Oracle ERPIntegrationservice automation using httppost

Oracle ERP Cloud – ERPIntegrationservice

Automation of ERP Cloud using ERPIntegrationservice

Objective:

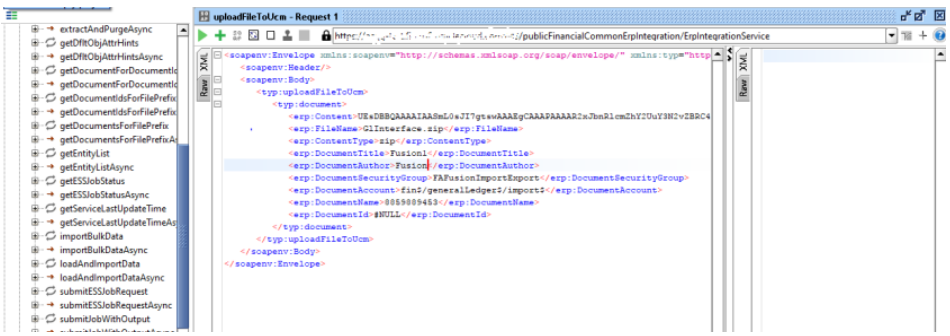
The objective of the below content is to achieve automation of File Based Data Import to UCM (Universal Content Management) using simple java code (HTTP Protocol). This approach can be used for both ERPCloud data migration and for ERPCloud UAT test automation. The required scenario specific test data can be created by end users and invoke the web service to automatically upload the data and verify the test scenarios are working as intended. The code can be extended in fetching the Test Results.

ERP Cloud modules offers FBDI (File Based Data Import) template for quick upload data integration strategy will have to invoke the ERPIntegrationservice using Java or from respective database. The below steps will explain on how to consume ERPIntegrationservice (SOAP) web service using standard Java code.

To initiate a web service using HTTP Protocol. The following are the understanding/details required to consume ERPIntegrationservice.

Step1 – Understanding the Payload of uploadFileToUCM.

Open SOAP UI and provide the WSDL and click on uploadFileToUcm



The below mentioned payload parameters will change based on the data content (In the above example we are uploading Journal data into UCM)

Payload Parameter Content: Content holds the actual data in FBDI. For e.g.: Journal details mentioned in FBDI need to be converted to base64 format. There are different options to convert the CSV or text content to base64.

Payload Parameter Document Name: Document Name should be unique for every run/upload.

Step2 – Import the certificate from the server into the client trust store.

```
C:\>cd C:\Program Files\Java\jdk1.8.0_144\bin

C:\Program Files\Java\jdk1.8.0_144\bin>keytool -list -v -keystore "C:\Program Files\Java\jdk1.8.0_144\jre\lib\security\cacerts"
```

```
C:\Program Files\Java\jdk1.8.0_144\bin>keytool -list -v -keystore "C:\Program Files\Java\jdk1.8.0_144\jre\lib\security\cacerts"
Enter keystore password:
```

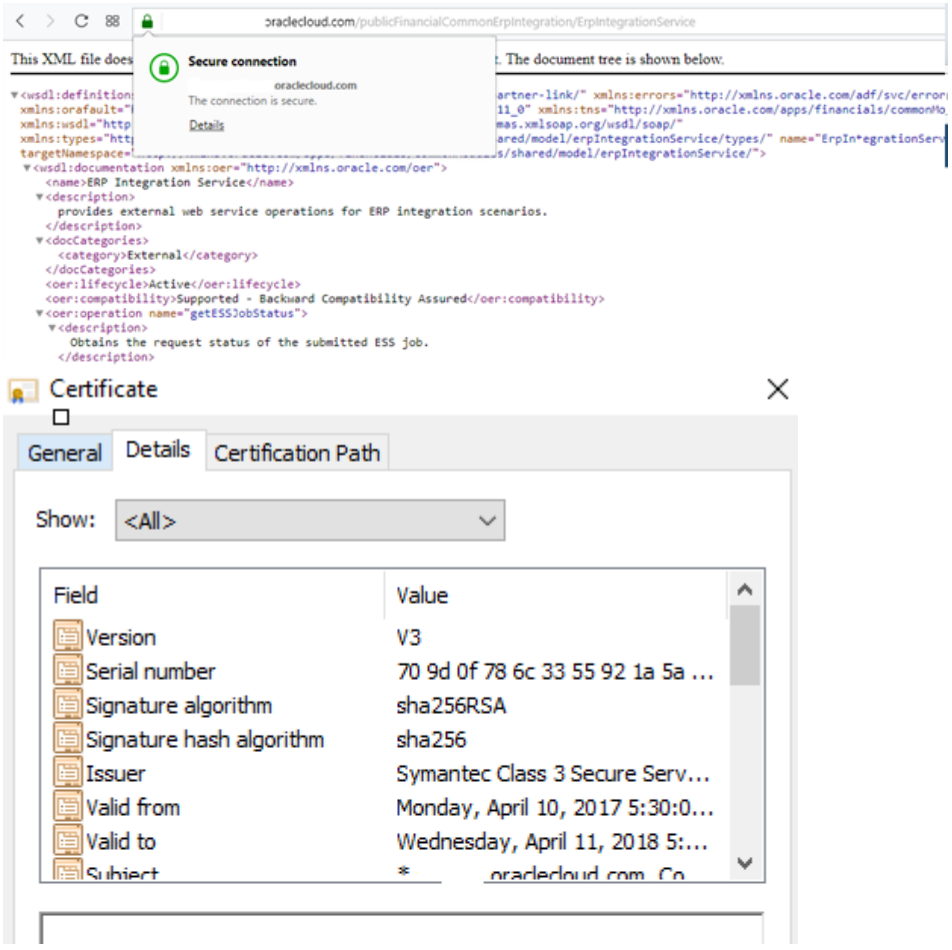
Default password is changeit

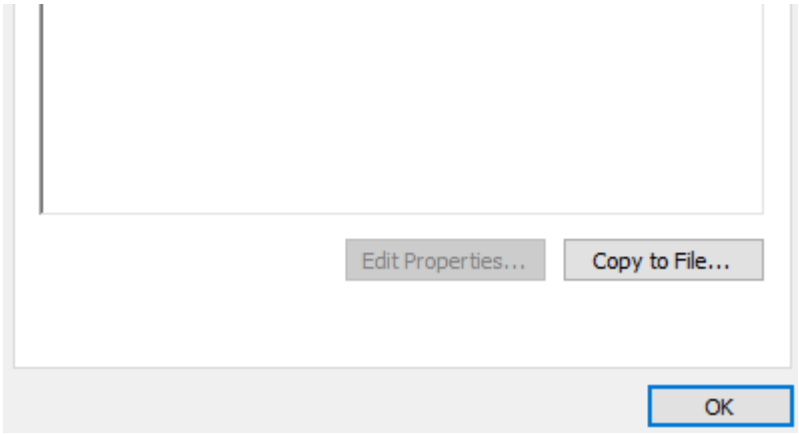
```
Alias name: addtrustclass1ca [jdk]
Creation date: Aug 25, 2016
Entry type: trustedCertEntry

Owner: CN=AddTrust Class 1 CA Root, OU=AddTrust TTP Network, O=AddTrust AB, C=SE
Issuer: CN=AddTrust Class 1 CA Root, OU=AddTrust TTP Network, O=AddTrust AB, C=SE
Serial number: 1
Valid from: Tue May 30 16:08:31 IST 2000 until: Sat May 30 16:08:31 IST 2020
Certificate fingerprints:
         MD5:  1E:42:95:02:33:92:6B:B9:5F:C0:7F:DA:D6:B2:4B:FC
         SHA1: CC:AB:0E:A0:4C:23:01:D6:69:7B:DD:37:9F:CD:12:EB:24:E3:94:9D
         SHA256: 8C:72:09:27:9A:C0:4E:27:5E:16:D0:7F:D3:B7:75:E8:01:54:B5:96:80:46:E3:1F:52:DD:25:76:63:24:E9:A7
         Signature algorithm name: SHA1withRSA
         Version: 3

Extensions:

#1: ObjectId: 2.5.29.35 Criticality=false
AuthorityKeyIdentifier [
KeyIdentifier [
0000: 95 B1 B4 F0 94 B6 BD C7   DA D1 11 09 21 BE C1 AF  ............!...
0010: 49 FD 10 7B                                        I...
]
[CN=AddTrust Class 1 CA Root, OU=AddTrust TTP Network, O=AddTrust AB, C=SE]
SerialNumber: [    01]
]
```

List of all certificates from trust store will be listed. This is to verify the certificate is not already added to your trust store.

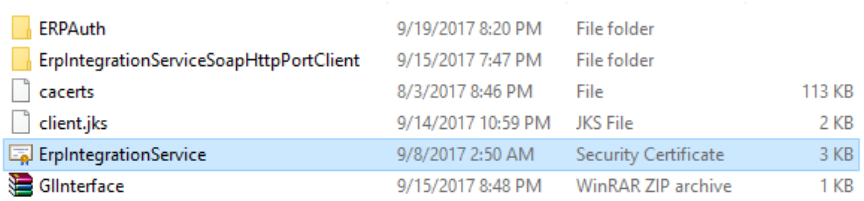
Now – Import Cloud certificate into cacert file to enable the authentication.

Place wsdl in respective browser and download the certificate to local machine.





Click Copy to File and save the .cer file to your local machine.



Note: you may find the exact steps to download certificate from your respective browser (Chrome/IE etc) by searching in Google.

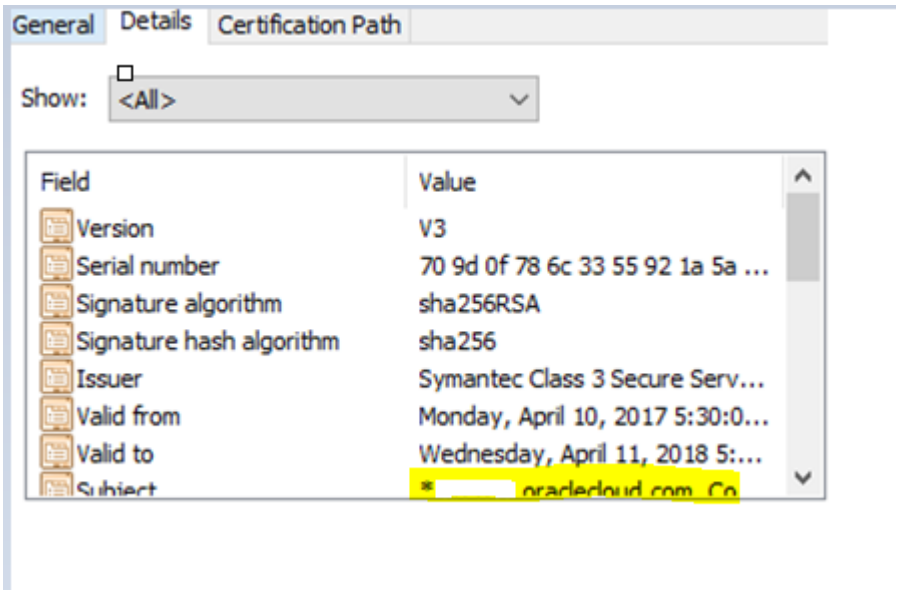
Downloaded security certificate to be included into trust store using the below keytool.

```
C:\Program Files\Java\jdk1.8.0_144\bin>keytool -import -file I:\SOAPfiles\ErpIntegrationService.cer -keystore "C:\Program Files\Java\jdk1.8.0_144\jre\lib\security\cacer
ts"
Enter keystore password:
```

Enter default password changeit and your certificate will be added to truststore.

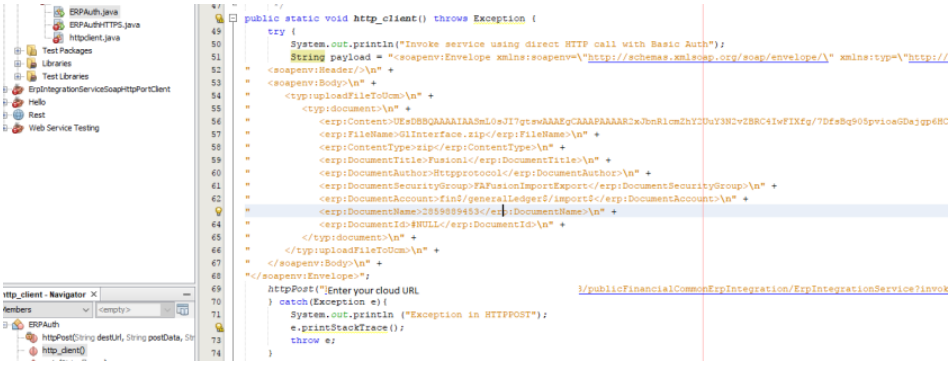
List's certificate from trust store and ensure it has been added successfully.

```
Alias name: mykey
Creation date: Sep 22, 2017
Entry type: trustedCertEntry

Owner: CN=*.oraclecloud.com, OU=Content Management Services IT, O=Oracle Corporation, L=Redwood City, ST=California, C=US
Issuer: CN=Symantec Class 3 Secure Server CA - G4, OU=Symantec Trust Network, O=Symantec Corporation, C=US
Serial number: 709d0f786c3355921a5ae8a453cbb1f5
Valid from: Mon Apr 10 05:30:00 IST 2017 until: Wed Apr 11 05:29:59 IST 2018
Certificate fingerprints:
         MD5:  FA:0D:B4:72:14:0E:54:85:DF:38:92:20:24:DB:79:D4
         SHA1: B4:9C:02:65:6C:CD:25:31:1F:31:0F:5E:B6:A8:22:76:8F:A6:D0:48
         SHA256: 29:B7:D0:BE:27:B7:A5:3D:83:84:DA:9C:D1:26:ED:2E:34:B1:77:AE:22:53:8F:45:4C:84:59:42:B4:41:0E:E4
         Signature algorithm name: SHA256withRSA
         Version: 3
```



Above image display the certificate name from browser vs from the list of certificates added to your trust store.

Any IDE can be used to execute the below shared java program. In this blog – Netbeans has been used to execute this script.



The above code snippet is available in this link – git. Replace the input string as per your FBDI data in all the Payload parameters. If you receive 500 – Internal Response error then either of the parameter is incorrect or url mentioned and login credentials mentioned in Httppost line might be incorrect.

Line number 69: Httppost(<Your cloud URL //publicFinancialCommonErpIntegration/ErpIntegrationService?

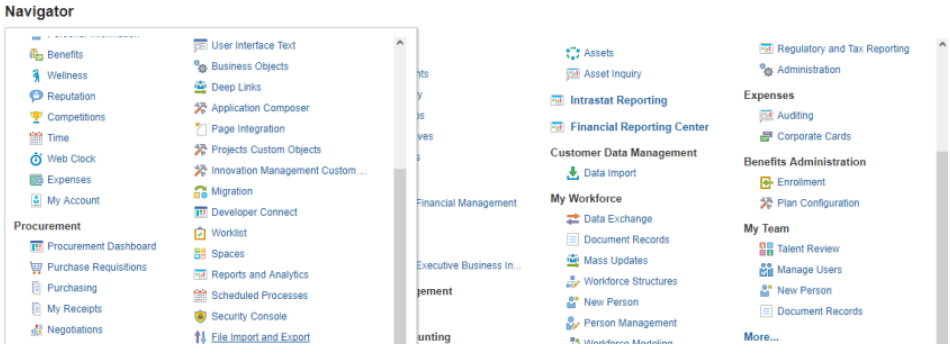
URL//publicFinancialCommonErpIntegration/ErpIntegrationService?invoke=>),Payload, “User Name:Password”)

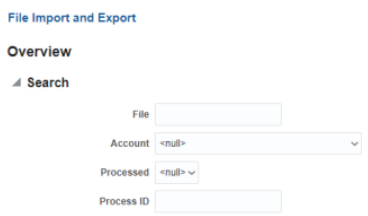
Note: Make sure this cloud user has Implementation role and Integration Specialist role.

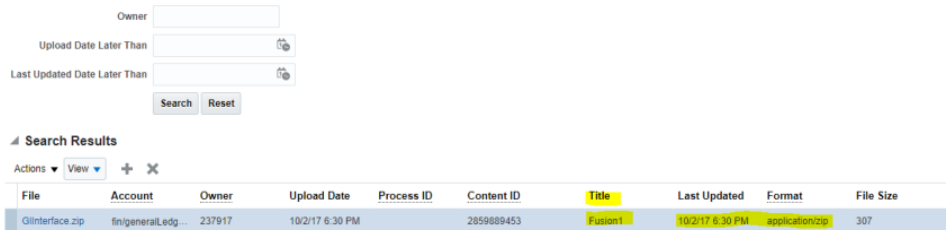
```
Invoke service using direct HTTP call with Basic Auth

post data size:1431
connection status: 200; connection response: OK
==================Service response: ================
------=_Part_66481_988002704.1506969002157
Content-Type: application/xop+xml;charset=UTF-8;type="text/xml"
Content-Transfer-Encoding: 8bit
Content-ID: <4665861e-dd6f-41e4-9b98-3e8c8cbe6765>

<?xml version="1.0" encoding="UTF-8" ?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://www.w3.org/2005/08/addressing">
------=_Part_66481_988002704.1506969002157--
BUILD SUCCESSFUL (total time: 18 seconds)
```

The above highlighted is the response received after uploading file to UCM. Please observe the following details in Payload: “Document Title – Fusion1”. Navigate to front end to verify the successful upload of this file.

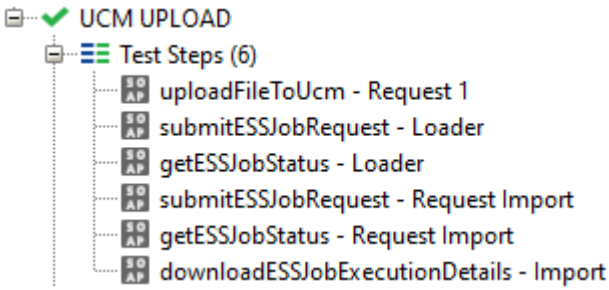
Navigate to File Import and Export to verify the upload







File has been successfully uploaded. The same approach can be extended in calling the below methods to achieve end to end File upload and spawning the respective ESS jobs to import the data into Interface and Base tables.



********** End of Content **********

Disclaimer:

This is a personal blog. Any views or opinions represented in this blog are personal and belong solely to the blog owner and do not represent those of people, institutions or organizations that the owner may or may not be associated with in professional or personal capacity, unless explicitly stated. Any views or opinions are not intended to malign any religion, ethnic group, club, organization, company, or individual.

All content provided on this blog is for informational purposes only. The owner of this blog makes no representations as to the accuracy or completeness of any information on this site or found by following any link on this site. The owner will not be liable for any errors or omissions in this information nor for the availability of this information. The owner will not be liable for any losses, injuries, or damages

from the display or use of this information.

October 5, 2017 / Oracle ERP Cloud, Uncategorized / Automate FBDI using webservice, Consuming Oracle SOAP service, ERPIntegrationservice, ERPIntegrationservice using HttpPost, Oracle ERP Cloud, Oracle SaaS, Oracle SaaS automation / Leave a comment