ELO Suite for SAP ArchiveLink® (SAP NetWeaver® & SAP S/4HANA®)

ELO Smart Link for SAP® ERP Secure login via SNC

Table of contents

ELO Smart Link for SAP® ERP

Secure login via SNC

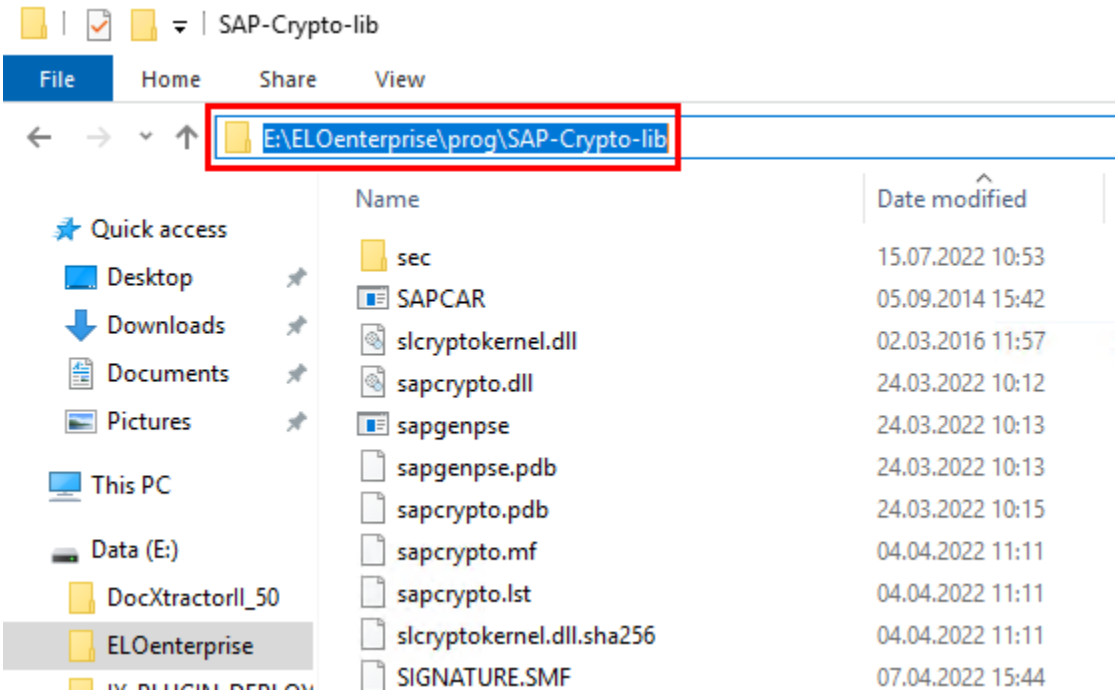
Secure login via SNC

To use secure transmission (SNC) with an RFC connection, you must take the following additional setup steps.

First, a cryptography library (*SAPCRYPTOLIB*) must be provided on the ELO server. You can also find information about this in SAP Note *1848999* in the SAP ONE Support Launchpad.

At the following link, you can download the current cryptography library that is suitable for the ELO server architecture: SAP ONE Support Launchpad.

Save and extract the downloaded library to a new folder in the *prog* directory of your ELO server. In our example, the path is *E:\ELOenterprise\prog\SAP-Crypto-lib*.



Set up Personal Security Environment

A Personal Security Environment (PSE) is required for SNC encryption. This is a keystore or truststore. It contains:

- your own public key data, including your own private key
- your own public key certificate
- all public key certificates of the trusted communication partners

The SAPCryptoLib contains the *sapgenpse* tool. This is used to set up the PSE. The tool works with the *SECUDIR* environment variable.

Create directory

A separate directory is needed to store the PSE file. In our example, the directory *sec* was created under the following path: *E:\ELOenterprise\prog\SAP-Crypto-lib\sec*.

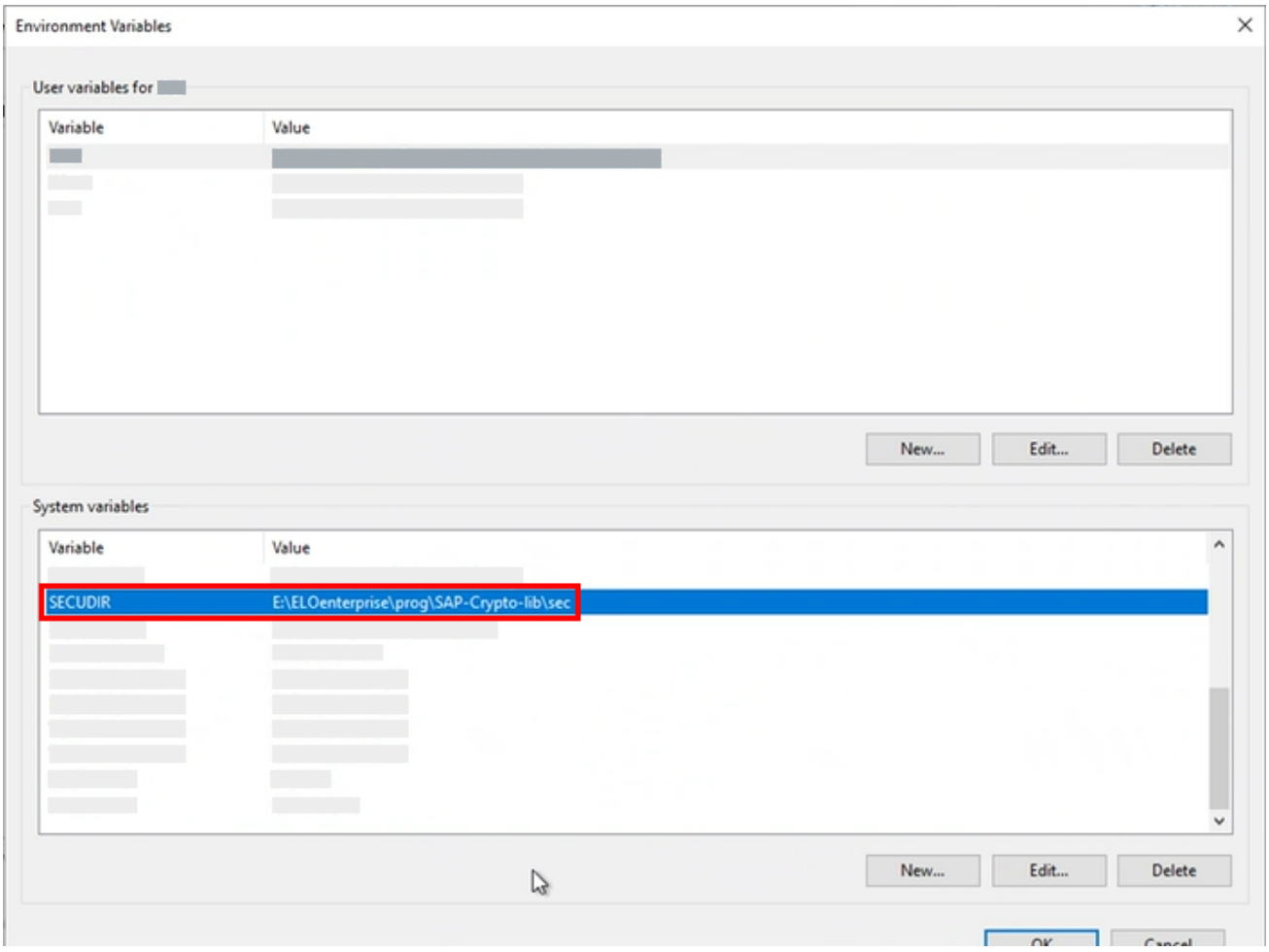
Set SECUDIR environment variable

The Tomcat needs to know where to find the PSE and credentials to use for SNC encryption.

You need to set the *SECUDIR* environment variable to the location of the PSE and credentials (*cred_v2*).

Windows system variable

SECUDIR is stored as a system variable in the Windows system properties under *Environment Variables*.



Now you need to create your own certificate store (Personal Security Environment; PSE).

Create PSE

There are two ways to create the PSE:

-

with an existing certificate
- while simultaneously generating a self-signed certificate

For this scenario, create the PSE *while simultaneously generating a self-signed certificate*.

Create PSE and generate self-signed certificate

The `get_pse` command is used to generate a PSE, a private key, and a public certificate. You need to specify the distinguished name.

```
sapgenpse get_pse -p <pse file> [other-options] [distinguished name]
```

The `-p` option specifies the file name of the PSE you are creating. The distinguished name consists of the following elements:

- `CN=<common name>`
- `OU=<organizational unit>`
- `O=<organization>`
- `C=<country>`

The full command could look like this:

```
sapgenpse get_pse -p ELOSLC.pse "CN=ELO Smart Link, C=DE"
```

In this case, you need to change the highlighted parameters to match your information.

When you execute the command, it requests the PIN for the PSE. Since the PSE is created at this point, you can choose any pin you like. The *sapgenpse* tool creates a PSE in the selected directory or in the *SECUDIR* directory. If you need to, you can generate a *SigningRequest* for the self-signed certificate. However, you can also do this afterwards.

You can then have the certificate signed by a certificate authority and updated in the PSE.

The Tomcat service user that contains ELO Smart Link needs an access right.

Create access right for the Tomcat

The Tomcat must have credentials at runtime to access the PSE. The `seclogin` command is used to open the PSE and create credentials.

```
sapgenpse seclogin -p <pse file> -O [<NT_domain>\]<user_ID>
```

The `-p` option specifies the file name of the PSE you are creating.

The `-O` option specifies the user name (possibly including domain) that you are creating SSO credentials for.

The full command could look like this:

```
sapgenpse seclogin -p ELOSLC.pse -O SYSTEM
```

When you execute the command, it requests the PIN for the PSE. The *sapgenpse* tool then creates a credentials file (*cred_v2*) in the *SEC* directory. If the file already exists, it is updated.

Exchange public key certificates

To enable SNC communication, ELO and the SAP system must be able to identify each other. This is done using the public key certificates that are stored in the PSE. To allow identification, the SAP public key certificate must be stored in the ELO PSE, and vice versa.

If SNC encryption is used to communicate with multiple SAP systems, the public key certificates for all of these SAP systems must be imported.

Export your own certificate

If you generated a new certificate with the *sapgenpse* tool, the public key certificate can be exported with the `export_own_cert` command.

```
sapgenpse export_own_cert -o <output file> -p <pse file>
```

The `-o` option specifies the file name of the exported certificate.

The `-p` option specifies the file name of the PSE you are creating.

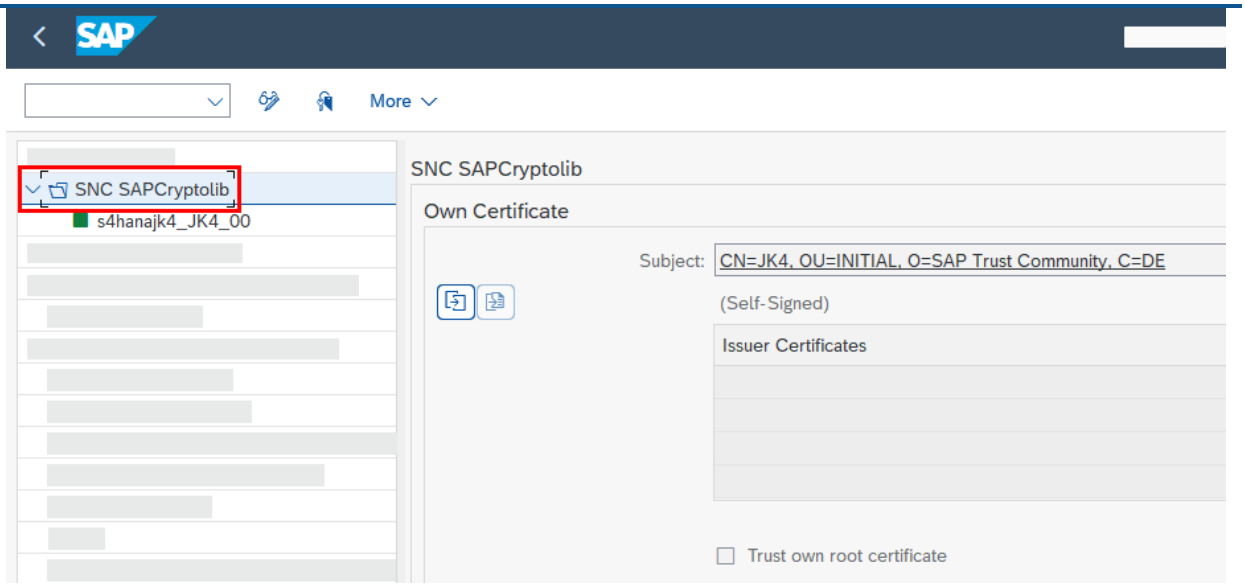
The full command could look like this:

```
sapgenpse export_own_cert -o smartLink.cer -p ELOSLC.pse
```

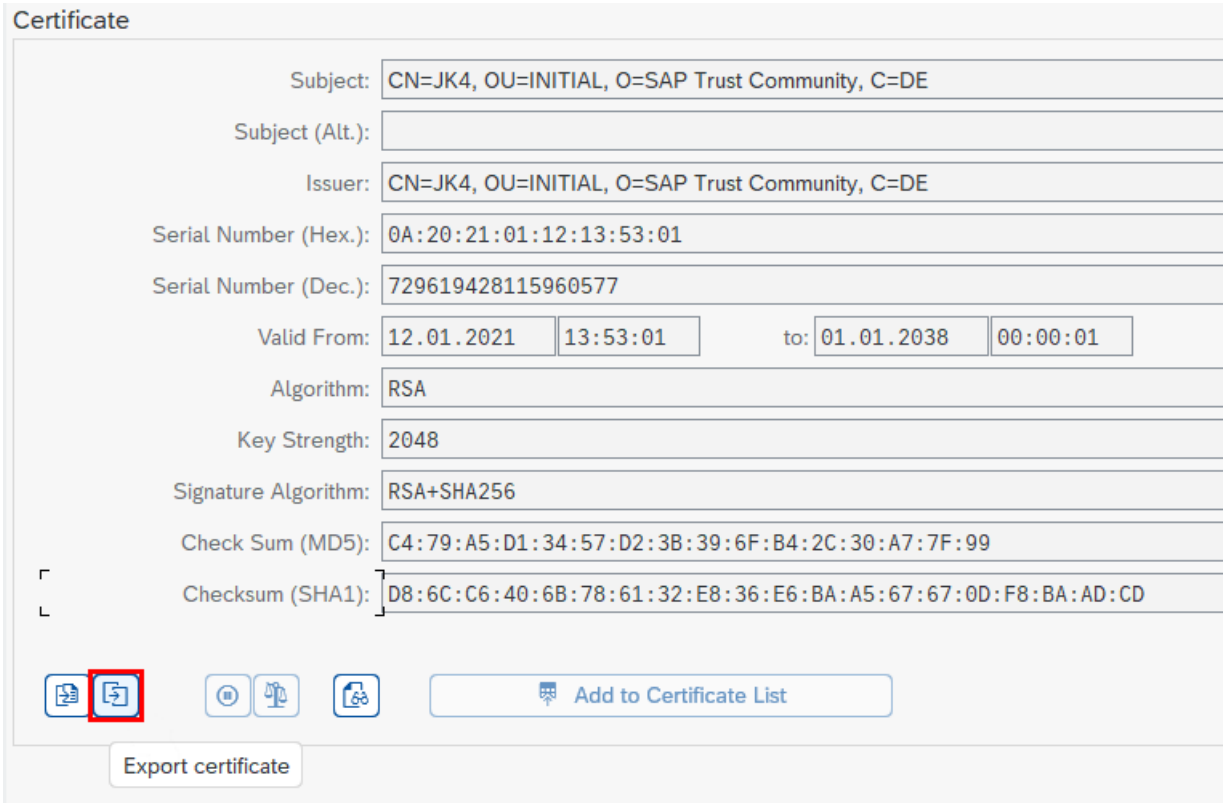
When you execute the command, it requests the PIN for the PSE. The *sapgenpse* tool then stores the public key certificate under the specified file name.

Export SAP certificate

1. In the SAP system, switch to the transaction *STRUST*.
2. Open the *SNC SAPCryptolib* menu item.



3. Double-click your own certificate to select it.
4. In the lower part of the certificate information, select the *Export certificate* button and enter an export path.



Import your own ELO certificate

Within the *STRUST* transaction in the SAP system, you also have the option to import the certificate you just exported from the ELO system.

1.

Select the *Import certificate* button.

2. Select the file path where you saved the certificate.
3. Select the *Add to Certificate List* button and save the steps afterwards.

Import certificate from SAP

The `maintain_pk` command is used to import the certificate from the SAP system.

```
sapgenpse maintain_pk -a <cert file> -p <pse file>
```

The `-a` option specifies the file name of the certificate being imported.

The `-p` option specifies the file name of the PSE you are creating.

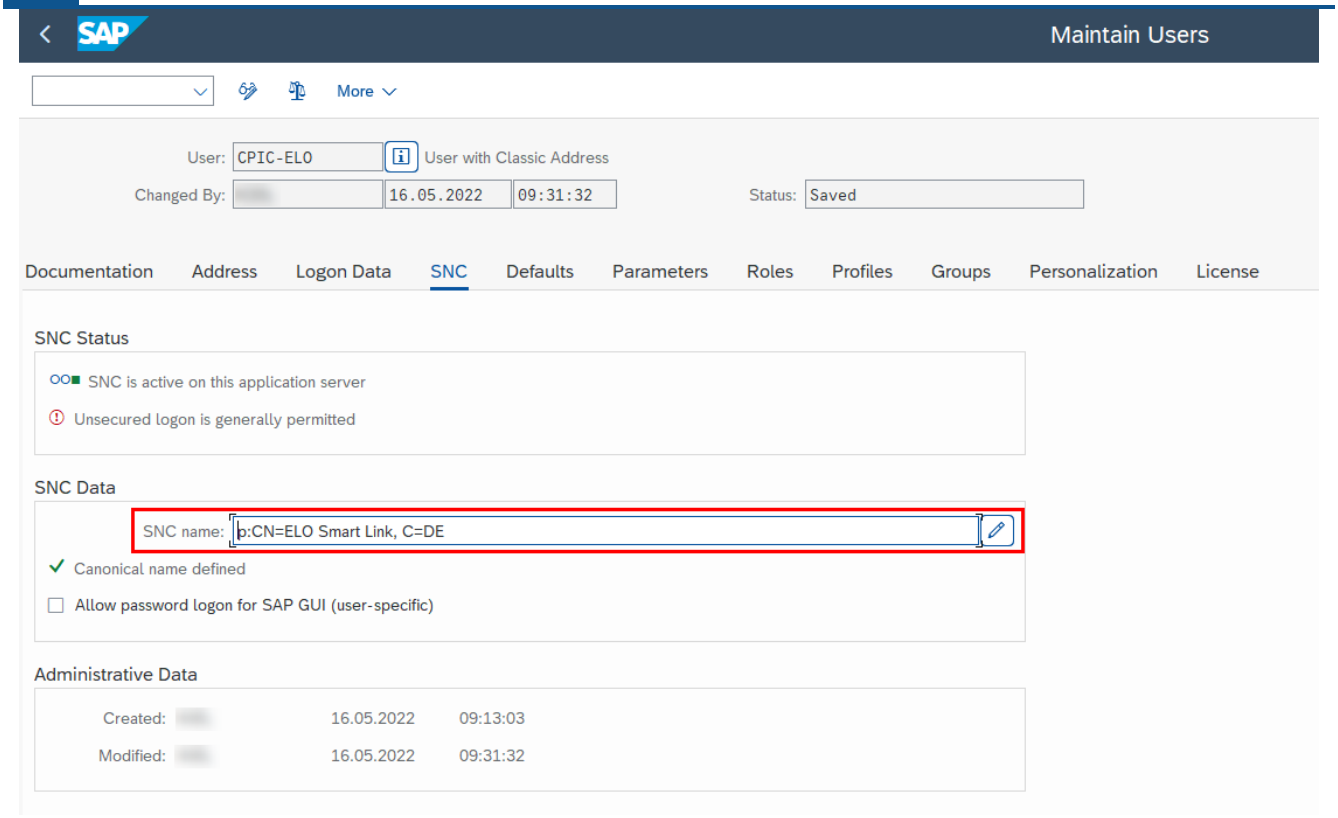
The full command could look like this:

```
sapgenpse maintain_pk -a JK4SNC.cer -p ELOSLC.pse
```

When you execute the command, it requests the PIN for the PSE. The *sapgenpse* tool then imports the public key certificate into the PSE.

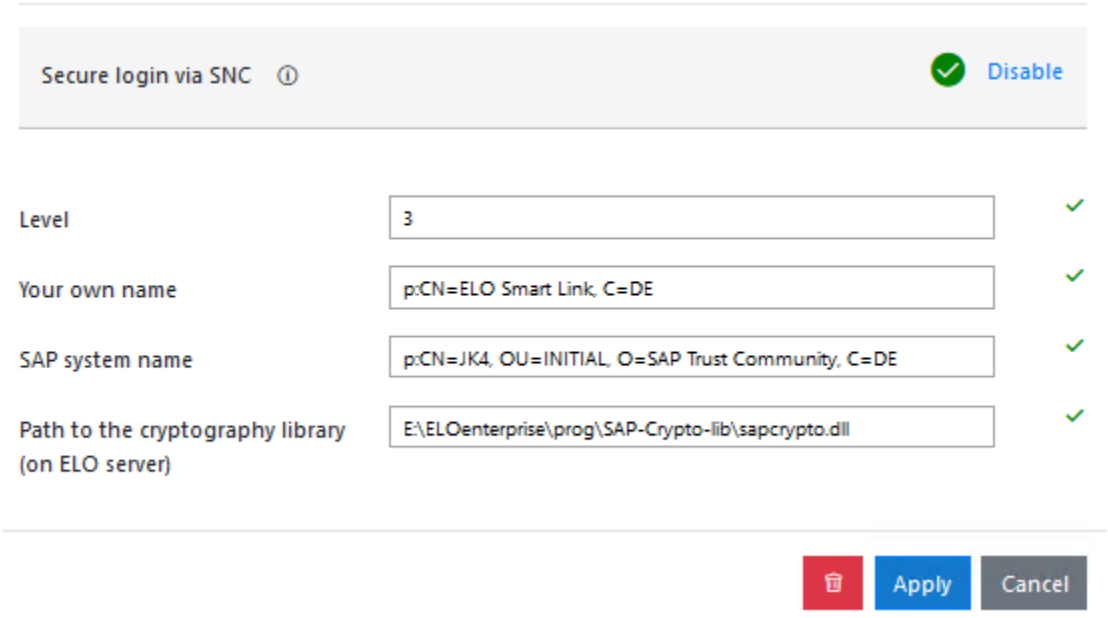
Enter the parameters in SAP

Open transaction *SU01* in SAP and call the corresponding ELO Smart Link user. Switch to the *SNC* tab and enter the SNC name in the *SNC name* field.



Enter the parameters in ELO Smart Link

Now enter the defined parameters.



Level: Specifies the security level to use for the connection. Possible values:

- 1: Authentication only
- 2: Integrity protection
-

3: Privacy protection

Own name: Your own SNC name can be read from the distinguished name in the public key certificate which was used when creating the PSE. The SNC name has the format `p:<distinguished_name>`.

SAP system name: The name of the SAP system. The name can be read from the distinguished name in the public key certificate of the SAP system. The SNC name has the format `p:<distinguished_name>`.

Path to the cryptography library (on ELO server): Specifies the cryptography library path and file name. It must be a valid path on the ELO server where the SAP Java Connector (JCo) can address the library.

If all parameters have been configured and the respective certificates have been exchanged, the ELO and SAP systems can now use SNC.