BQfbOLIyCXn8eUViBcnn.WcjUrWOtJlMNDOdAtI85zMa

IsAdmin : 0

Execute the following database command.

```
SQL> insert into smp_user (username, password, isadmin) values ('smp_user1',
'$2a$10$6nytsush2bqfboliycxn8euvibcnn.wcjurwotjlmndodati85zma', 0);
```

Example: Admin **Servicegroup** authenticated with **certificate**:

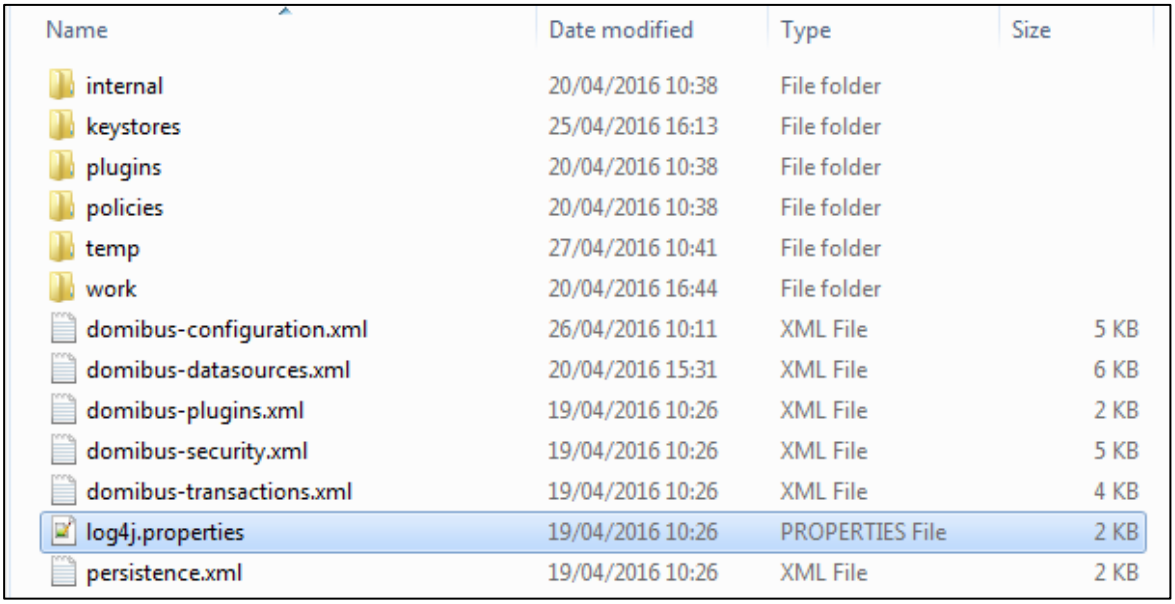
INSERT into SMP_USER (USERNAME, PASSWORD, ISADMIN) values
('CN=EHEALTH_AP_TEST_00000,O=European Commission,C=BE:23b207cb468b8519', '', 0);

12. LOGGING CONFIGURATION

12.1. Logging properties

The SMP logging properties are defined in the ***./WEBINF*/log4j.properties file** embedded in the SMP **.war** file.

It is possible to modify the configuration of the logs by editing the embedded **log4j properties**.



Name	Date modified	Type	Size
internal	20/04/2016 10:38	File folder	
keystores	25/04/2016 16:13	File folder	
plugins	20/04/2016 10:38	File folder	
policies	20/04/2016 10:38	File folder	
temp	27/04/2016 10:41	File folder	
work	20/04/2016 16:44	File folder	
domibus-configuration.xml	26/04/2016 10:11	XML File	5 KB
domibus-datasources.xml	20/04/2016 15:31	XML File	6 KB
domibus-plugins.xml	19/04/2016 10:26	XML File	2 KB
domibus-security.xml	19/04/2016 10:26	XML File	5 KB
domibus-transactions.xml	19/04/2016 10:26	XML File	4 KB
log4j.properties	19/04/2016 10:26	PROPERTIES File	2 KB
persistence.xml	19/04/2016 10:26	XML File	2 KB

In the example below, a **log4j.properties** file is shown:

```
# Direct log messages to stdout
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n

log4j.appender.file=org.apache.log4j.FileAppender
log4j.appender.file.file=${catalina.home}/logs/domibus.log
log4j.appender.file.layout=org.apache.log4j.PatternLayout
log4j.appender.file.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n


log4j.appender.atomikos=org.apache.log4j.FileAppender
log4j.appender.atomikos.file=${catalina.home}/logs/atomikos.log
log4j.appender.atomikos.layout=org.apache.log4j.PatternLayout
log4j.appender.atomikos.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L - %m%n

# In order to enable logging of request/responses please change the loglevel to INFO
log4j.logger.org.apache.cxf=WARN
# Root logger option
log4j.rootLogger=INFO, file, stdout

log4j.logger.com.atomikos=WARN, atomikos
```

In red: the parameters can be edited to modify the location of the log file, and the layout.

In green: the parameters can be edited to change the level of logging (3 levels definied: INFO, WARN, and ERROR).

13. SOAPUI TESTING

Soap UI can be used to create, update and delete Service Groups and Metadata.

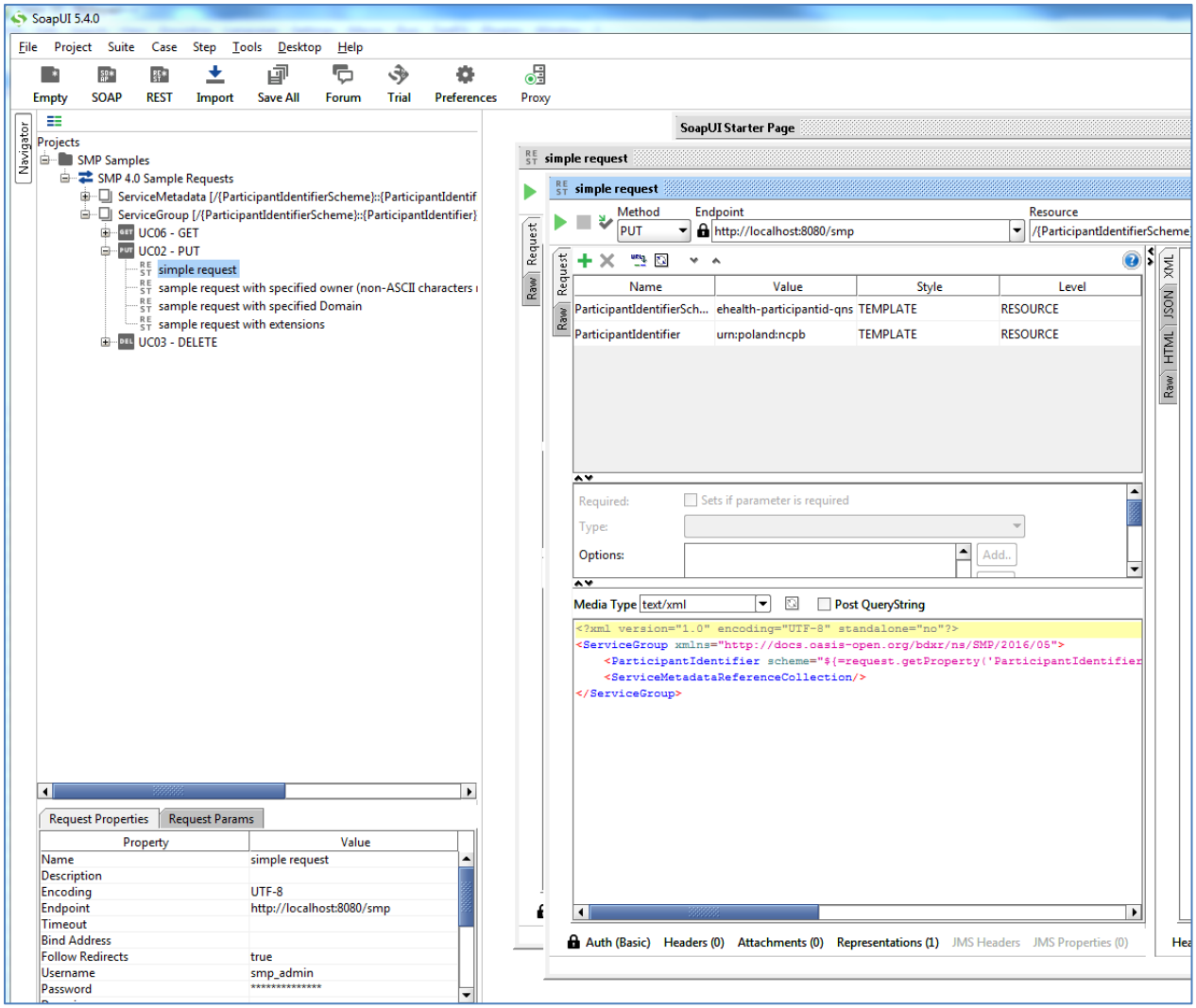
An SMP SoapUI project contains sample requests and is included in the zip file already downloaded.

The procedure to create, update or delete a Service Group is described in the next steps.

13.1. Creation, update and deletion of Service Groups.

13.1.1. *Create a Service Group*

In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:

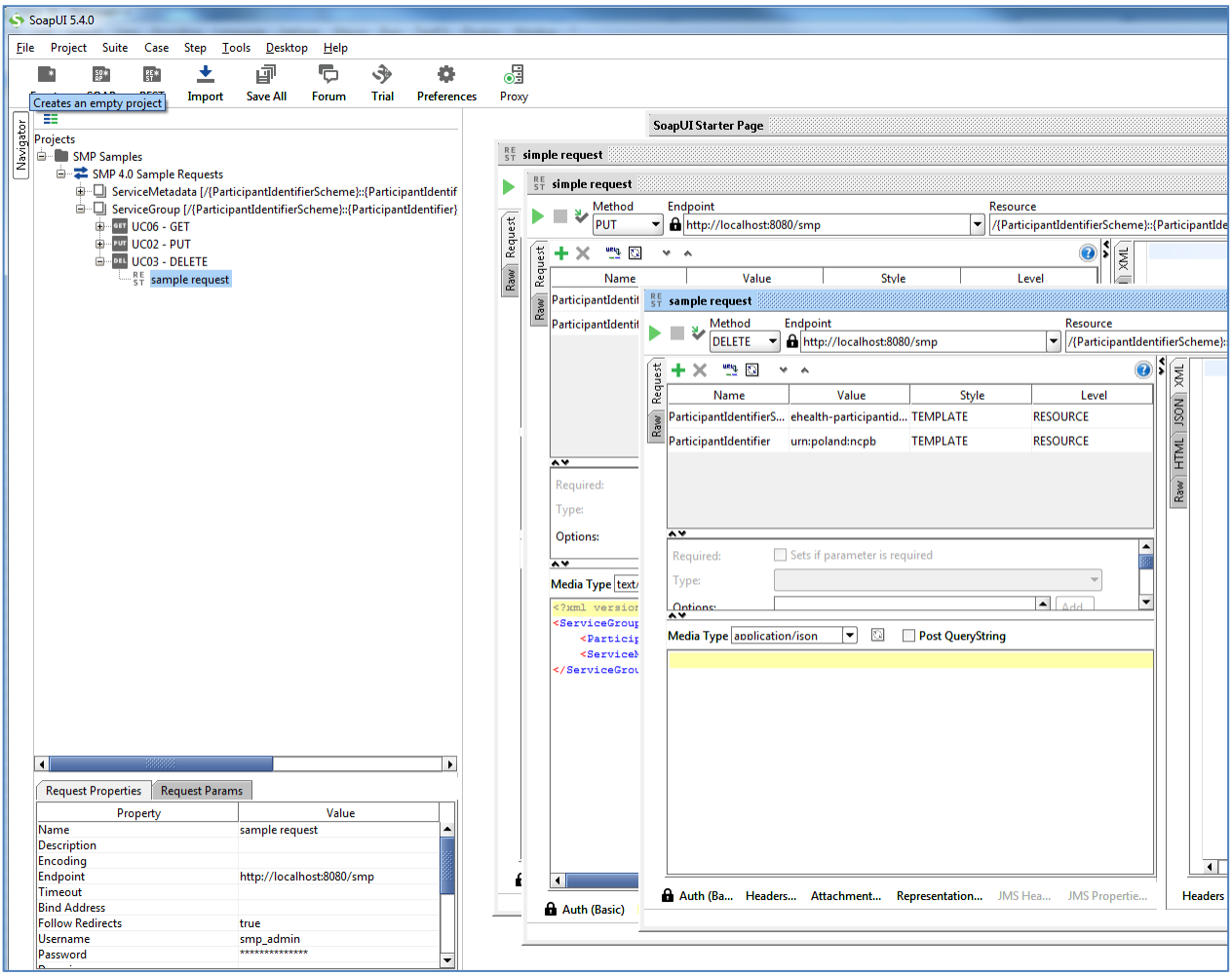


13.1.2. *Update a Service Group*

The REST method to update the **ServiceGroup** is the same as the one used for creating **ServiceGroup** described in the previous section.

13.1.3. *Delete a ServiceGroup*

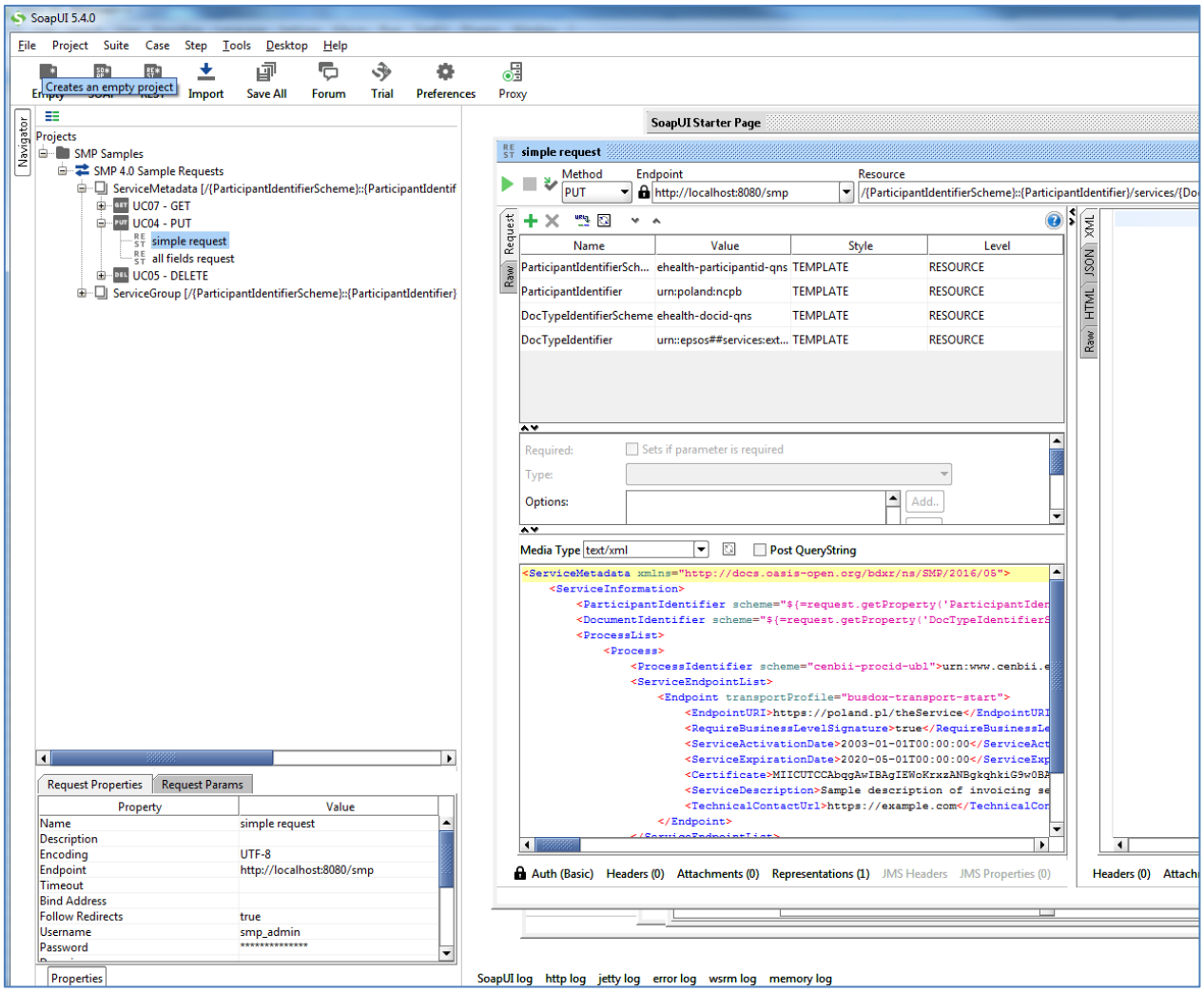
On the SoapUI interface on the left navigation panel, browse to the REST DELETE method as indicated below:



13.2. Creation, update and deletion of Service Metadata.

13.2.1. *Create a Service Metadata*

In the left navigation pane of the SoapUI interface, browse to the REST PUT method as shown below:

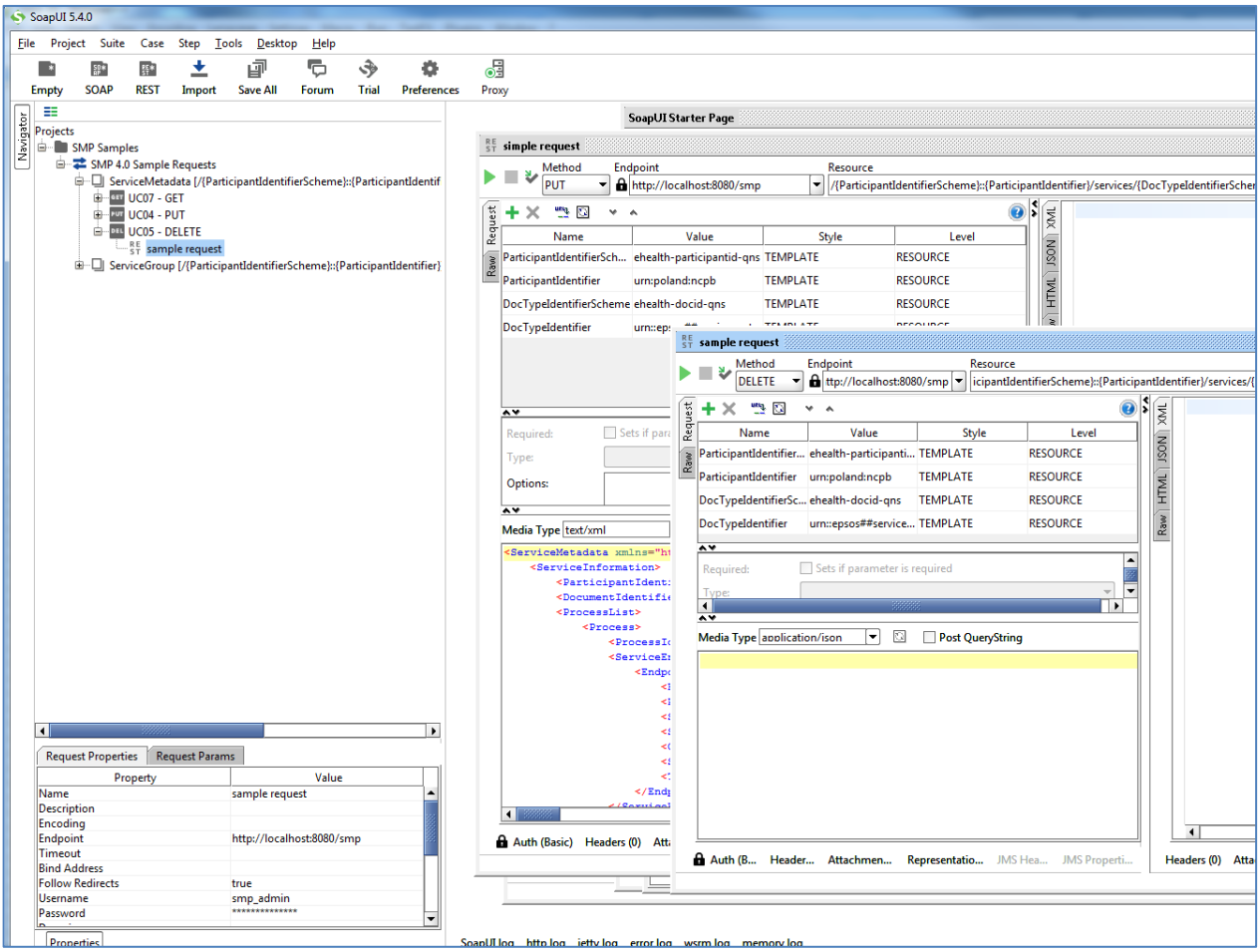


13.2.2. *Update Service Metadata*

The REST method to update **ServiceMetadata** is the same as the one use for creating **ServiceMetadata** as described in the previous section.

13.2.3. *Delete Service Metadata*

In the left navigation pane of the SoapUI interface, browse to the **REST DELETE** method as indicated below:


SoapUI 5.4.0
File Project Suite Case Step Tools Desktop Help
Empty SOAP REST Import Save All Forum Trial Preferences Proxy
Navigator
Projects
SMP Samples
SMP 4.0 Sample Requests
ServiceMetadata [/{ParticipantIdentifierScheme}::{ParticipantIdentif
UC07 - GET
UC04 - PUT
UC05 - DELETE
sample request
ServiceGroup [/{ParticipantIdentifierScheme}::{ParticipantIdentifier}
SoapUI Starter Page
simple request
Method
Endpoint
Resource
PUT
http://localhost:8080/smp
/{ParticipantIdentifierScheme}::{ParticipantIdentifier}/services/{DocTypeIdentifierSche
Request
Raw
XML
JSON
HTML
Name
Value
Style
Level
ParticipantIdentifierSch...
ehealth-participantid-qns
TEMPLATE
RESOURCE
ParticipantIdentifier
urn:poland:ncpb
TEMPLATE
RESOURCE
DocTypeIdentifierScheme
ehealth-docid-qns
TEMPLATE
RESOURCE
DocTypeIdentifier
urn::ep
sample request
Method
Endpoint
Resource
DELETE
ttp://localhost:8080/smp
icipantIdentifierScheme}::{ParticipantIdentifier}/services/{
Name
Value
Style
Level
ParticipantIdentifier...
ehealth-participanti...
TEMPLATE
RESOURCE
ParticipantIdentifier
urn:poland:ncpb
TEMPLATE
RESOURCE
DocTypeIdentifierSc...
ehealth-docid-qns
TEMPLATE
RESOURCE
DocTypeIdentifier
urn::epsos##service...
TEMPLATE
RESOURCE
Required:
Sets if parameter is required
Type:
Options:
Media Type
text/xml
application/ison
Post QueryString
Request Properties
Request Params
Property
Value
Name
sample request
Description
Encoding
Endpoint
http://localhost:8080/smp
Timeout
Bind Address
Follow Redirects
true
Username
smp_admin
Password
Properties
Auth (Basic)
Headers (0)
Auth (B...
Header...
Attachmen...
Representatio...
JMS Hea...
JMS Properti...
Headers (0)
SoapUI.log
http.log
jetty.log
error.log
wsrm.log
memory.log

14. THE SWAGGERUI INTERFACE

14.1. Introduction

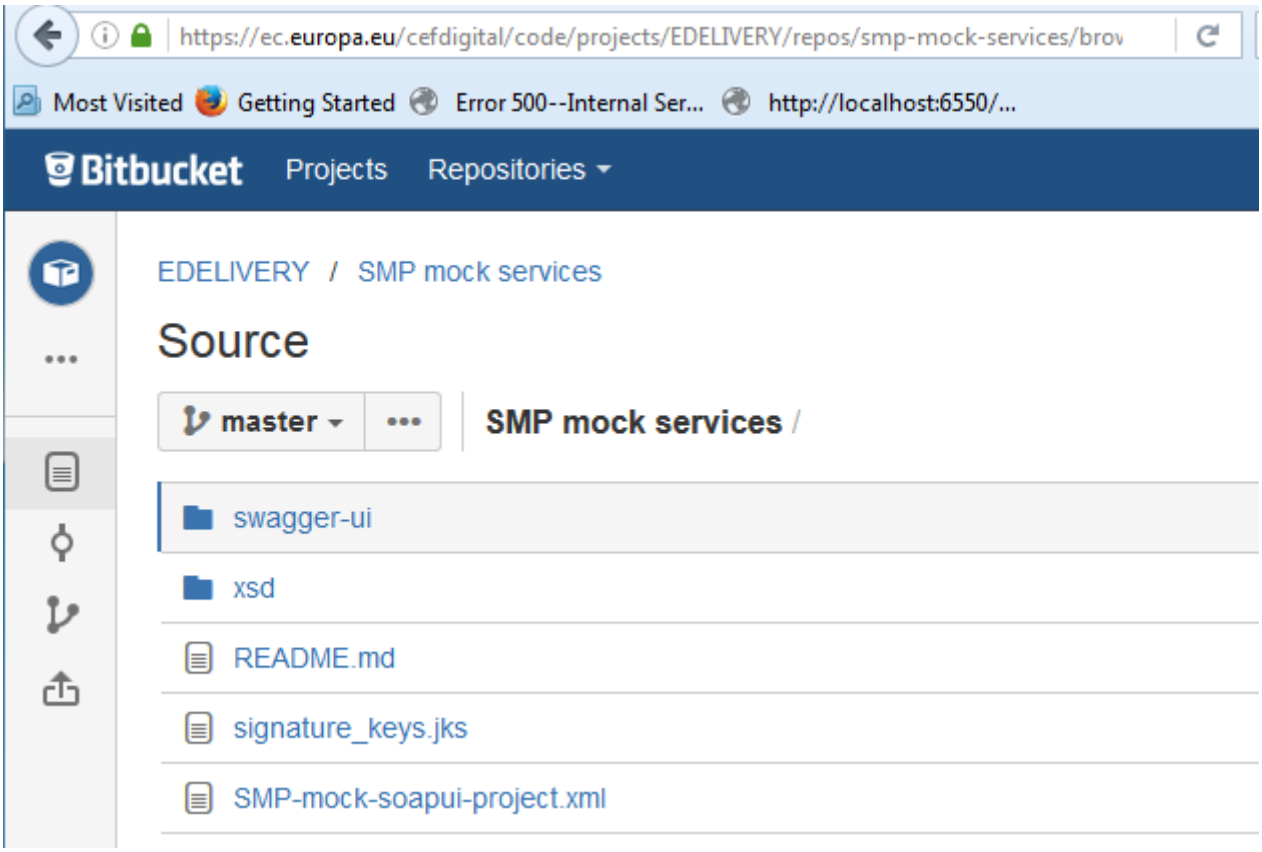
"***Swagger is an API developer tools for the OpenAPI Specification (OAS). It allows anyone (developers or end-users) to interact with the API's resources***"[2].

The SMP Web Client can be tested at: http://130.206.118.4/smp-swagger-ui and, as explained, is a WEB client configured to shoot (PUT, GET or DELETE) at the mocked SMP implementation Metadata.

14.2. Downloading the CEF SMP SwaggerUI web application project

The CEF SMP SwaggerUI web application project can be freely downloaded from the following location:

https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services/browse



Create a new **swagger_temp** temporary directory.

Within the previously created **swagger_temp** directory, execute the following command:

[2] Quote from: http://swagger.io/.

```
git clone https://ec.europa.eu/cefdigital/code/scm/edelivery/smp-mock-services.git
Cloning into 'smp-mock-services'...
remote: Counting objects: 133, done.
remote: Compressing objects: 100% (130/130), done.
remote: Total 133 (delta 50), reused 0 (delta 0)
Receiving objects: 100% (133/133), 823.54 KiB | 0 bytes/s, done.
Resolving deltas: 100% (50/50), Done.
```

The SMP **SwaggerUI** project is downloaded and saved the **smp-mock-services** directory:

```
ls
smp-mock-services
```

14.3. Configuring the SMP SwaggerUI

Navigate to the **swagger-ui** directory located under the **smp-mock-services** directory.

The contents is listed below:

```
ls
css  fonts  images  index.html  lib  smp.json  swagger-ui.js
```

Edit the **smp.json** file and modify it to target your SMP:

Replace:

```
{
  "swagger": "2.0",
  "info": {
    "description": "This WEB client is configured to shoot at the [mocked SMP implementation](http://smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu/ehealth-actorid-qns%3A%3Aurn%3Apoland%3Ancpb). After a few improvements (both on client and server side) it might be used also for shooting at TEST / PROD environments. You can find out more about Swagger at [http://swagger.io](http://swagger.io)",
    "version": "1.0.0",
    "title": "SMP 3.X WEB client (based on Swagger-UI)"
  },
  "host": "smp-digit-mock.publisher.ehealth.acc.edelivery.tech.ec.europa.eu",
  "basePath": "/",
  "externalDocs": {
    "description": "Find out more about SMP 3.X mock services",
```

With:

```
    "url": "https://ec.europa.eu/cefdigital/code/projects/EDELIVERY/repos/smp-mock-services"
{
  "swagger": "2.0",
  "info": {
    "description": "This WEB client is configured to shoot at [http://localhost:7003/ smp-4.X](http://localhost:7003/ smp-4.X). After a few improvements (both on client and server side) it might be used also for shooting at TEST / PROD environments. You can find out more about Swagger at [http://swagger.io](http://swagger.io)",
    "version": "1.0.0",
    "title": "SMP 4.X WEB client (based on Swagger-UI)"
  },
  "host": "localhost:7003",
  "basePath": "/ smp-4.X",
  "externalDocs": {
  },
```

14.4. Generating the Web Application Archive (.war file)

To generate the CEF SMP SwaggerUI Web Application archive (**.war** file), jus create a zip file of the content of the swagger-ui directory and rename it as **swagger.war**.

This can be performed using any **zip** utility (**winzip** on Windows or **zip** on Linux).

Example on Linux:

```
zip -r swagger.war swagger-ui/*
```

14.5. Deploy the SMP SwaggerUI war file

14.5.1. *On Tomcat*

Copy the **swagger.war** file to *cef_edelivery_path* **/webapps.**

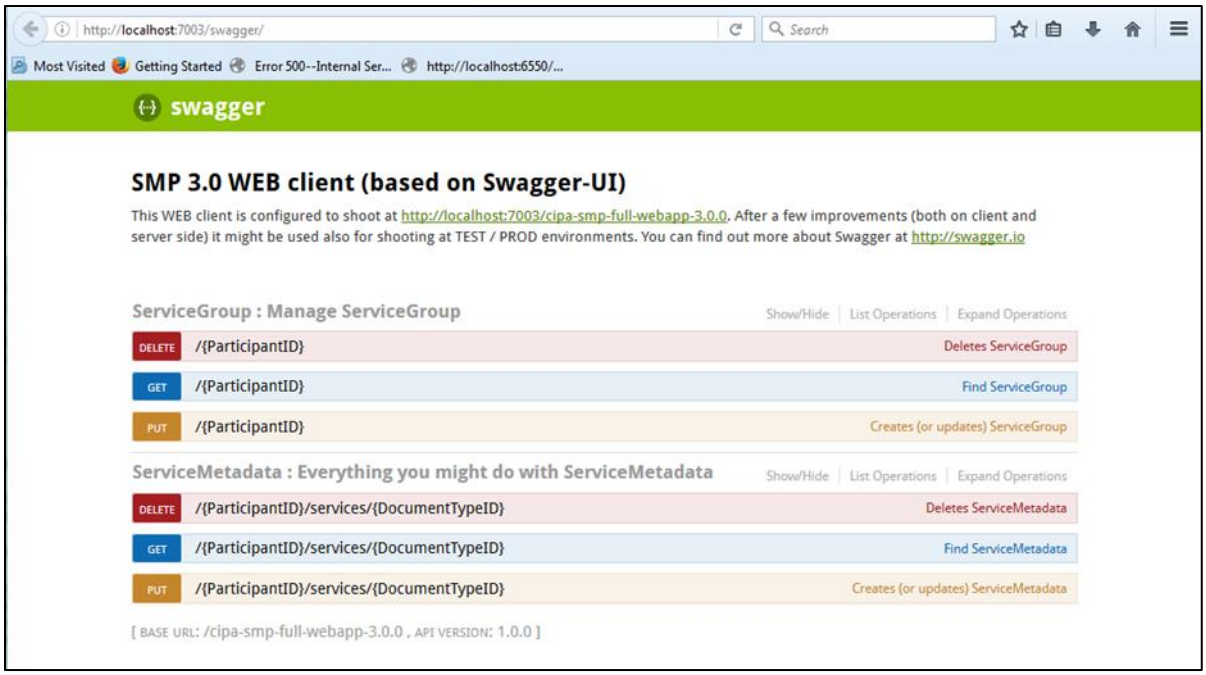
14.5.2. *On WebLogic:*

Deploy the **.war** file within WebLogic:

```
java weblogic.Deployer -adminurl
t3://${WebLogicAdminServerListenAddress}:${WebLogicAdminServerPort} \
-username ${WebLogicAdminUserName} \
-password ${WebLogicAdminUserPassword} \
-deploy -name swagger.war \
-targets ${SMP_ManagedServer} \
```

After starting the application, connect to http://locxalhost:7003/swagger.

A successful deployment should display the following page:



15. SMP COMPILATION

15.1. Compilation prerequisites

15.1.1. *Supported Operating System Platform*

CEF SMP can be built on the following OS platforms:

- Windows Workstation & Server
- Linux platform

15.1.2. *Software Requirements*

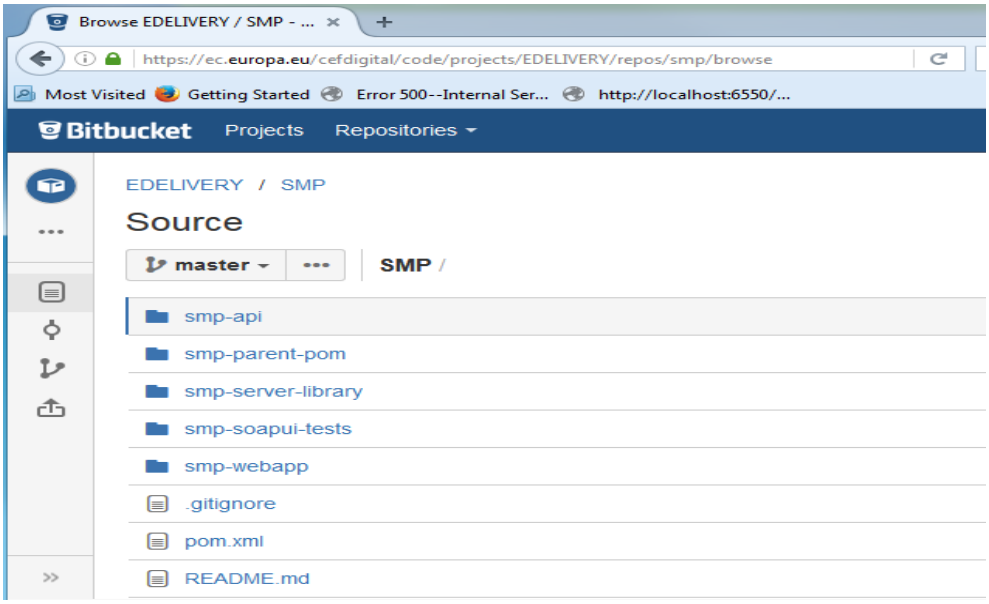
The following software components on the target system:

- Java Development Kit environment (JDK), version 7 or 8: http://www.oracle.com/technetwork/java/javase/downloads/index.html
- Maven 3.0 and above (https://maven.apache.org/download.cgi)
- GIT (optional: Git is only used to download the project sources but these sources can be downloaded from any system having Git installed and then just copied manually on the compilation platform).

15.2. Downloading the source code

The source code of SMP is freely available and can be downloaded from the following location:

https://ec.europa.eu/cefdigital/code/scm/edelivery/smp.git



15.3. Compilation

Create a new **comp_dir** temporary directory.

Within the previously created **comp_dir** directory, execute the following command:

```
git clone  https://ec.europa.eu/cefdigital/code/scm/edelivery/smp.git
Cloning into 'smp'...
remote: Counting objects: 52788, done.
remote: Compressing objects: 100% (15640/15640), done.
remote: Total 52788 (delta 25293), reused 47993 (delta 23387)
Receiving objects: 100% (52788/52788), 637.14 MiB | 2.06 MiB/s, done.
Resolving deltas: 100% (25293/25293), done.
```

Go to the newly created **smp** directory.

The directory contains the following:

```
ls
pom.xml  README.md  smp-api  smp-parent-pom  smp-server-library  smp-soapui-
tests  smp-webapp
```

Start the compilation by executing the following command:

```
mvn clean install -DskipTests
```

A successful compilation will result with the following:

```
mvn clean install -DskipTests
[INFO] Scanning for projects…
/..
../
[INFO] Installing /home/smpcomp/smp/smp/pom.xml to
/home/smpcomp/.m2/repository/eu/europa/ec/smp/3.X/smp-3.X.pom
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Summary:
[INFO]
[INFO] smp-parent-pom ..................................... SUCCESS [  0.120 s]
[INFO] smp-api ............................................ SUCCESS [ 32.375 s]
[INFO] smp-server-library ................................. SUCCESS [02:01 min]
[INFO] smp-webapp ......................................... SUCCESS [ 23.314 s]
[INFO] SMP Builder POM .................................... SUCCESS [  2.222 s]
[INFO] ------------------------------------------------------------------------
```

```
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 03:00 min
[INFO] Finished at: 2017-06-08T11:35:27+02:00
[INFO] Final Memory: 61M/726M
[INFO] ------------------------------------------------------------------------
```

The resulting will be a Web application Archive (.war file) named **smp-4.X.war** located in the **smp-webapp/target/** directory:

```
ls ./smp-webapp/target
smp-4.X  smp-4.X.war classes  generated-sources  generated-test-sources  maven-status  test-classes  webapp-classes
```

16. SMP CONFIGURATION FILE AND TABLE

16.1. Multitenancy and Multidomain Support

The SMP is able to support multiple certificates in the same SMP. This is very useful in the Acceptance environment where multiple domains like ISA ITB, eHealth and others are hosted.

The SMP has the capability of keeping a relationship between a particular **Service Group** and its related **domain**.

As a result of this feature, the SMP Administration has the option, if need be, to define extra domains for newly created **Service Groups** meaning that the SMP is able to handle multiple domains environments.

Remark:

*In normal circumstances, when any one SMP is used for only one Domain, the domain used is then considered as the "domain by default" (or "default domain") for configuration purposes. The domain, in this case, does not need to be specified in the **Service Group** definitions or other configurations of the SMP as in previous versions of SMP.*

The SMP configuration is performed in 2 different locations: in the **smp.config.properties** file as well as in the **smp_domain** table. The following section describes the details of the parameters that are included in the configuration.

16.2. The smp.config.properties file

The CEF SMP configuration is performed via the **smp.config.properties** file.

This file is delivered by default embedded within the SMP war file.

```
#
# Copyright 2017 European Commission | CEF eDelivery
#
# Licensed under the EUPL, Version 1.2 or - as soon they will be approved
by the European Commission - subsequent versions of the EUPL (the
"Licence");
# You may not use this work except in compliance with the Licence.
#
# You may obtain a copy of the Licence attached in file: LICENCE-EUPL-
v1.2.pdf
#
# Unless required by applicable law or agreed to in writing, software
distributed under the Licence is distributed on an "AS IS" basis,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the Licence for the specific language governing permissions and
limitations under the Licence.
```

```
#


# Blue Coat reverse-proxy authentication, by default disabled (false)
# Be careful with switching it to 'true' - do it only if you know what you are doing.
# Authentication with Blue Coat means that all HTTP requests having 'Client-Cert' header will be authenticated
# as username placed in the header.
# Never expose SMP to the WEB without properly configured reverse-proxy and active blue coat.
authentication.blueCoat.enabled=false


## Only set to false in PRODUCTION mode. This variable is used to clear the context path of the SMP
contextPath.output=true


## Most Java libraries and J2EE containers block encoded slashes in URL - For security reasons.
## Theoretically there are no restrictions on slash "/" characters in document or participant identifiers,
## but by default we block then as well.
## If slash "/" characters must be supported, than switch this property to "true".
## Remember that in such case the relevant change should be also applied on J2EE level
## I.e. for Tomcat it is handled by property: org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true
encodedSlashesAllowedInUrl=false


## Participant Identifier Schema of each PUT ServiceGroup request is validated against this schema
## I.e this regex: ^(?!^.{26})([a-z0-9]+-[a-z0-9]+-[a-z0-9]+)
## - limits length of scheme to 25 characters
## - forces pattern to consist of 3 alpha-numeric segments delimited with "-", i.e: aa1-bb2-cc3
## To turn validation OFF, set regex to "match all":  .*
identifiersBehaviour.ParticipantIdentifierScheme.validationRegex=^(?!^.{26})([a-z0-9]+-[a-z0-9]+-[a-z0-9]+)


## All Identifiers by default are CASE-INSENSITIVE.
```

```
## Specifies schemes of participant/document identifiers that must be considered CASE-SENSITIVE.

## List values (delimited by pipe character: "|" ) placed here are checked against runtime (request) schemes in the CASE-INSENSITIVE way

identifiersBehaviour.caseSensitive.ParticipantIdentifierSchemes=casesensitive-participant-scheme1|casesensitive-participant-scheme2

identifiersBehaviour.caseSensitive.DocumentIdentifierSchemes=casesensitive-doc-scheme1|casesensitive-doc-scheme2


# Switches ON/OFF the BDXL client - integrationwith with SML

bdmsl.integration.enabled=false


# SML URL (incl. the service name)

#bdmsl.integration.url=https://sml.peppolcentral.org/manageparticipantidentifier

#
bdmsl.integration.url=https://smk.peppolcentral.org/manageparticipantidentifier

bdmsl.integration.url=http://localhost:8080/manageparticipantidentifier

bdmsl.integration.keystore.path=

bdmsl.integration.keystore.password=


bdmsl.integration.proxy.server=

bdmsl.integration.proxy.port=

bdmsl.integration.proxy.user=

bdmsl.integration.proxy.password=


## XMLDSIG response signing:

xmldsig.keystore.classpath    = ../keystore/keystore.jks

xmldsig.keystore.password     = peppol


## JDBC configuration for DB

jdbc.driver = oracle.jdbc.OracleDriver

jdbc.url =
jdbc:oracle:thin:@olrdev3.cc.cec.eu.int:1597/EX1UDIGD_TAF.cc.cec.eu.int

jdbc.user = CIPA_ADMIN

jdbc.password = the_password

target-database = Oracle

jdbc.read-connections.max = 10
```

16.2.1. *Detailed SMP configuration file (smp.config.properties)*

The **WEB-INF/classes/smp.config.properties** file is used to configure various SMP properties, the following table describes them briefly:

Parameter	Default Value	Comment
authentication.blueCoat.enabled	false	Authentication with Blue Coat means that all HTTP requests having **Client-Cert** header will be authenticated as username placed in the header. # Never expose an SMP to the WEB without properly configured reverse-proxy and active blue coat.
contextPath.output	true	PRODUCTION mode. This variable is used to clear the context path of the SMP
encodedSlashesAllowedInUrl	false	Most Java libraries and J2EE containers block encoded slashes in URL, for security reasons. ## Theoretically there are no restrictions on slash "/" characters in document or participant identifiers, ## but by default we block then as well. ## If slash "/" characters must be supported, than switch this property to "true". ## Remember that in such case the relevant change should be also applied on J2EE level ## I.e. for Tomcat it is handled by property: org.apache.tomcat.util.buf.UDecoder.ALLOW_ENCODED_SLASH=true
bdmsl.integration.enabled	false	BDMSL (SML) integration ON/OFF switch

Parameter	Default Value	Comment
identifiersBehaviour.caseSensitive.ParticipantIdentifierSchemes	casesensitive-participant-scheme1|casesensitive-participant-scheme2	## All Identifiers by default are CASE-INSENSITIVE. ## Specifies schemes of participant/document identifiers that must be considered CASE-SENSITIVE. ## List values (delimited by pipe character: "|") placed here are checked against runtime (request) schemes in the CASE-INSENSITIVE way
identifiersBehaviour.caseSensitive.DocumentIdentifierSchemes	casesensitive-doc-scheme1|casesensitive-doc-scheme2	
bdmsl.integration.url	#https://sml.peppolcentral.org/manageparticipantidentifier http://localhost:8080/manageparticipantidentifier	The URL of the targeted SML (incl. the service name)
bdmsl.integration.keystore.path		The location of the keystore
bdmsl.integration.keystore.password		The password of the keystore
xmldsig.keystore.classpath	../keystore/keystore.jks	The location of the xmldsig keystore
xmldsig.keystore.password	peppol	The password of the xmldsig keystore
jdbc.driver	com.mysql.jdbc.Driver	Database Configuration: Driver MySQL: com.mysql.jdbc.Driver Oracle Database: oracle.jdbc.OracleDriver
jdbc.url	jdbc:mysql://localhost:3306/smp	Database Configuration: url MySQL : jdbc:mysql://dbhost:dbport/smp_database Oracle Database: jdbc:oracle:thin:@dbhost:dbport:smp_database jdbc:oracle:thin:@dbhost:dbport/smp_service

Parameter	Default Value	Comment
jdbc.user	smp	Database User/Password Configuration: User
jdbc.password	The_password	Database User/password Configuration: Password
target-database	MySQL	Target Database Backend type/Brand: For MySQL, use: MySQL For Oracle Database, use: Oracle
jdbc.read-connections.max	10	Database Configuration: Max Read Connection

16.2.1.1. bdmsl.integration.url

Only used if SML/DNS integration has been selected.

In case the integration with the SML/DNS is needed, this value has to be set to the address of the SML or the loadBalancer/proxy tagetting these SML instance(s).

16.2.1.2. XMLDSIG response signing

xmldsig.keystore.classpath = ../keystore/keystore.jks

xmldsig.keystore.password = peppol

16.2.1.3. Proxy Settings

When an SMP calls the SML and the SML is behind a proxy, then standard proxy settings need to be specified using the following variables in the configuration files:

- **bdmsl.integration.proxy.server**
- **bdmsl.integration.proxy.port**
- **bdmsl.integration.proxy.user**
- **bdmsl.integration.proxy.password**

16.3. smp_domain table configuration

This table is used to support the multi-tenancy feature of the SMP. Its parameters/fields are:

- **bdmslSmpId**: This is the SMP ID that must match the SMP ID registered within the SML.
- **bdmslClientCertHeader:** The SMP's certificate - needed only when accessing BDMSL directly through HTTP. The configured "Client-Cert" HTTP header will be added to each BDMSL request (bypassing SSL certificate verification made normally by SSL terminator) .
- **bdmslClientCertAlias**: This is the Domain scoped alias of the keystore private key used for authentication with the SML. The password is the same as xmldsig.keystore.password defined in the SMP configuration file.
- **signatureCertAlias:** This field points to the **Domain scoped** alias of the Keystore private key certificate, used by the SMP to sign GET Signed Service Metadata responses.

16.3.1. ***Example: Update the default single domain smp_domain table record:***

update smp_domain **set** ***bdmslSmpId***='SMP-MCB-ID14', ***bdmslClientCertAlias*** = 'smp_mock';

or

update smp_domain **set** ***bdmslSmpId***='SMP-MCB-ID14', ***bdmslClientCertHeader***= 'serial=000000000000000000009A195D2DD88C&subject=CN=SMP_1000000000,O=DG-DIGIT,C=BE&validFrom=Oct 21 02:00:00 2014 CEST&validTo=Oct 21 01:59:59 2016 CEST&issuer=CN=Issuer Common Name,OU=Issuer Organization Unit,O=Issuer Organization,C=BE' where domainId='default';

17. CONTACT INFORMATION

CEF Support Team

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

Support Service: 8am to 6pm (Normal EC working Days)