

A Novel Secure Cloud SAAS Integration for User Authenticated Information

C. V. Satyanarayana

Research Scholar, Rayalaseema University,
Kurnool, Andhra Pradesh, India

Dr. P. Suryanarayana Babu

Research Supervisor, Department of Computer
Science, Rayalaseema University,
Kurnool, Andhra Pradesh, India

ABSTRACT

Organizations are spending a lot of money to maintain their business data using legacy Enterprise content management systems [1], tools, hardware support and maintenance, which are not satisfying the consumers. Box is Novel ECM system which allows the users to store, view, search large value of data with lesser bandwidth to increase user experience for modern business. Modern business running on new enterprise cloud platforms like Amazon, eBay, Salesforce. Persistence of business user's data is simpler and faster with this platform and is really an emerging solution for the current industry. The real-time integration between cloud SaaS application with on-premises content management systems (SharePoint [2], box [3], Oracle Web Center [4], Nuxeo [5], Open Text [6], Dell Documentum [7], IBM FileNet p8[8], DocuShare [9], On Base [10], Laser Fiche [11], SpringCM [12], Seismic [13], Lexmark [14], M-Files [15], Alfresco [16], Media Bank [17], Veeva Vault [18], etc.)

In this paper, we are proposing a new paradigm of how Cloud SaaS applications integrate with novel ECM's for modern business. We performed real-time integration from SFDC to box with a business use case, we did bi-directional data synchronization between SFDC to box and vice versa.

I. INTRODUCTION

Enterprise Content Management used to store web content, forms processing, Rich media files (Audio, Video), email, confidential documents, host generated reports, fax and scanned files at a central repository. ECM has a lot of advantages like concurrent access, Email as a business asset, Quick Customer service, and documents in the central repository, difficult to contribute to websites. ECM allows people to connect with ERP, CRM, SCM, Desktop Applications,

Communities and any wireless communication devices. As per the international survey, 30% people are spending time for the content searching and there is 85% unstructured information. Globally 79% of IT companies have 2 repositories and 25% of companies have 15 and above. Every ECM must have basic capabilities (security, search, storage, Integration, collaboration, workflow, record management and personalized delivery) to access any information at any time. [19] Gartner stated IBM offers ECM and has significant growth. ECM is useful for big enterprises (US Army have 1,500,000 user's data with around 100,000 forms) to return good ROI by optimizing/automate the legacy process illustrated in the fig.1. (Courtesy: IBM Software Group).

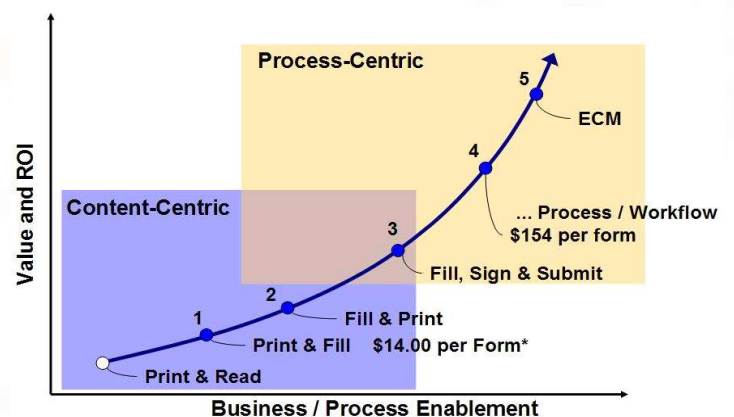


Fig 1: eForms (Source: IBM Software Group)

Box [20] – a modern cloud-based enterprise content management offers novel services (Content organization and delivery with context, simple workflow and governance). Deliver connect with connect handles content classification, stop endless searching. Workflow management improves the transparency, regulated process, and optimizes the

errors. Box provides self-service IT capability for business partners, vendors, customers.

II. DESIGN AND DEVELOPMENT

Use case:

Bi-Directional Synchronization Sequence Steps:

Step 1: Login into salesforce.com

Step 2: Create a new Account Record in Account Object, which navigates to its detailed page

Step 3: Create a new Note related list and enter Title, Description the Save

Step 4: 'Save' fires a pre-configured Trigger with two events (After insert / After Update)

Step 5: Trigger calls "Web service".

5.1: Once web service Callout then, constructor will cause execution of a class to get access token

5.2: Class sends a request and get response as key "Authorization Key"

5.3: Once Token generated as response, web service callout the box "end point URL"

5.4: Folder and File creation happen based on the Parent id (Account ID)

5.5: Once file created, web service receives "Folder & File" details as response

Step 6: Web service response updates the SFDC detailed record.

The enterprise Use case, design strategy as illustrated in the below diagram

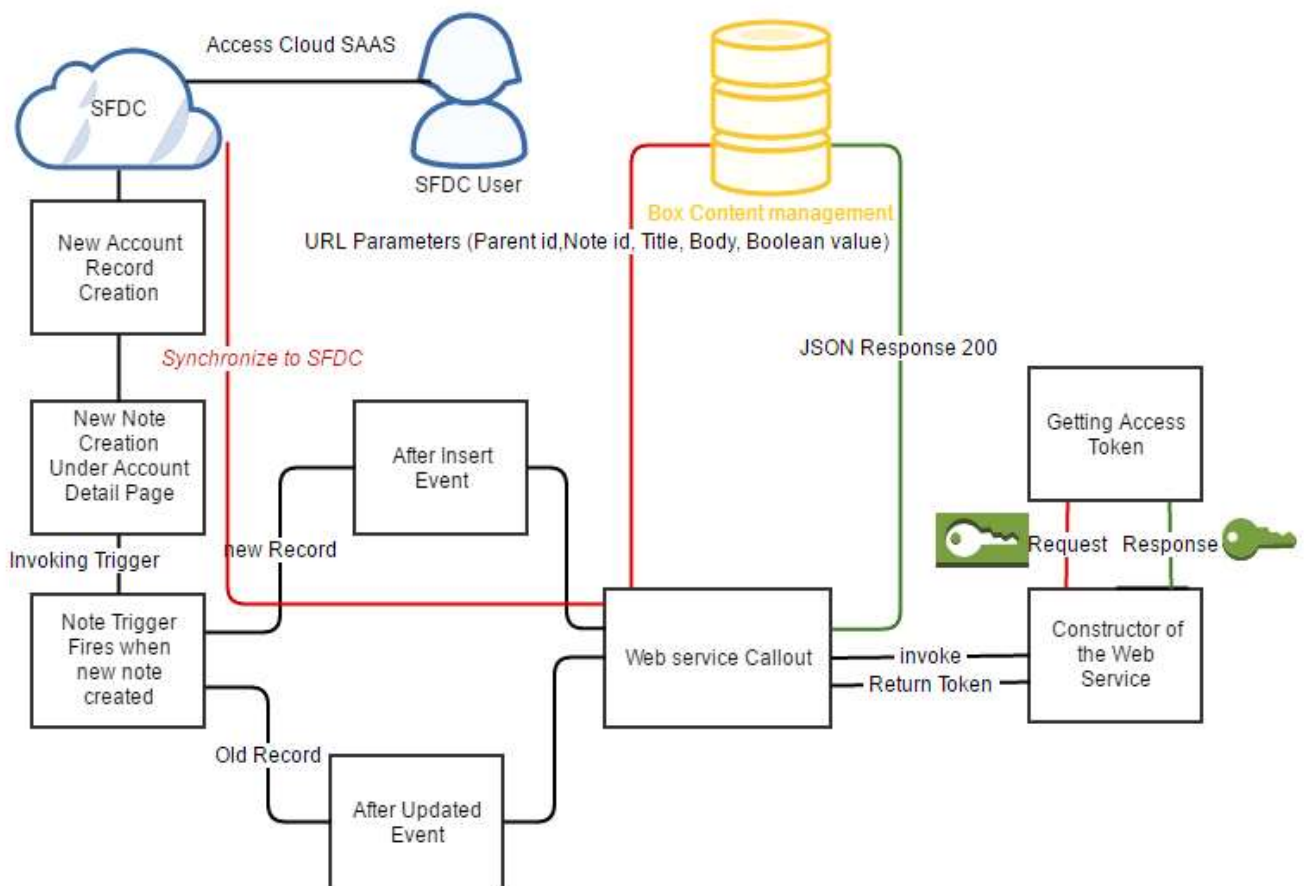


Fig.1. Architecture for SFDC integration with Box Content Management

Implementation of Use case

The overall scenario is implemented in several stages. Initial stage carried out the box ECM configuration to take care folder accessibility (Authorization) publicly. To enable folder accessibility public, 2048 bit RSA private key and the public key generated by OpenSSL using Cygwin in Windows environment. Private and public key generation done with below commands

```
openssl pkcs8 -topk8 -inform PEM -outform DER -in private_key.pem -out rsaprivkey.der -nocrypt →(1)
```

```
openssl pkcs8 -in private_key.pem -topk8 -nocrypt -out certificatename.pk8 →(2)
```

Once the private and public key generated and must submit public key in the box public key management which provides a KeyID "3n7jqgt7". The Header has three arguments, alg refers the algorithm (RS256, RS384, RS512), type stands type of token (default is JWT) and kid is public key id generated by box. The header looks like as follows:

```
string header='{ "alg": "RS256", "typ": "JWT", "kid": "3n7jqgt7"}'; →(3)
```

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAAuwU1D7VrNc3VugHAt/OK
awqorUjEorvY3C...
-----END PUBLIC KEY-----
```

Fig. Generated Public Key

```
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQC7BTUPTws1zdW6
-----END PRIVATE KEY-----
```

Fig. Generated Private Key

Trigger Implementation:

Once the Note created in Account detail page causes Trigger execution. The trigger is automatic functional task executed in two events, namely After Insert, After Update.

```
If(trigger.isinsert && trigger.isafter)
{
    For(Note nt:trigger.new){
        Id parentid= nt.parentId;
```

```
Schema.DescribeSOBJECTResult sa
=parentid.getSOBJECTType().getDescribe();
if(sa.Name.toLowerCase()=='account'){
    createfileandfoldersinbox.createdfilesand
folderinbox(parentid,nt.id,
nt.Title,nt.body,true);
}
```



```

if (sa.Name.ToLowercase() == 'opportunity'
|| sa.Name.ToLowercase() == 'contact') {

createfileandfoldersinbox.createdfilesand

folderother (parentid, nt.id,
nt.title, nt.body, true);
    }
}
}
}

```

Trigger call the method "createdfilesandfolderother" in the web service "createfileandfoldersinbox" by passing the parameters parented, id, Title, body, Boolean value (true / false). Createfileandfoldersinbox execute asynchronous way by made "Future" callout. Web service call external Utility class "boxaccessandrefereshToken" to get the access token and assign to the folderid.

```

Folderid= boxaccessandrefereshToken.
boxaccrecordid (parentid, noteid, tittle,
body, insertfile);

```

"Boxaccrecordid" method call another utility method called "Boxhalfuserauthenticationcls.Boxhalfuserauthentication" which returns the response as an authorization token. Once generate and submit the public key, you can able to send user OAuth and enterprise token by JWT (JSON Web Token) grant. Each and every JWT assertion have mainly three components namely Header, Claims and the Signature. The header has an algorithm support for JWT signature, Claims contain the information required to authenticate and generate a correct token, Signature searches the application and validated by the public key. The pseudo code of the utility class as follows:

```

String claim_set =
{'"iss": "9s9i33f74i3uth9pu6d7lfbfu3sffat6";
claim_set +=, "sub": "792309226";
claim_set +=, "box_sub_type": "user";

claim_set +=, "aud": "https://api.box.com/oauth2/tok
en";

claim_set +=, "jti": "" + jti + """;
claim_set +=, "exp": "" + exptime1 + ""};
String claim_set_Encode =
EncodingUtil.base64Encode(blob.valueOf(claim_set));

```

```

String Singature_Encode =
Header_Encode+'.'+claim_set_Encode;
string key='private key specified above';
blob privateKey = EncodingUtil.base64Decode(key);
Singature_Encode =
Singature_Encode.replaceAll('=', "");
Httprequest req = new HttpRequest();
HttpResponse res = new HttpResponse();

req.setEndpoint('https://api.box.com/oauth2/token')
;
req.setMethod('POST');
req.setHeader('ContentType', 'application/x-www-form-
urlencoded');
String grt = 'urn:ietf:params:oauth:grant-type:jwt-bearer';
List<string> urlParams = new List<string>
{'grant_type='+grt, 'client_id=' +
EncodingUtil.urlEncode('9s9i33f74i3uth9pu6d7lfbfu3sffat6
', 'UTF-8'), 'client_secret=' +
EncodingUtil.urlEncode('068tibjPovMcRr8qXNRhlq8wQKq
ZGSu', 'UTF-8'), 'assertion='+JWT};
string body = string.join(urlParams, '&');
system.debug('====body ====='+body);
req.setBody(body);
res = h.send(req);
string Response = res.getBody() +' '+ res.getStatusCode();

```

Finally, the response variable receives key value (Authorization token).

CONCLUSION

The modern enterprise business following cloud-based paradigm, which offers the shared services, pay and use the mechanism, repository, Infrastructure and working as Platform as well. Modern ECM allows you to maintain enterprise data through the cloud. The SAAS connectivity, data synchronization with ECM is a challenging task in terms of governor limitation. We proposed a novel approach to synchronize SAAS data with ECM with authorization user. The proposed technique provides a public/private key for the user to access ECM content seamlessly. We integrated Salesforce CRM SAAS application with modern ECM called box to evaluate the performance of the proposed solution by implementing custom web services, which outperform than other techniques in terms of speed and reliability.

REFERENCES

- 1) Enterprise content management , https://en.wikipedia.org/wiki/Enterprise_content_management.
- 2) SharePoint ,detailed information available at <https://products.office.com/en-us/sharepoint>.
- 3) *Box* Cloud storage company, detailed information available at <https://www.box.com>.
- 4) Oracle Web center , detailed information available at <http://www.oracle.com/technetwork/middleware/webcenter/suite/overview/index.html>.
- 5) Nuxeo , detailed information available at <https://doc.nuxeo.com>.
- 6) Open Text, detailed information available at <http://www.opentext.com/>.
- 7) Dell Documentum, detailed information available at <http://www.opentext.com/>.
- 8) IBM FileNet p8 , detailed information available at <http://www.ibm.com/support/knowledgecenter/SSGLW6>.
- 9) DocuShare, detailed information available at <https://docuShare.com/>.
- 10) On Base, detailed information available at <https://www.onbase.com/en/>.
- 11) Laser Fiche, detailed information available at <https://www.laserfiche.com/>.
- 12) SpringCM , detailed information available at <https://www.springcm.com/>.
- 13) Seismic, detailed information available at <https://seismic.com/>.
- 14) Alfresco content services, detailed information available at <https://www.alfresco.com/platform/content-services-ecm>.
- 15) Lexmark , detailed information available at https://www.lexmark.com/en_in.html.
- 16) M-Files, detailed information available at <https://www.m-files.com/>
- 17) Media Bank, detailed information available at <https://www.mediabank.me/>.
- 18) Veeva Vault, detailed information available at <https://www.veeva.com/products/enterprise-content-management/>.
- 19) Gartner Report <http://mediaproducts.gartner.com/reprints/emc/article5/article5.html>
- 20) Box enterprise content management <https://www.box.com/content-management>