



Configure Secure Network Communication (SNC)



In this exercise you will:

- Configure Secure Network Communication for RFC based communication

Configure Secure Network Communication (SNC) and Secure Sockets Layer (SSL)

Introduction.....	4
Business Systems	6
Configure SNC on the CI.....	7
Download the latest sapcryptolib from service market place:	7
Set environment variables	8
Install SAP Cryptolib on central instance	8
Maintain profile parameters	9
Restart System	11
Test startup.....	11
Test sapcryptolib	12
Delete the System PSE – not used.....	13
WebAS 6.20	13
SAP 4.6C	13
Create the System PSE – not used	14
WebAS 6.20	14
SAP 46C	15
Create the SNC PSE	16
WebAS 6.20	16
SAP 4.6C or lower.....	19
Generate Credentials.....	21
WebAS 6.20	21
SAP 46.C or lower only	21
Create the Certificate Request – not used.....	22
WebAS 6.20	22
SAP 46.C	23
Sign the SNC Certificate by a CA – not used.....	23
WebAS 6.20	23
SAP 46.C	23
Import the SNC Certificate signed by the CA– not used	23
WebAS 6.20	23
SAP 46.C	24
Activate SNC	24
Export the certificate of the partner server.....	24
WebAS 6.20	24

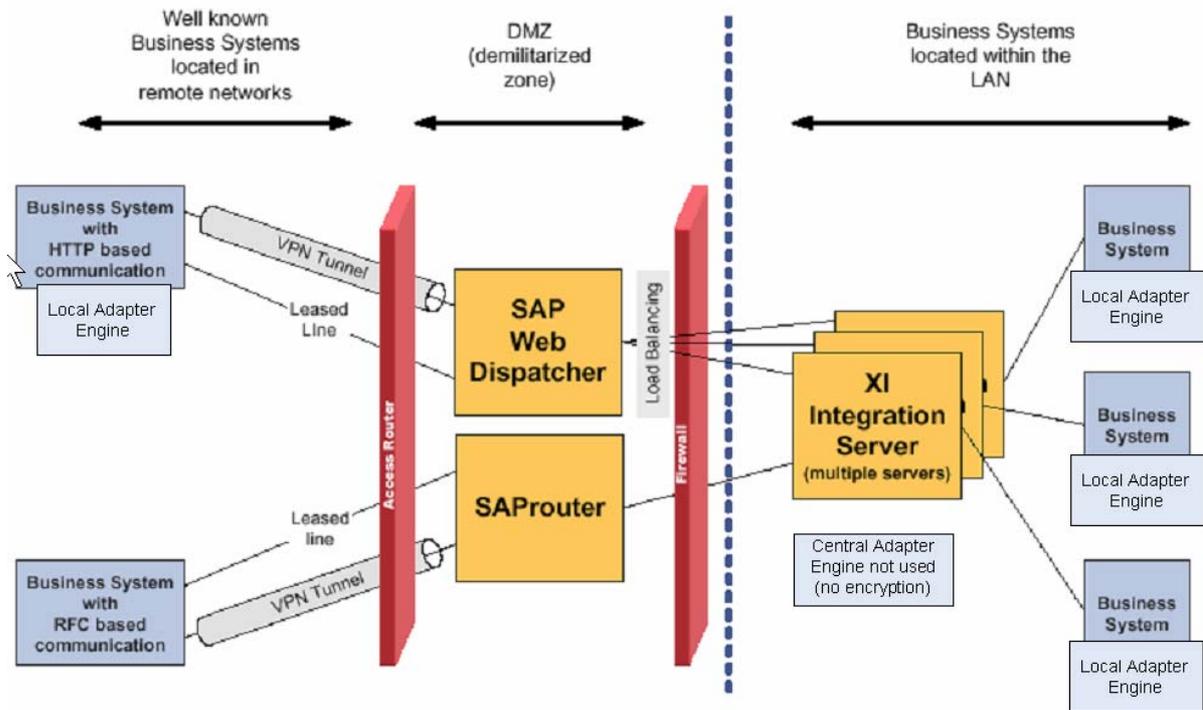
SAP 4.6C	25
Import the certificate of the partner server into your SNC PSE.....	26
WebAS 6.20	26
SAP 4.6C	27
Setup or change the RFC destination.....	29
Maintain SNC Access Control List.....	31
Activities in the communication partner system	32
Test the RFC destination.	33
Check SNC Names.....	33
Configure SNC on additional Dialog Server	36
Download the latest sapcryptolib from service market place	36
Set environment variables	36
Copy SAPSNCS.pse / <SID>SNCS.pse.....	36
Maintain Profile Parameters.....	36
Generate credentials	36
Activate SNC	37
Maintain the Access Control List (ACL)	37
Configure SNC for the RFC adapter.....	38
Configure RFC Destination used by the RFC Adapter for SNC	38
Configure the RFC Destinations that send data to the RFC Adapter for SNC	38
In the property file of the RFC Adapter	38
Configure SNC for the J2ee engine communication via the sap gateway	39
Configure SSL for the Integration Server	40
Create SSL Client Certificate	42
Activate SSL	42
Testing the connection for SSL Server Authentication.....	42
Change the Pipeline settings.	45
TA SICF	46
Create an Alias for HTTPs Calls of the Pipeline	46
Create an Alias for HTTP of the Pipeline.....	49
Configure SSL for a stand-alone adapter engine with a self-signed certificate	50
Business Systems with local adapter engine	50
Install the local adapter engine on the servers above.....	50
Installing the SAP Cryptographic Library on the server of the local adapter engine	50
Creating a PSE for the server of the local adapter engine using SAPGENPSE	51
without certificate request	51
Generate Credentials.....	52
Export the certificate of the partner server.....	53
On the Integration Server.....	53
On the Adapter Engine.....	54
Import the certificate of the partner server into your SSL Server PSE	55
Integration Server.....	55
Adapter Engine	57
Install the IAIK files on the local adapter engine	59
Maintain the configuration of the adapter engines	59
Maintain the services	59
HttpServer	59
GUIBrowserEngine – not used	60
Test the local adapter engine	61
Maintain properties files of all adapter instances	61
Maintain the endpoints of the Business Scenarios in the Integration Directory.....	61

Configure SSL for a stand-alone adapter engine with certificate from a CA.....	62
Installing the SAP Cryptographic Library on the server	62
Creating a PSE for the server using SAPGENPSE	62
Creating the Server's Credentials Using SAPGENPSE	62
Sign the Server Certificate	62
Export/Import the certificates between the servers	63
Transport.....	64
Related Notes and documents	65
Related Notes	65
Related Documents	65
Errors	66
Other comments	66

Introduction

In order to encrypt data sent via XI there are two ways.

- For RFC based communication over the RFC or Idoc Adapter SNC is used. The communication can be secured with a SAPRouter in addition.
- For all HTTP based communication SSL is used.



As currently all communication takes place in internal networks, no certificates from official Certificate Agencies are used for both SNC and SSL based communication.

For documentation reasons, the steps are mentioned but marked with (not used) in the heading.

Currently the document describes the procedure to set up SNC and SSL on Unix systems.

For windows systems there are slightly differences in the paths, the environment variables, and the rights for users and files. Please check the online documentation for that.

SNC:

For RFC based communication, the following types of RFC destinations can be encrypted.

SAP to SAP communication:

For the communication path between two SAP Systems when using RFC, the calling SAP System is the initiator of the communication and the SAP System defined as the RFC destination is the acceptor. Settings that are relevant for load balancing are made in the initiating system.

Internal RFC destinations:

For performance reasons, we do not recommend to use SNC for internal destinations. For incoming RFCs to internal destinations, the system does verify the entry in the SNCSYSACL table. This entry is automatically created as an internal destination (type = I) as start up, based on the information located in the profile parameters above mentioned.

ALE heavily uses internal RFC. For security reason internal RFC can be encrypted although the communication takes place on the same host.

RFC: TCP/IP connection to start an external program on an application server:
Not necessary to use SNC, because of the One-host installation of XI.

In addition the communication can be secured with a SAPROUTER.

SSL:

As currently all communication takes place in internal networks, currently no WebAS dispatcher is used in a DMZ.

Local Adapter Engines need to be installed on each Business System that requires Outbound Adapters based on the Adapter Engine.

Business Systems

System	Platform	Used adapter s	SSL	SNC	Adapter Engine
IET	HP-UX ?	SOAP, FILE	X		X
eCM	Unix ?	SOAP	X		X
SB1	Redhat 7.20 (CI) HP-UX 11.0 (DI)	File; RFC, IDOC	X	X	X
XID	HP-UX 11.11	SOAP, FILE, JDBC, RFC, IDOC	X	X	No
MS SQL Server	Windows 2000 Server SP ?	JDBC	?		?

Configure SNC on the CI

We use the SAP Cryptolib for SNC. The product meets the requirements of the GSS-API V2 Interface.

Download the latest sapcryptolib from service market place:

The product must provide the entire functionality defined in the standard interface, the GSS-API V2 (Generic Security Services Application Programming Interface Version 2). SNC uses this interface to communicate with the external security product.

See note 66687 for more information.

We use the SAP Cryptographic Library for SNC.

On the service marketplace <http://service.sap.com> specify the alias *download*.

Go to SAP Cryptographic Software → <your platform>

Important: The distribution of the SAP Cryptographic Library is subject to and controlled by German and US export regulations and can not be sent to all countries. In addition, the library may be subject to local regulations of your own country that may further restrict the import, use and (re-)export of cryptographic software.

We will use separate PSE for SNC and SSL.

System	Download for SNC
XI2	CI: SAP Cryptographic Library Linux Intel x86 – for testing purposes only
XID	CI: SAP Cryptographic Library HP UX 11.11
XIT	CI: SAP Cryptographic Library HP UX 11.11
XIP	CI: SAP Cryptographic Library HP UX 11.11
SB1	CI: SAP Cryptographic Library Linux Intel x86 DI: SAP Cryptographic Library HP UX 11.0

Unpack the CAR files to a temporary directory.

In the case of Linux there are the following files:

E.g. Under Linux:

```
[sbladm@sapa91 SNC]$ CAR -xvf ../linux_snc.car
processing archive ../linux_snc.car...
x Changelog.txt
x LEGAL.TXT
x LICENSE.TXT
x Ver555.pl14
x linux-glibc2.1.2
x linux-glibc2.1.2/libsapcrypto.so
x linux-glibc2.1.2/sapgenpse
```

x ticket

Set environment variables

These environment variables are valid for WebAS 620 and 46C also.
Check if they are already set for user <sid>adm in the profile .sapenv_<host name>.csh.
If not, set the following environment variables.

DIR_EXECUTABLE = /usr/sap/<SID>/SYS/exe/run

The environment variable DIR_EXECUTABLE determines the location where the PSE is stored. This means we use the directory /usr/sap/<SID>/SYS/exe/run.

USER= <sid>adm

LD_LIBRARY_PATH = /usr/sap/<SID>/SYS/exe/run

SECUDIR = /usr/sap/<SID>/DVEBMGS<Instance Number>/sec

Install SAP Cryptolib on central instance

On every server the files need to be installed. We start with the central instance.
The necessary steps on the application server are described in section *Configure SNC on additional Application Server*.

sapcryptolib

We use the /usr/sap/<SID>/SYS/exe/run directory to store the sapcryptolib .
This is determined by the environment variable DIR_EXECUTABLE:

Copy the files *libsapcrypto.sl / libsapcrypto.so* and *sapgenpse* to the directory /SYS/exe/run.
The *libsapcrypto.<so/sl>* has to be secured on OS level.

Only <sid>adm should be able to access the file.

Make sure that you change the authorizations to 700¹ for user <sid>adm and group sapsys.

sapgenpse

We use the /usr/sap/<SID>/SYS/exe/run directory to store the *sapgenpse file* .

Make sure that you change the authorizations to 700² for user <sid>adm and group sapsys.

ticket

Copy the ticket to the directory /usr/sap/<SID>/DVEBMGS<##>/sec/

Make sure that you change the authorizations to 700³ for user <sid>adm and group sapsys.

¹ Change to 700 as soon everything is working

² Change to 700 as soon everything is working

³ Change to 700 as soon everything is working

Maintain profile parameters

Make a backup of all profile files on OS level.

Set the following profile parameters in transaction RZ10.

If nothing is mentioned the parameters are valid for both 46C and WebAS 6.20.

Maintain the following parameters in the Instance-profile.

Some profile parameters are shown as XID (host sapsid) and SB1 (host sapp91) as an example.

Profile parameter	Value	Details
snc/enable	0 ⁴	0: not enabled 1: enabled
DIR_EXECUTABLE	/usr/sap/<SID>/SYS/exe/run	
sec/libsapsecu	HP-UX: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.sl Linux: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.so	Disables SAPSECULIB, as it cannot be used for SNC
ssf/ssfapi_lib	HP-UX: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.sl Linux: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.so	
ssf/name	SAPSECULIB	Although the SAPSECULIB is not used, the parameter has to be defined here.
snc/gssapi_lib	Linux: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.so HP-UX: /usr/sap/<SID>/SYS/exe/run/libsapcrypto.sl	<path and file name where the SAP Cryptolib is located>
WebAS 6.20 or higher only snc/identity/as	See list on next page: WebAS 6.20: Example for XID: p:CN=XID, OU=SD MA, O=SIEMENS DEMATIC, C=US Example for SB1: p:CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US	- The server's SNC name is the same for the CI and the Dialogservers of the SAP System ⁵ - Also see chapter Activities on Dialog servers
WebAS 6.20 or higher only: sec/rsakeylength default	1024	Use a key length of 1024 bit (only with kernel release 6.20 and higher), see note 509495. (512 (standard), 768, 1024, 2048)
snc/data_protectio	3	1: Authentication only

⁴ At that point the parameter is not activated. We activate it in the instance profile in one of the next steps.

n/max		2: Integrity protection 3: Privacy protection
snc/data_protection/min	1	1: Authentication only 2: Integrity protection 3: Privacy protection
snc/data_protection/use	3 ⁶	1: Authentication only 2: Integrity protection 3: Privacy protection
snc/accept_insecure_gui	1	0: Reject unprotected logons 1: Accept unprotected logons
snc/accept_insecure_rfc	1	0: Reject unprotected RFC 1: Accept unprotected RFC
snc/accept_insecure_cplic	1	0: Reject unprotected CPIC 1: Accept unprotected CPIC
snc/r3int_rfc_security	0	Protect RFC communications 0: Internal RFCs are unprotected 1: Internal RFCs are protected – persicht, ale
snc/r3int_rfc_qop	3	1: Authentication only 2: Integrity protection 3: Privacy protection 8: Use the value from snc/data_protection/use 9: Use the value from snc/data_protection/max
snc/accept_insecure_rfc	1	
snc/accept_insecure_r3int_rfc	1	This parameter enables RFC connections that were started by their own R3 System with internal destinations to be allowed without SNC security. Only effective if snc/accept_insecure_rfc = 0
snc/accept_insecure_start	1	If SNC is enabled, by default (value 0) the gateway does not start any programs that communicate without SNC
snc/force_logon_screen	0	0: The logon screen is displayed only when necessary 1. The logon screen is always displayed
gw/rem_start	REMOTE_SHELL	For Security reasons, start only programs on the computer where the gateway is located. Additionally the gateway passes the name of the external library onto the programs that it starts. → Value DISABLED not used yet. Needs to be tested

List of all SNC names

SID	Type (CI, Diag)	snc/identity/as ⁷	SNC enabled
XI2	CI	p:CN=XID, OU=SD MA, O=SIEMENS DEMATIC, C=US	
XIP	CI	p:CN=XID, OU=SD MA, O=SIEMENS DEMATIC, C=US	
XID	CI	p:CN=XID, OU=SD MA, O=SIEMENS DEMATIC, C=US	X
XIT	CI	p:CN=XIT, OU=SD MA, O=SIEMENS DEMATIC, C=US	
SB1	CI/DI	p:CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US	X

Restart System

Save the instance profile and restart the sap system

```
stopsap r3
```

```
startsap r3
```

Test startup

Check in /usr/sap/<SID>/DVEBMGS##/dev_w0 if errors occur during startup.

Errors during startup:

```
N The internal Adapter for the loaded GSS-API mechanism identifies as:
N Internal SNC-Adapter (Rev 1.0) to SECUDE 5/GSS-API v2
N *** ERROR => SncPSetNewName()==SNCERR_BAD_NT_PREFIX [sncxxall.c 2271]
N SncPImportPrName() parsing error
N name="sap00.sapxi2"
N <<- SncInit()==SNCERR_BAD_NT_PREFIX
```

Changed:

```
N SncInit(): found
snc/gssapi_lib=/usr/sap/XI2/SYS/exe/run/libsapcrypto.so
N File "/usr/sap/XI2/SYS/exe/run/libsapcrypto.so" dynamically loaded as
GSS-A
PI v2 library.
N The internal Adapter for the loaded GSS-API mechanism identifies as:
N Internal SNC-Adapter (Rev 1.0) to SECUDE 5/GSS-API v2
N SncInit(): found snc/identity/as=p:CN=sap00.sapxi2, OU=Test,
O=MyCompany, C
=DE
N *** ERROR => SncPacquireCred()==SNCERR_GSSAPI [sncxxall.c 1510]
N GSS-API(maj): No credentials were supplied
N GSS-API(min): SECUDE PSEDIR directory not found: /home/xi2adm/sec
($HOM
E)
N Could't acquire ACCEPTING credentials for
N
N name="p:CN=sap00.sapxi2, OU=Test, O=MyCompany, C=DE"
M *** ERROR => ErrISetSys: error info too large [err.c 945]
M Mon Oct 20 10:39:11 2003
M LOCATION SAP-Server sapxi2_XI2_00 on host sapxi2 (wp 0)
```

⁷ If you use a signed certificate, ask your certificate provider for the exact SNC name. In our case we only use self-signed certificates.

```

M ERROR          GSS-API(maj): No credentials were supplied
M GSS-API(min): SECUDE PSEDIR directory not found: /home/xi2adm/sec ($HO
M name="p:CN=sap00.sapxi2, OU=Test, O=MyCompany, C=DE"
M TIME           Mon Oct 20 10:39:11 2003
M RELEASE        620
M COMPONENT      SNC (Secure Network Communication)
M VERSION        5
M RC             -4
M MODULE         sncxxall.c
M LINE           1510
M DETAIL         SncPacquireCred
M SYSTEM CALL    gss_acquire_cred
M ERRNO
M ERRNO TEXT
M DESCR MSG NO
M DESCR VARGS    GSS-API(maj): No credentials were supplied;;;
M ;;;;GSS-API(min): SECUDE PSEDIR directory not found: /home/xi2adm/sec
($HO;;;
;
M ;;;;name="p:CN=sap00.sapxi2, OU=Test, O=MyCompany, C=DE"
M DETAIL MSG N
M DETAIL VARGS
M COUNTER        3
N SncInit(): Fatal -- Accepting Credentials not available!
N <<- ERROR: SncInit()==SNCERR_GSSAPI
N             sec_avail = "false"
M ***LOG R19=> ThSncInit, SncInitU ( SNC-000004) [thxxsnc.c    223]
M *** ERROR => ThSncInit: SncInitU (SNCERR_GSSAPI) [thxxsnc.c    225]
M in_ThErrHandle: 1
M *** ERROR => SncInitU (step 1, th_errno 44, action 3, level 1)
[thxxhead.c
8534]

```

→ You need to deactivated SNC to create the SNC PSE.

Test sapcrpytolib

It is important that the SAPCRYPTOLIB has the patch level 14 (5.5.5.C pl14)

Go to directory /usr/sap/<SID>/SYS/exe/run

You can check the patch level by calling sapgenpse on OS level.

The following result should appear:

```

[sbladm@sapa91 run]$ ./sapgenpse
Usage: sapgenpse [-h] <command> [-h] [sub-options] ...

```

```

Using default SAPCRYPTOLIB library name "libsapcrypto.so"

```

```

Platform:   Linux on Intel x86 32-bit
Versions:   SAPGENPSE    = 1.5.5   pl17   (Dec 11 2002)
            SAPCRYPTOLIB = 5.5.5.C pl14 (Dec 10 2002) MT-safe

```

```

USER="sbladm"

```

```

Environment variable $SECUDIR is defined:

```

```
"/usr/sap/SB1/DVEBMGS00/sec"
```

```
shared library search path defined by environment variable  
LD_LIBRARY_PATH=/usr/sap/SB1/SYS/exe/run:/oracle/SB1/817_32/lib
```

The following error occurred on Linux:

```
sapxip:xipadm 22> sapgenpse  
*****  
**   sapgenpse WARNING:  Environment variable "USER" not defined!  **  
** ----- **  
**   Please define the USER environment variable *AND* insert      **  
**   the definition into the startup script of your Unix shell,    **  
**   or you may get problems accessing credentials created        **  
**   through 'seclogin'!                                          **  
**                                                                 **  
**   Examples additions for your shell startup scripts:           **  
**                                                                 **  
**   (sh):  if [ "$USER" = "" ];then USER=`whoami`;export USER;fi **  
**   (csh):  if ( $?USER == 0 ) setenv USER `whoami`              **  
**                                                                 **  
**   You appear to have a csh-style login shell                   **  
*****
```

```
Usage: sapgenpse [-h] <command> [-h] [sub-options] ...
```

```
ERROR in unix_dlopen(): dlopen("libsapcrypto.sl") FAILED:  
"Mmap failed due to errno: 13."
```

```
Loading of shared library "libsapcrypto.sl" failed!  
You might need to define the shared library search path LD_LIBRARY_PATH
```

You need to do the following:

```
setenv USER xipadm
```

```
setenv LD_LIBRARY_PATH /usr/sap/XIP/SYS/exe/run  
→ Why needs LD_LIBRARY_PATH to be set under Linux?
```

Delete the System PSE – not used

Motivation:

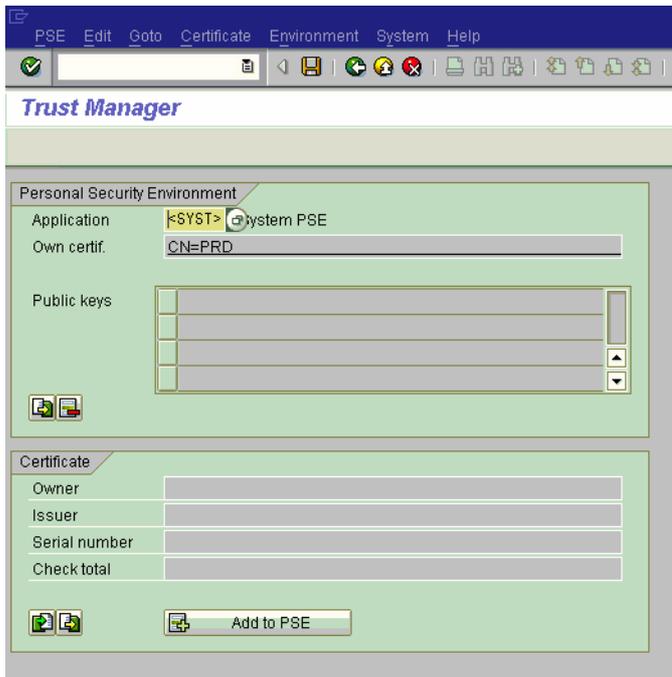
WebAS 6.20

Go to TA STRUST

High light *System PSE*

Choose in context menu *Delete*.

SAP 4.6C



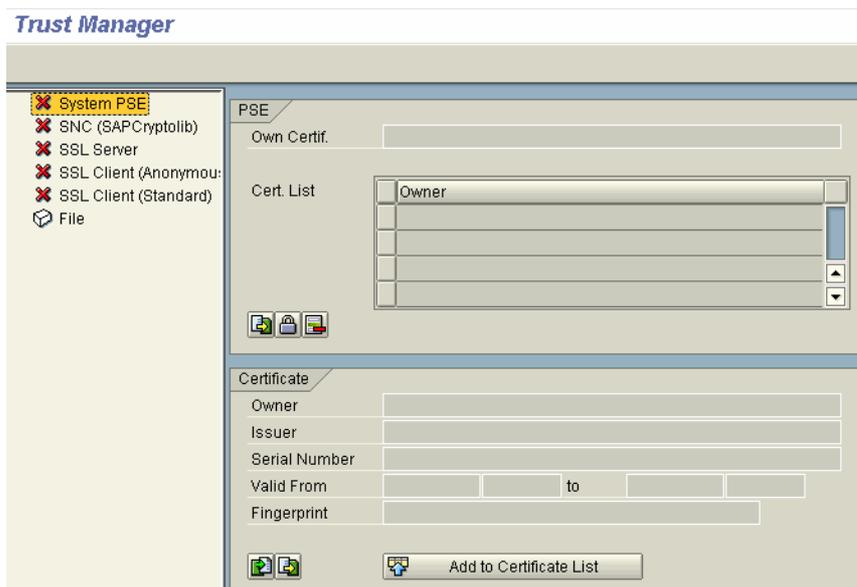
Therefore remove the file SAPSYS.pse in directory /usr/sap/<SID>/DVEBMGS##/sec
 On OS level
 rm SAPSYS.pse

Create the System PSE – not used

Use this functionality for enabling single-sign on the basis of logon-tickets between the systems.

WebAS 6.20

Start the trust manager with transaction STRUST.



High-light *System PSE*.
Choose *Create* in the context menu.

 Use that button to deactivate the suffix. The field CA gets greyed out and the field Country can be maintain.⁸

Make the following entries:

- Name: < SID >
- Org: SD MA
- Comp./Org.: SIEMENS DEMATIC
- Country: US

Choose *Enter*.

In directory \usr\sap\XI2\DVEBMGS00\sec the file <name>.PSE is created.

SAP 46C

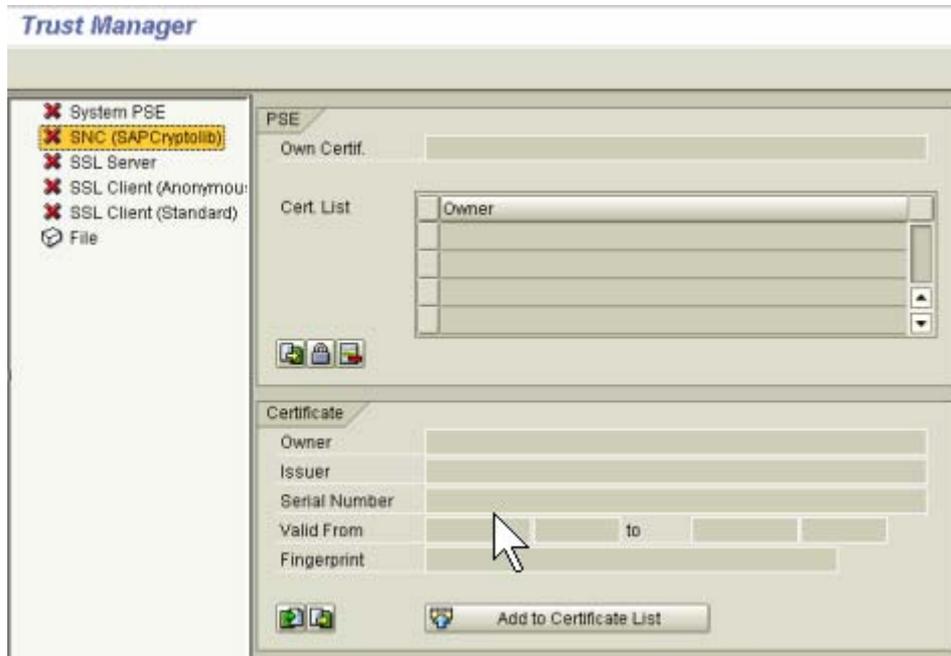
Not described.

⁸ The difference to SNC (where the SID is used for the name) is that there a whole SAP system is addressed. With SSL always a single server is referenced.

Create the SNC PSE

WebAS 6.20

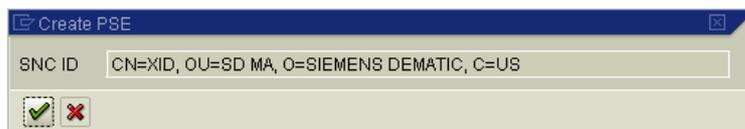
Start the trust manager with transaction STRUST.



High-light SNC (SAPCryptolib) and choose *Create* in the context menu.

High-light *SNC PSE*.

Choose *Create* in the context menu.



Choose *Enter*.

If a Pop up comes up for SNC name enter the following:

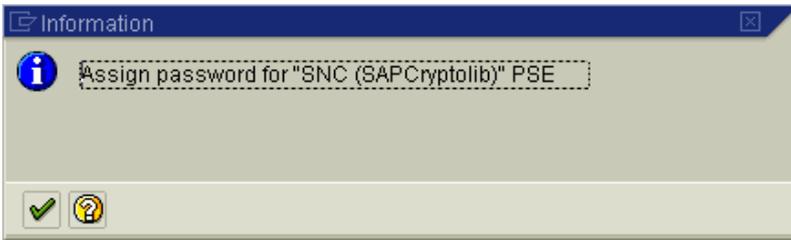
 Use that button to deactivate the suffix. The field CA gets greyed out and the field Country can be maintain.⁹

Make the following entries:

- Name: < SID >
- Org: SD MA
- Comp./Org.: SIEMENS DEMATIC
- Country : US

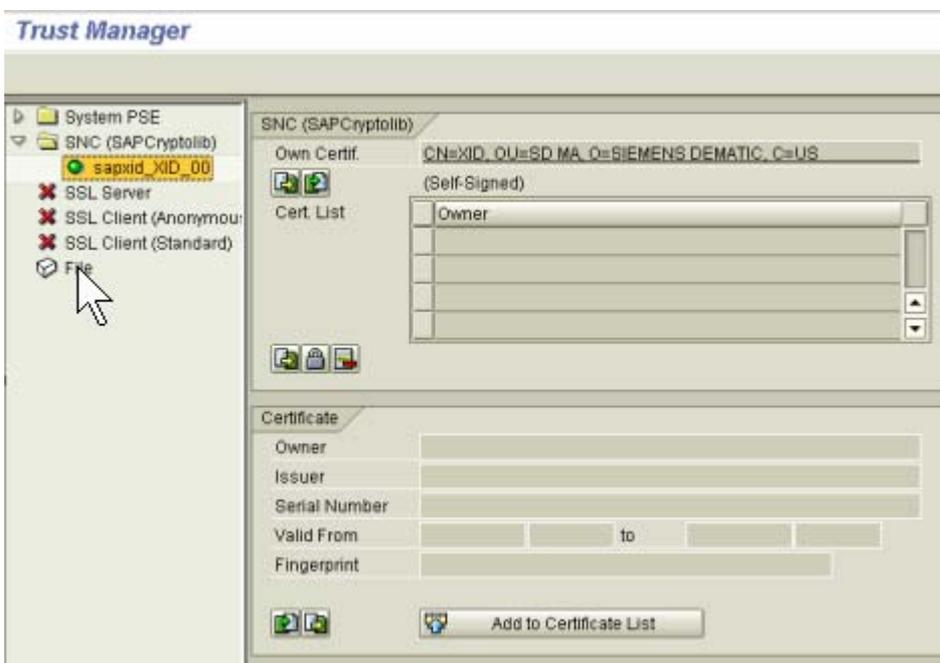
Choose *Enter*.

⁹ The difference to SNC (where the SID is used for the name) is that there a whole SAP system is addressed. With SSL always a single server is referenced.

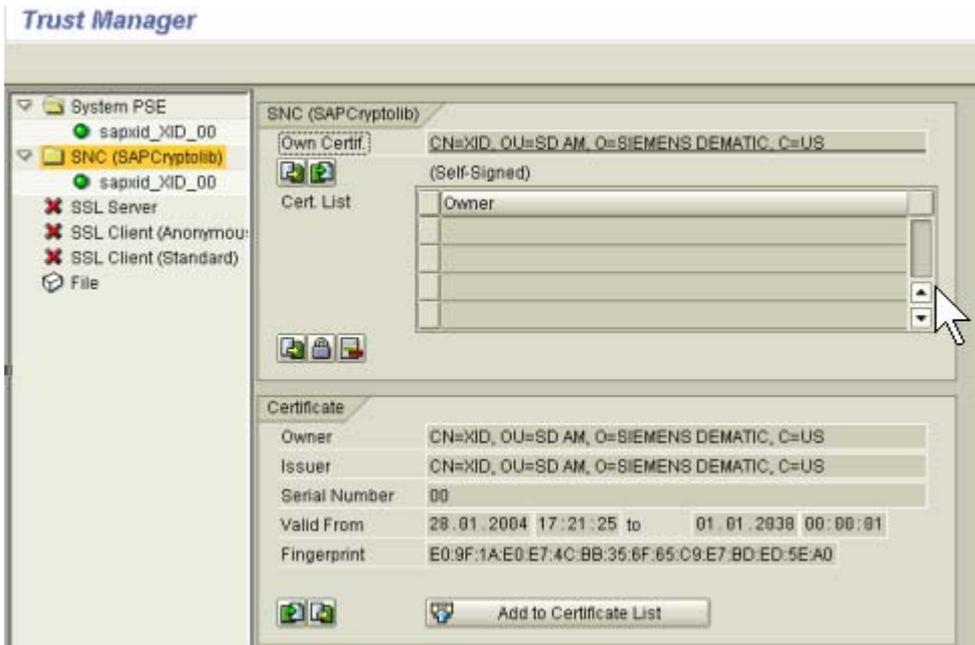


Choose *Enter*.

The result looks like this:



Double-click on SNC (SAP Cryptolib) → Double-click on *sapxid_XID_00*.
Double-click on *Own-certificate*.



Choose  to assign a password to the SNC PSE
 For testing purposes we use *empass*



You will get the message Temporary PSE encrypted.

TEMP*.pse created: Reason?

In directory /usr/sap/<SID>/DVEBMGS###/sec the file *cred_v2* is created.
 In the file *cred_v2* the password for the SNC PSE is stored in encrypted form. It is used for the SAP tools to log onto the SNC PSE automatically.

Make sure that **only the user under which the server runs** has access to this file (including read access).

Therefore make sure that you change the authorizations to 700¹⁰ for user <sid>adm and group sapsys

SNC does not require certificates signed by a CA.
 The PSE can use self-signed certificates.

Restart the SAP system

¹⁰ Change to 700 as soon everything is working

- stopsap r3
- startsap r3

SAP 4.6C or lower

Execute in directory /usr/sap/SB1/exe/run the command to create a SNC PSE using SAPGENPSE without certificate request:

```
sapgenpse get_pse <additional_options> [-p <PSE_name>] [-r
<cert_req_file_name>] [-x <PIN>] [DN]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-r	<file_name>	File name for the certificate request	Path description (in quotation marks, if spaces exist)	stdout
-x	<PIN>	PIN that protects the PSE	Character string	None
None	DN	Distinguished Name for the server The Distinguished Name is used to build the server's SNC name.	Character string (in quotation marks, if spaces exist)	None

Additional Options

Option	Parameter	Description	Allowed Values	Default
-s	<key_len>	Key length	512, 1024, 2048	1024
-a	<algorithm>	Algorithm used	RSA, DSA	RSA
-noreq	None	Only generate a key pair and PSE. Do not generate a certificate request.	Not applicable	Not set

-onlyreq	None	Generate a certificate request for the public key stored in the PSE specified by the -p parameter.	Not applicable	Not set
----------	------	--	----------------	---------

The SNC Distinguished Name consists of the following elements:

- CN = <SID>
- OU = SD MA
- O= SIEMENS DEMATIC
- C = US

The **Distinguished Name** is for example for SB1:

p: CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US

```
sapgenpse get_pse -s 1024 -a RSA -p /usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.pse
-noreq -x empass "CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US"
```

The result is the following.

```
[sbiadm@sapa91 run]$ pwd
/usr/sap/SB1/SYS/exe/run
[sbiadm@sapa91 run]$ more dwh
sapgenpse get_pse -s 1024 -a RSA -p /usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.pse -nore
q -x empass "CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US"

[sbiadm@sapa91 run]$ ./dwh

[sbiadm@sapa91 run]$ cdD
[sbiadm@sapa91 DVEBMGS00]$ cd sec
[sbiadm@sapa91 sec]$ ll
total 24
-rw----- 1 sbiadm sapsys 1255 Jan 29 13:38 SB1SNCS.pse
-rwx----- 1 sbiadm sapsys 374 Jan 26 09:20 ticket
[sbiadm@sapa91 sec]$
```

In directory /usr/sap/SB1/DVEBMGS00/sec/ the PSE SB1SNCS.pse is created.

Restart the SAP system.

- stopsap r3
- startsap r3

Check in /usr/sap/<SID>/DVEBMGS<Instance number>/work the file dev_w0 for errors.

Generate Credentials

WebAS 6.20

This step is not necessary as it is automatically done by STRUST.

The credentials are stored in file *cred_v2*.

SAP 46.C or lower only

```
sapgenpse seclogin <additional_options> [-p <PSE_name>] [-x <PIN>] [-O  
[<NT_Domain>]\<user_ID>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None
-O	[<NT_Domain>]\<user_ID>	User for which the credentials are created. (The user that runs the server's processes.)	Valid operating system user	The current user

Additional Options

Option	Parameter	Description	Allowed Values	Default
-l	None	List all available credentials for the current user.	Not applicable	Not set
-d	None	Delete PSE	Not applicable	Not set
-chpin	None	Specifies that you want to change the PIN	Not applicable	Not set

Creating Credentials for the Server

The following command line opens the application server's PSE (<SID> = SB1) that is located at /usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.pse and creates credentials for the user <sid>adm = sb1adm. The PIN that protects the PSE is empass.

```
sapgenpse seclogin -p /usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.pse -x empass -O sb2adm
```

The PSE is self-signed by sapgenpse. The file *cred_v2* is used to store the credentials and stored in directory /usr/sap/<SID>/DVEBMGS<Instance number>/sec.

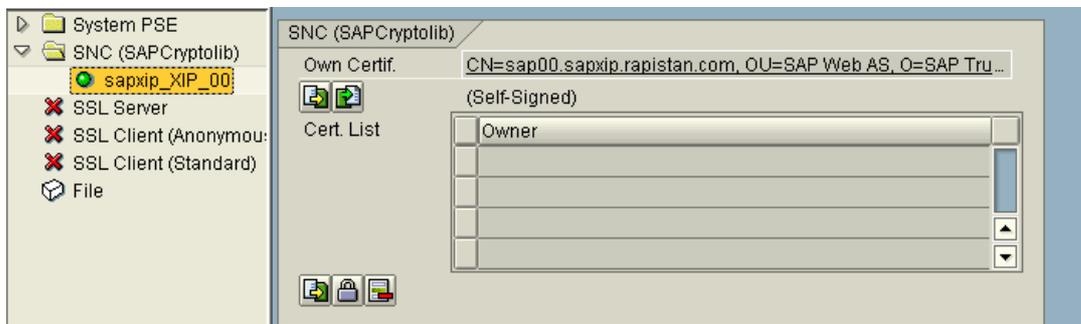
Restart the SAP system

- stopsap r3
- startsap r3

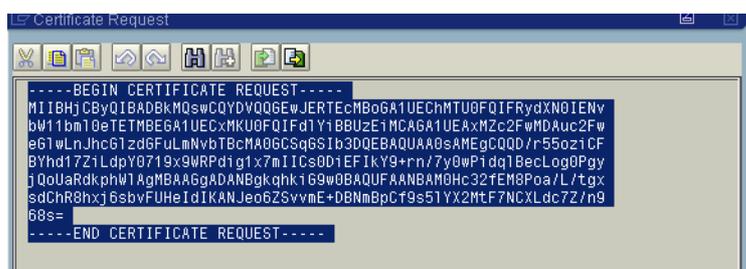
Create the Certificate Request – not used

WebAS 6.20

The following steps are for documentation only.



Choose 



Choose  to save the certificate request as local file.

Save it in the form <file name>.p10 if required.

As an alternative you can save it to the clipboard.

SAP 46.C

Not described here.

Sign the SNC Certificate by a CA – not used

WebAS 6.20

For testing purposes we use the SAP Certificate Infrastructure.

Go to the service marketplace <http://service.sap.com>.

Choose the Alias CTS.

If the alias is not working try <https://websmp105.sap-ag.de/SSLTest>

Choose SSL Server Test Certificates.

Save the response to a file.

As an alternative you can copy it to your clipboard.

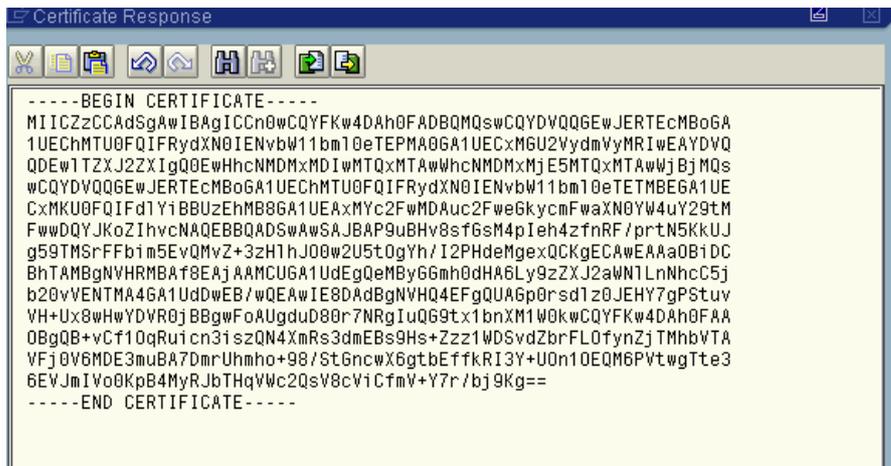
SAP 46.C

Not described here.

Import the SNC Certificate signed by the CA– not used

WebAS 6.20

The screenshot displays the SAP SNC (SAPCryptolib) interface. At the top, the title bar reads "SNC (SAPCryptolib)". Below it, the "Own Certif." field contains the text "CN=sap00.sapxp10.rapistan.com, OU=SAP Web AS, O=SAP Tru...". A "Cert. List" table is visible, with a single row containing the word "Owner". Below the table are three small icons: a folder, a document, and a key. The bottom section of the interface is titled "Certificate" and contains several input fields: "Owner", "Issuer", "Serial Number", "Valid From" (with a "to" field), and "Fingerprint". At the bottom of this section is a button labeled "Add to Certificate List" and another set of three small icons.



Paste in the response of the CA.
Choose the Green back button.



Save the data.

SAP 46.C

Not described here.

Activate SNC

In transaction RZ10:

Set the following parameter in the instance profile of the central instance.

snc/enable = 1.

Restart the SAP system.

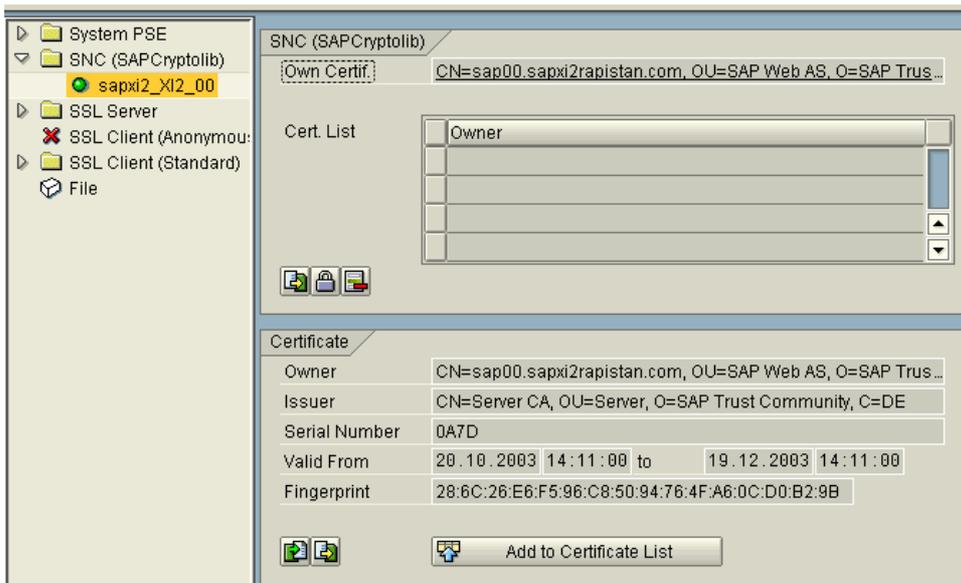
- stopsap r3
- startsap r3

Check the trace files that are written during startup, especially dev_w0 in the directory /usr/sap/<SID>/DVEBMGS##/work.

Export the certificate of the partner server

WebAS 6.20

In transaction STRUST high-light the SNC PSE.



Choose  Export Certificate

Save the certificate on the file system.

SAP 4.6C

Exporting the Application Server's Public-Key Certificate

```
sapgenpse export_own_cert -o <output_file> -p <PSE_name> [-x <PIN>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-o	<output_file>	Exports the certificate to the named file	Path description (in quotation marks, if spaces exist)	stdout
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None

The following command line exports the application server's public-key certificate (<SID> = SB1) to the file /usr/sap/SB1/DVEBMGS00/sec/ SB1SNCS.crt

```
sapgenpse export_own_cert -o /usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.crt -p /usr/sap/XI2/DVEBMGS00/sec/SB1SNCS.pse -x empass
```

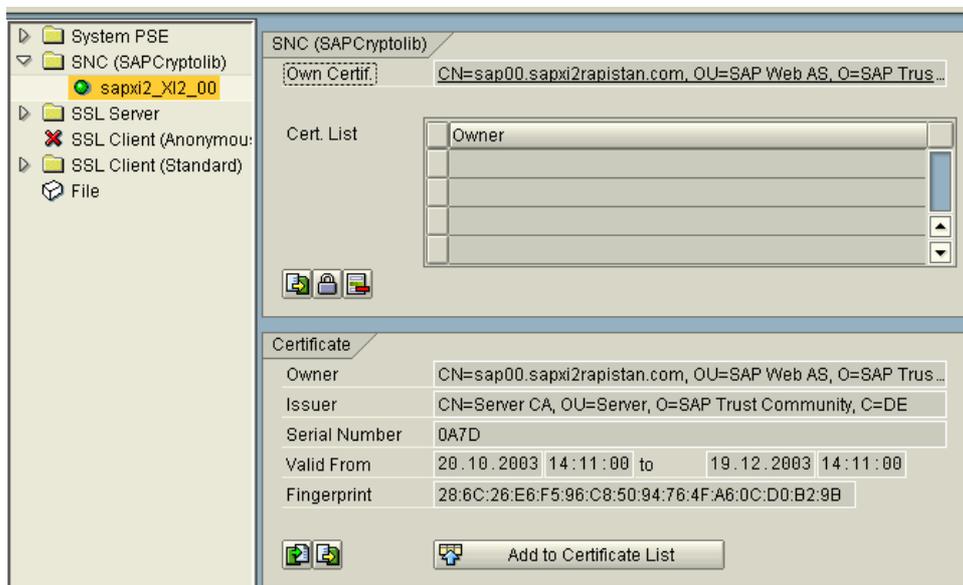
Import the certificate of the partner server into your SNC PSE

WebAS 6.20

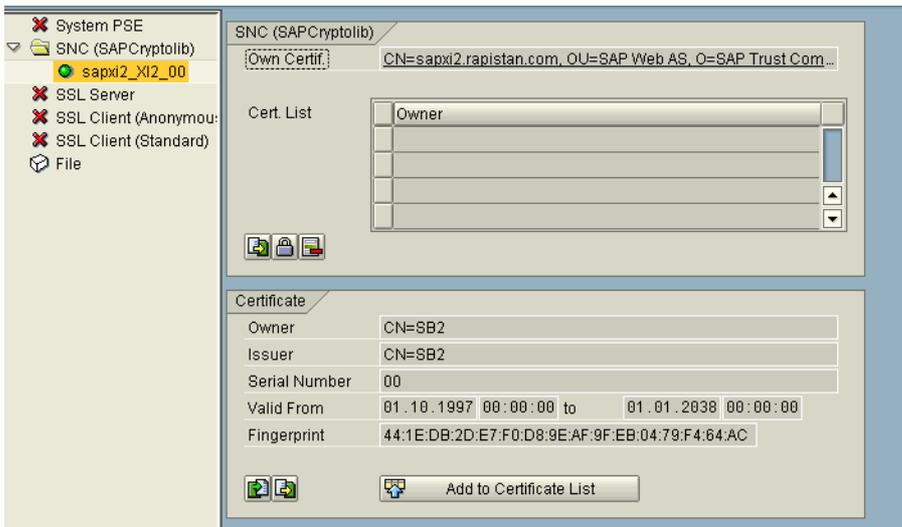
The certificates (public key) of all servers with which the server communicates have to be imported into the PSE.

In this case we will import the public key with a file.

In TA STRUST → Double-click on the SNC entry → Double-click on “Own Certf”



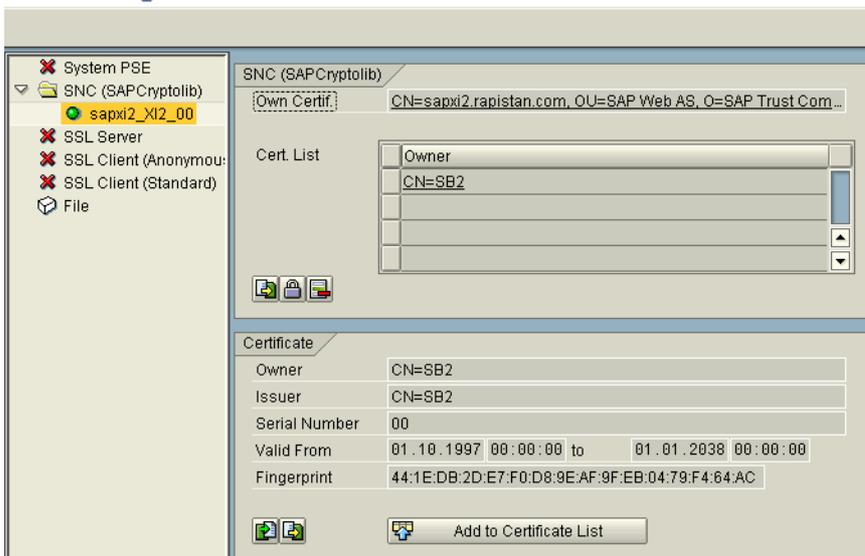
Choose *import Certificate*  to import the certificate's public key of all business Systems that use SNC.



Now choose *Add to Certificate List*.

The result is:

Trust Manager



SAP 4.6C

Use the tool's command `maintain_pk` to maintain the server's certificate list.

```
sapgenpse maintain_pk [<additional options>] [-a <cert_file>] [-d <number>]
-p <PSE_name> [-x <PIN>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
--------	-----------	-------------	----------------	---------

-a	<cert_file>	Add certificate from file <cert_file> to the certificate list.	Path description (in quotation marks, if spaces exist)	None
-d	<number>	Delete certificate number <number> from certificate list.	Numerical value	None
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None

Additional Options

Option	Parameter	Description	Allowed Values	Default
-cacert	None	The certificate to import is a CA root certificate.	Not applicable	None
-l	None	List existing certificate list	Not applicable	Not set

```
sapgenpse maintain_pk -a /usr/sap/XI2/DVEBMGS00/sec/SB1SNCS.crt -p
/usr/sap/XI2/DVEBMGS00/sec/SB1SNCS.pse
```

Setup or change the RFC destination

Use transaction SM59 to maintain the RFC destinations and their SNC options.

Before you can maintain the SNC information, the RFC destination must be defined and SNC activated on the application server.

When maintaining the SNC options specify the following information as defined in the instance profile of the application where the connection points to.

Naming convention in our case: <SID Target System>CLNT<Client>_SNC

Web AS 620

Select the *SNC Active* indicator

Choose Destination → SNC Options.

The Change View “*SNC extension: Details*” screen appears.

The screenshot displays the SAP SM59 transaction interface. At the top, the RFC destination is identified as 'SAPSB2' with a connection type of 'R/3 connection'. Below this, the 'SNC extension: Details' window is open, showing the following configuration:

- Destination:** SAPSB2
- QoP:** 8 (default (profile parameter snc/data_protection/use))
- SNC names:** (empty field)
- Description:** R/3 Connection
- Administrative data:** Created by JOCHENB, 27.11.2003 10:07:53

On the left side of the main window, the 'Security Options' section shows 'SNC' with the 'Actv.' radio button selected. The 'Logon' section shows 'Language: EN', 'Client: 050', and 'User: TSTSAPXI'.

Enter the Quality of protection in the QoP field. Set *QoP* = **8**.

This means the highest common security level of both systems is used.

Unless the destination is an external program that starts on the frontend workstation, enter the SNC name of the communication partner in the *SNC names* group.

To find out the SNC name of the communication partner:

In the partner System:

TA RZ10: Enter the value of profile parameter snc/identify/as

Save the data.

Maintain SNC Access Control List

Enter the communication partner in the SNC Access Control List.

Call transaction SNC0:

The screenshot shows the SAP transaction 'SNC0' in the 'New Entries: Details of Added Entries' view. The 'Type of ACL entry' is set to 'E'. The 'System ID' field is highlighted in yellow. The 'SNC name' field is empty and has an edit icon. Below these fields are five checkboxes: 'Entry for RFC activated' (checked), 'Entry for CPIC activated' (checked), 'Entry for DIAG activated' (unchecked), 'Entry for certificate activated' (unchecked), and 'Entry for ext. ID activated' (unchecked). At the bottom, the 'SNC data' section shows a warning icon and the message 'Canonical Name Not Determined'.

Enter the System ID of the communication Partner, in this case SB1.
(change: communication partner now XID)

Enter the SNC name of the communication Partner.

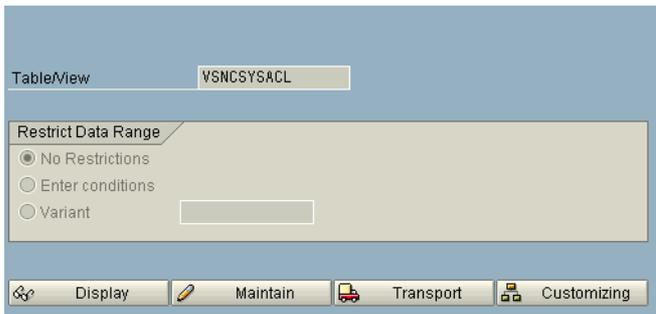
- In order to find out the SNC name of the communication partner use transaction RZ10 in the partner system:
Enter as SNC name the value of profile parameter *snc/identify/as*

Explanation of the options above:

- Entry for RFC activated: always checked in our case
- Entry for CPIC activated: always checked in our case
- Entry for DIAG activated: If you use the webgui service
- Entry for certificates activated: If users log on with X.509 client certificates
- Entry for external ID: If users log on using an external identity, for example, when using Pluggable Authentication Services

Go to Transaction SM30.

Choose the view VSNCSYSACL.



There you can maintain table SNCSYSACL.
Also see note 201417.

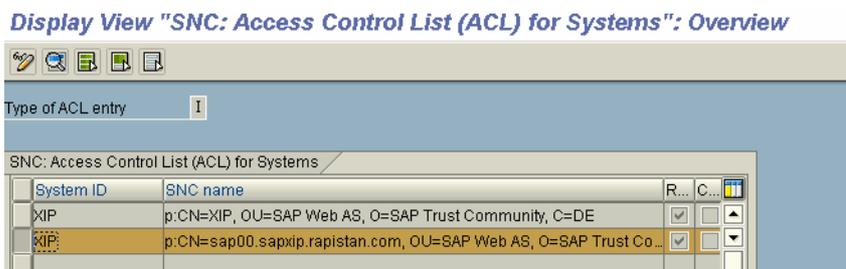
Choose *Maintain*.



Select E.

Here you can see entry you just created.
Go back.

Select I.



The internal entry is generated automatically.
If you change the PSE you might have to delete this entry and re-create it manually.

Other information: Instead of the ACL list a trust relationship between the two systems can be established.

Activities in the communication partner system

Repeat all steps of chapter Configure SNC for the ABAP Stack on the CI in the partner system.

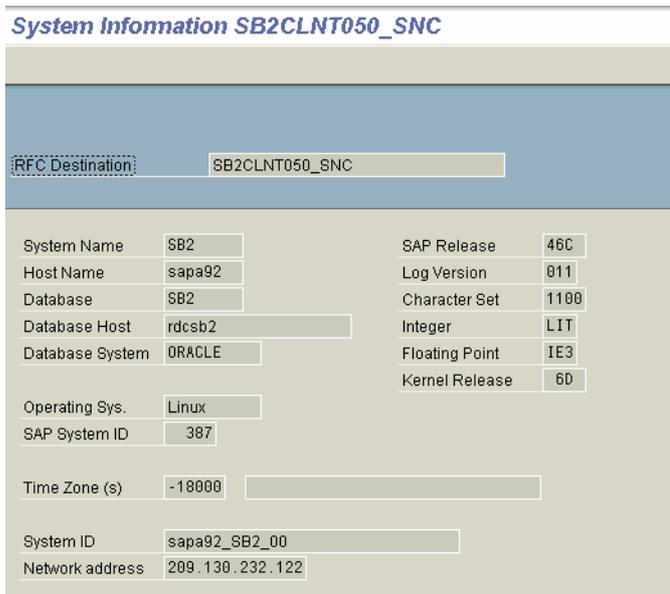
Test the RFC destination.

Choose in transaction SM59 the SNC-enabled RFC destination.

If you have a communication user maintained in the RFC destination

Choose System Information --> Target System.

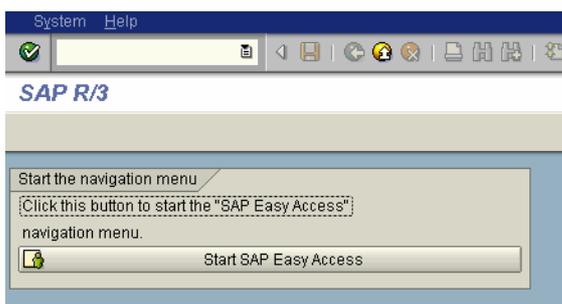
The result is:



If you have a dialog user in the RFC destination

Choose Remote Logon.

You must get the following screen without entering username and password:



For the activities on the partner system

- Do the same steps in the partner system as described above.

Check SNC Names

Execute Report RSSNCCHK in both systems.

Check and Clean-Up Canonical SNC Names

User names
 User with same SNC name
 SNC naming convention
 User extended
 RFC destinations
 CPIC destinations
 Systems
 Output device for SAPLPD

All Clients
 Display errors only
 Delete invalid canon. names

The result is e.g. on XI2:

Check and Clean-Up Canonical SNC Names

29.01.2004		Check and Clean-Up Canonical SNC Names		1
Table	USRACL			
Client	000			
\$ Action	Msg.	User	SNC name	
<input checked="" type="checkbox"/> empty		SAPJSF		
	Initial	passwd		
29.01.2004		Check and Clean-Up Canonical SNC Names		2
Table	USRACL			
Client	100			
\$ Action	Msg.	User	SNC name	
<input checked="" type="checkbox"/> empty		BAYERJ		
<input checked="" type="checkbox"/> empty		SAPJSF		
	Initial	passwd		
<input checked="" type="checkbox"/> empty		XIAPPLUSER		
	Initial	passwd		
<input checked="" type="checkbox"/> empty		XIDIRUSER		
	Initial	passwd		
<input checked="" type="checkbox"/> empty		XILDUSER		
	Initial	passwd		
<input checked="" type="checkbox"/> empty		XIREPUSER		
	Initial	passwd		
<input checked="" type="checkbox"/> empty		XIRWBUSER		
	Initial	passwd		
Statistics				
OK			0	
Change			0	
empty			8	
false			0	
Delete			0	
No check			0	

29.01.2004		Check and Clean-Up Canonical SNC Names		4
Table	RFCDESSECU			
\$ Action	Msg.	RFC destination	T SNC application server name	SNC message server name
<input checked="" type="checkbox"/> empty		SB2CLNT050_SNC	3	
Statistics				
OK			2	
Change			0	
empty			1	
false			0	
Delete			0	
No check			0	
29.01.2004		Check and Clean-Up Canonical SNC Names		4
Table	SNCSYSACL			
\$ Action	Msg.	T System ID	SNC name	
Statistics				
OK			2	
Change			0	
empty			0	
false			0	
Delete			0	
No check			0	

Correct the errors, e.g. replace initial password

Be careful when you maintain XI system user, as the passwords have to be changed in several places!

Configure SNC on additional Dialog Server

The following steps have to be repeated in the following sequence on each dialog server.

Download the latest sapcryptolib from service market place

Install the sapcryptolib and sapgenpse for the relevant operating system as described above.

Set environment variables

Set the environment parameters as described above.

Copy SAPSNCS.pse / <SID>SNCS.pse

Copy only the file SAPSNCS.PSE from the exe/run directory of the CI to the exe/run Directory of the dialog server

You could also reference one central SNC PSE for both Central Instance and dialog servers. This central SNC PSE needs to be accessible by all application servers via NFS. This is not used in our case due to security reasons.

Maintain Profile Parameters

Maintain the profile parameters as described above.

Generate credentials

Generate the credentials for user <sid>adm with sapgenpse on each dialog server with the tool sapgenpse as described above. Keep in mind that you use the paths for the dialog server.

In our example, it is:

```
sapgenpse seclogin -p D:/usr/sap/SB1/A00/sec/SB1SNCS.pse -x abcpin -O sbladm.
```

This creates the file cred_v2 in the directory /usr/sap/<SID>/A00/sec of the application server.

Restart the dialog server.

- stopsap r3
- startsap r3

Activate SNC

Activate SNC in the instance profile of the dialog server for which you are configuring SNC right now.

Restart the sap system of the dialog server.

- stopsap r3
- startsap r3

Important: Check dev_w0 if the start up was successful.

If the startup was not successful analyse the SNC errors in order to solve the problem.

Maintain the Access Control List (ACL)

Maintain the Access Control List as described above.

Create an entry for **each application server** from other SAP Systems that needs RFC access to this SAP System.

As we use the SID as part of the SNC name, we only need one entry here.

Not used in our case:

If you have multiple application servers in a remote SAP System that use different credentials (different SNC names), you need to make an entry for each application server in table SNCSYSACL.

Configure SNC for the RFC adapter

Configure RFC Destination used by the RFC Adapter for SNC

Name: AI_RFCADAPTER_JCOSERVER

As described above.

Configure the RFC Destinations that send data to the RFC Adapter for SNC

As described above.

In the property file of the RFC Adapter

Make for the following entries for each SAP sender System.

Thereby you specify the RFC adapter the SNC names as defined in *snc/identity/as* or in the SNC PSE and the patch to the SNC PSE.

```
RfcAdapter.SB1.sncName=p:CN=SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US
RfcAdapter.SB1.sncLib=/usr/sap/SB1/DVEBMGS00/sec/SB1SNCS.pse
RfcAdapter.SB1.sncQop=8
RfcAdapter.SB1.sncAcl=*
#RfcAdapter.SB1.sncAcl=peerB6A peerB6Q
#RfcAdapter.SB1.sncAcl.peerB6A=p:CN=B6A, O=SAP-AG, C=DE
#RfcAdapter.SB1.sncAcl.peerB6Q=p:CN=B6Q, O=SAP-AG, C=DE
```

Configure SNC for the J2ee engine communication via the sap gateway

If the J2ee part of the WebAS 6.20 communications with the ABAP part of the WebAS 6.20 and visa versa the communication takes place over the SAP gateway.

Within XI this functionality can be used to secure the communication of user SAPJSF that is used for the communication between the XI components.

Currently we do not encrypt these connections because we have a one host installation of XI.

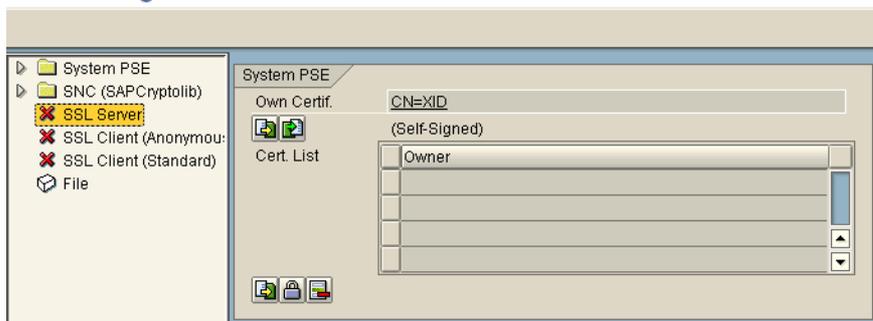
Configure SSL for the Integration Server

In this section we will enabling SSL on the ICM.

Add the following profile parameters in additional to the profile parameters already maintained for SNC:

Profile parameter	Value	Details
ssl/ssl_lib	Linux: /usr/sap/XI2/SYS/exe/run/libsapcrypto. so HP-UX ssl/ssl_lib /usr/sap/XI2/SYS/exe/run/libsapcrypto.sl	
icm/plugin_<xx> > Not used any more ¹¹	< blanc >	<xx> is the number of the ICM plug-in. By default 0 is used for HTTP, and 1 for HTTPS
Icm/server_port _0	PROT=HTTP,PORT=8000,EXTBIND=1	Sets the HTTP Port
icm/server_port _1	PROT=HTTPS,PORT=8443,EXTBIND=1	Sets the HTTPS Port
icm/server_port _2	PROT=SMTP,PORT=0	Sets the SMTP Port
sec/rsakeylength default	1024	Use a key length of 1024 bit (only with kernel release 6.20 and higher), see note 509495. (512 (standard), 768, 1024, 2048)
icm/HTTPS/verifi fy_client	1	0/1 (Default) / 2. F you want to suppress/allow/force the user logon by client certificate in the SSL log.

Trust Manager



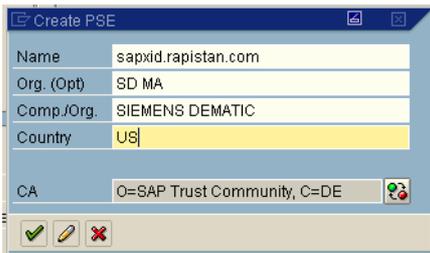
High-light *SSL Server*.
Choose *Create* in the context menu.

¹¹ Still necessary? Not in note 510007 any more

 Use that button to deactivate the suffix. The field CA gets greyed out and the field Country can be maintain.

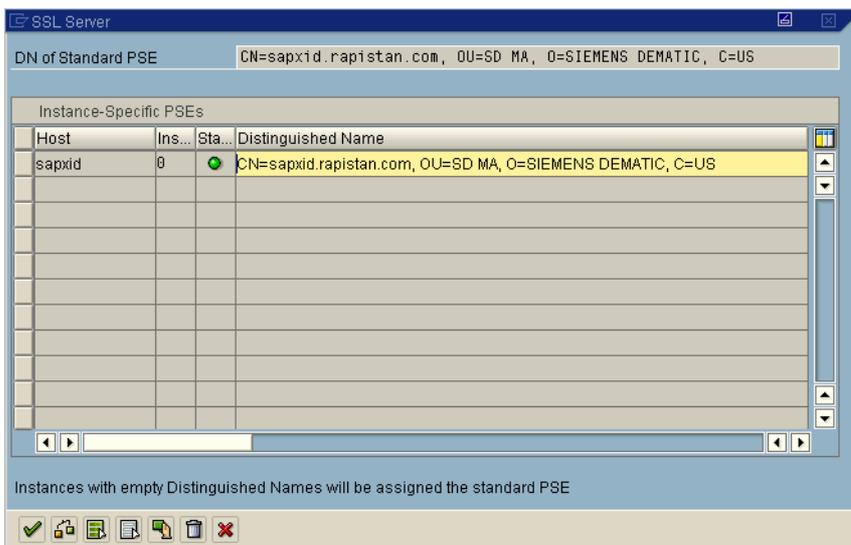
Make the following entries:

- Name: <here you have to define the **fully qualified domain name of the Server**¹² >
- Org: SD MA
- Comp./Org.: SIEMENS DEMATIC
- Country: US



Choose *Enter*.

The result is:



Choose *Enter*.

Choose *Save*.

¹² The difference to SNC (where the SID is used for the name) is, that there a whole SAP system is addressed. With SSL always a single server is referenced.

Create SSL Client Certificate

The PSE is used in the SSL log if the Web AS issues a HTTPS request a s client. For technical reasons, there must always be a SSL client PSE even if the system does not issue any client requests. The reason is that the SSL implementation cannot be started if the PSE is missing.

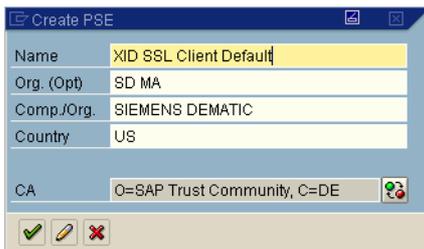


When creating it specify the following name:

Name: <SID> SSL client default

 Use that button to deactivate the suffix. The field CA gets grayed out and the field Country can be maintain.

Make the following entries:



Choose *Enter*.

Choose *Save*.

Activate SSL

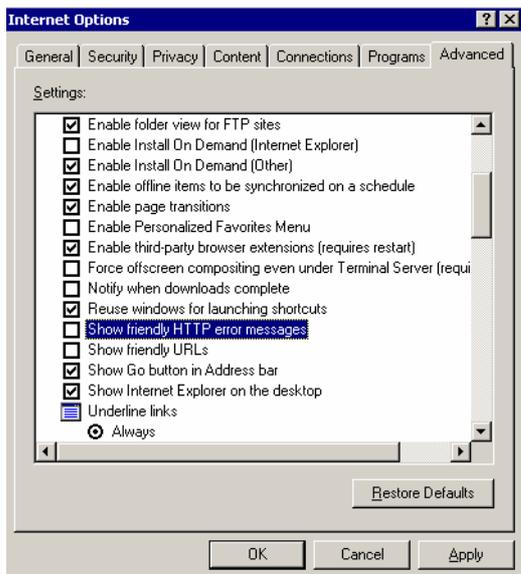
Restart the ICM in order to activate the SSL.

In TA SMICM → Administration → ICM → Exit Hard.

Testing the connection for SSL Server Authentication

Prepare the Internet Explorer properties.

In the menu choose *Extras* → *Internet Options* → *Advanced*.



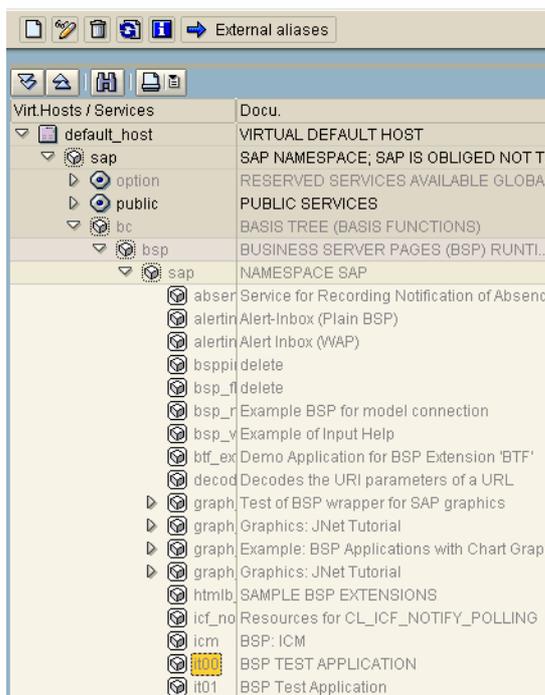
Disable *Show friendly HTTP error messages*
 Disable *Show friendly URLs*

Choose *OK*.

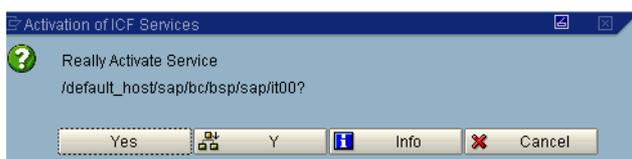
TA SICF

Activate service it00.

Maintain service



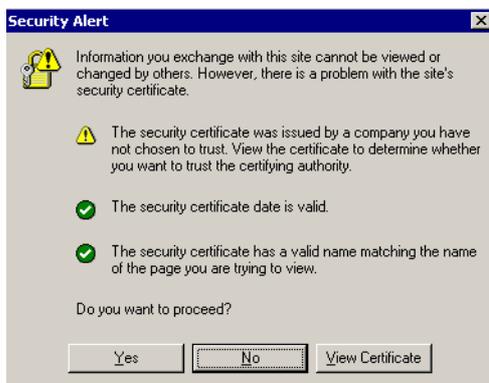
Choose *Activate Service* in the context menu.



Choose *Yes*.

Call the service with the following link:

<https://sapxid.rapistan.com:8443/sap/bc/bsp/sap/it00/default.htm>

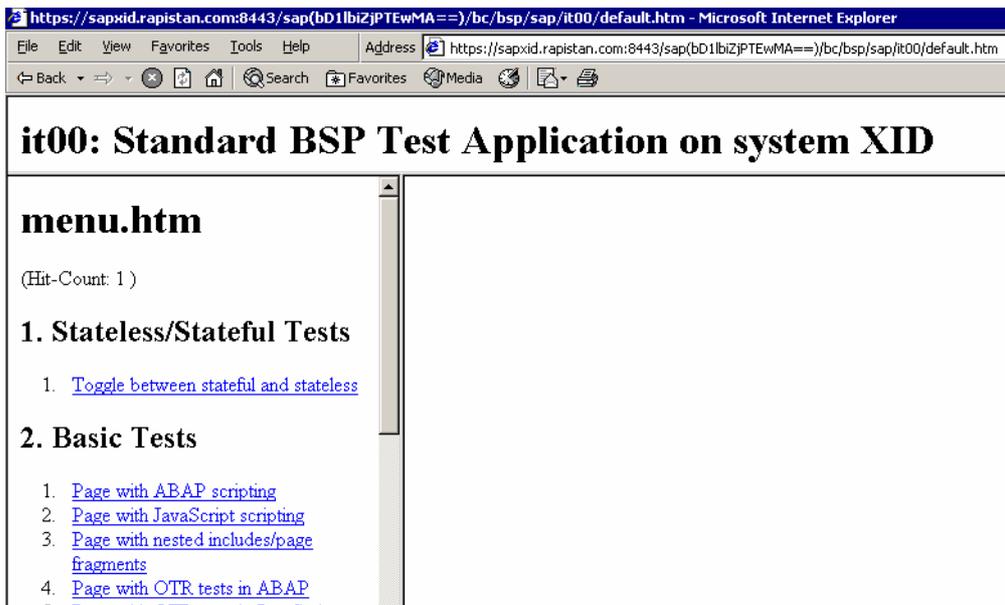


Choose *Yes*



Specify your *SAP User name* and *password* and press *OK*.

The result is the following:



Calling the page from Offenbach returns the following.

```
<BODY o
```

This might be due to the proxy settings.

Change the Pipeline settings in the SLD.

Pipeline is called from an adapter engine

No changes need to be made so far in the pipeline settings. The pipeline settings in the SLD are not required because the link to the pipeline is maintained directly in the adapter property files (see below).

Pipeline is called with a PROXY

If you want to force the use of SSL set the parameter IS_URL in the TA SXMB_ADM to the following value.

<http://sapxi2/8443/sap/xi/engine/entry?action=execute&pipelineid?=entry>

→ Needs to be tested

As an alternative, you can change the pipeline settings in the SLD content maintenance

<http://<host>:<HTTPport>/sap/xi/engine/?type=entry>

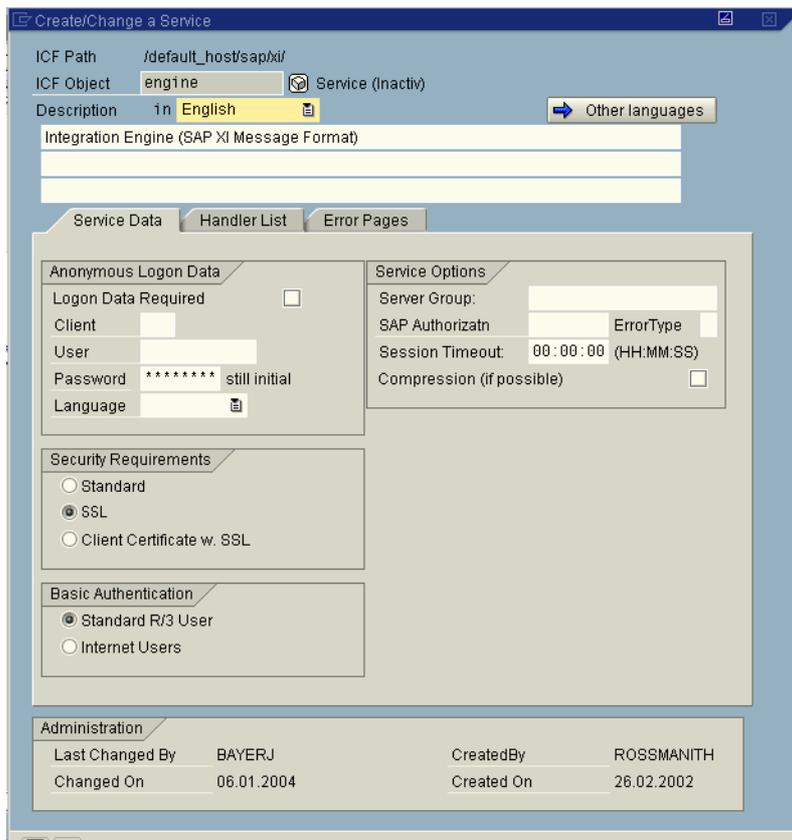
Creating two parameters IS_URL is not possible: Creating a new HTTP Server Port with Name Pipeline_Integration_Server_BS_XI2_CLNT100 was not possible:

CIM_ERR_ALREADY_EXISTS: Instance already exists:
SAP_HTTPServicePort.CreationClassName="SAP_HTTPServicePort",Name="Pipeline_Integration_Server_BS_XI2_CLNT100",SystemCreationClassName="SAP_BCMessageServer",SystemName="XI2.HostName.sapxi2.MessagePort.3600"

TA SICF

Make changes in the service of the Pipeline service in order to use SSL.

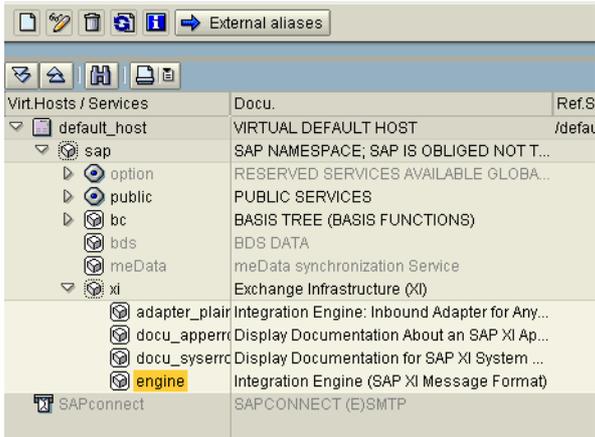
In our case: Do not make direct changes to the service /default_host/sap/xi:



Create an Alias for HTTPs Calls of the Pipeline

The use of this Alias in the Adapter Configuration Files is for all adapters that go to production later on mandatory.

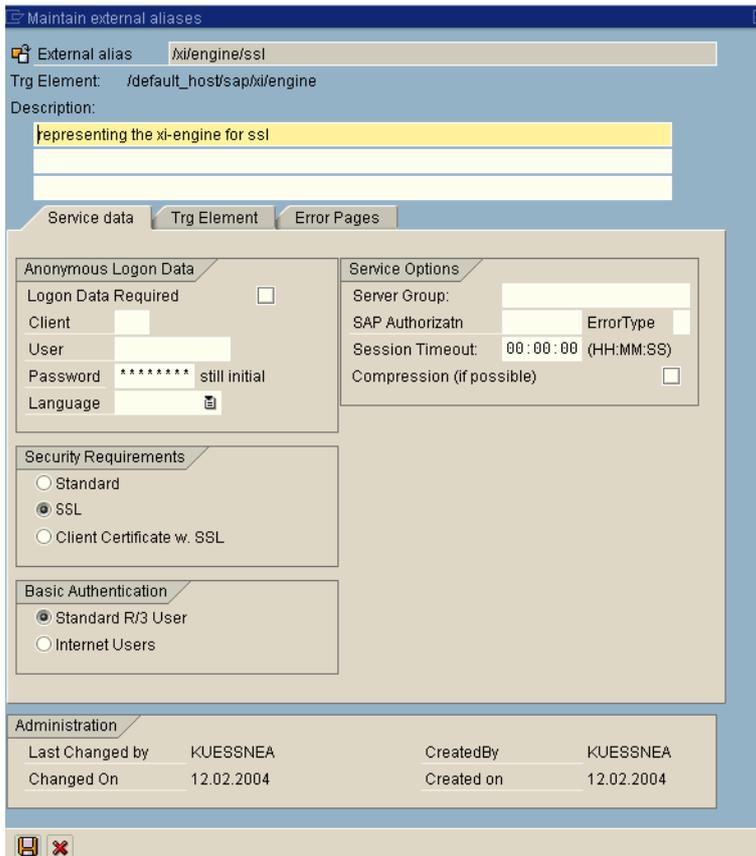
Maintain service



Choose *External aliases*

High-light *default_host*

Choose Create.

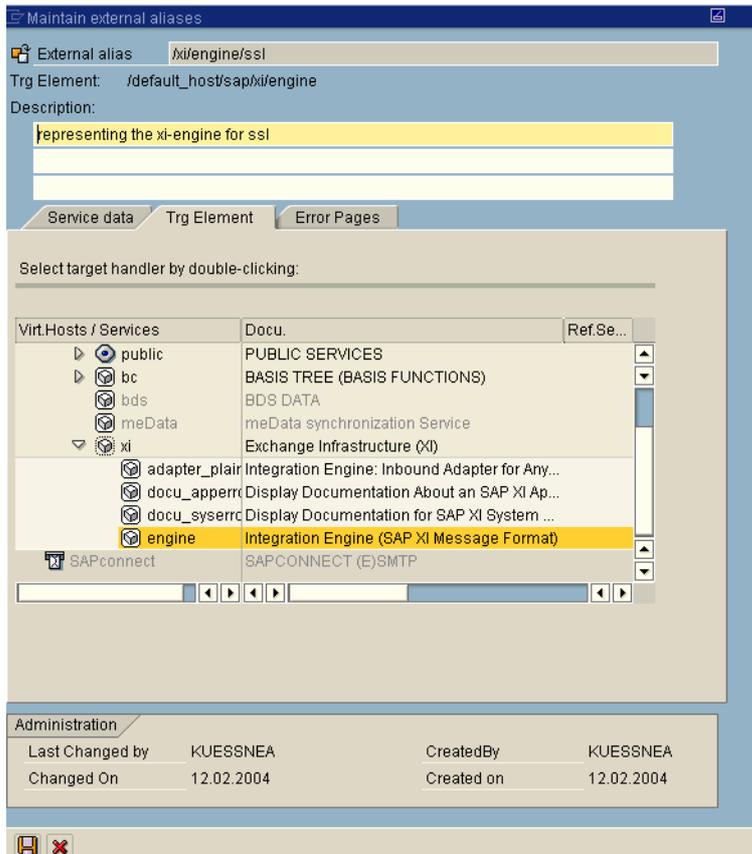


Name: /xi/engine/ssl

Enter a documentation

Check SSL

Go to tab *Trg Element*



Double click on *engine*
Choose *Save*.

Choose *Save*.

Test the SSL Alias:

On XID use the following link:
<https://sapxid:8443/xi/engine/ssl?type=entry>

The following result should appear:



This means that the XI pipeline on XID was called successfully via https and encrypted with SSL.

Create an Alias for HTTP of the Pipeline

Use calls to the Pipeline. Only use this alias for testing purposes.

Test the HTTP Alias:

On XID use the following link:

<http://sapxid:8000/xi/engine/http?type=entry>

The following result should appear:



This means that the XI pipeline on XID was called successfully via http.

Configure SSL for a stand-alone adapter engine with a self-signed certificate

Business Systems with local adapter engine

On the the following Business Systems a local adapter Engine is installed:

System	Operating System	Address local adapter engine
IET	Unix (HP-UX?) - Dialog server	http://rdcsb1:8200
MS SQL Server		
SB1		
XID		

Currently we do not use a certificate signed by a Certification Authority (CA).
Organizational procedures

Install the local adapter engine on the servers above.

See Adapter Engine documentation.

You find the documentation for example on the central adapter engine:

Open <http://<host name integration server>:<Port adapter engine, 8200 standard>>.

Set environment variables

In our example we use the environment variables that were already configured for SNC.
For detailed information see Chapter *SNC - Set environment variables*.

If you install the adapter engine on a non-SAP System you have to adjust the parameters in this steps according to your needs.

Installing the SAP Cryptographic Library on the server of the local adapter engine

In our example we generate a PSE for an adapter engine on a SAP system,
Therefore we use the settings that already exist for SNC.
For detailed information see Chapter *SNC - Install SAP Cryptolib on central instance*.

If you install the adapter engine on a non-SAP System you have to adjust the parameters in this and the following steps according to your needs.

Creating a PSE for the server of the local adapter engine using SAPGENPSE without certificate request

Execute in directory /usr/sap/<SID>/exe/run the command to create a SNC PSE using SAPGENPSE without certificate request:

```
sapgenpse get_pse <additional_options> [-p <PSE_name>] [-r
<cert_req_file_name>] [-x <PIN>] [DN]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-r	<file_name>	File name for the certificate request	Path description (in quotation marks, if spaces exist)	stdout
-x	<PIN>	PIN that protects the PSE	Character string	None
None	DN	Distinguished Name for the server The Distinguished Name is used to build the server's SNC name.	Character string (in quotation marks, if spaces exist)	None

Additional Options

Option	Parameter	Description	Allowed Values	Default
-s	<key_len>	Key length	512, 1024, 2048	1024
-a	<algorithm>	Algorithm used	RSA, DSA	RSA
-noreq	None	Only generate a key pair and PSE. Do not generate a certificate request.	Not applicable	Not set
-onlyreq	None	Generate a certificate request for the public key stored in the PSE specified by the -p parameter.	Not applicable	Not set

The SSL Distinguished Name for the adapter engine consists of the following elements:

- CN = ADAPTERENG_<SID>
- OU = SD MA
- O= SIEMENS DEMATIC
- C = US

The Distinguished Name for the local adapter engine for example on SB1 is:

p: CN=ADAPTERENG_SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US

```
sapgenpse get_pse -s 1024 -a RSA -p
/usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse -noreq -x empass
"CN=ADAPTERENG_SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US"
```

The result is the following.

In directory /usr/sap/SB1/DVEBMGS00/sec/ the PSE ADAPTERENG_SB1.pse is created.

Generate Credentials

```
sapgenpse seclogin <additional_options> [-p <PSE_name>] [-x <PIN>] [-O
[<NT_Domain>]\<user_ID>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None
-O	[<NT_Domain>]\<user_ID>	User for which the credentials are created. (The user that runs the server's processes.)	Valid operating system user	The current user

Additional Options

Option	Parameter	Description	Allowed Values	Default
-l	None	List all available credentials for the current user.	Not applicable	Not set
-d	None	Delete PSE	Not applicable	Not set
-chpin	None	Specifies that you want to change the PIN	Not applicable	Not set

Creating Credentials for the Server

The following command line opens the adapter engine's PSE (ADAPTERENG_<SID>) that is located at /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse and creates credentials for the user <sid>adm = sb1adm. The PIN that protects the PSE is empass.

```
sapgenpse seclogin -p /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse -x  
empass -O sb1a
```

The PSE is self-signed by sapgenpse. The file created is adaptengxi2.crt.

The file cred_v2 is used to store the credentials and stored in directory /usr/sap/<SID>/DVEBMGS<Instance number>/sec.

Restart the SAP system

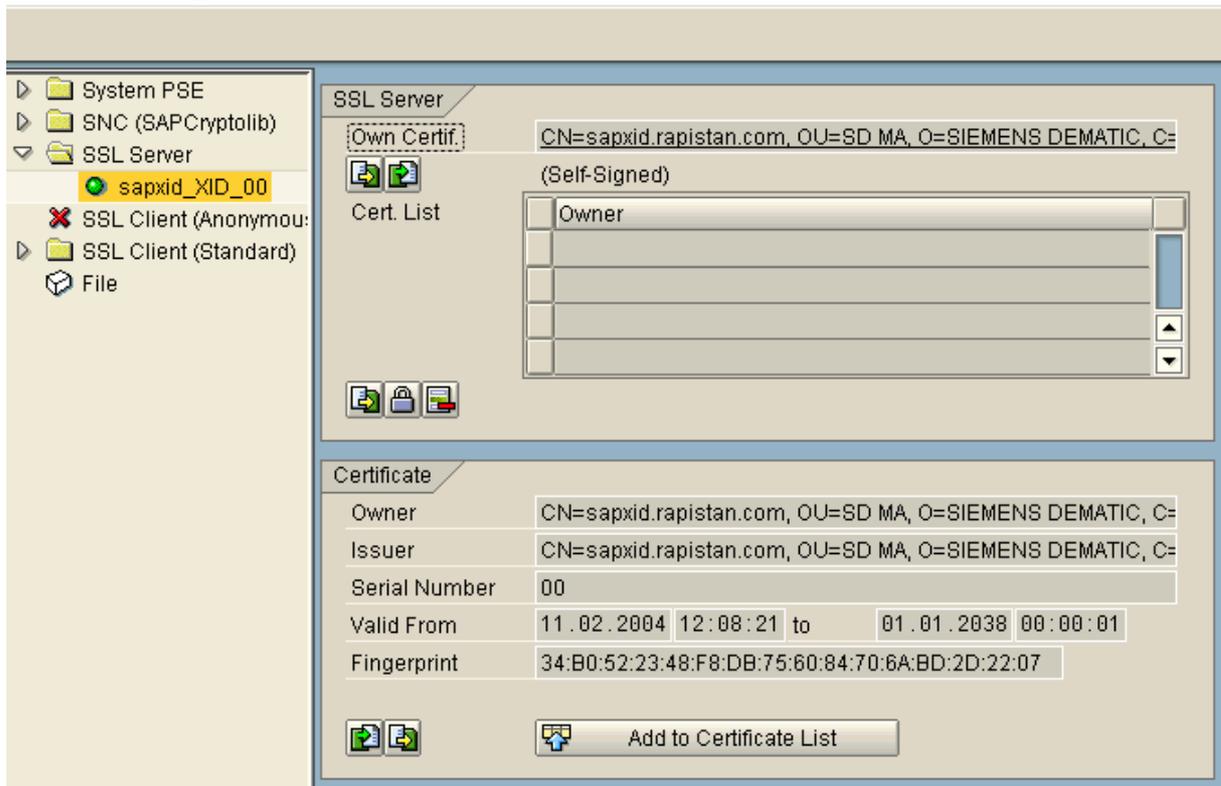
- stopsap r3
- startsap r3

Export the certificate of the partner server

On the Integration Server

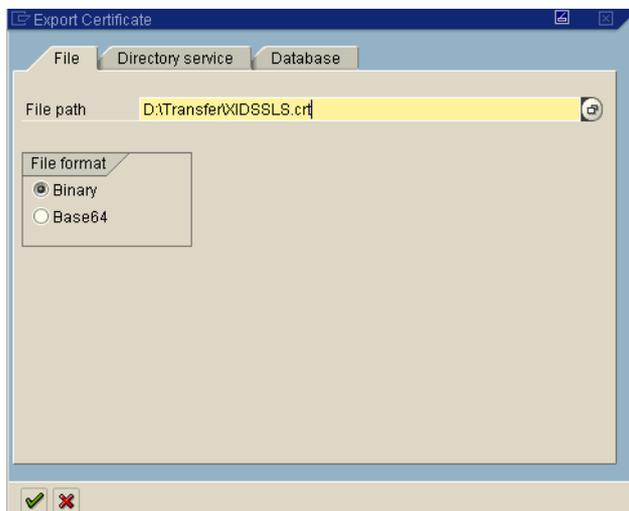
On the integration server, go to transaction STRUST double-click on SSL Server. Double-click on the entry in *Own Certif.*

Trust Manager



Choose  Export Certificate

Save the certificate on the file system.



On the Adapter Engine

Exporting the Integration Server's Public-Key Certificate

Export the public key certificate of the adapter engine with the following command:

```
sapgenpse export_own_cert -o <output_file> -p <PSE_name> [-x <PIN>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-o	<output_file>	Exports the certificate to the named file	Path description (in quotation marks, if spaces exist)	stdout
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None

The following command line opens the adapter engine's PSE that is located at /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse and creates credentials for the user <sid>adm = sb1adm. The PIN that protects the PSE is empass.

The certificate is thereby stored in the file /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse

```
sapgenpse export_own_cert -o /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.crt  
-p /usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse -x empass
```

Import the certificate of the partner server into your SSL Server PSE

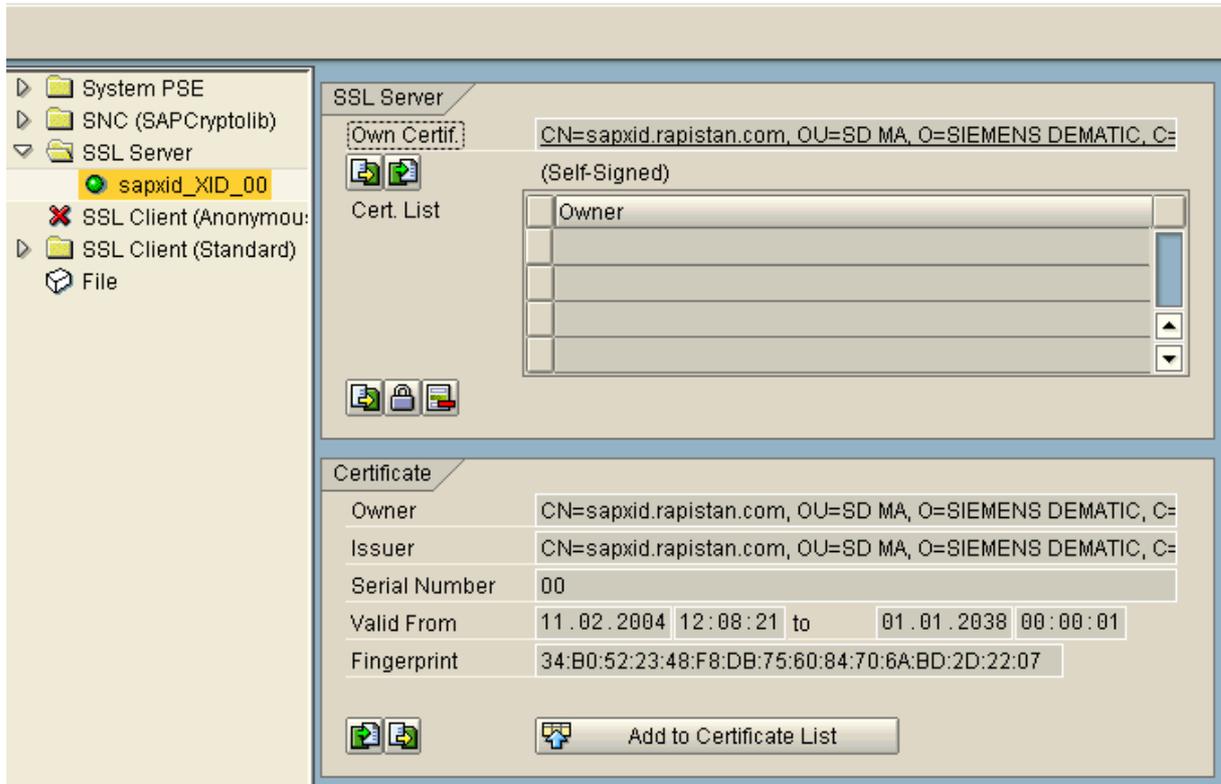
Integration Server

The certificates (public key) of the adapter engines that communicate with the Integration Server XID have to be imported into the PSE of Integration Server.

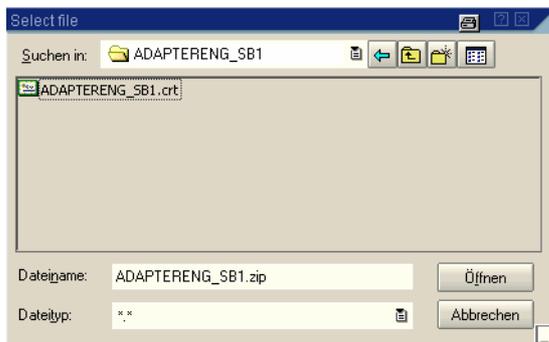
Import the public key with a file

In TA STRUST → Choose SSL Server → Double-click on sapxi2_XI2_00.
Double-click on the entry on *Own Certif.*

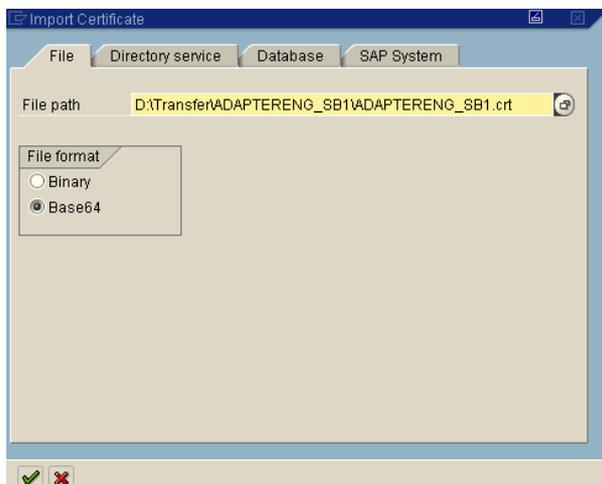
Trust Manager



Choose *import Certificate*  to import the certificate's public key of all adapter engines that use SSL. Browse to the exported certificate of the adapter engine.



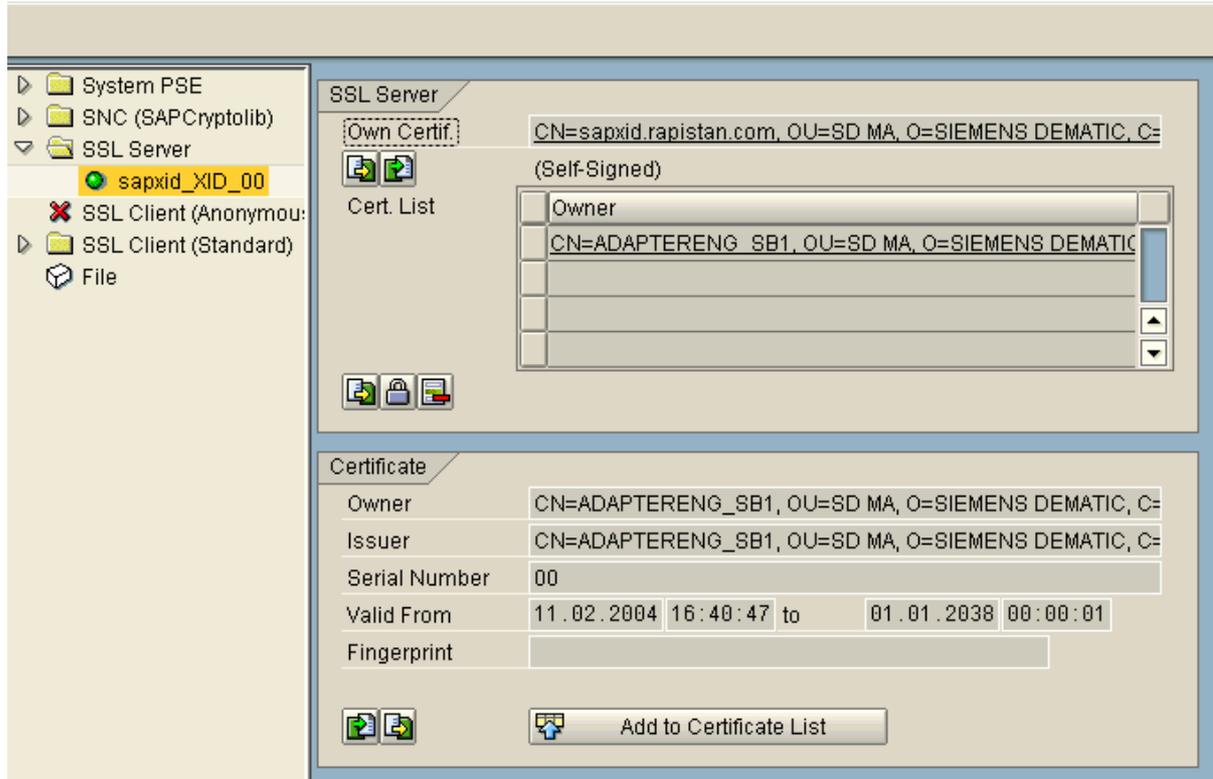
Choose 'Base64' for the import:



Now in the 'Trust Manager' choose *Add to Certificate List*.

The result is:

Trust Manager



Adapter Engine

The certificate (public key) of the Integration Server engines that communicate with adapter engines has to be imported into the PSE of the adapter engines.

Use the tool's command `maintain_pk` to maintain the server's certificate list.

```
sapgenpse maintain_pk [<additional options>] [-a <cert_file>] [-d <number>]
-p <PSE_name> [-x <PIN>]
```

Standard Options

Option	Parameter	Description	Allowed Values	Default
-a	<cert_file>	Add certificate from file <cert_file> to the	Path description (in quotation marks, if	None

		certificate list.	spaces exist)	
-d	<number>	Delete certificate number <number> from certificate list.	Numerical value	None
-p	<PSE_name>	Path and file name for the server's PSE	Path description (in quotation marks, if spaces exist)	None
-x	<PIN>	PIN that protects the PSE	Character string	None

Additional Options

Option	Parameter	Description	Allowed Values	Default
-cacert	None	The certificate to import is a CA root certificate.	Not applicable	None
-l	None	List existing certificate list	Not applicable	Not set

```
sapgenpse maintain_pk -a /usr/sap/SB1/DVEBMGS00/sec/XIDSSLS.crt -p
/usr/sap/SB1/DVEBMGS00/sec/ADAPTERENG_SB1.pse -x empass
```

Install the IAIK files on the local adapter engine

If the adapters are to communicate using HTTPS, additional libraries to implement the SSL protocol are required.

Download the Java libraries *iaik_jce.jar* and *iaik_ssl.jar* from the SAP Service Marketplace (<http://service.sap.com>).

Use the alias *download*.

Choose *SAP Cryptographic Software*, but ensure that you follow the regulations outlined there.

Select JAVA CryptoToolkit (your release).

You must copy both libraries to the Java CLASSPATH after installation.

In our case we copy *iaik_jce.jar* and *iaik_ssl.jar* them to the *tech_adapter* directory of the local adapter engine.

Now you have to add these jar files in *run_adapter.sh* and *run_adapter.cmd*.

If the local adapter engine should use HTTPS protocol to communicate with XI, you must configure the HTTP server correspondingly for the adapters. This is achieved using the *HttpServer* service.

If the browser is to be connected to the Adapter Engine configuration screen using HTTPS protocol, you must configure the *GuiBrowserEngine* service correspondingly.

We currently do not use this option.

Maintain the configuration of the adapter engines

Maintain the services

This is the service for replacing the password tokens in the adapter configurations. If required, you can change the beginning and end characters of the tokens centrally here.

Parameter	Value used	Comments
PWD.beginSeparator	<%!	PWD.beginSeparator can be any character string. Existing replacements of tokens in the adapter configurations must then be adjusted correspondingly.
PWD.endSeparator	%!>	PWD.endSeparator can be any character string. Existing replacements of tokens in the adapter configurations must then be adjusted correspondingly.

HttpServer

The HTTP server is used by the outbound adapters. You have the option of making the

following settings during the configuration of the HTTP server:

Parameter	Value used	Comments
http.authentication	basic	Define whether user-authentication is performed. In the basic setting, the HTTP client of the Integration Server must log on to the Adapter Engine with a valid user that has been assigned the role HTTP Server User. In the setting none , no authentication takes place. The default is none . This can be used for testing purposes only. Values: none basic
HTTP.transmission	SSL	Defines whether the Integration Server and the HTTP server of the Adapter Engine are to communicate using HTTP (plain) or HTTPS (SSL). The default is plain . We use SSL instead. Values: plain SSL
HTTP.SSLcertificate	<SSL distinguished name defined above>	SSLcertificate specifies the complete file name of a password-protected certificate
HTTP.SSLcertificatePassword	<Password for the SSL PSE >	SSLcertificatePassword specifies the corresponding password (the password can be protected by using the token concept, described above). To make the installation HTTPS-enabled you must install additional Java libraries that are available from the SAP Service Marketplace (http://service.sap.com). These IAIK libraries must be located in the Java CLASSPATH.

GUIBrowserEngine – not used

Parameter	Value used	Comments
port	<portNo	The GUI browser engine represents a separate HTTP server with a configurable port. This is the HTTP port that the browser can log on to. The default value is 8200 and must not be changed unless it has already been reserved
zones	root	Must not be changed under any circumstance!
rootDirectory	Administration	Must not be changed under any circumstance!
HTTP.transmission	plain	Defines whether the browser and the Adapter Engine are to communicate using HTTP (plain) or HTTPS (SSL). Values: plain SSL
SSLcertificate	<p12-	SSLcertificate specifies the complete file

	certificate name	name of a password-protected certificate.
SSLcertificatePassword	<p12-certificate password>	SSLcertificatePassword specifies the corresponding password (the password can be protected by using the token concept, described above).

Test the local adapter engine

Once settings are applied, restart the local Adapter Engine.

Check in the adapter engine log files for errors.

Maintain properties files of all adapter instances

Change the call of the Pipeline in all adapter configuration files to
XMB.TargetURL=https://< **fully qualified domain name of XI host** >:8443/xienginessl

In the productive all data has to be encrypted using SSL.

Maintain the endpoints of the Business Scenarios in the Integration Directory

Find out which of the endpoints use an adapter of the local adapter engine.

Change the protocol from HTTP to HTTPS.

Now test all business scenario that have an endpoint on that local adapter engine.

Configure SSL for a stand-alone adapter engine with certificate from a CA

Currently we do not use certificates for the stand-alone adapter engine.

Installing the SAP Cryptographic Library on the server

Creating a PSE for the server using SAPGENPSE

Execute in exe/run the command:

```
sapgenpse get_pse <additional_options> [-p <PSE_name>] [-r <cert_req_file_name>] [-x <PIN>] [DN]
```

```
sapgenpse get_pse -s 1024 -a RSA -p /usr/sap/XI2/DVEBMGS00/sec/ADAPTERENG_SB1.pse -r /usr/sap/XI2/DVEBMGS00/sec/ADAPTERENG_SB1req -x empass "CN=ADPTERENG_SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US"
```

Creating the Server's Credentials Using SAPGENPSE

Execute in the directory exe/run the command:

```
sapgenpse seclogin <additional_options> [-p <PSE_name>] [-x <PIN>] [-O [<NT_Domain>\]<user_ID>]
```

```
sapgenpse seclogin -p /usr/sap/XI2/DVEBMGS00/sec/ADAPTERENG_SB1.pse -x empass -O xi2adm
```

The following result appears:

```
running seclogin with USER="xi2adm"
  creating credentials for yourself (USER="xi2adm")...
  Added SSO-credentials for PSE
"/usr/sap/XI2/DVEBMGS00/sec/ADAPTERENG_SB1.pse"
  " CN=ADPTERENG_SB1, OU=SD MA, O=SIEMENS DEMATIC, C=US"
```

You can see on operating system level that cred_v2 was changed.

Sign the Server Certificate

Sign the certificate request ADAPTERENG_SB1req by your certificate provider.

As a type I specified other *WebServer*.

Name the file with the ending .crt.

In this example it is

1 Enter Certificate Request 2 Import Certificate

```
Import Certificate into Webserver.
Below is the test certificate for your Webserver . Please copy & paste the text beginning
CERTIFICATE -----" into a local text file on the server.
-----BEGIN CERTIFICATE-----
MIICp jCCAg+gAwIBAgICDUUwDQYJKoZIhvcNAQEFBQAwUDELMakGA1UEBhMCREUx
HDAaBgNVBAoTElNBUCBUcnVzdCBDb21tdW5pdHkxZzANBgNVBAstB1Nlcn21cjES
MBAGA1UEAxMJU2VydWVyeTENBMB4XDTAOMDEwNTE2Mjk1NFoXDTAOMDMwNTE2Mjk1
NFowVjELMakGA1UEBhMCREUxHDAaBgNVBAoTElNBUCBUcnVzdCBDb21tdW5pdHkx
EzARBgNVBAstB1NBUCBUcnVzdCBDb21tdW5pdHkxZzANBgNVBAstB1NBUCBUcnVzdCBDb21tdW5pdHkx
CSqGSIb3DQEBAQUAA4GNADCBiQKBgQD+2iK1/3i0AxUns8ywYFYL23MQ54qXYwM
PNqUJLaS2udc9dyrrA/0Jq/y3w1g/44ZLN1LRoi8tWryAIk01pTK+5YsXR6UkSj
25TgtMgBpY+hLakjCxG1GJ2Jwm2DL4EiIppcEKcesw3VwBHh5VfvxX9t80nHU5GC
pgL5/DrRGwIDAQABo4GIMIGFMAwGAlUdEwEB/wQCMAAwJQYDVRO5BB4wHIYaaHR0
cDovL3Nlcn21cjE2Uuc2FwLmNvb39UQ1MwDgYDVRO5BAQH/BAQDAgTwMBOGAlUdDgQW
EBTtOEFT07pgLM36U+Ry9dpNc1mujAfBgNVHSMEGDAwqB3B24Pz3vs1GAi5Ab23
HVudczVbSTANBgkqhkiG9w0BAQUFAAOBgQCp31HGnd577q7NqDfVtsoTy2or23g
gF6qSIVh0Z/VJ20cFXAm9/V531cCVYb0kTcx0JBWylbwJdgoPXwvJq025n3m717
Pmxy3UsX1irL/PmUj17P94TJ7nk+TnE.Ff5pYAuDQ9HLfHZrhybfYXRf+f43TY/
QszxRo/DLjHXcw=
-----END CERTIFICATE-----
```

Import the certificate into the PSE

```
sapgenpse seclogin additional_options> [-p <PSE_name>] [-x <PIN>]
[-O [<NT_Domain>\]<user_ID>]
```

In this example:

```
sapgenpse seclogin -p -p /usr/sap/XI2/DVEBMGS00/sec/ADAPTERENG_SB1.pse -x
-O xi2adm
```

Export/Import the certificates between the servers

See above.

Transport

Nothing of this configuration is transported. It has to be maintained in each system.

Related Notes and documents

Related Notes

578377
354819 install the newest version of SAPSECULIB
397175 use SAPCRYPTOLIB instead of SAPSECULIB
509495
5100 Setting up SSL on the Web Application Server

Related Documents

Service Marketplace

Documents can be found on the service market place under service.sap.com
Alias Security → Security in Detail → Secure System Management

Using the SAP Cryptographic Library for SNC

SNC User Guide

Using the SAP Cryptographic Library for SNC

SAP Web Application Server Security

Configuring the Use of SSL on the SAP J2ee Engine

Documents can be found on the service market place under service.sap.com
Alias xi → Media Library

XI Security Guide

XI Configuration Guide

Online Help

Help.sap.com

→ Netweaver → WebAS 6.20 SP25

→ Security

 → Using the SAP Cryptographic Library for SNC

 → Configuring SNC for Using the SAPCRPYTOLIB Using SAPGENPSE

Other XI Documentation

Adapter Engine Documentation

You find the adapter engine documentation on the XI server in the directory
`/usr/sap/<SID>/SYS/global/tech_adapter/Administration/Documentation.`

Errors

Test sm59: XI2:

Error occurred when calling remote function. SNC n

Message no. SR000

Test sm59: XIP

SNCERR_INVALID_FRAME A received frame is invalid/t

Message no. SR000

Solution:

- The error only occurs in Unicode Systems.
- Kernel Patch 1300 or higher.
- See Note 695205

Other comments

SNC:

- beide System einspielen
- Problem falls glib anders als bei entpackten files
- Applicationsserver: eigenes Zertifikat
- Profil parameter
- Sapcryptolib: <sid>adm
- Funktion ticket

SSL:

- Client Certificate
- 32 Bit also?
- DIR_EXECUTABLE to store PSE
- Sapcryptolib: <sid>adm
- Funktion ticket
- SECUDIR to the sec sub-directory

Report RSPFPAR

- Values of all profile parameters in START, DEFAULT, INSTANCE profile

Report TU02

- Values of all profile parameters of all servers in the system