BlackLine SAP Core

External API configuration

Valid from version 2102

Prepared By: BlackLine Support
Release Date: 2021-02-25
Document Version: 2.1

Contents

1. Revision History

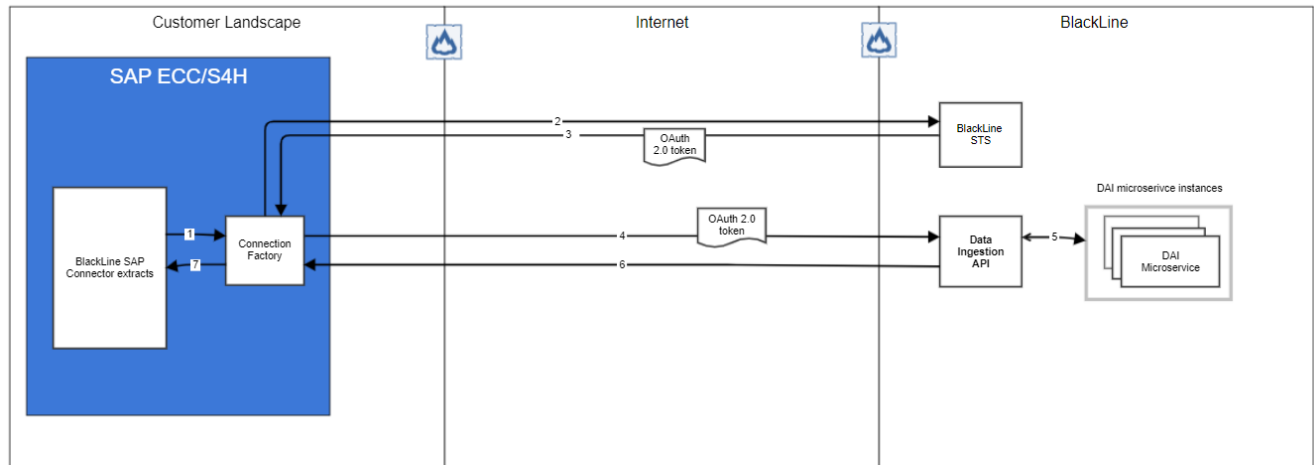| Version	Date	Writer/Editor	Description/Notes
1.0	15-05-2020	Arjan Aalbers	Initial for release EC2005.00.00 from SAP Technology group
1.1	17-08-2020	Guido Buitendijk	Removed Release version from the first page, and added the compatible Release S2009 to par.4
1.2	18-08-2020	Arjan Aalbers	Updates for Logical File and pre-req Firewall + DNS
1.3			Changes in par.6.3
2.0	19-01-2021	Arjan Aalbers	rewrite for EC2102 release
2.1	25-02-2021	Arjan Aalbers	Update certificate upload added currency rates api for v2105

[Intentionally Left Blank]

2. Overview

This document explains how to setup the API access from the Customer SAP system to the BlackLine Cloud API's. It is meant for Implementation Consultants.

3. Process flow

This process flow is from the SAP customer landscape point of view. The processing within BlackLine has been simplified.

Step	Description
1	The SAP Connector query the data and send result to the Connection factory
2	The connector presents API key to BlackLine STS to authenticate via HTTPS POST
3	BlackLine STS validates API key and returns an OAuth 2.0 token to the Connection Factory. This token is valid for the duration of the token's TTL (time-to-live) and can be used for subsequent communication with the BlackLine Data Ingestion API.
4	The Connection Factory executes an HTTPS POST/PUT/GET operation passing the OAuth 2.0 token inserted into the authorization header of the request.
5	The Data Ingestion API hands the payload over for processing. An ImportID is generated
6	The result HTTP 202 Accepted + ImportID is returned in the response
7	The Connection Factory reports the result to SAP Connector Extract program for logging

4. Message Flow

This process flow is from the SAP customer landscape point of view. The processing within BlackLine has been simplified.
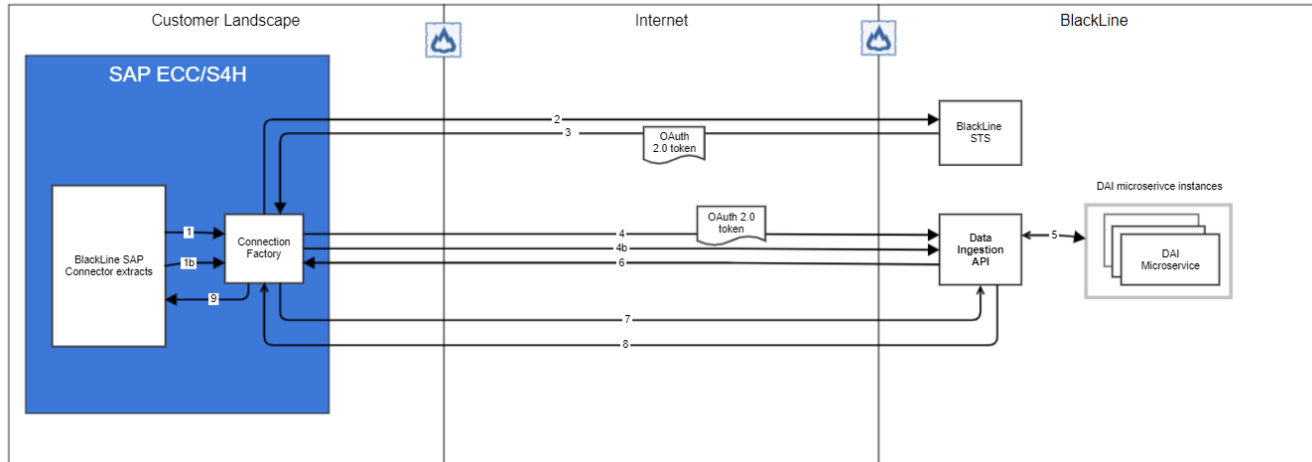
Step	Description
1	The SAP Connector extract program queries the data in SAP and send result to the Connection factory
1b	For Open Items and Transactions, the body is split into chunks of 50 MB and multiple calls to the Connection Factory are made
2	With the First call the connector presents API key to BlackLine STS to authenticate via HTTPS POST
3	BlackLine STS validates API key and returns an OAuth 2.0 token to the Connection Factory. This token is reused for calls 4, 4b and 7. If the token expires in the meanwhile, a new token will be fetched
4	The Connection Factory executes an HTTPS POST/PUT/GET operation passing the OAuth 2.0 token inserted into the authorization header of the request.
4b	For the Transaction Matching API (Data source) multiple call must be made if the payload exceeds 50MB
5	The Data Ingestion API hands the payload over for processing. An ImportID is generated
6	The result HTTP 202 Accepted + ImportID is returned in the response
7	The Connection Factory queries the Status of the import via the Status API using the ImportID. If the returned status (8) = PENDING or SYSTEM INFORMATION , the connection factory waits for 2 seconds and tries again with a maximum of 10 tries.
8	The status API return the import status: SUCCESS, SUCCESSFULLY COMPLETED, DATA_ERROR, PENDING or SYSTEM INFORMATION
9	The import status is returned to the SAP Connector extract program

5. Prerequisites

- Check with the Network team or SAP Basis team in your organization that a Firewall must allow traffic from your SAP system to the BlackLine Instance on Port 443
- Check with the Network team or SAP Basis team in your organization that the target host (*{environment}.*api.blackline.com) can be resolved to an IP address from your SAP server. This might need a change to your DNS or host file configuration.

- SAP Connector version EC2005.00 or S2009.00 or higher is installed on the customer system.
- BLG version 2102 is installed on the customer system.

- An SAP user is available with SAP_ALL or the following authorizations:
    - transaction SM59
    - transaction STRUSTSSO2
    - transaction /BLG/EC_MAINT_ECON
    - transaction FILE when the customer has a Runtime Database License or has a SolEx contract
- The BlackLine Implementation Consultant has provided for your BlackLine instance:
    - The API and Token endpoint URL
    - The Client ID   (token server logon user)
    - The Client Secret   (token server logon password)
    - The Client Guid

        [Example on next page]

From: BlackLine - Do Not Reply <donotreply@blackline.com>
Date: 8 May 2020 at 18:34:50 CEST
To: …..
Subject: Case # 00269443: API - Enable All API has been updated.

*** CASE UPDATE ALERT ***

Company: Blackline, Inc.
Case: 00xxxxxx: API - Enable All API
BlackLine Owner: ……

Case Comment: Hello,
The reporting, task, dataingestion, and the matching API has been enabled to
https://xyz.us2.blackline.com

Below are the connection information for Reporting, and Dataingestion.

Dataingestion is for dataingestion, matching, and task.
Regards
……

Reporting API info:
****
Environment = US2
ClientID   = my_bl_user
Client Secret = 1Q3.6;!EsL#u
Scope   = ReportsAPI instance_6610EFC3-2F5C-4E82-BB96-D8442CFA5946
URL to request token = https://my_bl.api.blackline.com/authorize/connect/token
****

Dataingestion API Info:
****
Environment = US2
ClientID   = my_bl_user
Client Secret = 1Q3.6;!EsL#u
Scope   = DataIngestionAPI instance_BEA772D9-890A-4710-96E10-3F6CEB98BB
URL to request token = https://my_bl.api.blackline.com/authorize/connect/token
****

Click on the link to access the case: https://blackline.my.salesforce.com/…...

Sincerely, The BlackLine Support Team

BLACKLINE

The world's most trusted solution for Finance Controls and Automation.

6. Preparation

6.1 Setup API user

- Go to the user admin grind on your BlackLine instance and create an API user

- Note the Login ID (example api_user)
- Generate the API key for that user and copy the Guid for later use

- Assign the correct API authorizations in Authorized Roles
  Apart from the API Access option also check the boxes for the domains in which this user must upload data

6.2 Get SSL certificate

Note: It's recommended to use Chrome or Microsoft Edge for this step. The screenshots below are made with the Chrome browser.

- The SSL client certificate of the BlackLine webserver must be installed on the SAP system. To obtain the certificate, open a browser and enter the token endpoint url (https://*{environment}.*api.blackline.com/authorize/connect/token Or
  https://{environment}.api.blackline.com/sts/diagnostics/status). (Applies to the new APIgee framework hosted on Google Cloud Platform GCP)
  Ignore the error message and click on the lock icon to see the certificate details.
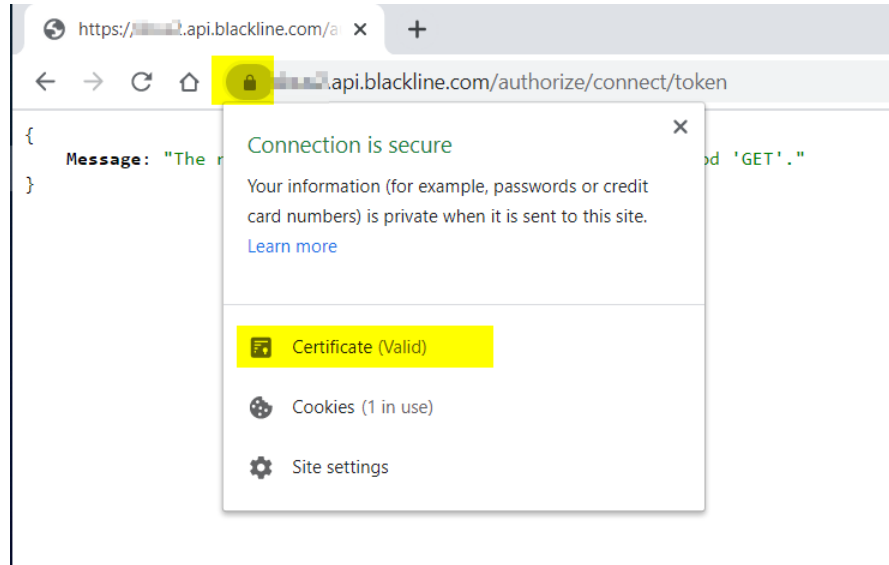
- Click on "Certificate (Valid)" to display the certificate details and switch to the "Details" tab
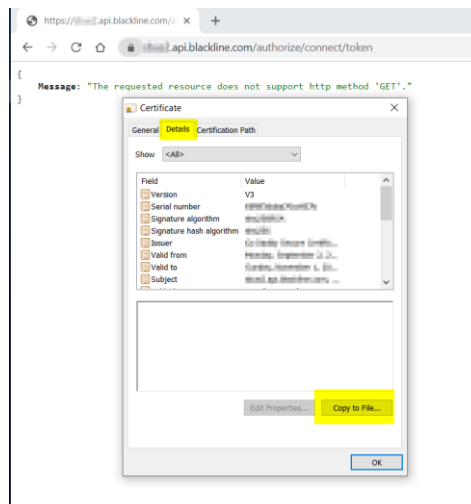
- Use the "Copy to File..." option to export the certificate as "Base-64 Encoded X.509 (.cer)"
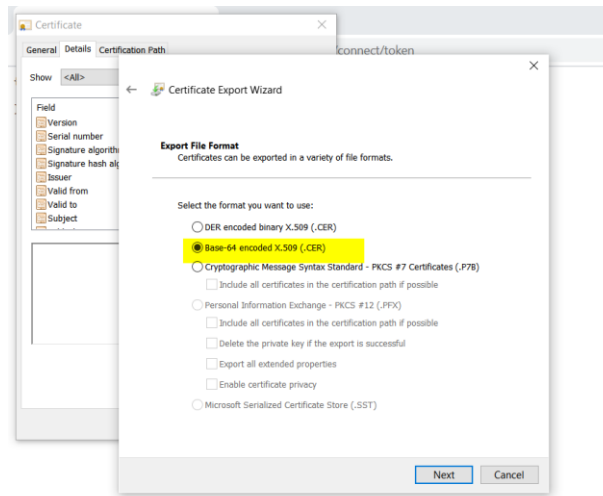
- Save the file to your local PC for later use.
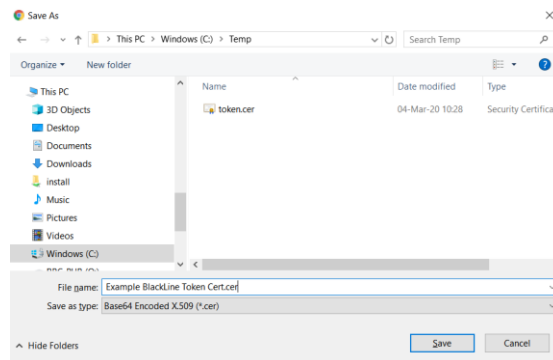
- Repeat this for the whole certificate chain.
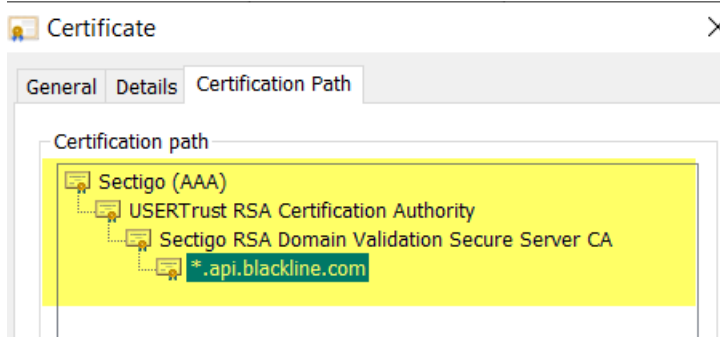
In this example 4 certificates must be downloaded.

7. Setup

7.1 Logical File (optional)

In case of "*Runtime Database License"* the configuration is not stored in Database tables but in a configuration file on the SAP application server. For this the Logical File and Path must be configured. **For all other licenses, skip chapter 6.1.**

- Start transaction /BLG/EC_MAINT_ECON
  If you get an error message:

  "Logical file '/BLG/EC_CON_CONFIG_FILE' p1 'ConfigHeader' validation failed. Please check trx FILE"
  Or
  "Please maintain logical path '/BLG/EC_CON_CONFIG_FILE' for storage of config files"

  Then continue with the following, else **skip chapter 6.1**

- Start transaction FILE

  - On newer SAP systems you might get the question to choose between the new SFILE and the old FILE transaction. Choose in this case for the Old FILE option.
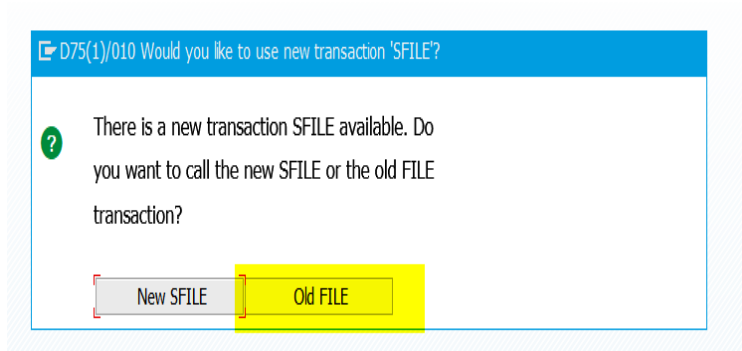
- Select node "Logical File Path Definition" + position to navigate to path "/BLG/EC_CON_CONFIG_FILE"
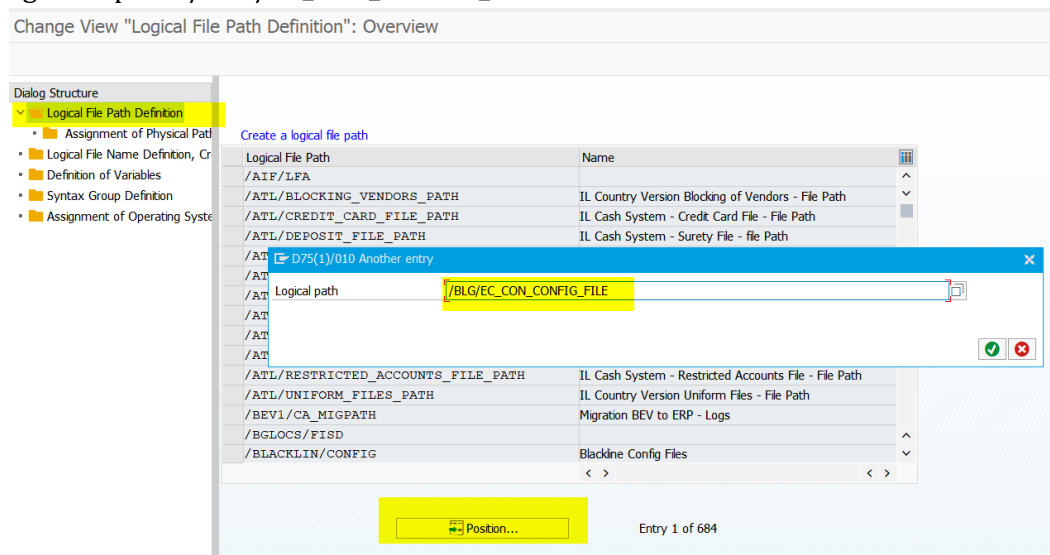
- Select the entry and create a new entry for "Assignment of Physical Paths to Logical Path".
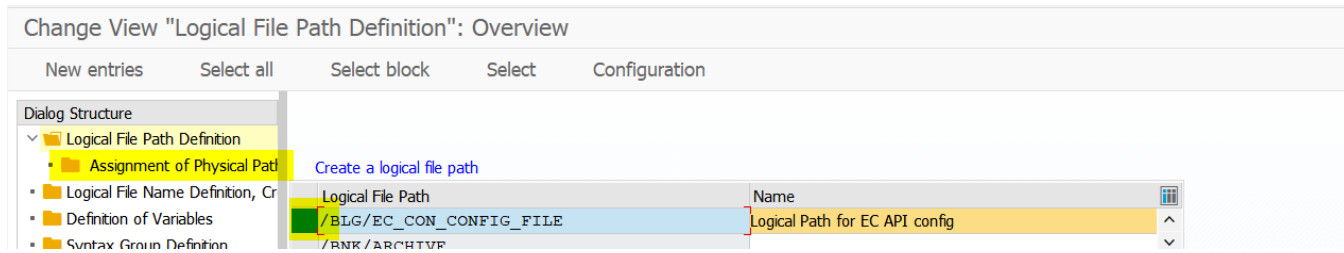
- According to your OS (Windows, Unix,…), choose a suitable folder and use "<FILENAME>" as the placeholder.
  Example for Unix:
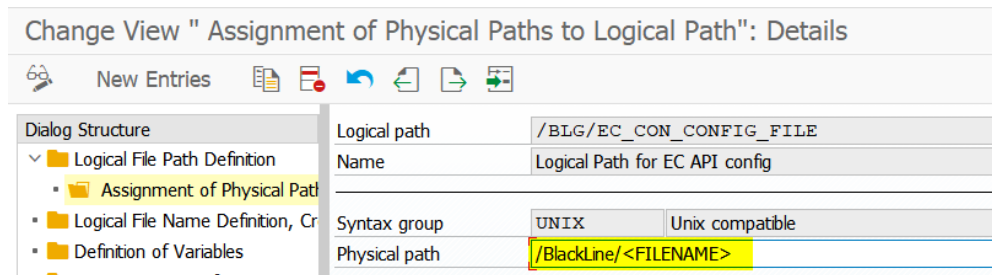
Picture 7-1

- Save your entry and assign a workbench request when prompted.

7.2 **Upload SSL Certificate**

For https traffic to the BlackLine API and Token endpoint the client certificate must be uploaded in SAP.

- Start transaction STRUST or STRUSTSSO2

- Switch to change mode when needed and open Folder "SSL client SSL Client (Standard)" and select your SAP instance
  (You can also use a custom defined PSE to store the Blackline Certificates)
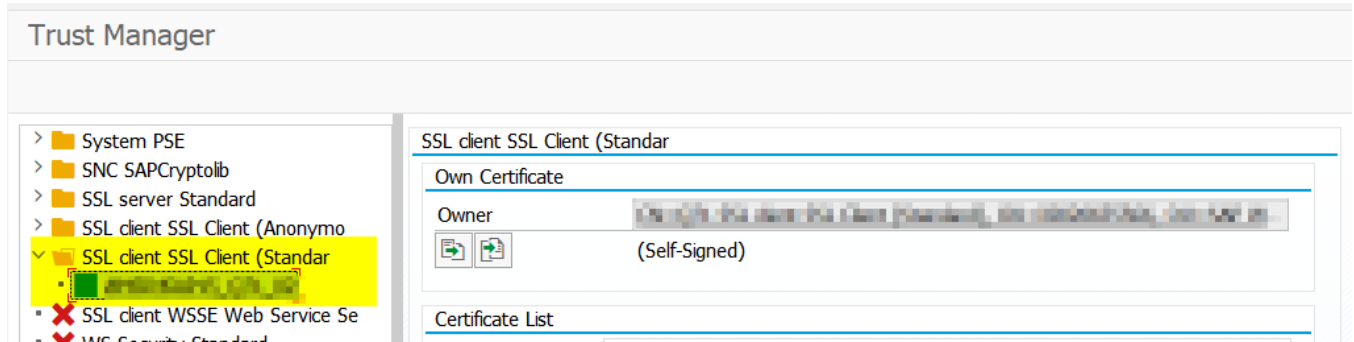
- Use the import button to upload the .cer file downloaded in the Preparation step
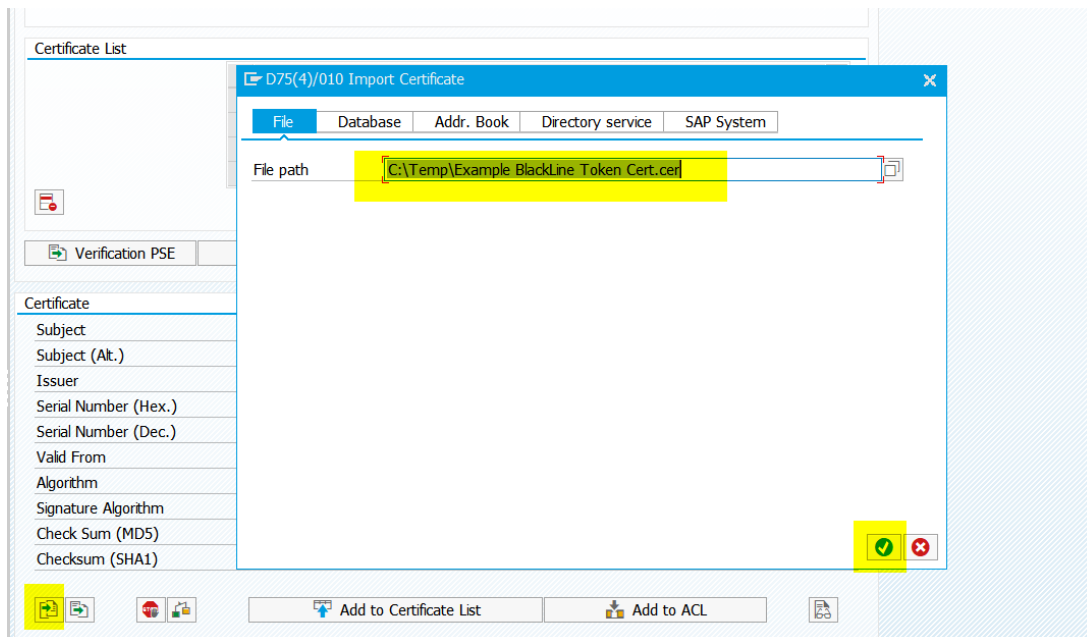
Use the "add to Certificate List" to add the certificate to the list.

- Save

- Repeat the upload for the other certificates in the chain.
  In this example 4 certificates in total are added to the certificate list
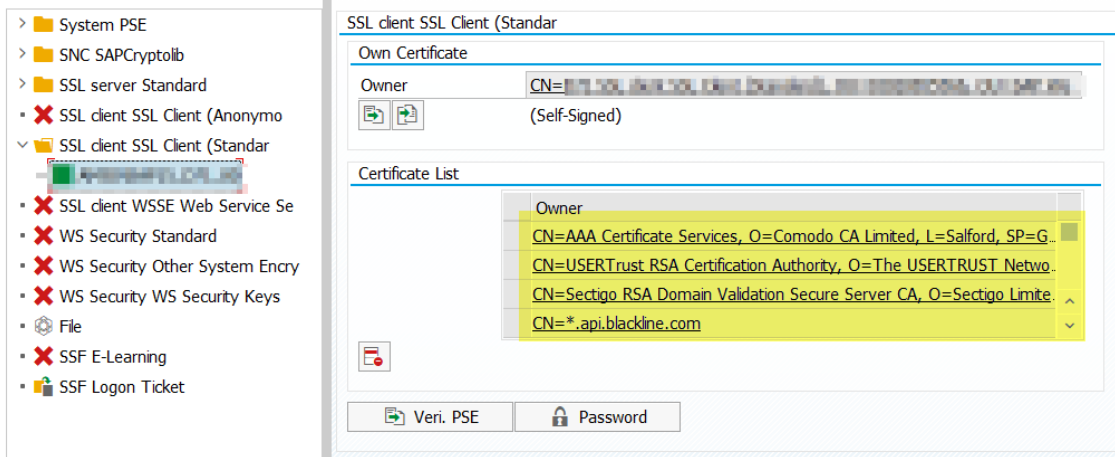
7.3 Create HTTP destinations

Use transaction SM59 to create 2 HTTP destinations:

7.3.1 HTTP Destination for OAUTH

- Enter a name for the HTTP destination. For example: BL_OAUTH_[PROD/SANDB/TEST/…]
  Choose type = G
- Enter a description (1) and press enter

  Ignore the message "HTTP connections may not be secure"

- From the Url to request token copy the hostname (without https:// or suffixes) into Target Host
- Leave the Service No. empty
- Enter rest of the url into the field Path Prefix

- Switch to the Logon and Security Tab
- Switch to Basic Authentication
- Use the ClientID provide in the email for User
- Use Client Secret provide in the email as Password

- Set SSL to active and select the Certificate list containing the uploaded BlackLine Certificates - DFault SSL Client (Standard)

- Switch to the Special Options Tab
- Set HTTP 1.1 active
- Set Compression to Active(depends on Mime Type)
- Set Compressed response to Yes
- Set Accept Cookies to Yes (All)
- SAVE the HTTP destination

7.3.2 HTTP Destination for API

- Enter a name for the HTTP destination. For example: BL_API_[PROD/SANDB/TEST/…]
- Choose type = G
- Enter a description (1) and press enter
  Ignore the message "HTTP connections may not be secure"

- From the Url to request token copy the hostname (without https:// or suffixes) into Target Host
- Leave the Service No. empty
- Leave Path Prefix **empty**

- Switch to the Logon and Security tab
- Choose logon – Do not use a User
- Set SSL to active and select the Certificate list containing the uploaded BlackLine Certificates - DFault SSL Client (Standard)

- Switch to the Special Options Tab
- Set HTTP 1.1 active
- Set Compression to **InActive**
- Set Compressed response to Yes
- Set Accept Cookies to Yes (All)
- SAVE the HTTP destination

7.4 Generate API configuration

- Start transaction /BLG/EC_MAINT_ECON and switch to change mode

- Use "Generate all API config" to start the wizard
- Choose Data Ingestion Wizard in the popup, leave "*With OAuth 2.0 support*" checked.

Enter a Prefix to generate unique names

- Choose the HTTP destination for API create before (6.3.2)
- Choose the HTTP destination for Oauth token created before (6.3.1)
- Enter the user created in the user admin grid
- Enter the generated API key from the user admin grid
- Enter the API Scope provided in the Email

- Select all the API's needed for your integration with BlackLine

- Recommended selection is:
  - BL_DAI_DS   BlackLine DAI Data Sources
  - BL_DAI_GL   BlackLine DAI Accounts
  - BL_DAI_MCB   BlackLine DAI Multi Currency Balances
  - BL_DAI_SBL   BlackLine DAI Subledger
  - BL_DAI_STS   Blackline DAI Status
  - BL_DAI_CRR   Blackline DAI Currency Rates *(for version v2105 and higher)*

- Note: The API's *BlackLine DAI Status* and *BlackLine DAI OAuth Token* will always be generated, regardless of the checkbox.

The result should look like this:

7.5 Individual maintenance

The wizard is convenient to generate all configuration in a single step.

You can also view, edit and create individual API – or Authentication configuration

Example 1: manual creation of an API configuration

- On the main screen (in change mode) choose "new configuration"

- Choose the predefine API Type in type popup (in this example "Accounts")
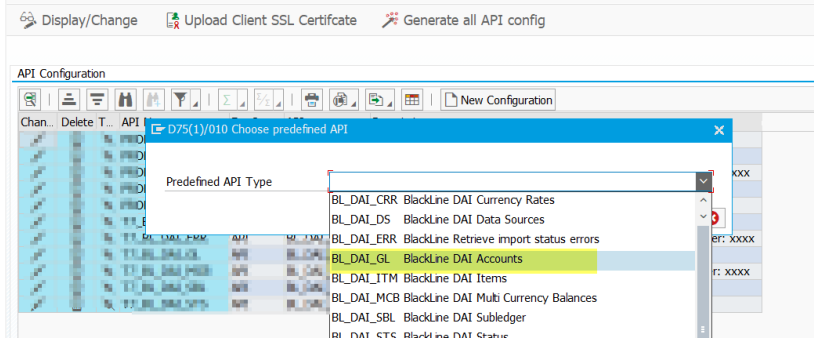
- API Detail Screen

1. Configuration name   – *Fill out with a logical name*
2. Description   – *Add a meaning full description*
3. Http Connection   – *choose the connection configured in step 7.3.2*
4. Path – endpoint of the API
5. Method used to call the API
6. Content type
7. Authentication method
8. API Scope   - *Fill out the correct Guid provided to you*
9. Response is Binary
10. OAuth Configuration   - *Select the OAuth configuration*
11. Import Status API   - *Select the API to retrieve the Job status*
12. Check configuration   - *Checks if all the mandatory fields are filled*
13. Save/Cancel

Example 2: Changing the OAuth configuration

You can change the OAuth configuration, when for example the API key has expired and needs to be regenerated and must be updated in SAP.

1. On the main screen select the edit button for the OAuth entry
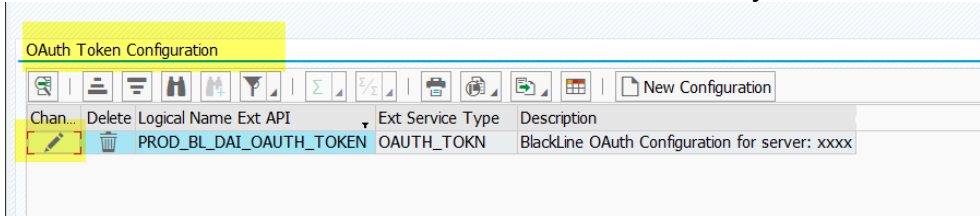
2. The fields on the Details screen

1. Configuration name
2. Description
3. HTTP destination configured in step 7.3.1
4. Method used to call the Token Service
5. Type of Grant
6. Logon id of the user created in the user admin grid (See chapter 6.1)
7. API key (re-)generated in the user admin grid
8. Hide/Show API key
9. Check configuration   - *Checks if all the mandatory fields are filled*
10. Save/Cancel

8. Test

After generating the API configuration, you can test if connection to API's end point works.
The connection test works from the main screen or in display mode from the API detail screen

- Main Screen:
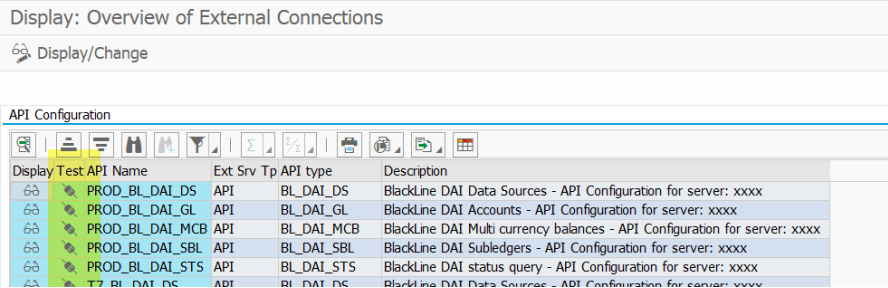
- Or Open one of the API detail screen by pressing the display icon
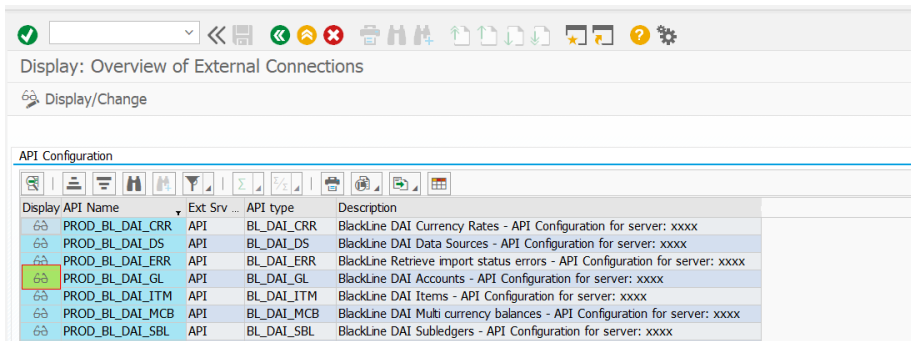
- Use the "Test Connection button"
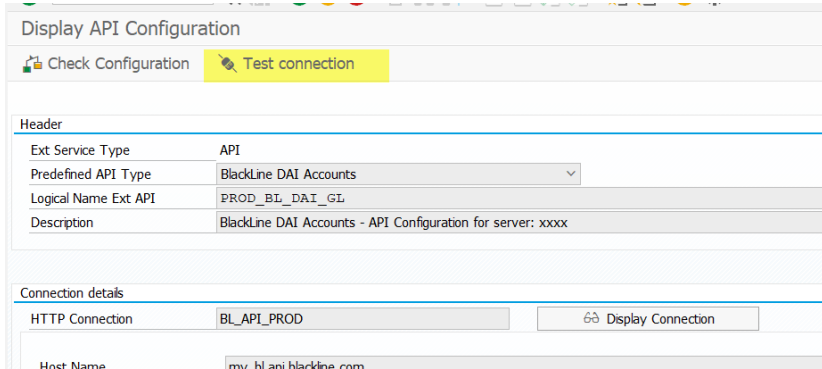
- The test will call first the OAuth service to get a token
  Secondly a call is made to the API using invalid data. This prevents that actual postings are made via the API's
  An error message "invalid_format" can be expected.

- Some API's use import parameters in the URL. In a popup you can enter a value for this parameter

9. Troubleshooting

Common errors:
- SSL error- Most Likely an uploaded certificate has expired.
  - Please check STRUSTSS02 for the validity of the BlackLine certificates

- Failed to get OAUTH Token: 400 Bad Request - invalid grant - Authentication failed.
  The API key has expired or entered incorrectly into the SAP OAUTH configuration
  - Generate a new key in the BlackLine user admin grid and copy into the SAP OAUTH configuration
  - Make sure the use can also do has the roles to upload the data in the specific domain

Potential error codes:

Status Code	Error Code	Description
400	invalid client	SM59 -> Token service destination -> password(client secret) invalid
400	invalid grant + authentication failed	API key is invalid. Go to BLC user admin grid and regenerate new key or username is invalid
400	invalid grant	- grant_type must be "password" - API key not valid/has expired
400	invalid scope	check scope is correct filled in the API configuration
400	version_required	User has not supplied an import version
400	version_not_supported	The version supplied by the user is not supported
400	no_data	User has not supplied any data for import
400	invalid_format	The data was send is in the wrong format.
401	unauthorized	The token request was send with wrong credentials
401	unauthorized	Authentication failed due to invalid authentication credentials
403	Forbidden	Authorization failed due to insufficient permissions The user in the BlackLine Cloud instance has insufficient API authorizations
404	Not Found	Check host/url/path in SM59 destination
405	method not allowed	Check API uses POST or PUT
429		User has made more requests per second than the allowed threshold
500	Internal Server error	Server has encountered an internal error and cannot complete the request.
500	Native SSL error	The TLS version on older ABAP systems might be too low. Use tls 1.2
500	SSL Peer Certificate untrusted	Check STRUSTSSO2 if the BlackLine Certificates are uploaded correctly and still valid
503	Service Unavailable	The web server is currently unavailable