eSign ASP Integration Guide **CUSTOMER**

Document Version: 1.1 – 2017-08-14

Integration between Digital Compliance Service and eSign Application Service Provider

Integration package version 1.1.0

Typographic Conventions

Type Style	Description
Example	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.
Example	Emphasized words or expressions.
`EXAMPLE`	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE`	Keys on the keyboard, for example, `F2` or `ENTER`.

Document History

Version	Date	Change
1.0	2017-07-24	First release of the eSign ASP integration document
1.1	2017-08-14	Document enhancement with additional information

Contents

1 Introduction

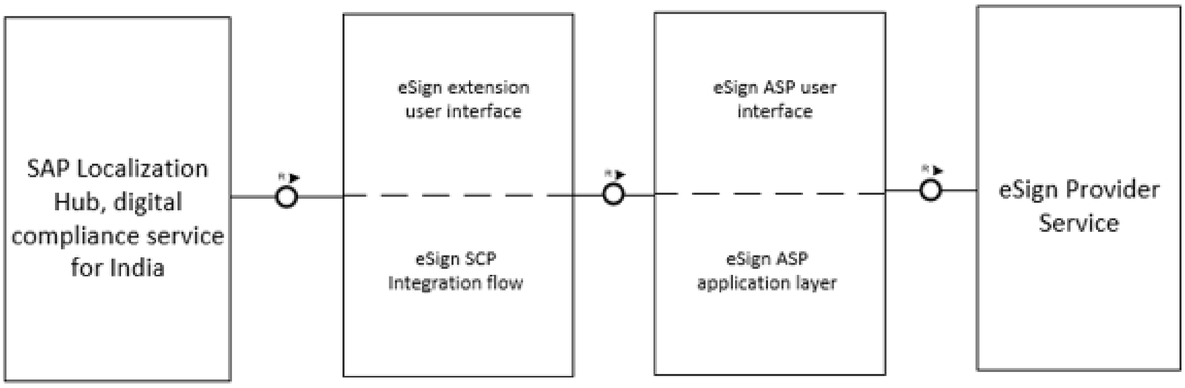
You integrate SAP Localization Hub, digital compliance service with the eSign application service provider (eSign ASP) to digitally sign filing document content before filing returns. The SAP Cloud Platform Integration Service integrates the SAP Localization Hub, digital compliance service with the eSign application service provider (eSign ASP). This document provides steps to establish communication between SAP Localization Hub, digital compliance service and eSign ASP application.

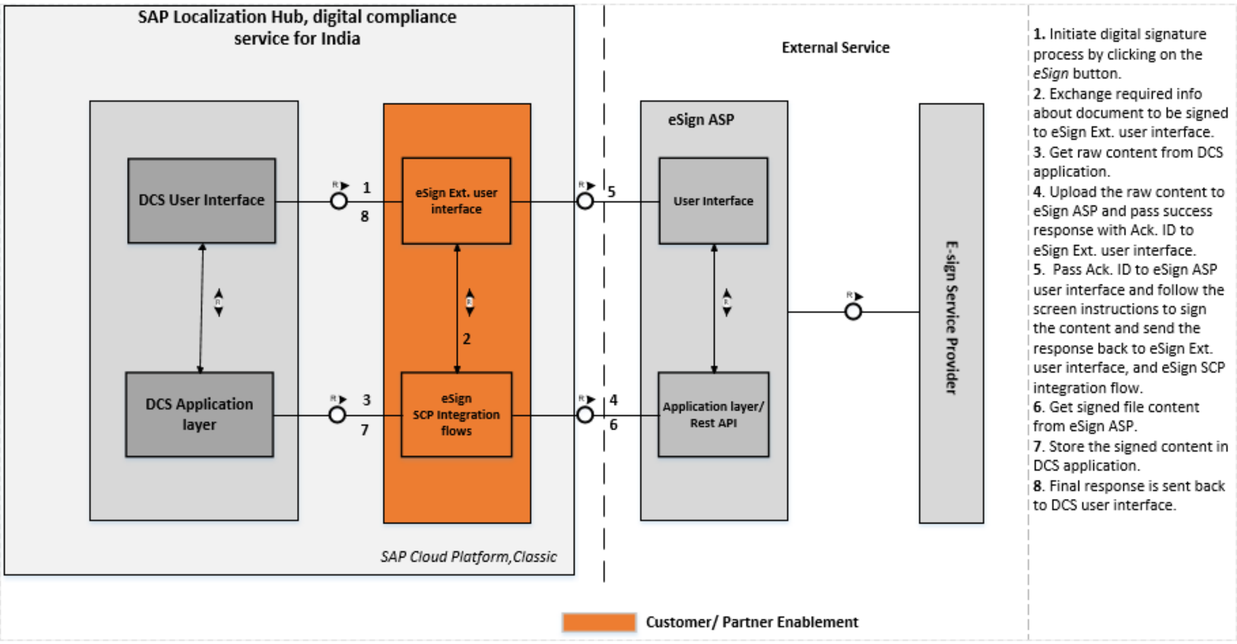
> Note
>
> This documentation and configuration would be required only if you want to digitally sign the document content via the online process as mentioned in the **SAP Localization Hub, digital compliance service: Digital Signature Process** documentation. Alternately, you can use the offline process using Digital Signature Certificate(DSC) USB device to digitally sign the filing content.

eSign Service Provider (ESP), also known as certificate authority or *Certifying Authorities* (CA), is an entity that issues digital signature certificates for electronic authentication of users.

eSign Application Service Provider (eSign ASP) is an entity that signs the contract with the ESP to provide electronic signature Service. eSign ASP uses eSign service as part of their application to digitally sign the content. eSign ASP provides the interface software platform for users to sign documents using Aadhaar + OTP or using Digital Signature Certificate (DSC) USB device.

The below diagram provides an overview of the integration between SAP Localization Hub, digital compliance service and eSign ASP.


SAP Localization Hub, digital compliance service for India
eSign extension user interface
eSign SCP Integration flow
eSign ASP user interface
eSign ASP application layer
eSign Provider Service


SAP Localization Hub, digital compliance service for India
DCS User Interface
DCS Application layer
eSign Ext. user interface
eSign SCP Integration flows
SAP Cloud Platform,Classic
External Service
eSign ASP
User Interface
Application layer/ Rest API
E-sign Service Provider
R
1
8
2
3
7
5
4
6
Customer/ Partner Enablement

1. Initiate digital signature process by clicking on the *eSign* button.

2. Exchange required info about document to be signed to eSign Ext. user interface.

3. Get raw content from DCS application.

4. Upload the raw content to eSign ASP and pass success response with Ack. ID to eSign Ext. user interface.

5. Pass Ack. ID to eSign ASP user interface and follow the screen instructions to sign the content and send the response back to eSign Ext. user interface, and eSign SCP integration flow.

6. Get signed file content from eSign ASP.

7. Store the signed content in DCS application.

8. Final response is sent back to DCS user interface.

2 Prerequisites

Before you start with the activities mentioned in this document, ensure that the following prerequisites are met:

1. Installed the *SAP Localization Hub, digital compliance service for India* solution in your *SAP Cloud Platform, Neo* test and/or productive landscape. For detailed information, refer SAP note 2460667.
2. Obtained the DSC USB device.
3. Have a registered PAN, as recommended by the GSTN, at the time of filing digitally signed content.
4. Provisioned live SCP Integration Service test and/or productive tenants.
5. Chosen eSign ASP or eSign gateway service providers and completed the registration process.
6. You have received the following information or documents after on boarding with *eSign ASP*:

- eSign ASP integration or setup manual guide
- JavaScript SDK
- Certificate required for establishing SSL handshake with eSign ASP.

Note

After you complete integration with the eSign ASP mentioned in this guide, you should develop and configure digital compliance service (DCS) eSign ASP SAPUI5 application which acts as interface between the DCS application screen and the eSign ASP/ Web service screen, by exchanging data between the screens or the application. For more details, see document **SLH DCS eSign ASP User Interface guide.**

3 eSign ASP Integration content

The content catalog package *Integration between Digital Compliance Service to GST Suvidha Provider and eSign Application Service Provider* contains the following iFlows:

iFlow Name in WebUI	Project Names/Artifact Names
GSP Integration Template	com.sap.slh.dcs.gsp.template
eSign ASP Integration Template	com.sap.slh.dcs.esp.template

eSign ASP integration flow is a generic integration template and should be modified or enhanced for integration with eSign ASP. This template contains process calls to set up communication with the DCS application. The process call for communication with eSign ASP must be added based on the details received after registering with eSign ASP.

Procedure

To find the integration flow in the content catalog, perform the following steps:

1. In your browser, go to the WebUI of the SCP integration tenant using the url:

   `<SCP Integration Tenant URL>/itspaces`.
2. To logon, enter your **P user** or **S user**.

   If you get the *HTTP Status 403* error, contact your tenant administrator.
3. After successful login, from the menu in the upper left corner, choose *Discover*.
4. In the subsequent screen, search for *Integration between Digital Compliance Service to GST Suvidha Provider and eSign Application Service Provider*, and select the package.

4 DCS RESTful API for eSign Integration

SAP Localization Hub, digital compliance service exposes two RESTful APIs to setup eSign Integration with an external application. The two RESTful APIs are the following:

- dcsdownload

  Provides plan or unsigned filing document content in base64 encoded format.
- dcsupload

  Accepts signed filing document content along with type of signature and signatory ID.

Note

eSign ASP integration flow template has built-in communication setup for DCS RESTful API and DCS oAuth service. You should configure DCS application URL and oAuth token endpoint URL while deploying the Integration flow.

Structure of both these APs are detailed below.

4.1 DCS RESTful API: dcsdownload.

This RESTful API accepts query parameters to select filing document in database and provide the same in response with base64 encoded format along with the response status code.

URL: `<dcsmain application baseURL>/gstr/esignservice/api/v1/dcsdownload`

Method: `GET`

Headers: `Content-Type: application/json`

The following table provides a list of query parameters:

Parameter name	Data type	description	example
gstin	String	GST Identification number	
repCatId	String	GST Return type	IN_GSTR1
fp	String	Reporting period	022017 where 02 indicates reporting month and 2017 indicates reporting year

Parameter name	Data type	description	example
repActivityId	String	Activity for which filing is done.	GSTR1FILE

Response:

```
{
"code": <status>,
"content": <filing document content in base64 format>
}
```

The following table details the Response JSON attributes:

Attribute	Data type	description	example
code	String	Status code	SUCCESS or FAILED
Content	String	Selected filing document content for requested query parameters	

4.2 DCS RESTful API: dcsupload

This RESTful API accepts signed filing document content, type of signature and signatory ID as part of the request body, and header or query parameters to update the request content to the respective GSTR in the database. The API returns the response content status and the document reference ID for content stored.

The API also converts the received request content to JSON format as prescribed by the GSTN for filing returns. For more details, refer here.

URL: `<dcsmain application baseURL>/gstr/esignservice/api/v1/dcsupload`

Method: `POST`

Headers: `Content-Type: application/json`

The following table details the list of header parameters or query parameters:

Parameter name	Data type	description	example
gstin	String	GST Identification number	
repCatId	String	GST Return type	IN_GSTR1

Parameter name	Data type	description	example
fp	String	Reporting period	"022017"where 02 indicates reporting month and 2017 indicates reporting year
repActivityId	String	Activity for which filing is done.	GSTR1FILE

Request body payload:

```
{
        "sign": <signed filing document content in pkcs7 format>,
        "st": "<DSC>",
        "sid": "<PAN Number>"
}
```

The following table details the Body JSON attributes:

Attribute	Data type	description	example
sign	String	Stores signed content	
st	String	Signature Type	Either DSC or ESIGN
sid	String	Signer Id	If *st* is **DSC**, then *sid* will be **PAN**. If *st* is **ESIGN**, then *sid* will be **Aadhaar number**.

Response:

```
{
        “code”: SUCCESS or FAILED
        “content”: <Document Id if SUCCESS>
}
```

The following table details the Response JSON attributes:

Attribute	Data type	description	example
code	String	Status code	Either SUCCESS or FAILED
content	String	Content that is returned from DCS App	Contains Document ID of DCS Application

5 Integration content setup

SAP Cloud Platform Integration (Cloud Integration) for process integration facilitates the integration of business processes spanning different companies, organizations, or departments within an organization.

For more details, see SAP Cloud Platform Integration.

> ➡ Recommendation
>
> Perform the below mentioned steps in your **TEST** and **PRODUCTION** SCP Integration cockpit account.

5.1 Import SSL Certificates to SCP Integration Tenant

To set up an SSL connection between the SCP integration tenant and eSign ASP, you must import the required security certificates into the SCP integration tenant JAVA keystore. For more details, see Tenant Client Keystore.

Procedure

1. Create a new Keystore and import the eSign ASP trusted certificates into the keystore.

   > i Note
   >
   > You can use tools like *Keystore Explorer* to create a new keystore.

2. Add keystore entries to the JAVA keystore of the SCP integration tenant by following the process mentioned here.

   > i Note
   >
   > To perform the above operation, you need to be a tenant administrator with role **AuthGroup.Administrator**.

   > ➡ Recommendation
   >
   > To check the connectivity with eSign ASP, run connectivity test using *Monitor-> Manage Security -> Connectivity Tests.*

5.2 Adapt eSign ASP Integration Content Template

An integration flow is a graphical representation of how the integration content can be configured to enable the flow of messages between two or more participants using SAP Cloud Platform Integration, and thus ensure successful communication. For more details, see Creating Integration Project for an Integration Flow.

The *Integration between Digital Compliance Service to GST Suvidha Provider and eSign Application Service Provider* content catalog contains pre-delivered integration template for eSign ASP. You can use iFlow eSign ASP Integration Template to build the specific eSign ASP integration flow.

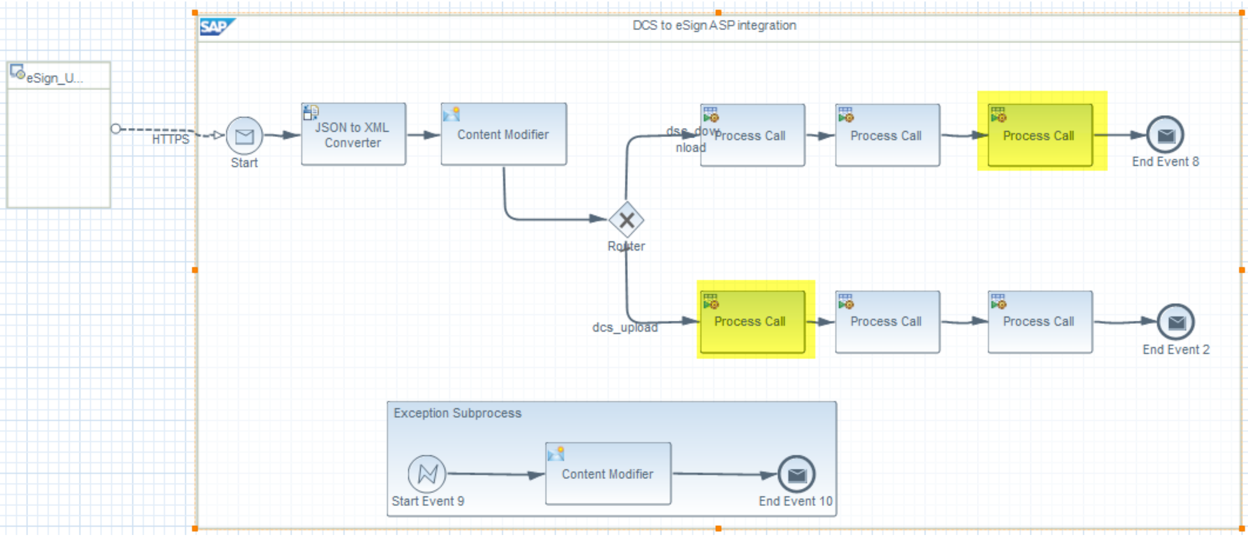
Procedure

To find the integration flow in the content catalog, perform the following steps:

1. In your browser, go to the WebUI of the SCP integration tenant using the url:

   `<SCP Integration Tenant URL>/itspaces`.
2. To logon, enter your **P user** or **S user**.

   If you get the *HTTP Status 403* error, contact your tenant administrator.
3. After successful login, from the menu in the upper left corner, choose *Discover*.
4. In the subsequent screen, search for *Integration between Digital Compliance Service to GST Suvidha Provider and eSign Application Service Provider*, and select the package.
5. Click on the entry and in the subsequent screen, choose *Copy*.
6. From the menu in the upper left corner, choose *Design*.
7. Click on *Integration between Digital Compliance Service to GST Suvidha Provider and eSign Application Service Provider* catalog and in the subsequent screen, choose *ARTIFACTS*.
8. Select integration flow **eSign ASP Integration Template**.
9. Choose *Actions -> Download*. The system downloads the integration content as a ***.zip** file.
10. Import the downloaded zip file to eclipse and modify the content to adapt eSign ASP integration. For more details, see Developing Integration content flow using the Eclipse Integration Designer.



i Note

You need to adapt the eSign ASP communication and authentication for highlighted process call as shown in the above picture.

11. Configure the integration flow parameters as defined below:

Receiver: DCS_server_get_Auth_token

- Parameter name: oauthasservices_url
- Description: SCP account oAuth Token Endpoint
- Value: In your web browser, log on to the cockpit, and select an account.
  - Go to *Security -> OAuth -> Branding section -> oAuth URL's*
  - Select and copy *Token Endpoint* value.

    Example: https://oauthasservices-example.com/oauth2/api/v1/token
  - Maintain the value without prefixing **https**.

    Sample value to be entered: *<oauthasservices-example.com/oauth2/api/v1/token>*

- Parameter name: Credential Name/dcsoAuthtokencredentials
- Description: user credential for client ID and client secret key for oAuth authentication.
- Value: get the ID and secret key for the registered oAuth client **gstrapp** in SCP cockpit. For more details refer here.
  - Use this ID and secret key as name and password while creating user credentials as mentioned in step 12 below.

    User credentials act as alias or reference to client ID and secret key maintained in security materials.
  - Enter the *<user credential>* created.

Receiver: DCS_server_download/DCS_server_upload

- Parameter: dcsapplurl
- Description: DCS main application URL
- Value: In your web browser, log on to the cockpit, and select an account.
  - Go to *Connectivity -> Destinations* and select ACRS_CORE_DEST.
  - Under *Destination Configuration*, copy URL value.

    For example, https://gstrappxxxxx.example.com/acrscore
  - Maintain the value without prefixing **https**.

    For example, *<gstrappxxxxx.example.com/acrscore>*

Receiver: eSign_ASP_server_upload

- Parameter: eSignASPGatewayUploadURL
- Description: eSign ASP gateway URL to upload raw content
- Value: Enter received eSign ASP gateway URL to upload raw content.
  For example, https://esigngatewayurlupload.com
- Maintain the value without prefixing **https**.

For example, *<esigngatewayurlupload.com>*

- Parameter name: Credential Name/ eSignASPcredentials
- Description: user credential for eSign ASP client ID and client secret key
- Value: You would have received eSign ASP client ID and client secret key after eSign registration**.**
  - Use this ID and secret key as name and password while creating user credentials as mentioned in step 12 below.

    User credentials act as alias or reference to client ID and secret key maintained in security materials.
  - Enter the *<user credential>* created.

Receiver: eSign_ASP_server_download

- Parameter: eSignASPGatewaydownloadURL
- Description: eSign ASP gateway URL to download signed filing document content
- Value: Enter received eSign ASP gateway URL to download signed filing document content.

  For example, https://esigngatewayurldownload.com
- Maintain the value without prefixing **https**.

  For example, *<esigngatewayurldownload.com>*

12. Create user credentials in Web UI of the SCP integration tenant by performing the following steps:
    - In the SCP integration Web UI, from the menu, choose *Menu* -> *Monitor*.
    - Click *Security Material*.
    - Choose the *Add* button, and select *User Credentials*.
    - In the *Add User Credentials* screen, enter the following details:

Field name	User action and values
Name	Enter Credential Name
Description	Free text
User	Enter username/client_id
Password	Secret key
Repeat password	Secret key
SuccessFactors	Un-selected

    - Choose *OK* to save the credentials.

13. Deploy the modified integration flow.

Note

After the deployment, check if the integration flow is in **Started** state. You can check this by choosing *Monitor-> Manage Integration Content*.

After successful deployment, the endpoint URL is https://<SCP Integration Tenant *URL>>/http/dcs/esign*.

www.sap.com/contactsap

Material Number: