Encrypt all the things; don't forget your SAP communication!

Encryption is the word recently, especially in the post-snowden time we live in. And there is good reason for that; Your SAP system stores your business-critical data and might very well need additional protection when flowing through your SAP landscape.

The topic of SAP communication encryption is wide, it includes for example communication between SAP servers and the Database, between clients and servers and between SAP servers. This article only discusses the latter, encryption between SAP servers via the RFC protocol. Although there are other ways to realize RFC encryption (for example via certificates), we will zoom in on the method via the SAP crypto library. This description assumes the windows operating system, steps for other operating systems can differ but the general procedure is similar.

Install the required software

To use RFC encryption between SAP servers, perform the below steps on all SAP Servers you want to communicate between. This step might not be not needed as recent systems have this already in place.

Note: any GSSAPI v2.0 compliant software is useable. However, this instruction assumes that the SAP COMMONCRYPTOLIB library is used as it's free and easy to use.

Start by downloading the necessary software from the SAP Service Marketplace.The required files are platform specific but always include at least two files:

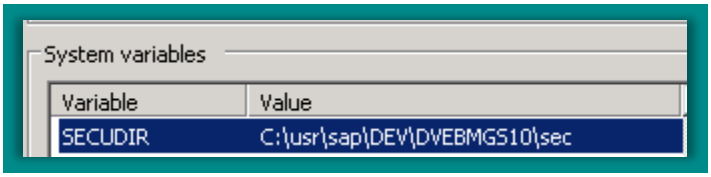
- A sapcrypto shared library
- The sapgenpse program

These files can be unpacked to any place on the server, but the directory set by the profile parameter DIR_EXECUTABLE is recommended (For example C:\usr\sap\DEV\DVEBMGS10\exe). This also makes sure the required environment variables are set correctly.

Note: in windows-only environments there is also the option of working with the Kerberos dll's as mentioned in note 352295, although this works, this solution is not supported by SAP.

Operating system adjustments

Set the environment variable 'SECUDIR' to <drive>/usr/sap/<SID>/<instance>/sec. For example:



Make sure that to add the secure entries for sapdp<XX>s and sapgw<XX>s in /etc/services, for example:

```
sapdp10s        4710/tcp
sapgw10s        4810/tcp
```

(This is automatically done after installation of recent SAP versions.)

Required parameters

To make SNC active it is needed to set some specific parameters in the instance profile of the SAP system.

snc/enable
Set this value to '1' to enable SNC communication.

snc/gssapi_lib
This parameters points to the cryptographic library used by the SAP system. Include the full path and filename (or when placed in the $(DIR_EXECUTABLE) folder, use the profile parameter name). For example *C:\windows\system32\gx64krb5.dll*
Note: on newer versions of SAP (Web AS 6.20 and newer, also include the following parameters, pointing to the same file:

- sec/libsapsecu
- ssf/ssfapi_lib

ssf/name
This is the name of the used library, use value 'SAPSECULIB'.

snc/identity/as
This value contains the SNC name of the server. It is in the form: 'p:SAPService<SID>@<domain>'.
For example: snc/identity/as = p:SAPServiceDEV@mooo.com.

SNC defines three levels of data protection:

1: Secure authentication only: Only the sender of the communication is verified, no protection of actual data.

2: Data integrity protections: The authenticity of the sender is verified, as well as whether the actual data was changed during transfer.

3: Data privacy protection: The authenticity of the sender is verified, and the integrity of the data is verified and the data is encrypted so it is unreadable during transport.

snc/data_protection/min
This parameter defines the minimum security level that the server requires. Communication with a lower security level are not accepted. The default value is 2 (data integrity protection), set this to 3 (data privacy protection).

snc/data_protection/max
This parameter defines the maximum security level that the server accepts. Communications with a higher security level are not accepted. The default value is 3 (data privacy protection). set this to the highest value (3).

snc/data_protection/use
This parameter defines the default security level that is used by the server for RFC and CPIC connections. Set this to value 9.

Handling of non-SNC connections

By default, an SNC enabled system will not accept non-SNC communication anymore. Although this sounds as a good option, it might be wise to, at least for now, keep support of non-secure communication to not break business processes.

The following parameters permit non-SNC communication to the SAP system if their value is set to 1. The default value is 0 (don't permit)..

snc/accept_insecure_gui
If this value is set to 1, non-SNC connections from SAPGUI are accepted.

snc/accept_insecure_rfc
If this value is set to 1, non-SNC RFC connections are accepted.

snc/accept_insecure_cpic = 1
If this value is set to 1, non-SNC CPIC connections are accepted.

snc/r3int_rfc_secure
If this value is set to 1, the SAP system will use SNC for all internal RFC and CPIC connections. Note that there is a performance penalty for enabling this option. Try to set it to 1. If performance remains acceptable, it is good to keep it, otherwise set it to 0.

snc/accept_insecure_r3int_rfc
If this parameter is set to 1, then the SAP system will accept non-SNC RFC connection from other internal destinations even if acceptance of non-SNC RFC connections is not allowed (i.e. snc/accept_insecure_rfc is set to value 0). Only set this value to 1 if it is required.

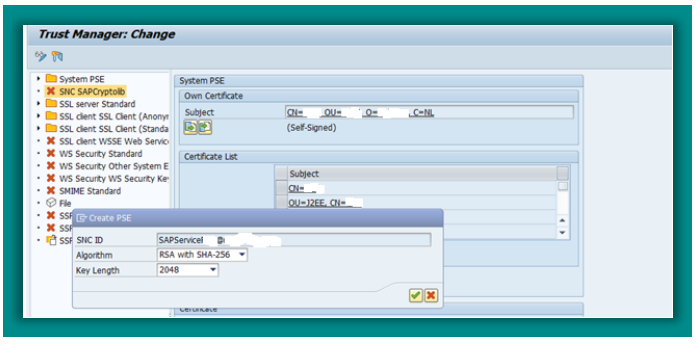
snc/permit_insecure_start
By default, once SNC has been activated (snc/enable = 1), the gateway refuses to start programs when the communications are not SNC protected.

Note: restart the SAP system to activate the new profile parameters. HOWEVER as the PSE is not yet created, SNC won't work and SAP won't start. **Therefore set parameter *snc/enable* to 0 for now**.

Create the PSE for the server

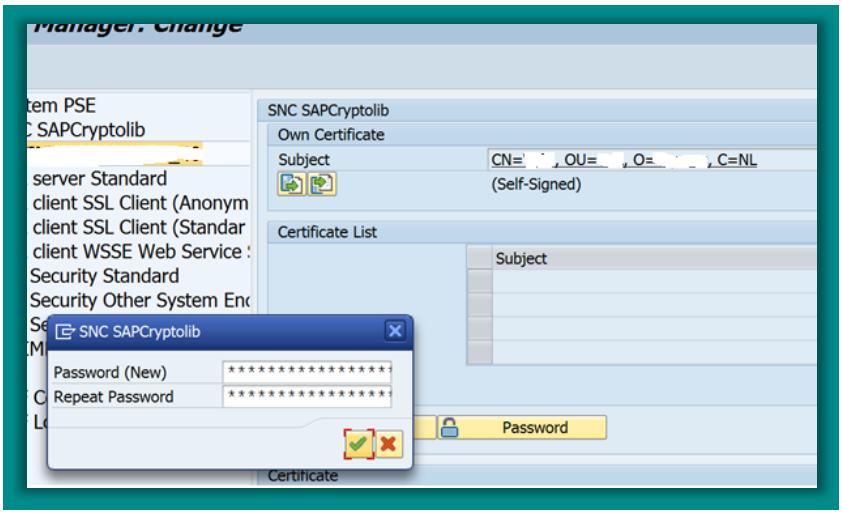
Note: when working with the windows Kerberos dll's solution, this step is not needed.

Create the Personal Security Environment (PSE) for the SAP system. Log on to the system and call transaction STRUST. Go into CHANGE mode first. Right click on the entry 'SNC (SAP Cryptolib)' and choose 'Create'.

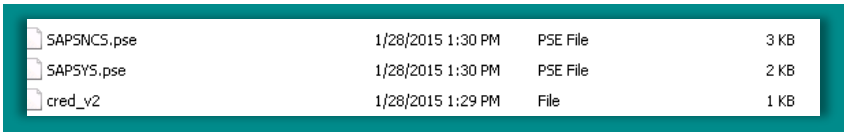


Confirm the SNC ID. It is set based on the parameter snc/identity/as and choose the key length.

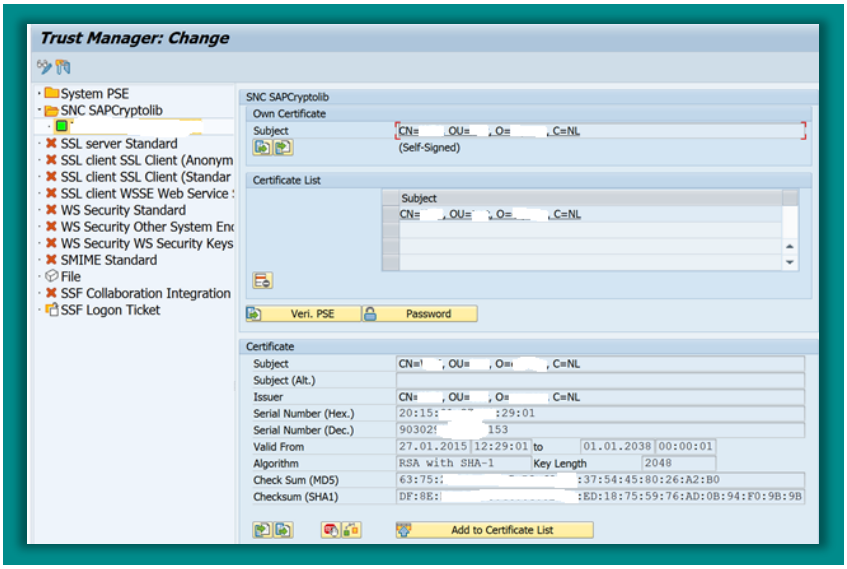
Set a password on the SNC SAPcryptolib, this creates the credentials (don't forget to press 'Save').



On the operating system these files get created:



In transaction STRUST, double-click on the 'SNC SAPcryptolib' and then double-click on the SNC ID. This displays the certificate in the bottom of the screen.



Click the button 'Add to Certificate List'. This adds the certificate to the certificate list. Now set parameter **snc/enable** to **1** and restart SAP. The application server should now start with SNC enabled.

SNC RFC configuration

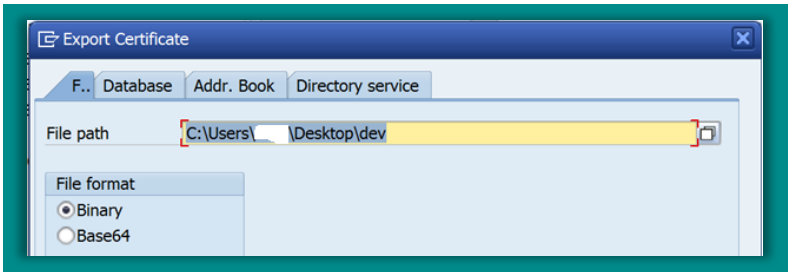
Adapting the RFC's for usage of SNC is done via these three steps:

- Exchange profiles between source and destination systems
- Allow SNC RFC connections from the initiator in the acceptor system.
- Adjust RFC settings

Exchange profiles between source and destination systems

Note: when working with the windows Kerberos dll's solution, this step is not needed.
Note: execute these steps on both the source and destination system.

Log on to the system and call transaction STRUST. Choose from the tree 'SNC (SAPCryptolib) → instance. Double-click the certificate name. Choose 'Certificate' → 'Export'.

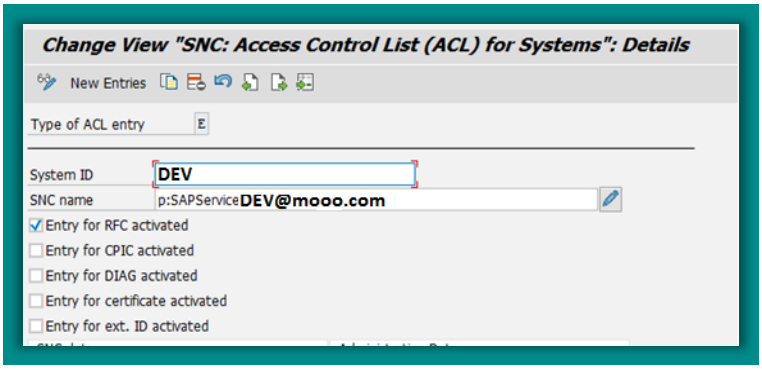


Now, on the other system, log in and choose transaction STRUST. Choose 'Certificate' → 'Import' and choose the file you just exported. Click 'Add to certificate list'. Choose 'Save'.

Allow SNC RFC connections from the initiator in the acceptor system.

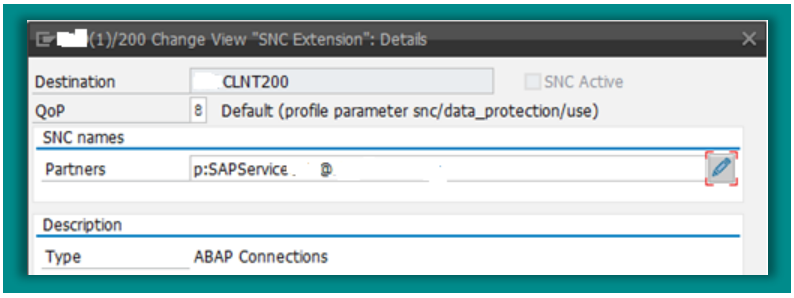
To have the receiving system accept incoming connections via SNC perform these steps:

1. Log into the acceptor system through SAPGUI and start Transaction **SNC0**
2. Choose **E** for the Type of ACL entry.
3. Enter System ID and SNC name (do not forget the **p**: in front of the **DN**).
4. Check the boxes 'Entry for RFC activated' and 'Entry for CPIC activated'.
5. Save the entry.



Adjust RFC settings

In the RFC itself now the RFC settings must be reflected. Go to SM59 and change the SNC settings for the connection. Go into change mode, and choose the tab 'Logon & Security'. Click the SNC button and enter the name from the profile parameter snc/identity/as:



Click ‘Continue’. Click ‘Continue’ and ‘Save’. Now set ‘Status of Secure Protocol’ to ‘Active’. Click ‘Save’ and test the connection to check it works.

Firewall target system

When things are not working, adapting the firewall policies by allowing traffic to port 48XX might help.

How Protect4S can help

Protecting SAP communication is just one of the many topics in the complex field of securing SAP platforms. Making sure all these topics are covered and configured correctly is impossible for most SAP running businesses. For that purpose we developed Protect4S; our SAP Security scanner. For more information see https://protect4s.com/.