SAP Basis Configuration - Server

System description

1.4.2

2024-01-15

SEAL Systems

Copyright

SEAL Systems AG
Lohmühlweg 4
D-91341 Röttenbach
Germany

Contents

1 Introduction

subject - connection external system/SAP

SEAL Systems offers a range of integration modules for the output management systems or DPF from SEAL Systems as external system and SAP including add-ons for SAP PDM, creation of digital paper or plot reproduction services for SAP. For data exchange between the external system and the SAP system, the two systems must recognize each other.

The following documentation describes the necessary steps to establish connections between the external system and the SAP system to exchange data.

alternatives

The following alternatives are available for data exchange between SAP and the external system, the configuration of which is discussed in this documentation:

- RFC destination
- HTTP connection, for example in combination with a REST interface
- Web service with the SAP system as provider or consumer

SAP gateway

RFC servers logon on the SAP gateway.

Application servers communicate with their local SAP gateways via an optional central gateway.

message server

A message server is used for load balancing for RFC clients.

 reference

You may find information about a secure RFC communication via SNC and SSO and about the HTTPS support in [SAP_BASECONF_SNC_TEC] and [SAP_BASECONF_HTTPS_TEC]

in this chapter

This chapter deals with the following topics:

Conventions in this Documentation

path specification

The path information given in this documentation is relative to the installation directory of PLOSSYS netdome. This is usually the home directory of the `plossys` user. The path information is indicated in Windows notation only in most cases. This corresponds to the Linux directory structures unless noted otherwise.

typography

The following table lists the typographical conventions employed in this documentation.

Typographical Convention	Meaning
`Consolas`	File names, paths, commands, menu items, keywords, special values, short scripts and examples
`Consolas italic`	Parameters; variables that must be replaced by current values

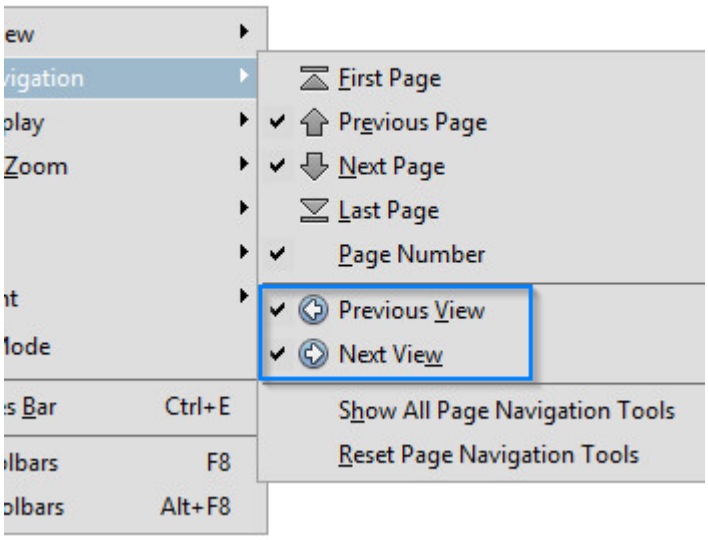
Activate the Retrace of your Viewing Path in PDF

description

Adobe Reader provides buttons to retrace your viewing path of PDF pages that you viewed earlier. This makes reading easier and helps to keep the central theme.

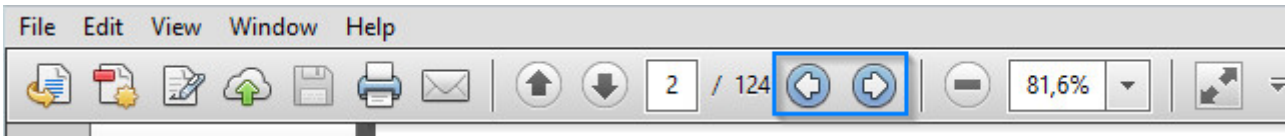
instructions

This is how you activate the buttons for retracing your viewing path in the PDF documentation as of Adobe Reader 10:

Step	Action
1	Activate the following options via the menu: • View - Show/Hide - Toolbar Items - Page Navigation Show Tools • Previous View: Activate • Next View: Activate 

result

Adobe Reader offers the following buttons to allow you jumping forward and backward in the document while showing pages you viewed in the reverse order that you viewed them:



Variables in this Documentation

meaning of *ModuleGlobal*

The variable *ModuleGlobal* represents the general module short cuts:

- pls
- dvs

meaning of *ModuleSelect*

The variable *ModuleSelect* represents the individual modules:

- convserv
- convservdpf
- dvsviewserv
- filecheck
- rfcserver
- rfcserver

meaning of *Action*

The variable *Action* represents the actions:

- start
- status
- stop

Overview of Contents

structure

This documentation has two parts: a description and a reference. The first part describes the functionality and the installation process using figures, step-by-step-procedures and explanatory texts. The second part serves as a detailed reference guide, containing configuration settings, keywords etcetera.

description

The description deals with the following topics:

→ *Overview - Data for RFC Connection*, Page 15

This chapter offers an overview of the relevant configuration files and their correlation in the context of communication via RFC connection.

→ *RFC Destination - Configuration*, Page 17

This chapter describes the configuration settings required for communication via RFC destinations.

→ *Static RFC Destinations for SNC Support*, Page 56

This chapter explains how the connections for the data exchange between servers and SAP systems are established and tested in case of communications via static RFC destinations. Static RFC destinations offer the following advantages:

- SNC is supported for sapftp/saphttp.
- Static RFC destinations can be explicitly enabled/restricted at the gateway.

Static RFC destinations are available for:

- RFC client:
  JSAPcli
- RFC server:
  DPF
  DMS Loader/ABAP
  XSA
  (DMS View Server is not affected))

→ *HTTP Connection - Configuration*, Page 67

This chapter describes the configuration settings required for communication via HTTP connections, which are used in combination with a REST interface, for example.

→ *Integration via Web Service - SAP as Provider*, Page 79

This chapter describes the establishing of an integration via Web service, if SAP serves as provider.

→ *Integration via Web Service - SAP as Consumer*, Page 89

This chapter describes the establishing of an integration via Web service, if SAP serves as consumer.

reference

The reference contains the following chapters:

→ *Configuration Files - Reference*, Page 103

Overview of Contents, Continuation

This chapter explains the configuration files which are evaluated to establish a connection between the OM server and the SAP system in case of communication via RFC destination.

→ *Configuration Tables - Reference*, Page 151

This chapter contains an alphabetically sorted list of all configuration tables relevant for the basis configuration in SAP.

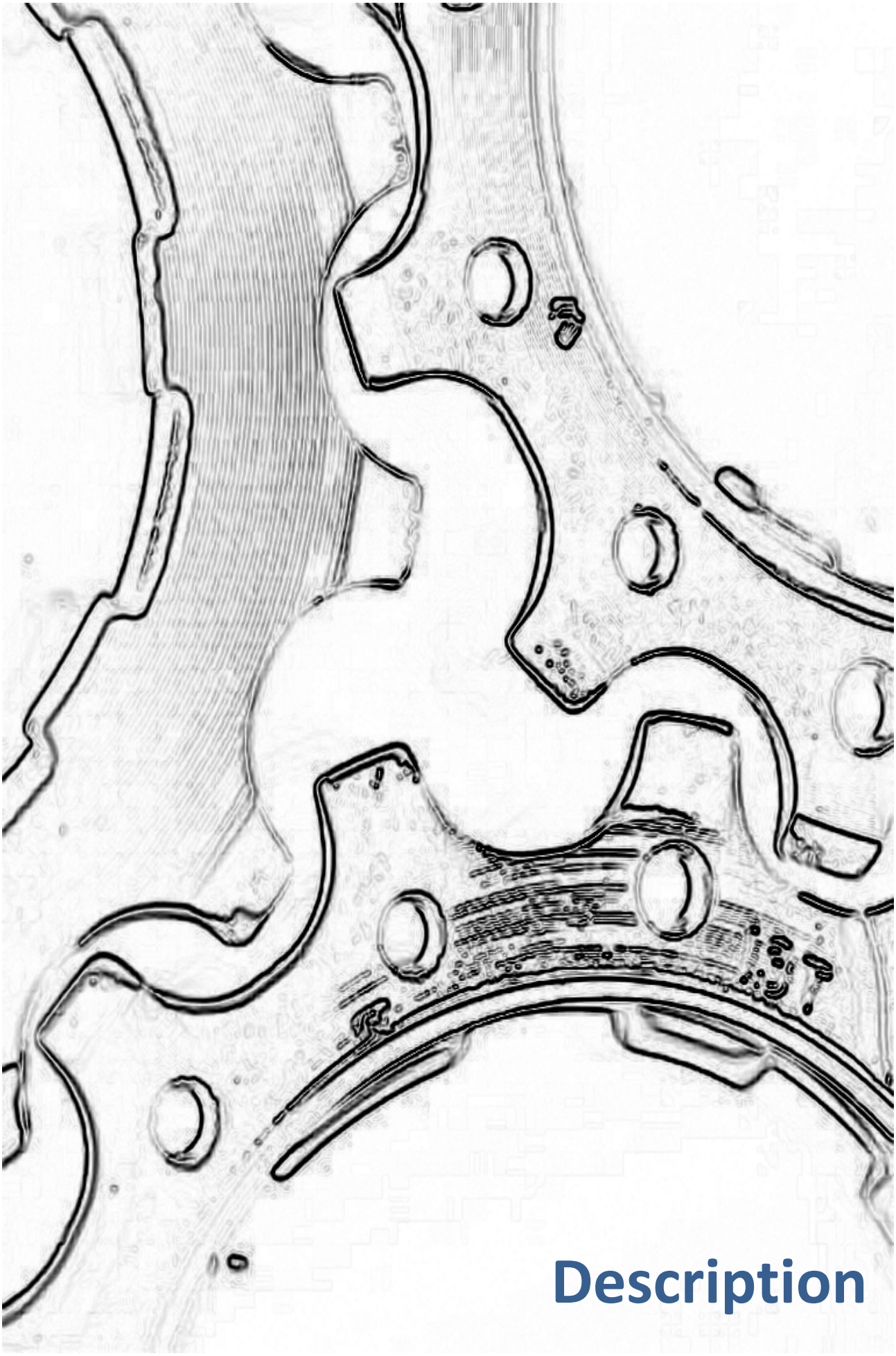
changes

→ *Changes*, Page 157

This chapter describes the most important changes for each released module version.

lists

For an easier overview, a bibliography, terminology list, abbreviation list, and index are included at the end of the documentation.



Description

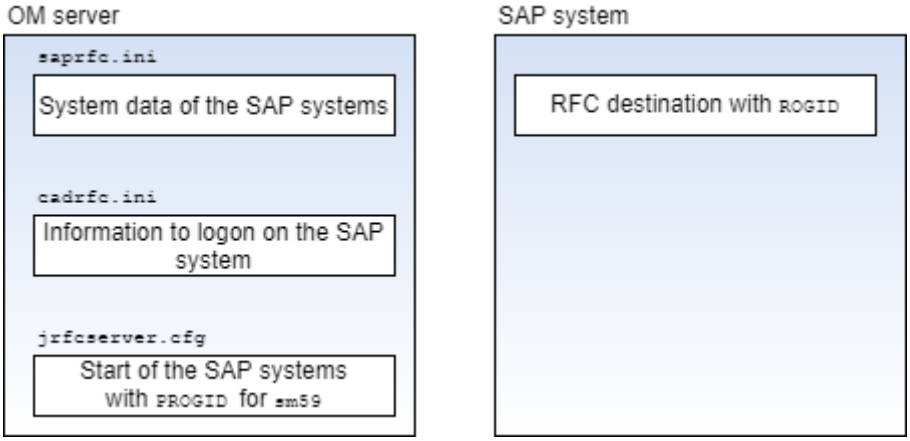
2 Overview - Data for RFC Connection

introduction

This chapter offers an overview of the relevant configuration files and their correlation in the context of communication via RFC connection.

data for the connection via RFC

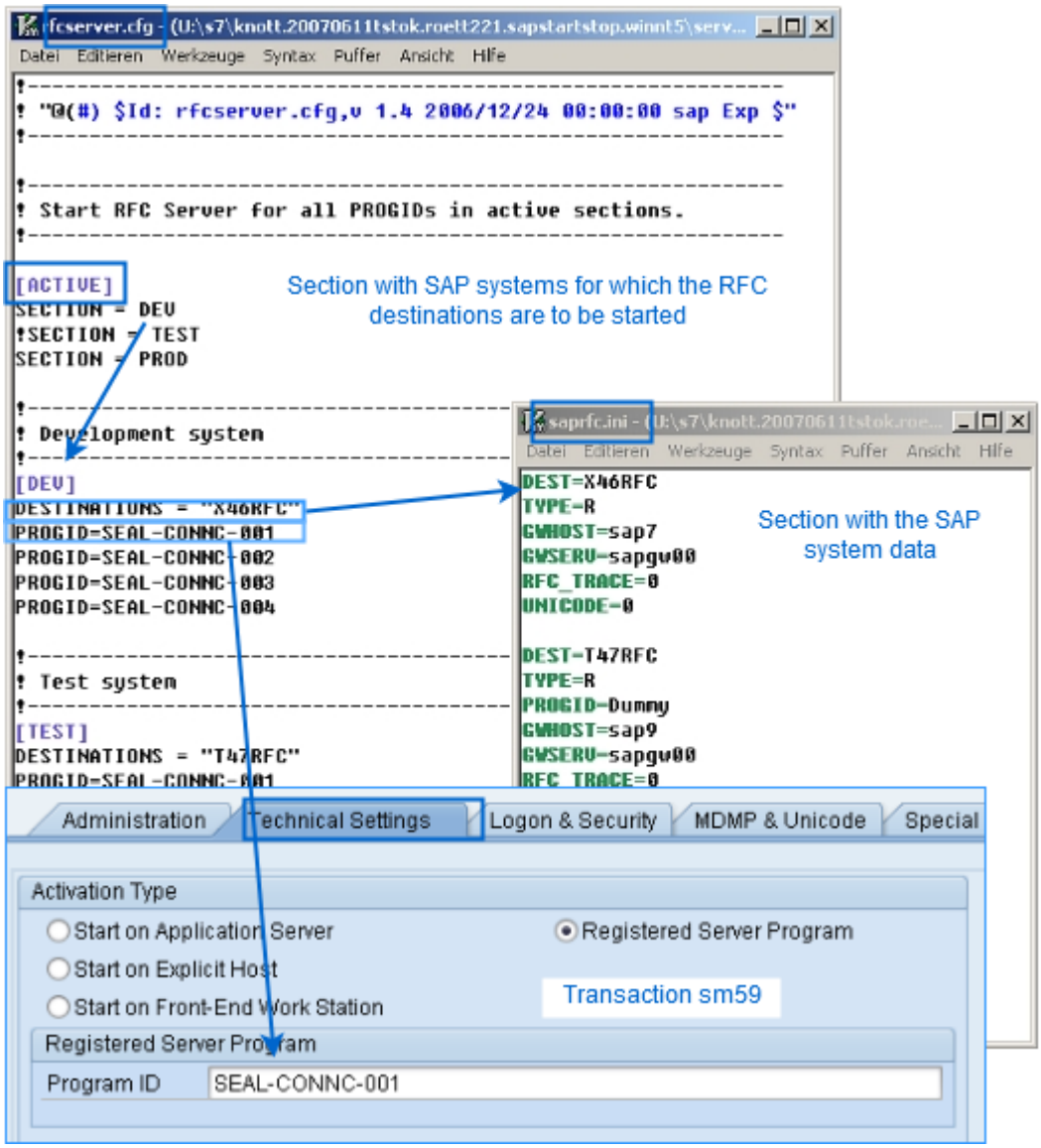
The data necessary for the RFC connection between OM servers and SAP systems is saved in the following files/transactions:



Overview - Data for RFC Connection, Continuation

coherence be-tween the data

The following important correlations exist between these settings:



3 RFC Destination - Configuration

This chapter describes the configuration settings required for communication via RFC destinations.

introduction

Alternatively, the following transfer types are available:

→ *HTTP Connection - Configuration*, Page 67

alternative

This chapter deals with the following topics:

→ *RFC Destination - Configuration on Server*, Page 18

→ *Establish the RFC Destination on the SAP System*, Page 24

→ *Security Configuration for RFC and SAP Destinations*, Page 29

→ *Start and Test RFC Destination*, Page 39

→ *Troubleshooting - RFC Destination*, Page 45

in this chapter

→ *Static RFC Destinations for SNC Support*, Page 56

 related topics

3.1 RFC Destination - Configuration on Server

SEAL Setup Suite

The SEAL Setup Suite module queries the required data and inserts the data into the configuration files. Alternatively, you can customize the configuration files. The following chapters describe the customizing.

general proceed-ing

The configuration on the external server includes the following steps:

Step	Description
1	Define the system data of the SAP systems to which the RFC desti-nations are to be established in `saprfc.ini` → *saprfc.ini - Connection Data*, Page 116
2	Define the RFC destinations to be started in `rfcserver.cfg` → *rfcserver.cfg*, Page 146
3	Specify the information for the first-time logon in: `cadrfc.ini` → *cadrfc.ini - Logon Information*, Page 104 Hint: You can modify the `cadrfc.ini` file on the installation directory. Afterwards you can distribute this file to the required directories.

in this chapter

This chapter deals with the following topics:

Unicode SAP System - Transfer with sapftp/saphttp

Description

SAP provides different versions of the programs `sapftp` and `saphttp` for the data exchange with Unicode SAP systems and non-Unicode SAP systems. The correct variant is started via the `sapftp`/`saphttp` wrapper program from SEAL Systems..

requirement - saprfc.ini

At the configuration of the SAP systems, you have to specify with `UNICODE` in `saprfc.ini` whether the SAP system is a Unicode or a non-Unicode system.

requirement - message server

For each application server, which is used by the message server, a section has to exist in `saprfc.ini` with `TYPE R`, where beside the net address of the application server as `GWHOST` also `UNICODE` have to be specified correctly.

Unicode SAP systems only

If Unicode SAP systems are used only (no mixed operation with non-Unicode SAP systems, determined via `UNICODE` in `saprfc.ini`), `sysinit` automatically copies the Unicode version of `sapftp` and `saphttp` from SAP to `sapftp` and `saphttp` and replaces the wrapper program. In this case, you may skip this chapter.

requirement - wrapper program

The `sapftp`/`saphttp` wrapper program from SEAL Systems must be located in the `tools\bin_xxx` directory for mixed operation with Unicode SAP systems and non-Unicode SAP systems. Do not replace this program by the `sapftp`/`saphttp` program from SAP!

This is how you ensure that the program is the wrapper program:

Step	Action
1	Determine the program version with: `sapftp -V` and `saphttp -V`

2	The wrapper program from SEAL Systems displays this result:
	`**********`
	`saphttp Version 1.0.0.3 of 2015-07-06 - $Revision: 1.11`
	`(C) 2013 SEAL Systems`
	`This programm calls the UNICODE/NON-UNICODE version of`
	`depending on command line args and saprfc.ini settings.`
	`----------`
	`Environment:`
	`RFC_INI Path to saprfc.ini.`
	`HTTP_TRACE=2 Write more debug output`
	`RFC_TRACE_DIR Log file directory with`
	`dev_http_seal.log`
	`SAPUNICODE=Y|N Force unicode on/off,`
	`do not parse saprfc.ini`
	`**********`

Unicode SAP System - Transfer with sapftp/saphttp,

Continuation

Step	Action

3	Replace the program if the display looks different, for example:
	`SAPFTP Non-Unicode`
	
	`@(#) $Id: //bas/721_REL/src/krn/ftp/ftpmain.c#7 $ SAP`
	`@(#) $Id: //bas/721_REL/src/krn/ftp/ftp.c#6 $ SAP`
	`@(#) $Id: //bas/721_REL/src/krn/ftp/ftpcmd.c#2 $ SAP`
	
	`inifilename =`
	
	`x¿ÌB, ¹.ini`
	`open Failed`
	`SAP release: 721  SAP release no: 7210`

requirement - DLLs

All required DLLs must exist, for instance under Windows in the directory `tools\bin_xxx` `icuuc.dll`, `librfc32u.dll`, `libsapu16vc.dll`, `libsapucum.dll`, `sapnwrfc.dll`.

effect and process

The value of `UNICODE` defined in `saprfc.ini` is read when starting the RFC destination and the correct variant of `sapftp` and `saphttp` is started.

Level	Processing
1	The `sapftp`/`saphttp` wrapper program from SEAL Systems reads `UNICODE` from `saprfc.ini`.
2	If `UNICODE=1`: The wrapper program starts the `sapftp_uc`/`saphttp_uc` program from SAP.
3	If `UNICODE=0`: The wrapper program starts the `sapftp_nuc`/`saphttp_nuc` program from SAP.

saprfc.ini - RFC Client - Example

example - without SNC

This example shows items in `saprfc.ini` for a RFC client destination, for example JSAPcli or DMS Rlist, without SNC:

```
DEST=W74
TYPE=A
SYSNR=01
ASHOST=roesap005.sealsystems.local
RFC_TRACE=0
ABAP_DEBUG=0
USE_SAPGUI=0
UNICODE=1
SNC_MODE=0
```

SAP router

An SAP router can also be specified with :

```
GWHOST=/H/saprouter.com/H/roegw003.sealsystems.local
```

example - with SNC

This example shows items in `saprfc.ini` for a RFC client destination, for example JSAPcli or DMS Rlist, with SNC:

```
DEST=W74
TYPE=A
SYSNR=01
ASHOST=roesap005.sealsystems.local
RFC_TRACE=0
ABAP_DEBUG=0
USE_SAPGUI=0
UNICODE=1
SNC_MODE=1
SNC_MYNAME=p:CN=SEALRFC,OU=SEALSAP,O=SEAL,C=DE
SNC_QOP=3
SNC_PARTNERNAME=p:CN=roesap005.sealsystems.local, OU=SEALSAP, O=SEAL-
SYSTEMS, C=DE
```

related topics

→ *saprfc.ini - Connection Data*, Page 116

saprfc.ini - RFC Server - Example

example - without SNC

This example shows items in `saprfc.ini` for a RFC server destination without SNC:

```
DEST=W74RFC
TYPE=R
GWHOST=roesap005.sealsystems.local
GWSERV=sapgw01
RFC_TRACE=0
SEAL_TRACE=0
UNICODE=1
SNC_MODE=0
```

SAP router

An SAP router can also be specified with :

```
GWHOST=/H/saprouter.com/H/roegw003.sealsystems.local
```

example - with SNC

This example shows items in `saprfc.ini` for a RFC server destination with SNC:

```
DEST=W74RFC
TYPE=R
GWHOST=roesap005.sealsystems.local
GWSERV=sapgw01
RFC_TRACE=0
SEAL_TRACE=0
UNICODE=1
SNC_MODE=1
SNC_MYNAME=p:CN=SEALRFC,OU=SEALSAP,O=SEAL,C=DE
SNC_QOP=8
```

related topics

→ *saprfc.ini - Connection Data*, Page 116

saprfc.ini - Message Server - Example

example - without SNC

This example shows items in `saprfc.ini` for a connection via message server without SNC:

```
DEST=W74
TYPE=B
MSHOST=roesap005.sealsystems.local
R3NAME=W74
GROUP=PUBLIC
RFC_TRACE=0
ABAP_DEBUG=0
USE_SAPGUI=1
UNICODE=1
SNC_MODE=0
```

SAP router

An SAP router can also be specified with :

```
GWHOST=/H/saprouter.com/H/roegw003.sealsystems.local
```

example - with SNC

This example shows items in `saprfc.ini` for a connection via message server with SNC:

```
DEST=W74
TYPE=B
MSHOST=roesap005.sealsystems.local
R3NAME=W74
GROUP=PUBLIC
RFC_TRACE=0
ABAP_DEBUG=0
USE_SAPGUI=1
UNICODE=1
SNC_MODE=1
SNC_MYNAME=p:CN=SEALRFC,OU=SEALSAP,O=SEAL,C=DE
SNC_QOP=3
SNC_PARTNERNAME=p:CN=roesap005.sealsystems.local, OU=SEALSAP, O=SEAL-
SYSTEMS, C=DE
```

related topics

→ *saprfc.ini - Connection Data*, Page 116

3.2 Establish the RFC Destination on the SAP System

Description

An RFC destination can be used to transfer data from SAP to an external system.

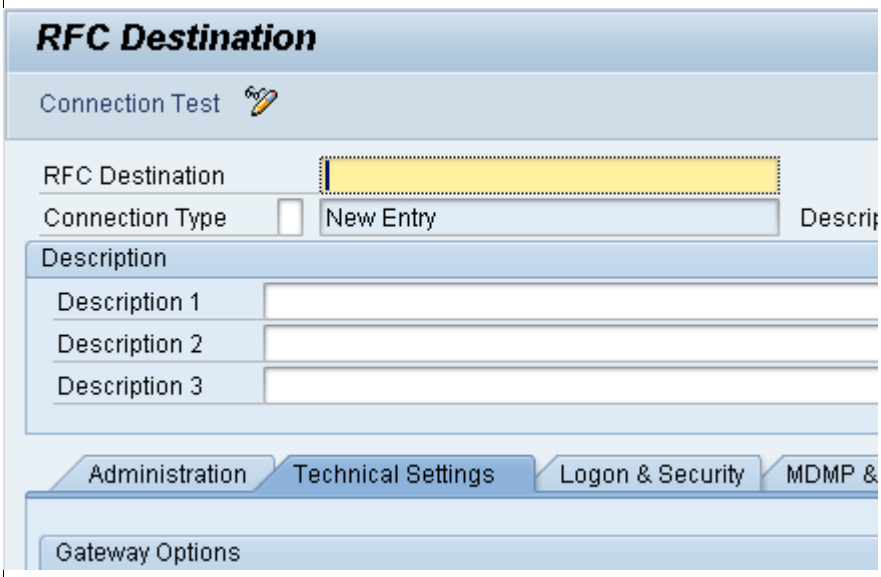
alternative

Alternatively, the following transfer types are available:

→ *HTTP Connection - Configuration*, Page 67

procedure - over-view

This is how you create an RFC destination:

Step	Action
1	Start the `sm59` transaction.
2	Click `Create`: 
3	→ *RFC Destination - Basis Data*, Page 25
4	→ *RFC Destination - Technical Settings*, Page 27
5	→ *RFC Destination - Unicode Settings*, Page 28
6	Save the settings.

RFC Destination - Basis Data

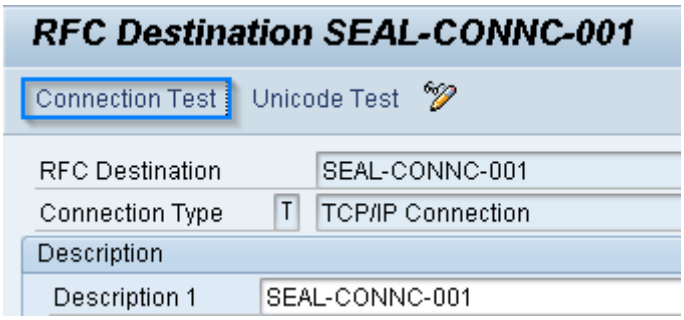
Enter the following basis data:

necessary settings

Parameters	Value
RFC Destination	*Unique identifier on the SAP system* Example: SEAL-CONNC-001 Hint - name identical to the program ID: For reasons of simplicity, the values of RFC destination and Program ID at the technical settings should be identical. Hint - exception: When using Conversion Server, a different name must be specified in order to implement load balancing/ reliability, see scenario 1 in: → *Hints for the Load Balancing*, Page 48 Hint - ambiguous identifier: If the identifier is ambiguous the SAP system establish the connections to the server where the RFC server with this identifier is started first.
Connection Type	T
Description	*Describing text*
Gateway Options - Gateway Host	*GWHOST* as in saprfc.ini Hint - exception: When using Conversion Server, this value may be empty in order to implement load balancing/reliability, see scenario 1 in: → *Hints for the Load Balancing*, Page 48
Gateway Options - Gateway Service	*GWSERV*as in saprfc.ini Hint - exception: When using Conversion Server, this value may be empty in order to implement load balancing/reliability, see scenario 1 in: → *Hints for the Load Balancing*, Page 48

RFC Destination - Basis Data, Continuation

example



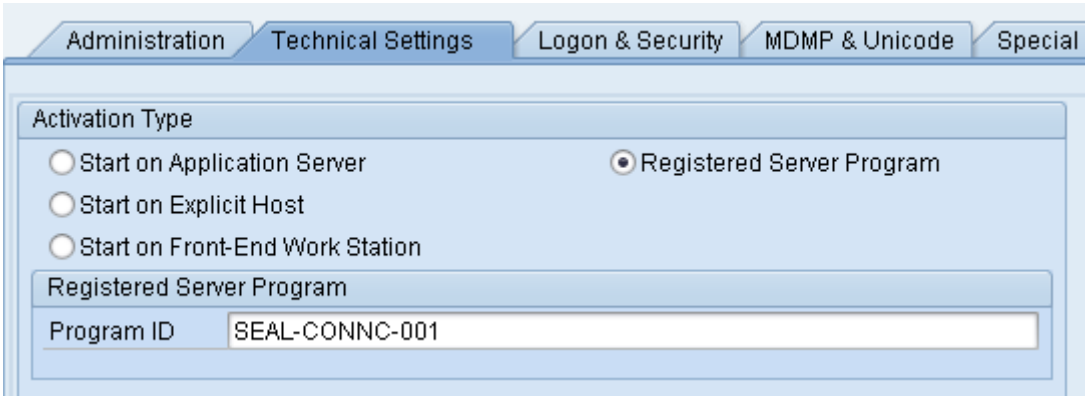
RFC Destination - Technical Settings

Switch to the `Technical settings` tab. requirement

Enter the following technical settings: necessary settings

Parameter	Value
`Activation Type`	`Registered server program`
`Registered Server Program - Program ID`	*`PROGID`* identifier in the configuration file like `rfcserver.cfg` on the SEAL server Example: `SEAL-CONNC-001` Caution - upper/lower cases: The upper/lower case letters are evaluated! Caution - allow RFC destinations: As of EhP7 or kernel 721, destinations are no longer accepted automatically but have to be allowed manually in `reginfo`/`secinfo`: → *Security Configuration for RFC and SAP Destinations*, Page 29 Hint - naming convention: The identifier can include letters, numbers, '+', '.', '-', and '_' characters!

example



RFC Destination - Unicode Settings

requirement

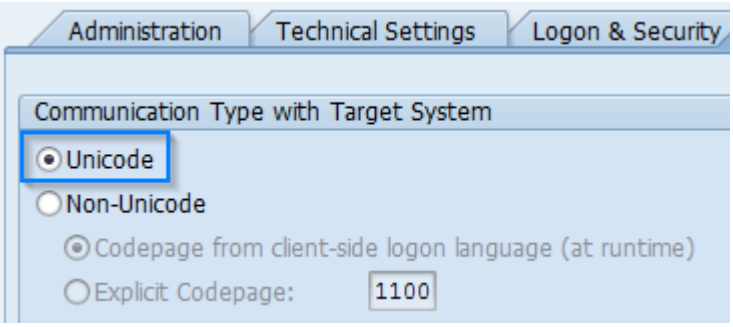
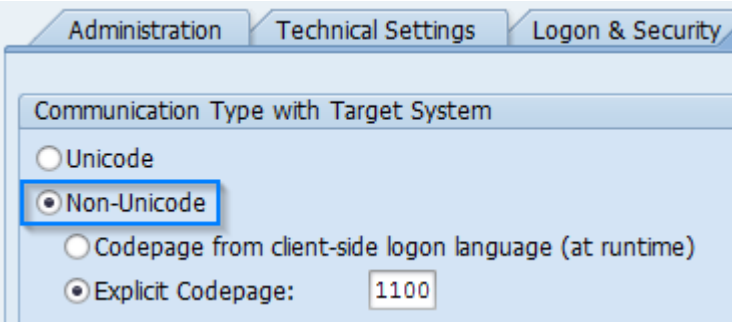
Switch to the tab:

- `Unicode`(as of ECC 6.0)
- `MDMP & Unicode` (up to release 6.40)
- `Special Options` (up to release 4.7)

necessary settings

Enter the following options:

Parameter	Value

as of release 6.40:
`Communication Type with Target System`
 	
up to release 4.7:
`Character Width in Target System`
	• For JRFC Server:
	`Unicode`
	 
	
	• For RFC Server and Conversion Server:
	`Non-Unicode`
	`Explicit Code Page` (as of ECC 6.0)
	 
	
	Caution - Conversion Server:
	The `ConvUtil` and `ConvServSamp` programs on the server are not Unicode-enabled! Therefore, `Non-Unicode` has to be activated!

3.3 Security Configuration for RFC and SAP Destinations

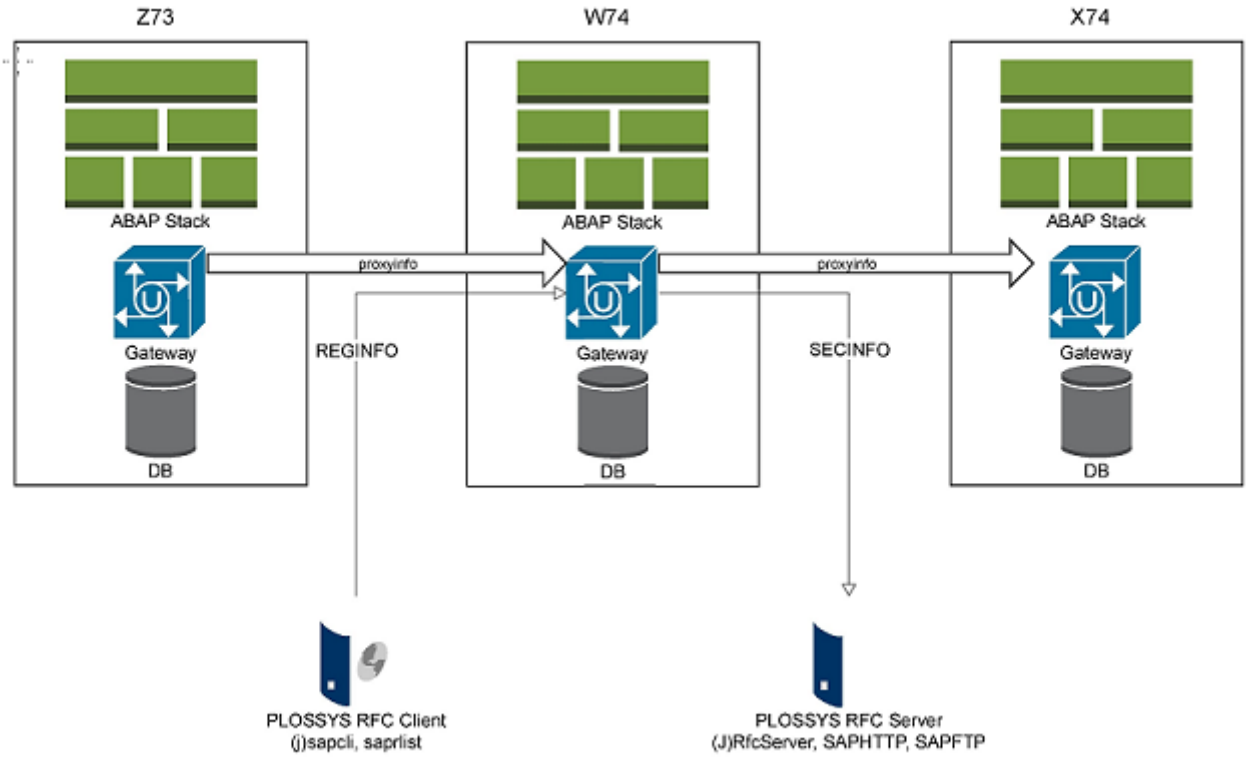
reason

Versions before EhP7 or kernel 721 automatically accept all destinations by default. As of EhP7 or kernel 721, destinations are no longer accepted automatically but have to be allowed manually.

connection types

The security configuration in SAP can be used in order to avoid that unauthorized programs or users log on to SAP or connect to outside. The security configuration distinguishes the following connection types:

- `reginfo` - incoming connections
  (registrations and communication with registered programs)
- `secinfo` - outgoing connections
  (start attempts)
- `proxyinfo` - connections from SAP to SAP
  (not relevant in the context of SEAL Systems)



alternatives

The following alternatives are provided at the security configuration:

- Explicit maintenance of `secinfo`/`reginfo`
  See description below
- Profile parameter `gw/acl_mode` (transaction: `rz11`)
  Evaluated if `secinfo`/`reginfo` are not maintained
  If `gw/acl_mode=0`, all connections are allowed.
  If `gw/acl_mode=1`, all internal connections are allowed.

in this chapter

This chapter deals with the following topics:

Security Configuration for RFC and SAP Destinations, Continuation

→ *Configure Security for Incoming Connections - reginfo*, Page 31
→ *Configure Security for Outgoing Connections - secinfo*, Page 33
→ *Security Configuration - Important Hints*, Page 35
→ *Activate Simulation Mode for Security Configuration*, Page 36
→ *Specify Prefix for Generic RFC Destinations*, Page 37
→ *Security Configuration - Background Knowledge*, Page 38

Configure Security for Incoming Connections - reginfo

This is how you specify the accepted registered programs on the SAP gateway in `reginfo` for SAP NetWeaver Application Server 7.45 or newer:

instructions - >= NW 7.45

Step	Action
1	Start the `smgw` transaction.

2	Select
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Maintain ACL Files`
	`→Reginfo File`

3	Enter the following line: `P TP=* HOST=<IP address of the SEAL server> CANCEL=* ACCESS=*` ⚠ Caution - important hints: Note the following hints when you enter the security configuration: → *Security Configuration - Important Hints*, Page 35

This is how you specify the accepted registered programs on the SAP gateway in `reginfo` for SAP NetWeaver Application Server older than 7.45:

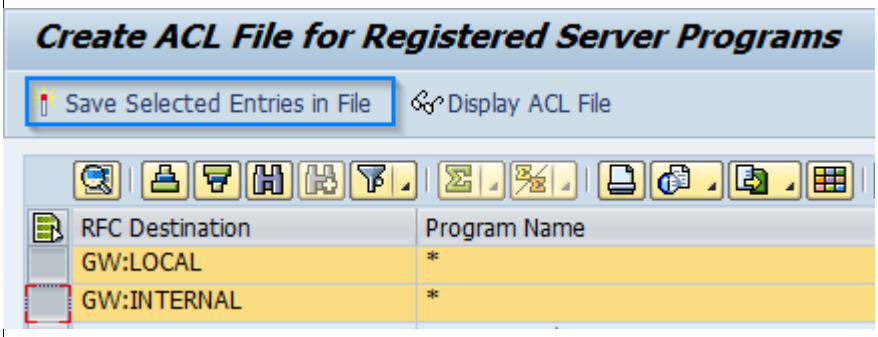
instructions - < NW 7.45

Step	Action
1	Start the `smgw` transaction.

2	Select
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Display (reginfo)`

Configure Security for Incoming Connections - reginfo, Continuation

Step	Action

3	If the file does not exist, you can create it via:
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Create (reginfo)`
	Select the destinations, which are to be accepted, and save them via `Save Selected Entries in File`.
	Example:
	 

4	Enter the following lines in the `usr\sap\`*`system`*`\dveb-mgs00\data\reginfo.dat` file: `P TP=* HOST=<IP address of the SEAL server> CANCEL=* ACCESS=*` Caution - important hints: Note the following hints when you enter the security configuration: → *Security Configuration - Important Hints*, Page 35

Configure Security for Outgoing Connections - secinfo

This is how you specify the accepted programs to be started on the SAP gateway in `secinfo` for SAP NetWeaver Application Server 7.45 or newer:

instructions - >= NW 7.45

Step	Action
1	Start the `smgw` transaction.

2	Select
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Maintain ACL Files`
	`→Sec info`

3	Enter the following line: `P TP=* USER=* USER-HOST=<IP address of the SEAL server> HOST=<IP address of the SEAL server>` ⚠ Caution - important hints: Note the following hints when you enter the security configuration: → *Security Configuration - Important Hints*, Page 35

This is how you specify the accepted programs to be started on the SAP gateway in `secinfo` for SAP NetWeaver Application Server older than 7.45:

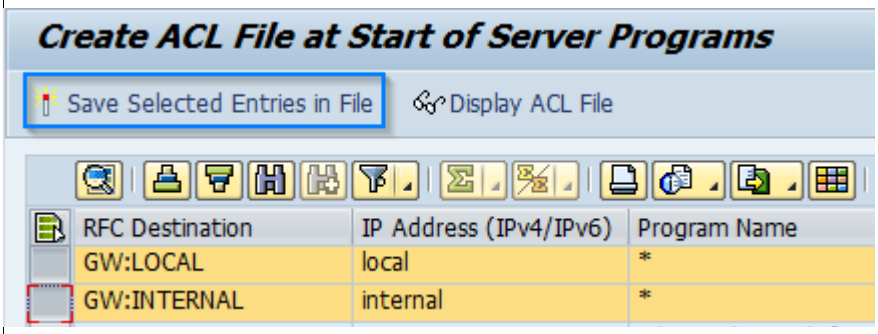
instructions - < NW 7.45

Step	Action
1	Start the `smgw` transaction.

2	Select
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Display (secinfo)`

Configure Security for Outgoing Connections - secinfo, Continuation

Step	Action

3	If the file does not exist, you can create it via:
	`Goto`
	`→Expert Functions`
	`→External Security`
	`→Create (secinfo)`
	Select the destinations, which are to be accepted, and save them via `Save Selected Entries in File`.
	Example:
	 

4	Enter the following lines in the `usr\sap\`*`system`*`\dveb-mgs00\data\secinfo.dat` file: `P TP=* USER=* USER-HOST=<IP address of the SEAL server> HOST=<IP address of the SEAL server>` Caution - important hints: Note the following hints when you enter the security configuration: → *Security Configuration - Important Hints*, Page 35

Security Configuration - Important Hints

note

Note the following issues at the security configuration:

- The first line has to start with: `#VERSION=2`
- Each line has to describe a complete rule, which starts with:
  - `P`: Permit
  - `D`: Deny
- If not all destinations/programs are accepted via `TP=*`, all destinations, which are used by SEAL Systems products, have explicitly to be specified, for instance `ConvUtil.exe` and `ConvServSamp.exe`.
  You also may specify a prefix for generic RFC destinations and only allow RFC destinations with this prefix:
  → *Specify Prefix for Generic RFC Destinations*, Page 37
- The items depend on the order. The first matching rule is used and the reading of the rules is canceled.
  For instance, if a rule is found, which denies SEAL Systems connections, the search is canceled. A rule below, which permits SEAL Systems connections, is not read and is therefore without effect.
- A rule at the end, which denies all connections, is not required, because it is set automatically by the system.
  Exception: If the simulation mode is activated all connections are allowed.
- The rules have to be reread after changed in order to become effective.
- Activate the simulation mode before major changes in order to check the effects of the changes.
  → *Activate Simulation Mode for Security Configuration*, Page 36
  Deactivate the simulation mode as soon as you are sure that the security settings are correct.

Activate Simulation Mode for Security Configuration

purpose

The simulation mode makes the creating of the security configuration easier. It specifies a rule at the end, which allows all connections. These are logged with a specific identifier. The security settings can be modified by means of these items.

The simulation mode is intended for analysis purposes only and does not increase in safety because the registration of non-specified connections is allowed, while a communication via these connections is denied.

instructions

This is how you activate the simulation mode:

Step	Action
1	Start the transaction: `smgw`
2	Select: `Goto` →`Expert Functions` →`Logging`
3	Configure the `Log Events` area according to your wishes and activate in the `Simulation Mode` area: • `On`: Activate Hint - effects: The connection is allowed or denied if a matching rule is found. If there is no explicit rule for the connection the connection is allowed and logged in the gateway log file with `Z` as identifier.
4	Activate the change with `Edit`→`Activate`.
5	Deactivate the simulation mode as soon as you are sure that the security configuration is correct.

alternative

Alternatively, you can set the `gw/sim_mode` profile parameter (transaction: `rz11`):

- If `gw/sim_mode=0` (default) all connections without any explicit rule are denied. The simulation mode is deactivated.
- If `gw/sim_mode=1` all connections without any explicit rule are allowed. The simulation mode is activated.

Specify Prefix for Generic RFC Destinations

description

The server starts generic RFC destinations (`sapftp` and `saphttp`) at the check-out of files from the SAP system. For these, you can specify a prefix, for instance `SEAL`, in order to explicitly allow all destinations with this prefix.

required if

These steps are only required as of EhP7 or kernel 721, if you do not want to allow generally all destinations with `*` in `secinfo`/`reginfo` but you want to restrict this setting as much as possible.

instructions

This is how you specify the prefix for generic RFC destinations on the SAP system:

Step	Action
1	Start the `se16n` transaction with the `sdokprof` table.
2	Enter: • `Key`: `RFC_PREFIX` • `Secondary Key`: Empty or `SAPFTP,SAPHTTP` • `Contents`:`SEAL_` (example) Example: Key | User | Client | Secondary key | Contents RFC_PREFIX | | 800 | | SEAL_ Hint - SAP note: For further information see SAP note `750877`.

Security Configuration - Background Knowledge

tips & tricks

The `smgw` transaction provides following useful functions additionally:

- Gateway monitor: `Goto→Trace→Gateway→Display File`
  Hints regarding all connections
- Clients logged on on SAP: `Goto→Logged on Clients`
  Overview which clients/systems are logged on with which identification

background knowledge - SAP notes

Important SAP notes regarding the security configuration on `https://support.sap.com/notes`:

- General:
  - `1408081` - Basic settings for reg_info and sec_info
  - `1525125` - Update #1 to security note
  - `1105897` - GW: reginfo and secinfo with permit and deny ACL
  - `1425765` - Generating of sec_info reg_info
  - `2061464` - GW: Several smaller corrections in gateway
  - `2090489` - GW: Problems with registrations
- secinfo
  - `614971` - GW: Changes to the ACL list of the gateway (secinfo)
- recinfo
  - `1069911` - GW: Changes to the ACL list of the gateway (reginfo)
  - `1592493` - GW: Problems during the reginfo configuration
- proxyinfo
  - `910918` - GW: Parameter gw/prxy_info
- Gateway simulation mode
  - `1689663` - GW: Simulation mode for reg,sec and prxy_info

background knowledge - SAP WIKI

SAP WIKI items regarding the security configuration on `https://wiki.scn.sap.com/wiki`:

- Gateway Access Control List:
  - `https://wiki.scn.sap.com/wiki/display/SI/Gateway+Access+Control+Lists`
- SAP Network Interface and ACL Control
  - `https://wiki.scn.sap.com/wiki/display/SI/SAP+Network+Interface+and+ACL+control`

3.4 Start and Test RFC Destination

introduction

This chapter explains how the connections for data exchange between the server and the SAP system are established and tested in the case of communication via dynamic RFC connection.

start script

You can use the following alternatives as start script:

- *ModuleGlobal*`start`, for instance `dvsstart`
- `sysstart` *System*, for instance `sysstart SAP`
- *ModuleSelect*`start`, for instance `rfcserverstart`

process at the connection start

The connections between the servers and the SAP systems are established as follows:

Level	Processing
1	→ *Start the RFC Destinations on the Server*, Page 40
2	The start script determines the RFC destinations which are to be started: → *Determine RFC Destinations to be Started via the [ACTIVE] Section*, Page 41 → *Determine RFC Destinations to be Started via Call Parameters*, Page 42
3	The start script determines the SAP system data for the RFC destinations which are to be started in `saprfc.ini` (`DESTINATIONS` in *ModuleSelect*`.cfg` like `rfcserver.cfg` and `DEST` in `saprfc.ini`).
4	The start script starts the RFC destinations to the SAP systems.
5	The SAP systems establish the connection to the SEAL servers via `PROGID` as specified in the `sm59` transaction: → *Test the RFC Destination on the SAP System*, Page 43

in this chapter

This chapter deals with the following topics:

Start the RFC Destinations on the Server

start on the server

You can start the RFC destinations on the server as follows:

- `plsstart` or `sysstart PLOSSYS netdome`
  Starts PLOSSYS netdome and all required RFC destinations
- `dvsstart` or `sysstart SAP`
  Starts all required RFC destinations
- `rfcserverstart`, `jrfcserverstart`, `convservstart`, `convservdpfstart`, `filecheckstart`, `dvsviewserverstart`
  Starts only the required RFC connections from the corresponding configuration file, for example `rfcserver.cfg`

status check on the server

You can check the status of the RFC destinations on the server as follows:

- `plsstatus` or `sysstatus PLOSSYS netdome`
  Displays the status of PLOSSYS netdome and of all required RFC destinations
- `dvsstatus` or `sysstatus SAP`
  Displays the status of all required RFC destinations
- `rfcserverstatus`, `jrfcserverstatus`, `convserverstatus`, `convservdpfstatus`, `filecheckstatus`, `dvsviewserverstatus`
  Shows the status of the required RFC connections from the corresponding configuration file, for example `rfcserver.cfg`, on

problems at start

Sometimes the RFC server can not be started, if a registration already exists on the SAP system. Stop the RFC server and test the connection on the SAP system to delete the registration; after that it should be possible to start the RFC server on the server again.

Determine RFC Destinations to be Started via the [ACTIVE] Section

format

The `[ACTIVE]` section in *`ModuleSelect`*`.cfg` like `rfcserver.cfg` contains the section names with the RFC destinations which are to be started. Each section which is to be considered is assigned to a server name or to the general keyword `SECTION`:

- `SECTION=`*`SectionName`*
  If a section is assigned to the `SECTION` keyword it is considered for all servers.
- *`HostName`*`=`*`SectionName`*
  If a section is assigned to a server it is considered only for this server.

evaluated by

The `[ACTIVE]` section is evaluated by all start scripts.

exception

The `[ACTIVE]` section is ignored on these conditions:

- The `SAP_START_SYSTEM` environment variable is specified.
- Section names are specified as parameters at the program start.

 example 1

The RFC destinations of the `DEV`, `TEST` and `PROD` sections are always to be started independently of the current server:

`rfcserver.cfg`:

```
[ACTIVE]
SECTION=DEV
SECTION=TEST
SECTION=PROD
```

 example 2

The following scenario is to be emulated:

- The RFC destinations of the `DEV` section are always to be started independently of the current server.
- The RFC destinations of the `DEV-EXT` and `PROD` sections are to be started only if `SEALSAP1` is the current server.
- The RFC destinations of the `DEV-EXT`and `TEST` sections are to be started only if `SEALSAP2` is the current server.

`rfcserver.cfg`:

```
[ACTIVE]
SECTION=DEV
SEALSAP1=DEV-EXT
SEALSAP2=DEV-EXT
SEALSAP2=TEST
SEALSAP1=PROD
```

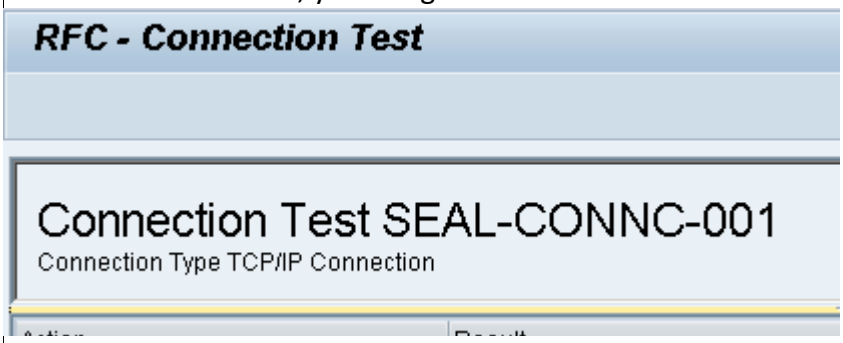
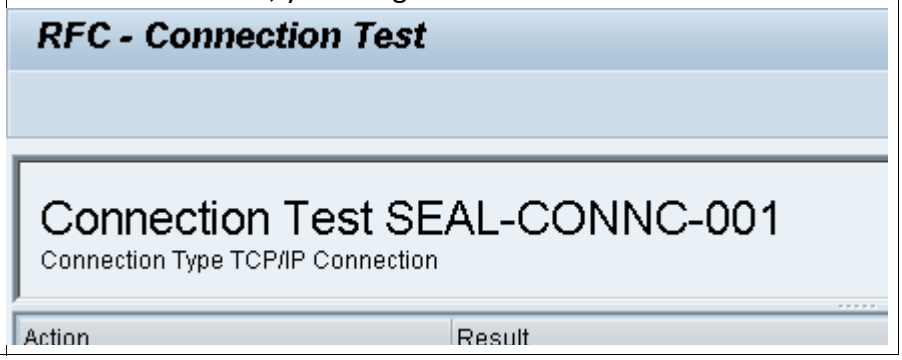
Determine RFC Destinations to be Started via Call Parameters

format	The sections with the RFC destinations which are to be started are specified directly as parameters at the start. Multiple section names can be specified separated by blanks.
evaluated by	The call parameters are only evaluated by the *ModuleSelectAction* scripts, for example `rfcserverstart`. The parameters specified at the start have the highest priority. The `SAP_START_SYSTEM` environment variable and the `[ACTIVE]` section are ignored.
exception	The parameters specified at the start are not evaluated by the *ModuleGlobalAction* scripts, like `dvsstart`, or `sysstart` *System*, like `sysstart SAP`.
example	`rfcserverstart PROD`

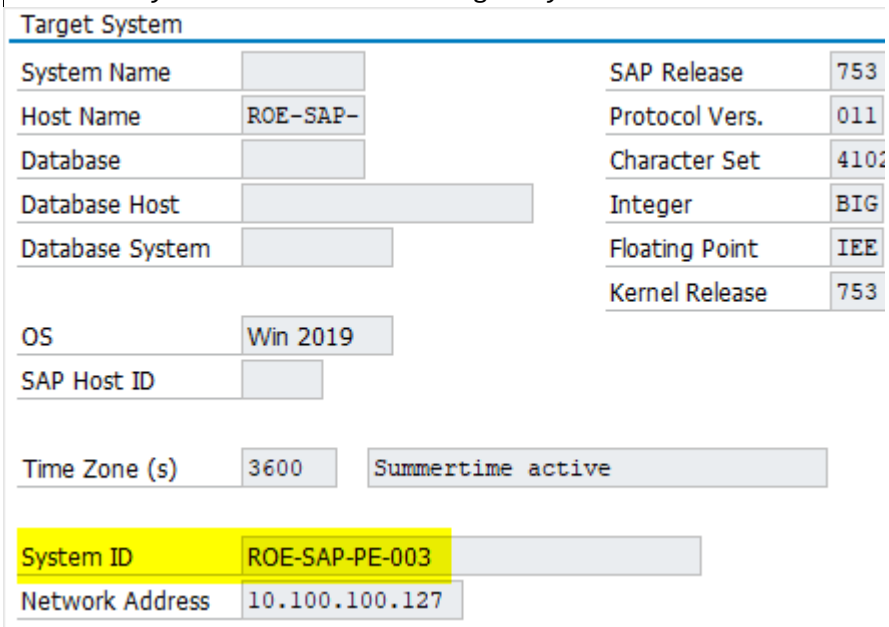
Test the RFC Destination on the SAP System

This is how you test the RFC destination on the SAP system:

instructions - connection test on the SAP system

Step	Action
1	Start the `sm59` transaction.
2	Double-click the RFC destination which you want to test in the list `TCP/IP connections`.
3	Click `Connection Test`. 
4	In the case of success, you will get this result: 
5	In the case of error, you will get this result: 

Test the RFC Destination on the SAP System, Continuation

Step	Action
6	In the case of error, check if the RFC server on the server is actually started and `PROGID` in `rfcserver.cfg` matches the `Program ID` in `sm59`. Hint - connection data: You can check the current system ID of the target system with: `Extras→System Information→Target System` 

3.5 Troubleshooting - RFC Destination

typical problems and their solutions

The following table illustrates typical problems (P) and their approaches (A):

P:	How is the correct connection determined when sapftp/saphttp is started?
L:	→ *saphttp/sapftp Start - Background Knowledge*, Page 47
P:	Establishing several application server - what do you need to know?
L:	→ *Hints for the Load Balancing*, Page 48
P:	Supporting a central SAP gateway - what needs to be considered?
L:	→ *Support Central SAP Gateway*, Page 65
P:	Additional information is required for troubleshooting.
L:	→ *Additional Messages and Debugging*, Page 50
P:	Transfer with sapftp/saphttp is very slow.
L:	→ *Activate DLL Version of sapftp/saphttp in Case of Bottlenecks*, Page 51
P:	The check-out of files returns an error. Possibly, the following error is logged on Unicode systems: `[E] E171 Only available with the RFC library from 4.0C onwards`
L:	→ *Define the USE_GWHOST Parameter*, Page 52
P:	RFC destinations cannot be established. Possible cause is the update to EhP7 or kernel 721, where the destinations are no longer accepted automatically.
L:	→ *Security Configuration for RFC and SAP Destinations*, Page 29 The simulation mode may simplify the troubleshooting: → *Activate Simulation Mode for Security Configuration*, Page 36
P:	RFC destinations - sporadic connection errors or `program ProgamId not registered` error A network component, for example a firewall, closes the TCP/IP connection without informing the external registered RFC server program. The external RFC server program waits endlessly for incoming RFC requests.

Troubleshooting - RFC Destination, Continuation

L:	Activate logging for the SAP gateway to find the cause of the error (`JCO_ERROR_CANCELLED`): → *Activate the SAP Gateway Logging*, Page 53 Activate the RFC trace to find the cause of the error (`JCO_ERR_COMMUNICATION`): → *Activate RFC Trace*, Page 54 Avoid idle time in the network, for example by pings at regular intervals to prevent the TCP-IP connection from being closed, see SAP note 1332022 or 1494757.
P:	Problems with: sapftp/saphttp
L:	Activate logging for sapftp/saphttp: → *Activate Trace Functions for sapftp/saphttp*, Page 55

saphttp/sapftp Start - Background Knowledge

description

When starting saphttp/sapftp, the current source node and the service for the correct connection must be determined, depending on whether an SAP router, an application server or a message server is used.

The following figure provides an overview of this determination.

Hints for the Load Balancing

description

If an SAP system consists of several application servers, for instance in order to distribute the workload (load balancing), one of these servers must be specified as the gateway so that only a single RFC server is started for this connection.

specify

The host name of the application server is specified as `Gateway Host` and the gateway service as `Gateway Service` for the load balancing when creating the RFC destination on the SAP system.

select the appli-cation server

It is advisable to select the application server with the least downtime as `Gateway host`. This should also be the system's database server.

Caution - copy of the SAP system

If the SAP system is copied, for instance for validation purposes, these gateway options must be customized for the new (copied) system to use the RFC destination of the second system!

overview

SEAL Systems products support load balancing/reliability in different ways:

Product	Load Balancing/Reliability
Conversion Server	Scenario 1: Several application servers with one message server which convert independently - even in case of failure of another application server. Solution: • Establish the RFC destination (`sm59` transaction) without any gateway option. • Establish one RFC server connection of type `R` with different names and application server but identical program ID for each application server. Scenario 2: Load Balancing/Reliability Solution: This functionality is provided by the SAP standard conversion. Several RFC destinations (converter numbers) can be specified for each conversion.

Hints for the Load Balancing, Continuation

Product	Load Balancing/Reliability
RFC Server: • `DMS View Server` • `Core Convert`	The functionality described as scenario 2 is supported by Conversion Server. Alternative when using JRFC Server: You may assign several JRFC Server via the same RFC destination.  Hint - restriction: The JRFC Server which will execute the function is selected by random. The execution of the individual functions must therefore be independently. This is only valid for parts of the BC XDC Interface Implementation (assignment and status query can be executed on different servers).
RFC client	RFC clients support the SAP load balancing via message servers (type `B`).
CD Installations	CD installations from SEAL Systems (type `A`) do not support the automatic load balancing. The configuration files, for instance `saprfc.ini` and `rfcserver.cfg`, must be modified and saved manually.

Additional Messages and Debugging

description	You can activate additional files with trace messages, if the RFC destinations cause difficulties. The debugging of system functions can be activated for RFC client destinations alternatively.
activate trace messages	You activate the trace messages with `RFC_TRACE` in `saprfc.ini`. Alternatively, you can also activate the RFC trace or SAP Gateway logging within the SAP system: → *Activate the SAP Gateway Logging*, Page 53 → *Activate RFC Trace*, Page 54
debugging of system functions	You activate the debugging of system functions with `ABAP_DEBUG` in `saprfc.ini`.

Activate DLL Version of sapftp/saphttp in Case of Bottlenecks

required if

These steps are only required if you want to use the DLL variant of the `sapftp` and `saphttp` programs due to performance reasons.

requirements

The following requirement must be fulfilled:

- The `sapftp.dll` and `saphttp.dll` files must be located in the `tools\bin_winxx` directory.
- In the SAP system, the `/seal/bas_dm_be_checkoutviewx` function must exist.

instructions

This is how you activate the DLL variant of the `sapftp` and `saphttp` programs:

Step	Action
1	Edit the following file: `plossys.ini`

2	Enter:
	`[rlistsap]`
	`CAD_CHECKOUT_DLL = Y`

3	Enter:
	`[rlistsap]`
	`BAPI_DOCUMENT_CHECKOUTVIEWX = /SEAL/BAS_DM_BE_CHECKOUTVIEWX`
	`ARC_CHECKOUT_RETRY_ERROR=Y`

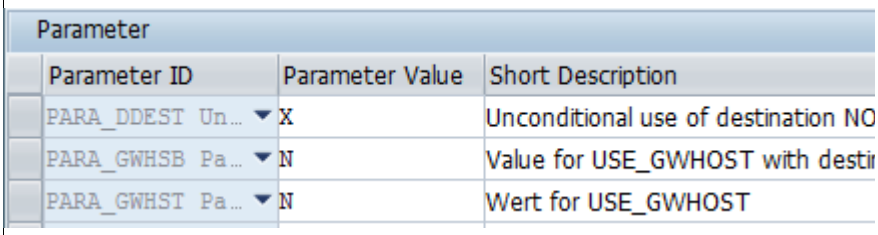
Define the USE_GWHOST Parameter

description

The check-out of files from the SAP systems returns an error. One of the reasons could be that a wrong host name is used at the start of `sapftp` and `saphttp` - required for the check-out - when using several application servers. The determination of the host name can be influenced by the `USE_GWHOST` parameter.

instructions

This is how you specify the value for the `USE_GWHOST` parameter:

Step	Action
1	Start the `/seal/img` transaction.
2	Click  at `Basis Configuration` →`Define Parameters` (table: `/seal/bas_cr142`)
3	Specify the value of the `USE_GWHOST` parameter with: • RFC server destinations: `PARA_GWHST`: `Y` or `N` • RFC client destinations: `PARA_GWHSB`: `Y` or `N`  Example:  Default is for both parameters: `N`

Parameter

Parameter ID	Parameter Value	Short Description
PARA_DDEST Un…	X	Unconditional use of destination NO
PARA_GWHSB Pa…	N	Value for USE_GWHOST with desti
PARA_GWHST Pa…	N	Wert for USE_GWHOST

background knowledge

When using sapftp/saphttp, the host name is passed via the `GWHOST` parameter and transferred to `librfc23`, for instance:

`sapftp -x`*`HostName`*

The *`HostName`* is previously determined via the `Z_SYS_INFO` function for RFC server destinations and via the `SYSTEM_GET_CURRENT_GATEWAY` function for RFC client destinations.

`librfc32` uses the transferred value or determine the host name again for itself. This depends on the value of the `USE_GWHOST` parameter. With `USE_GWHOST Y` the transferred value is used, with `USE_GWHOST N` the host name is determined again.

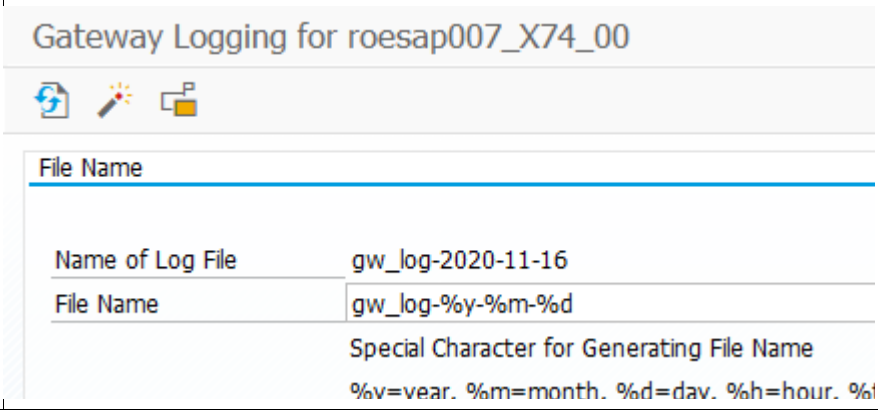
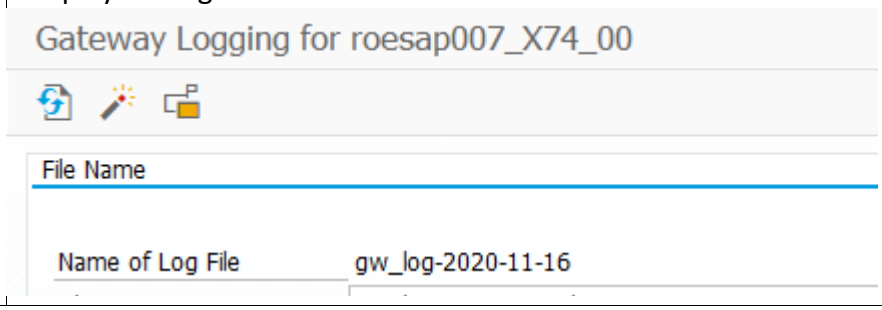
Activate the SAP Gateway Logging

description

If there are problems with RFC destinations, you can activate the RFC trace to search for the cause of the error in the generated trace files.

instructions

This is how you activate the SAP gateway logging:

Step	Action
1	Start the transaction: `smgw`
2	Select: `Goto` →`Expert Functions` →`Logging`
3	Activate the following options in the `Log Events` area: • `Network`: Activate • `Security - (denied accesses without rules)`: Activate • `Registered Programs`: Activate 🔍 Example: 
4	Display the log to search for the cause of the error: 

Activate RFC Trace

description

If there are problems with RFC destinations, you can activate the RFC trace to search for the cause of the error in the generated trace files.

alternative

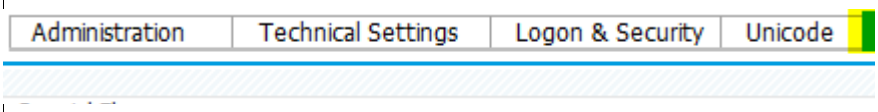
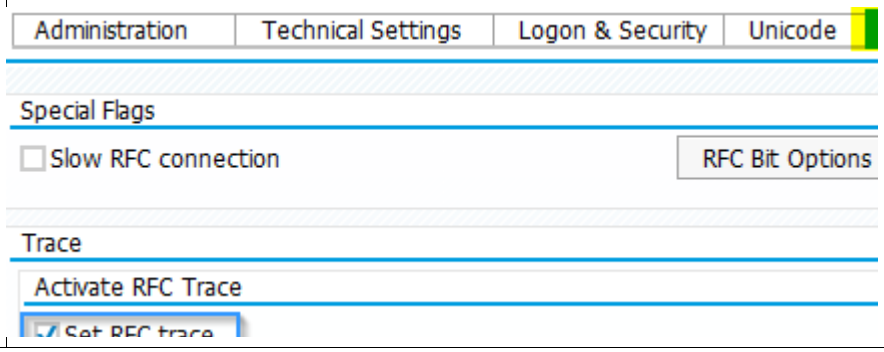
Alternatively, you can also activate the RFC trace on the server side:

→ *RFC_TRACE Parameter*, Page 133

→ *SEAL_TRACE Parameter*, Page 134

instructions

This is how you activate the RFC trace:

Step	Action
1	Start the transaction: `sm59`
2	Switch to the tab: `Special Options` 
3	Activate the following option in the `Trace` area: • `Set RFC Trace`: Activate Hint - effects: This automatically activates the `Export Trace` option in the `Trace Export Methods` section and overrides the `gw/export_trace_level` profile parameter. See also SAP note 558254. Example: 

result

The following trace files are created:

- `dev_jco_rfc.log`
- `jco_rfc`*`PID_UUID`*`.trc`

deactivate trace

After successful cause analysis, deactivate the option `Set RFC Trace` again.

Activate Trace Functions for sapftp/saphttp

description

You can use environment variables to activate trace functions for sapftp/saphttp.

When sapftp/saphttp is started by JRFC Server, `SEAL_TRACE=1` (or higher) is available in `saprfc.ini` as an alternative.

When sapftp/saphttp is started by JSAPcli, the `-ft` parameter or the `SAPCLI_FULL_TRACE=1` environment variable is available as an alternative.

instructions

This is how you activate the trace functions for sapftp/saphttp if they are not started by JRFC Server or JSAPcli:

Step	Action
1	Set following environment variable: • For sapftp: `FTP_TRACE=2` • For saphttp: `HTTP_TRACE=2`
2	Activate the changed environment variables to make them available in the context of the SEALServiceuser: • `sitwelcome` • `sysstop` • Close SEAL shell • Open SEAL shell • sysstart Restart at least all processes, such as `rlist`, whose start is to be analysed by sapftp/saphttp.

4 Static RFC Destinations for SNC Support

introduction

This chapter explains how the connections for the data exchange between servers and SAP systems are established and tested in case of communications via static RFC destinations. Static RFC destinations offer the following advantages:

- SNC is supported for sapftp/saphttp
- Static RFC destinations can be explicitly enabled/restricted at the gateway.

Static RFC destinations are available for:

- RFC client:
  JSAPcli
- RFC server:
  DPF
  DMS Loader/ABAP
  XSA
  (DMS View Server is not affected))

in this chapter

This chapter deals with the following topics:

Establish Templates for sapftp/saphttp (sm59)

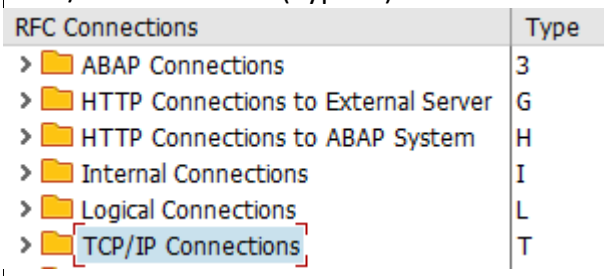
description

The following templates must be established manually for the static sapftp/saphttp connections:

- For RFC client (for example, for use by JSAPcli):
  - `SEAL-CLT-SAPFTP`
  - `SEAL-CLT-SAPHTTP`
- For RFC server (for example, for use by DMS View Server):
  - `SEAL-SRV-SAPFTP`

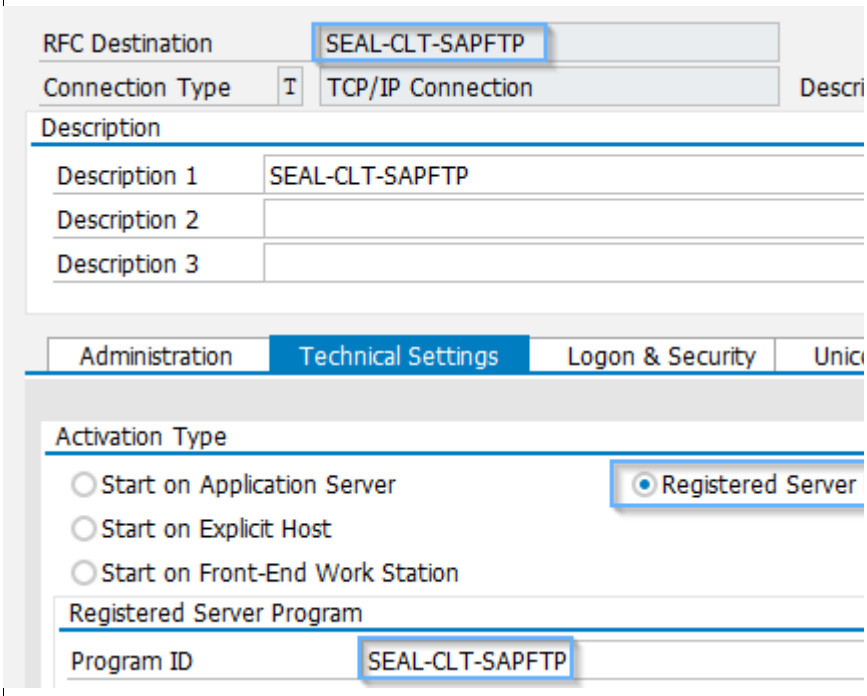
instructions

This is how you establish the necessary templates for the static sapftp/saphttp connections:

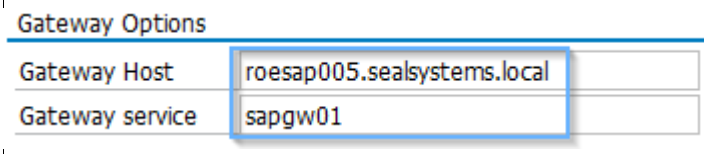
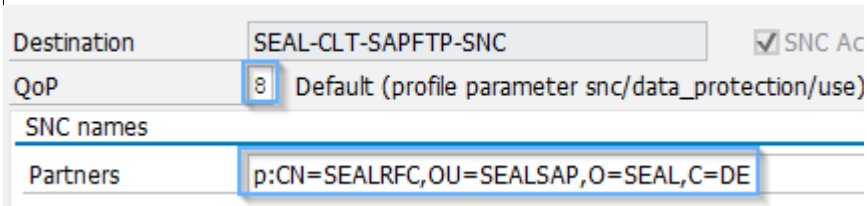
Step	Action
1	Start the transaction: `/n/seal/img`
2	Click  at `Basis Configuration` →`Static Destinations` →`Establish RFC Destinations`  Hint - alternative: Start the `sm59` transaction.
3	Select the connection type on the left: TCP/IP Connections (Type: `T`) 

Establish Templates for sapftp/saphttp (sm59), Continuation

Step	Action

4	Create three items with the following settings in the `Technical Settings` tab:
	• For RFC client - SAPFTP:
	  • `RFC Destination`: `SEAL-CLT-SAPFTP`
	  • `Registered Server Program`: Activate
	  • Program ID: `SEAL-CLT-SAPFTP`
	• For RFC client - SAPHTTP:
	  • `RFC Destination`: `SEAL-CLT-SAPHTTP`
	  • `Registered Server Program`: Activate
	  • Program ID: `SEAL-CLT-SAPFHTP`
	• For RFC server - SAPFTP:
	  • `RFC Destination`: `SEAL-SRV-SAPFTP`
	  • `Registered Server Program`: Activate
	  • Program ID: `SEAL-SRV-SAPFTP`
	Example - RFC client - SAPFTP:
	 
	Hint - copy existing RFC destination:
	Generally, you already have an RFC destination in use. You can copy this and enter the settings mentioned above:
	`Connection→Copy (Ctrl+F12)`

Establish Templates for sapftp/saphttp (sm59), Continuation

Step	Action
5	On the `Technical Settings` tab, enter the gateway options: • `Gateway Host`: *`GWHOST`* from `saprfc.ini` • `Gateway Service`: *`GWSERV`* from `saprfc.ini` 🔍 Example:  Background info: JSAPcli uses this information for static sapftp/saphttp destinations.
6	For SNC destinations, enter in the `Logon & Security` tab under `Security Options`: • SNC: SNC parameter, enter • `Active`: Activate 🔍 Example - SNC parameter:  ⚠ Caution - `saprfc.ini`: The SNC settings in `saprfc.ini` must also be activated on the server: • `SNC_MODE=1` • `SNC_MYNAME=p:CN=SEALRFC,OU=SEALSAP,O=SEAL,C=DE` • `SNC_QOP=8` The system ID used is defined via `rfcserver.cfg` (RFC server) or `cadrfc.ini` (RFC client).

Create Multiple sapftp/saphttp Destinations

description

If you use multiple RFC servers, for example for DPF, DMS View Server, which are to use static sapftp destinations, you must create a corresponding number of sapftp destinations.

For the RFC client, it makes sense to create several static sapftp/saphttp destinations for parallel mode, for example multiple JSAPcli calls at the same time.

instructions

This is how you create multiple sapftp/saphttp destinations based on the created templates:

Step	Action
1	Start the transaction: `/n/seal/img`
2	Click  at `Basis Configuration` →`Static Destinations` →`Copy Template for Static sapftp/saphttp`

Create Multiple sapftp/saphttp Destinations, Continuation

Step	Action
3	Enter the following values for RFC client and then create the destinations with : • RFC client: • `Template SAPFTP`: `SEAL-CLT-SAPFTP` Template for RFC client - SAPFTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57 • `Template SAPHTTP`: `SEAL-CLT-SAPHTTP` Template for RFC client - SAPHTTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57 • `Number`: Desired number of static destinations Example: **Create SAPFTP and SAPHTTP Destinations** Template SAPFTP: SEAL-CLT-SAPFTP Template SAPHTTP: SEAL-CLT-SAPHTTP Number: 3 • RFC server: • `Template SAPFTP`: `SEAL-SRV-SAPFTP` Template for RFC server - SAPFTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57 • `Template SAPHTTP`: Empty • `Number`: Desired number of static destinations Example: **Create SAPFTP and SAPHTTP Destinations** Template SAPFTP: SEAL-SRV-SAPFTP Template SAPHTTP: Number: 4 Background knowledge - no SAPHTTP required: RFC servers have an integrated saphttp destination so that none needs to be created.

When copying from the source destination to the target destination, the following values are set: background knowledge

Create Multiple sapftp/saphttp Destinations, Continuation

- SNC parameter are passed
- The target destination is always configured as an Unicode SAP system
- Whitelist, timeout parameters and others are not passed, which is why the target destination must be checked again afterwards

RFC Client - Assign Static sapftp/saphttp Destination

description

As soon as you perform the configuration described here, static instead of dynamic RFC destinations are started for the specified user - or for all users if the user is empty.

Dynamic RFC destinations are only used if no item exists for the user.

instructions

This is how you assign the templates for the RFC client:

Step	Action
1	Start the transaction: `/n/seal/img`
2	Click  at `Basis Configuration` →`Static Destinations` →`Create Static Destination for RFC Client` (table: /`seal/bas_cr114`)
3	Enter: • `User Name`: User with which the RFC client logs on to SAP from `cadrfc.ini`, or empty (default) • `SAPHTTP`: `SEAL-CLT-SAPHTTP` Template for RFC client - SAPHTTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57 • `SAPFTP`: `SEAL-CLT-SAPFTP` Template for RFC client - SAPFTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57 • `Number`: Number of possible sapftp/saphttp destinations including the template You can create this number of destinations with: → *Create Multiple sapftp/saphttp Destinations*, Page 60  Example: Static destinations for RFC Client User Name | SAPHTTP | SAPFTP | Number SEAL-CLT-SAPHTTP | SEAL-CLT-SAPFTP | 3

RFC Server - Assign Static sapftp/saphttp Destination

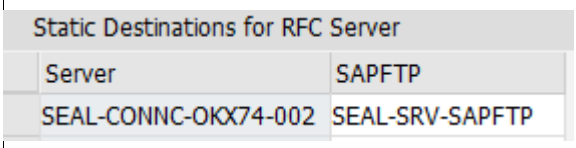
description

As soon as you perform the configuration described here, static instead of dynamic RFC destinations are started for the specified RFC server - or for all RFC servers if RFC server is empty.

Dynamic RFC destinations are only used if no item exists for the RFC server.

instructions

This is how you assign the templates for the RFC server:

Step	Action
1	Start the transaction: `/n/seal/img`
2	Click  at `Basis Configuration` →`Static Destinations` →`Create Static Destination for RFC Server` (table: /`seal/bas_cr113`)
3	Enter: • `Server`: RFC destination, for example from `rfcserver.cfg`, which should use a static sapftp destination Depending on how many servers are to use static sapftp destinations, you must create this number of destinations with: • `SAPFTP`: `SEAL-SRV-SAPFTP` Template for RFC server - SAPFTP from → *Establish Templates for sapftp/saphttp (sm59)*, Page 57  Example: 

Support Central SAP Gateway

The destination via a central SAP gateway is only supported together with static RFC destinations. requirement

The following restrictions apply to the destination via a central SAP gateway: restriction

- SNC is only supported on request.
- The connection via SAProuter is only supported on request.

This is how you support a central SAP gateway: instructions

Step	Description
1	On the server, open: `saprfc.ini`

2	Only for RFC client destinations, such as JSAPcli
	In the section for the message server (`B` type), also enter the name of the section containing the data of the central SAP gateway with:
	`ASCS=<section of the central SAP gateway of type R>`
	Example:
	`DEST=W74`
	`TYPE=B`
	`ASCS=W74RFC`

3	Enter the data of the central SAP gateway in the section of type `R` that you specified in step 2 for `ASCS`.
	Example:
	`DEST=W74RFC`
	`TYPE=R`
	`GWHOST=roesap005.sealsystems.local`
	`GWSERV=sapgw01`
	`RFC_TRACE=0`
	`SEAL_TRACE=0`
	`UNICODE=1`
	Hint - RFC client/RFC server:
	For RFC client destinations, both sections are required.
	For RFC server destinations, the section with type `R` contains the data of the central SAP gateway. No further section is required.

This example shows items in `saprfc.ini` for a connection via a central SAP gateway: example

Support Central SAP Gateway, Continuation

```
DEST=W74
TYPE=B
MSHOST=roesap005.sealsystems.local
R3NAME=W74
GROUP=PUBLIC
ASCS=W74RFC
RFC_TRACE=0
ABAP_DEBUG=0
USE_SAPGUI=1
# Enter data of the central SAP gateway as type R
DEST=W74RFC
TYPE=R
GWHOST=roesap005.sealsystems.local
GWSERV=sapgw01
RFC_TRACE=0
SEAL_TRACE=0
UNICODE=1
```

note

For RFC client destinations, both sections are required.

For RFC server destinations, the section with type `R` contains the data of the central SAP gateway. No further section is required.

5 HTTP Connection - Configuration

introduction

This chapter describes the configuration settings required for communication via HTTP connections, which are used in combination with a REST interface, for example.

alternative

Alternatively, the following transfer types are available:

→ *RFC Destination - Configuration*, Page 17

in this chapter

This chapter deals with the following topics:

→ *HTTP Connection - Configuration on Server*, Page 68

→ *Create an HTTP Connection on the SAP System*, Page 72

5.1 HTTP Connection - Configuration on Server

in this chapter

This chapter deals with the following topics:

Create PSE

The certificates have to be provided in the X.509 Base64 format. requirement

This is how you create a PSE for HTTPS on the external server, using Windows as an example: instructions

Step	Action
1	Logon on the server as user who start the processes which use `saphttp`, for instance the SEALService user.
2	Switch to the `SECUDIR` directory.
3	Create the SAPSSLC PSE: • `sapgenpse.exe gen_pse -noreq -p SAPSSLC.pse` ⚠ Caution - PIN: A PIN has to be entered for the creation. Remember the PIN for further configuration. Hint - PSE user: Note the format at ‚distinguished name of PSE owner', for instance: `CN=SEAL, OU=SEALSAP, O=SEAL, C=DE` `CN` is mandatory, all other parameters could be empty.
4	Integrate the root certificate and the intermediate certificate from Web server into the SAPSSLC PSE, for example: • `sapgenpse.exe maintain_pk -p SAPSSLC.pse -m`*`RootCertificate`*`.crt` • `sapgenpse.exe maintain_pk -p SAPSSLC.pse -m`*`Webserver-Certificate`*`.crt`
5	Copy the SAPSSLC PSE if `SAPSSLS.pse` does not exist, with: `cp SAPSSLC.pse SAPSSLS.pse`

6	Create the `cred_v2` file for the SEALService user via the execution of `sapgenpse.exe`, with the PIN from step 3, for instance:
	`sapgenpse.exe seclogin -O SEALServiceUser -p SAPSSLC.pse -x PIN`
	`sapgenpse.exe seclogin -O SEALServiceUser -p SAPSSLS.pse -x PIN`

7	Check the imported certificates:
	`sapgenpse.exe maintain_pk -p SAPSSLC.pse -l`
	`sapgenpse.exe maintain_pk -p SAPSSLS.pse -l`

Activate Logon with Basic Authentication and SSL

description

The recommended logon method is to logon with user and basic authentication.

instructions

This is how you activate basic authentication and SSL as logon method on the external server:

Step	Action
1	Open a SEAL shell and start the command: `sysinit dpf` Enter: • `Do you want to configure DPF: Y` • `DPF Web authentication type: basic` • `User name for DPF Web basic authentication [wsuer]:` RETURN • `New password:` Enter new password for `wsuser` and confirm again • `Do you want to configure fastlogin: Y`

2	Enter:
	`sysinit -auto`
	Result:
	As a result, the following items exist:
	• File: `applications\server\web\apache\conf\extra\httpd-ssl.conf`
	`JkMount /rest/* seal-worker`
	`JkMount /rest seal-worker`
	• File: `applications\server\web\apache\conf\httpd.conf`
	`Include d:/SEAL/customer/server/web/conf/auth-basic-dpf.conf` (example)

3	Enter in the include file of the last step: • File: `customer\server\web\conf\auth-basic-dpf.conf`: Add to `LocationMatch`: `|rest/` Example: `<LocationMatch "/(dpf4c-sercive-v1.3/|pdflls-service-v1.0/|cgi-bin/testAuth/|/rest/dpf/v1/|/rest|rest)">`
4	Enter: • sysstop -full web

Activate Logon with Basic Authentication and SSL,

Continuation

Step	Action
5	Enter: • `Do you want to configure Apache Web Server: Y` • `Should the web server support the secure HTTPS protocol: Y` • `Enter the SSL certification file [conf/sealsystems-ca]:` Basis name of the certificate without path and without file extension • `Should access only be allowed via HTTPS: Y` • `Do you want to configure fastlogin: Y`
6	Enter: • `sysstart web`

5.2 Create an HTTP Connection on the SAP System

description

An HTTP connection can be used to transfer data from SAP to an external system, for example in combination with a REST interface.

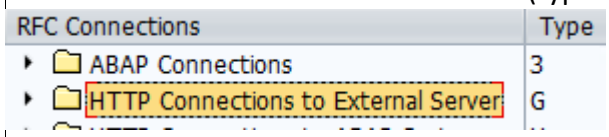
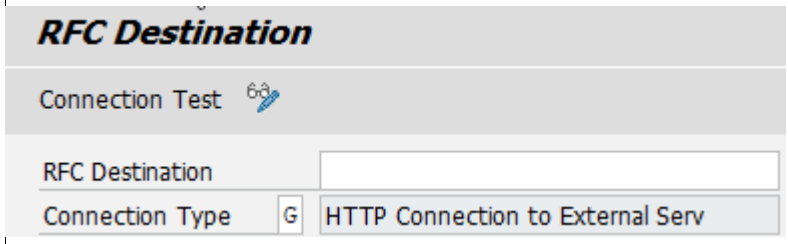
alternative

Alternatively, the following transfer types are available:

→ *Establish the RFC Destination on the SAP System*, Page 24

procedure - over-view

This is how you create the HTTP connection required for the REST interface:

Step	Action
1	Start the `sm59` transaction.
2	Select the connection type on the left: `HTTP Connection to External Server` (type: `G`) 
3	Click `Create`: 
4	→ *HTTP Connection - Basis Data*, Page 73
5	→ *HTTP Connection - Technical Settings*, Page 74
6	→ *HTTP Connection - Logon & Security*, Page 75
7	→ *HTTP Connection - Special Options*, Page 77
8	Save the settings.

9	Activate the service required for HTTP transfer if it is deactivated:
	• Transaction:
	`sicf`
	• Start the display via with:
	`Hierarchy Type: SERVICE`
	• Open tree display on the left:
	`default_host - sap - bc`
	• In the context menu check if service is active for:
	`rest`

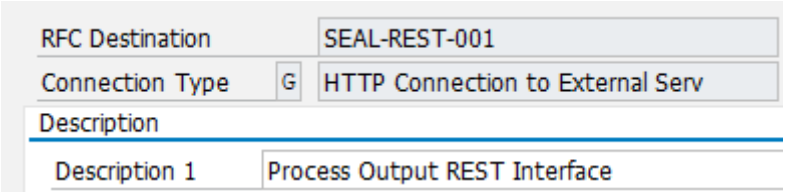
HTTP Connection - Basis Data

Enter the following settings:

necessary settings

Parameters	Value
RFC Destination	*Unique identifier on the SAP system* Example: SEAL-REST-001
Connection Type	G
Description	*Describing text*

example



HTTP Connection - Technical Settings

requirement

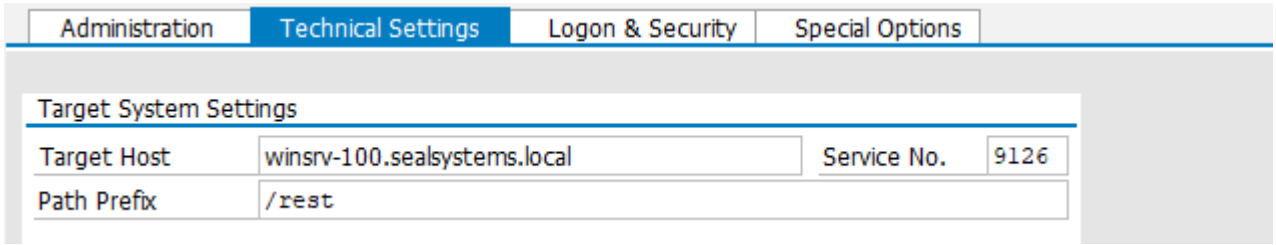
Switch to the `Technical settings` tab.

necessary settings

Enter the following settings:

Parameter	Value
`Target Host`	IP addresses of the host with web service, for example: `10.100.53.47`
`Service Number`	`9125` (HTTP), `9126` (HTTPS)
`Path prefix`	`/rest`

example



HTTP Connection - Logon & Security

Switch to the `Logon & Security` tab.

requirement

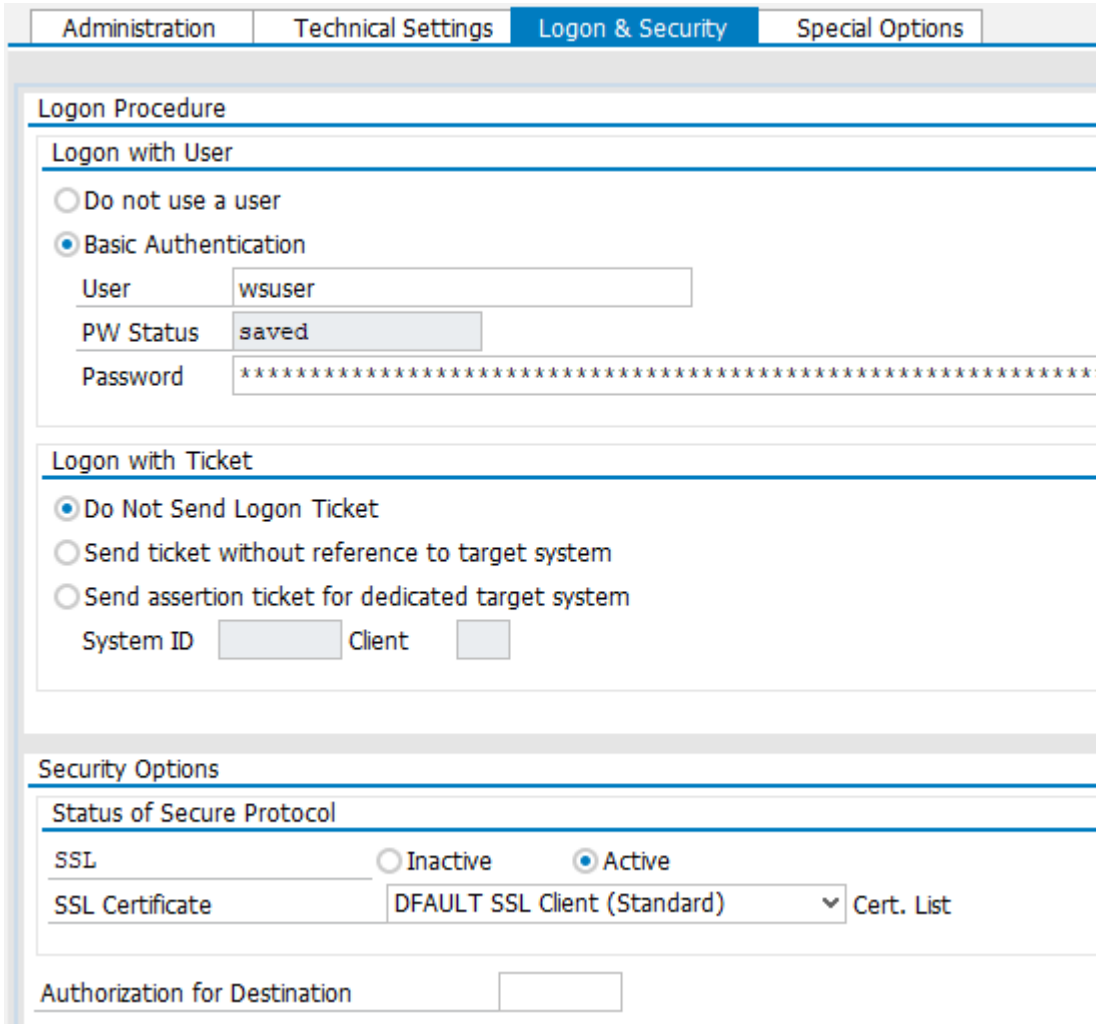
Enter the following settings:

necessary settings

Parameter	Value
`Logon with User`	• Basic Authentication: Activate • user: user for logon • password: password for logon Hint - requirement: → *Activate Logon with Basic Authentication and SSL*, Page 70
`Logon with Ticket`	• `Do Not Send Logon Ticket`: Activate
`Security Options`	• `SSL`: Activate • `SSL Certificate`: `DFAULT SSL Client (Standard)`

example

HTTP Connection - Logon & Security, Continuation



HTTP Connection - Special Options

Switch to the `Special Options` tab. — requirement

Enter the following settings: — necessary settings

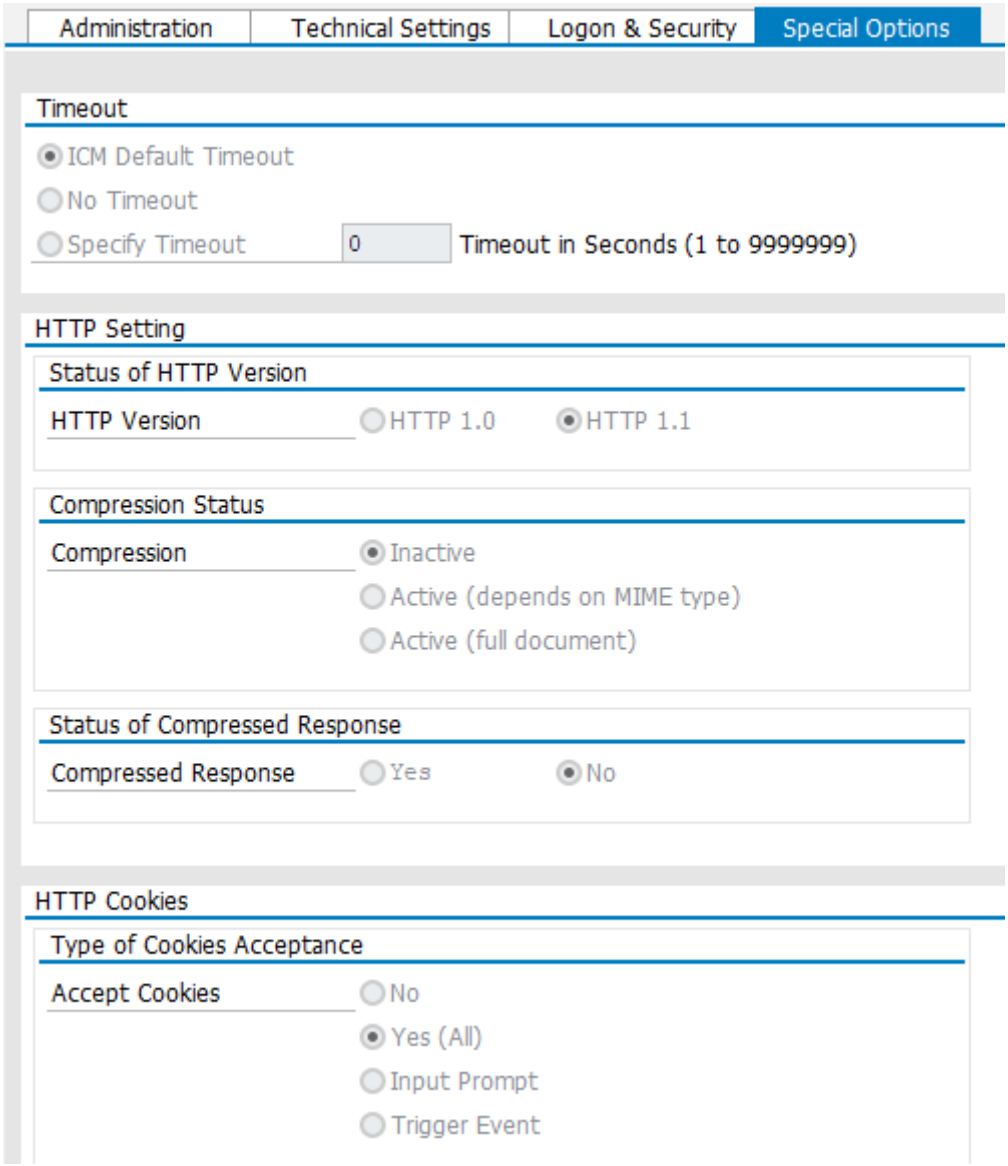
Parameter	Value
`Timeout`	• `ICM Default Timeout`: Activate Maximum response time for the connection when sending an HTTP request.
`HTTP Setting` `Status of HTTP Version`	• `HTTP 1.1`: Activate Protocol version of HTTP requests

`HTTP Setting`
`Compression Status`
	• `Compression`
	  • `Inactive`: Activate
	• `Status of Compressed Response`
	  • `No`: Activate

`HTTP Cookies`	• `Yes (All)`: Activate Handling of received cookies:

HTTP Connection - Special Options, Continuation

example



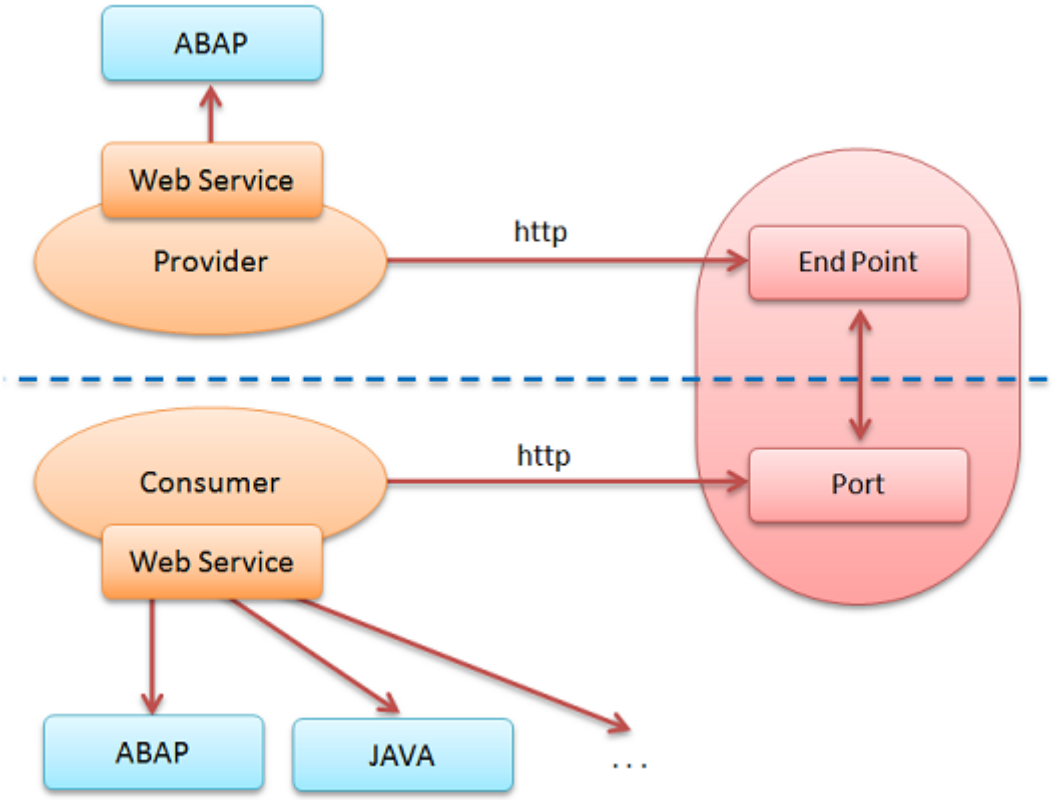
6 Integration via Web Service - SAP as Provider

This chapter describes the establishing of an integration via Web service, if SAP serves as provider. introduction

Functions that are implemented in different SAP software components may be combined via Web service to one process. This chapter describes the steps required to use the integration via Web service for SEAL Systems products. description

→ *Integration via Web Service - SAP as Consumer*, Page 89 related topics

SAP as provider: overview



This chapter deals with the following topics: in this chapter

6.1 Required Authorizations

requirement - roles

The required authorizations have to be assigned to the users via the following roles (transaction: `su01`; maintenance of the authorizations via the profile generator with the `pfcg` transaction):

- User, who administrates the Web service via the SOA manager:
  `SAP_BC_WEBSERVICE_ADMIN_TEC`
- SAP as provider - user, who is used as ABAP service user for the endpoint on the provider system:
  `SAP_BC_WEBSERVICE_SERVICE_USER`
- SAP as provider - user, who want to debug the Web service requests:
  `SAP_BC_WEBSERVICE_DEBUGGER`
- SAP as consumer - user, who starts the Web service:
  `SAP_BC_WEBSERVICE_CONSUMER`

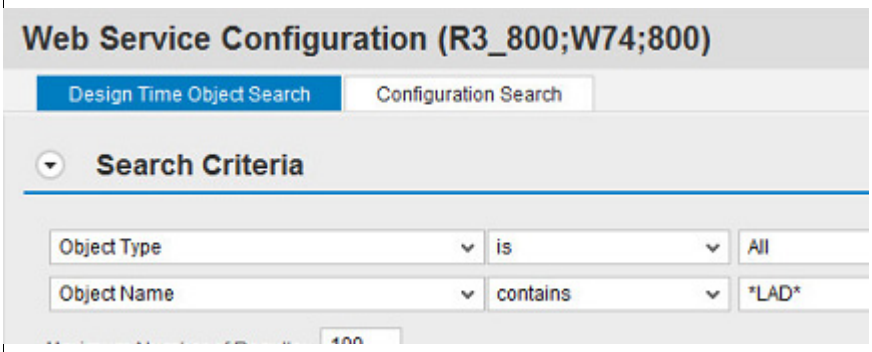
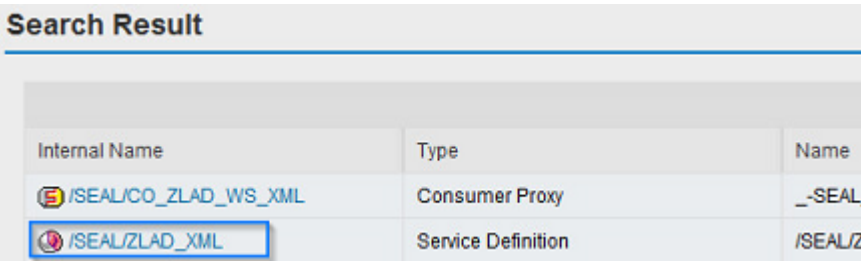
6.2 Configuration on the Provider System (SAP)

This chapter deals with the following topics: in this chapter

Select the Service

instructions

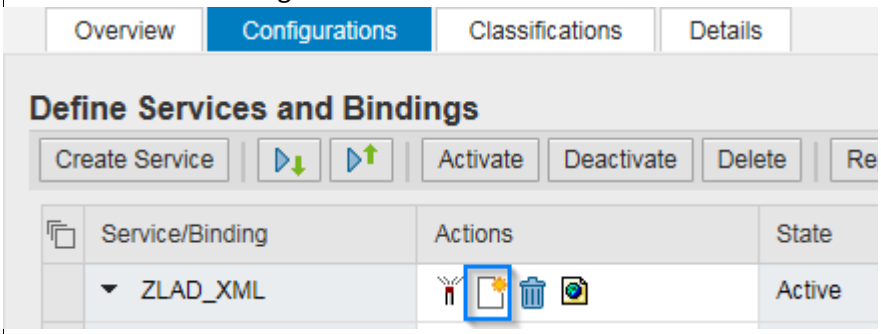
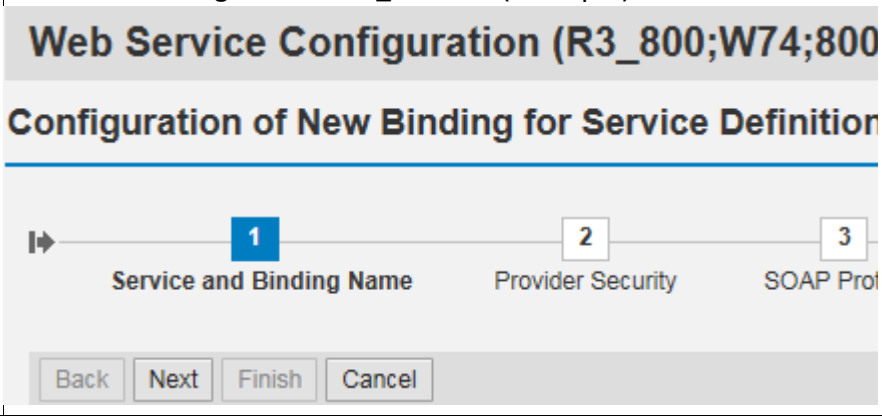
This is how you select the service which is provided by SEAL Systems and for which you want to establish the integration:

Step	Action
1	Start the SOA manager via the `soamanager` transaction.
2	Switch to the `Service Administration` tab.
3	Click: `Web Service Configuration` 
4	Search for the desired service: Click `Search`. 
5	Select the required service of type `Service Definition`: Click the internal name of the service. 

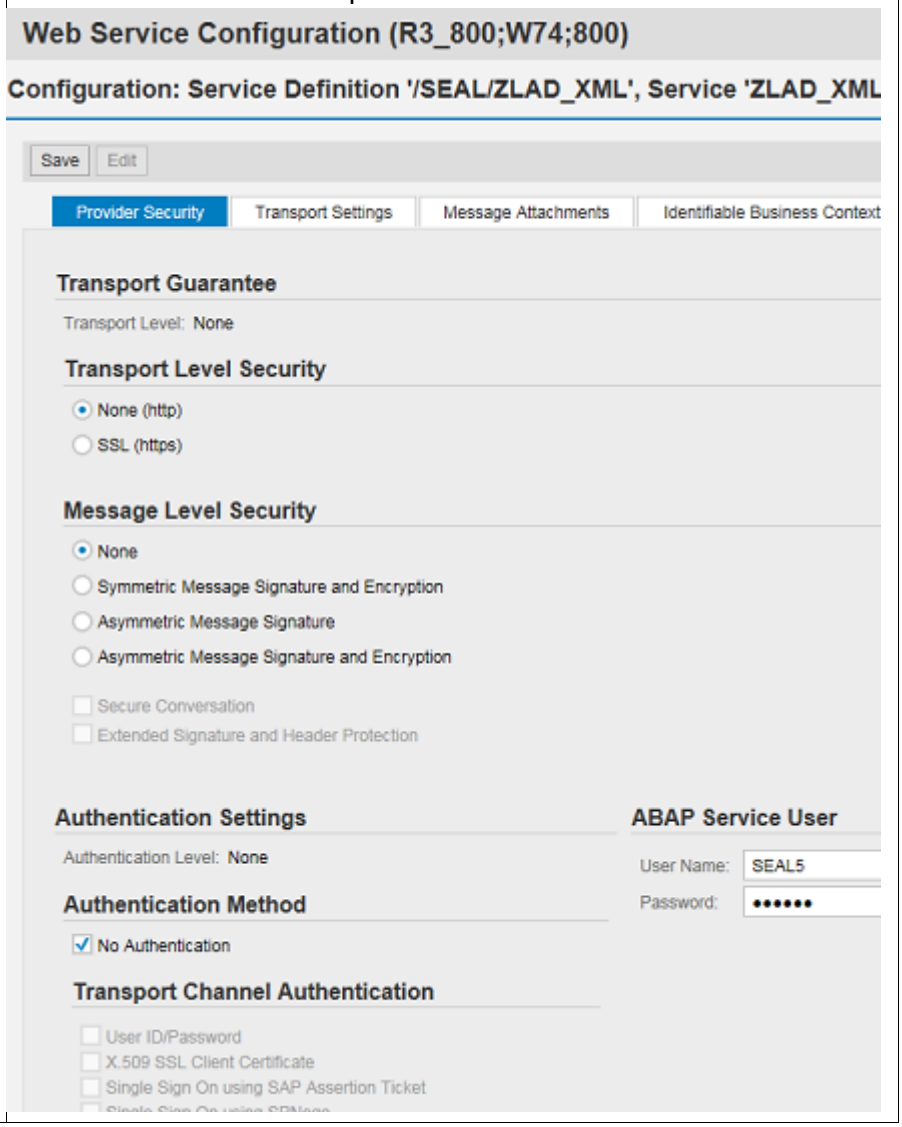
Create the Binding on the Provider System

A binding has to be created on the provider system in order to provide the service. description

This is how you create a binding on the provider system: instructions

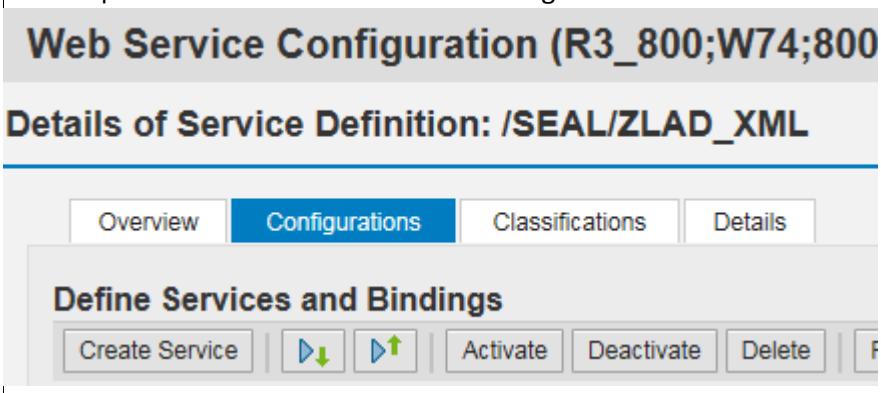
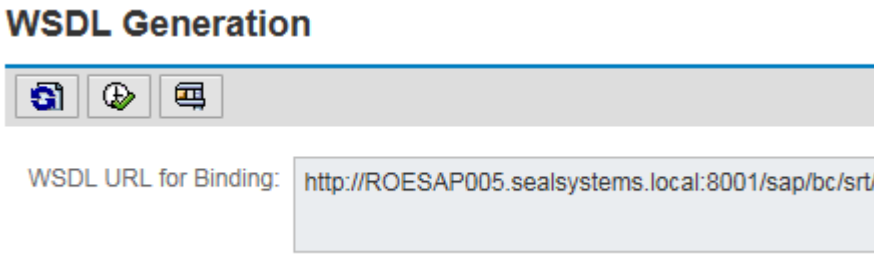
Step	Action
1	→ *Select the Service*, Page 82
2	Switch to the `Configurations` tab.
3	Click `Create Binding`. 
4	Enter the name of the binding and click `Next`: • `Service Name`: `SEAL_ZLAD_XML` (example) • `Description`: Binding for `SEAL_ZLAD_XML` (example) • `New Binding Name`: `SEAL_ZLADXML` (example) 

Create the Binding on the Provider System, Continuation

Step	Action
5	Specify the required settings and click `Finish`: • `Communication Security`: `None (HTTP)` activate • `Authentication Method`: `No Authentication` activate • `ABAP Service User`: Enter user name and password 

Determine URL for WSDL Access

This is how you determine the URL for the WSDL access on the provider system, which you need for the creation of the logical port on the consumer system: instructions

Step	Action
1	→ *Select the Service*, Page 82
2	Switch to the `Configurations` tab.
3	Click `Open WSDL Generation for Binding`. 
4	Copy the URL. 

6.3 Troubleshooting - Analysis of Web Service Requests

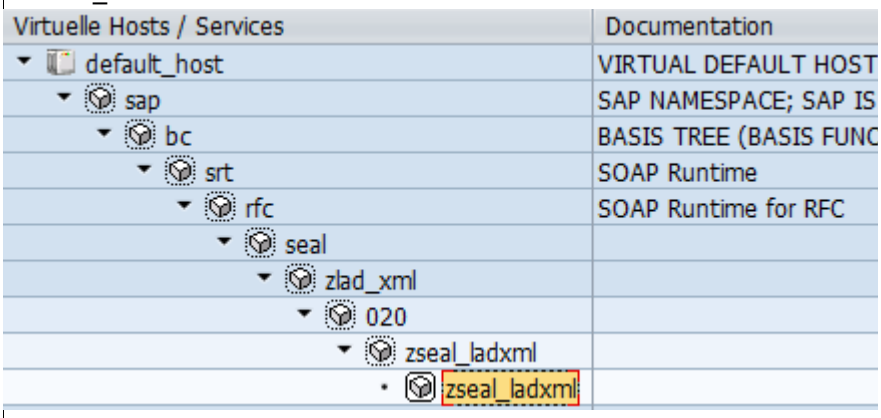
in this chapter

This chapter deals with the following topics:

Record and Display Web Service Requests

instructions

This is how you use the recorder in order to record and display incoming and outgoing Web service requests on the provider system:

Step	Action
1	Start the `sicf` transaction.
2	Open the service tree for the `SERVICE` hierarchy type: `sap/bc/srt/rfc/sap/`*`Web service Function/Client/Web service Name/Binding`* Example: `/sap/bc/srt/rfc/seal/zlad_xml/020/zseal_ladxml/zseal_ladxml` 
3	Select the menu: • `Edit→Recorder→Activate Recording` for incoming requests on the provider system • `Client→Recorder→Activate Recording` for outgoing requests on the consumer system
4	Enter the desired settings and click `Activate`.
5	Start the Web service.
6	Display the recording with: • `Edit→Recorder→Display Recording` for incoming requests on the provider system • `Client→Recorder→Display Recording` for outgoing requests on the consumer system

trace messages

Activate additional trace messages for the troubleshooting on the provider system via `Edit→Trace→Activate Trace`.

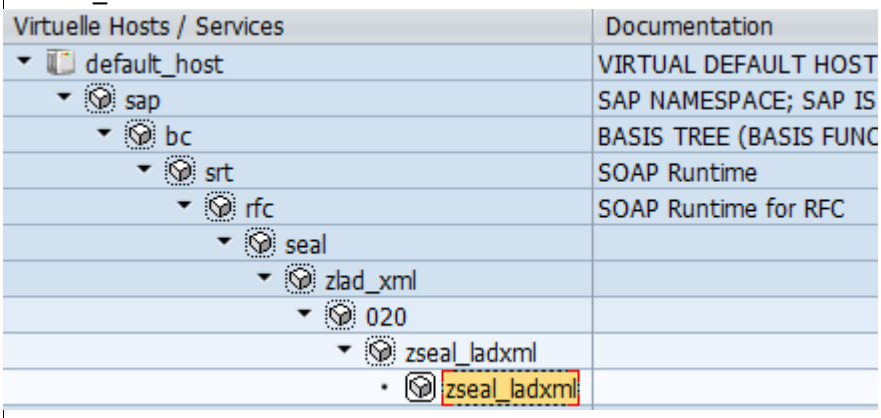
Debug Web Service Requests

requirement

→ *Required Authorizations*, Page 80

instructions

This is how you debug Web service requests on the provider system:

Step	Action
1	Start the `sicf` transaction.
2	Open the service tree for the `SERVICE` hierarchy type: `sap/bc/srt/rfc/sap/`*`Web service Function/Client/Web service Name/Binding`* Example: `/sap/bc/srt/rfc/seal/zlad_xml/020/zseal_ladxml/zseal_ladxml` 
3	Select the menu: `Edit→Debugging→Activate Debugging`
4	Start the Web service.
5	Debug the Web service in the debugger.

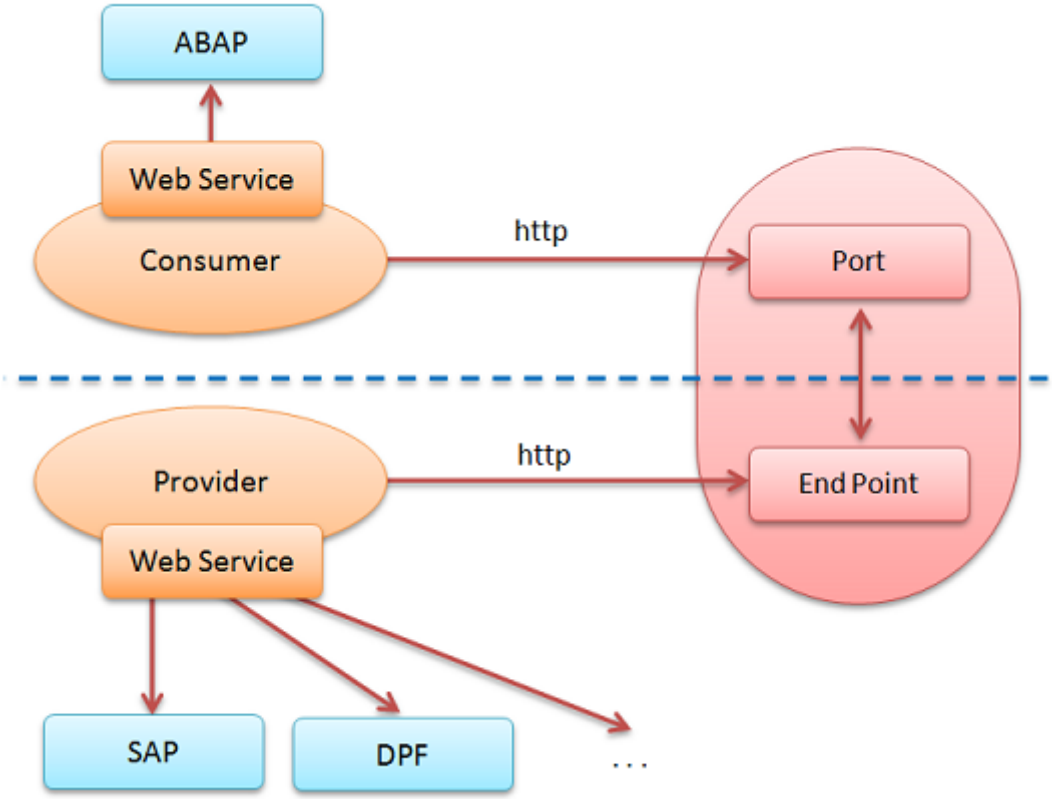
7 Integration via Web Service - SAP as Consumer

This chapter describes the establishing of an integration via Web service, if SAP serves as consumer. introduction

External functionality can be provided via Web service in order to be used by SAP as consumer. description

→ *Integration via Web Service - SAP as Provider*, Page 79 related topics

SAP as consumer: overview



This chapter deals with the following topics: in this chapter

7.1 Required Authorizations

requirement - roles

The required authorizations have to be assigned to the users via the following roles (transaction: `su01`; maintenance of the authorizations via the profile generator with the `pfcg` transaction):

- User, who administrates the Web service via the SOA manager:
  `SAP_BC_WEBSERVICE_ADMIN_TEC`
- SAP as provider - user, who is used as ABAP service user for the endpoint on the provider system:
  `SAP_BC_WEBSERVICE_SERVICE_USER`
- SAP as provider - user, who want to debug the Web service requests:
  `SAP_BC_WEBSERVICE_DEBUGGER`
- SAP as consumer - user, who starts the Web service:
  `SAP_BC_WEBSERVICE_CONSUMER`

7.2 Configuration on the Consumer System (SAP)

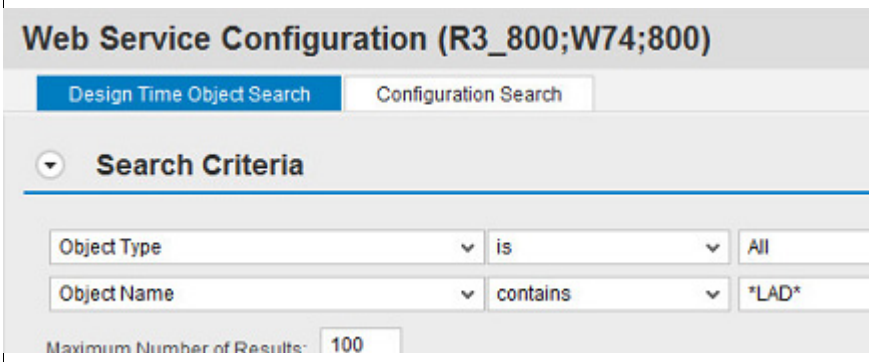
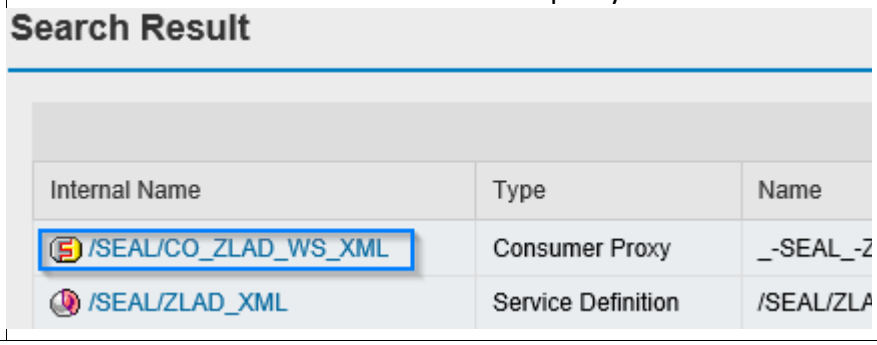
The Web service, which you want to use from SAP, is started externally. requirement

This chapter deals with the following topics: in this chapter

Select the ABAP Proxy

instructions

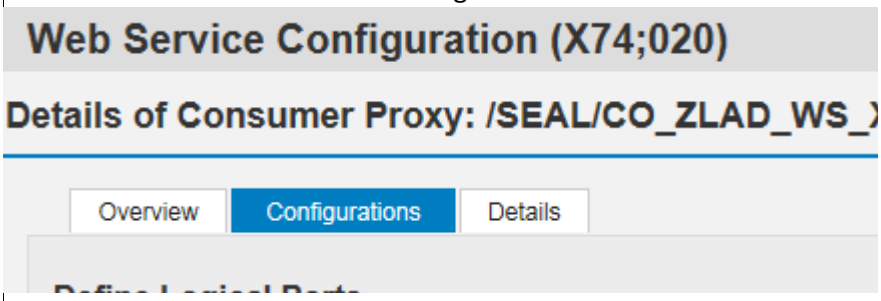
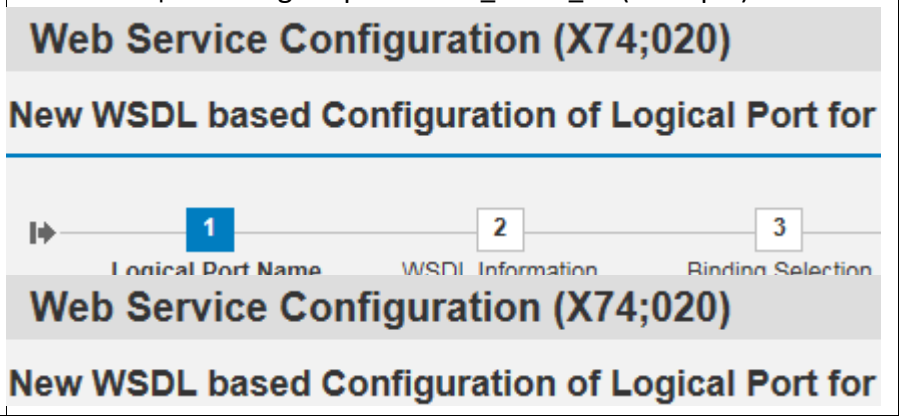
This is how you select the ABAP proxy for which you want to establish the integration:

Step	Action
1	Start the SOA manager via the `soamanager` transaction.
2	Switch to the `Service Administration` tab.
3	Click: `Web Service Configuration` 
4	Search for the desired service: Click `Search`. 
5	Select the desired consumer proxy from the `Consumer Proxy` type: Click the internal name of the consumer proxy. 

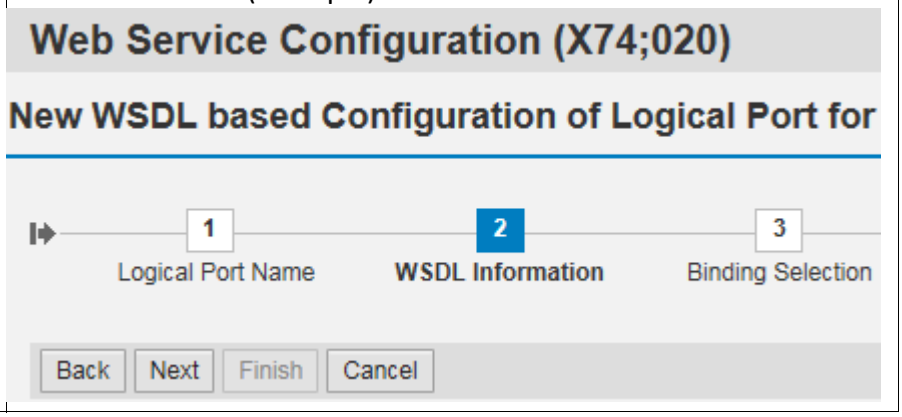
Create a Logical Port

A logical port has to be created for the communication between the service consumer and the external Web service. description

This is how you create a logical port: instructions

Step	Action
1	→ *Select the ABAP Proxy*, Page 92
2	Switch to the `Configurations` tab.
3	Click `Create - WSDL based Configuration`. 
4	Enter the data for the logical pro and click `Next`: • `Logical Port Name`: `WS_DPF4C_OK` (example) • `Logical Port is Default`: activate • `Description`: Logical port for `WS_DPF4C_OK` (example) 

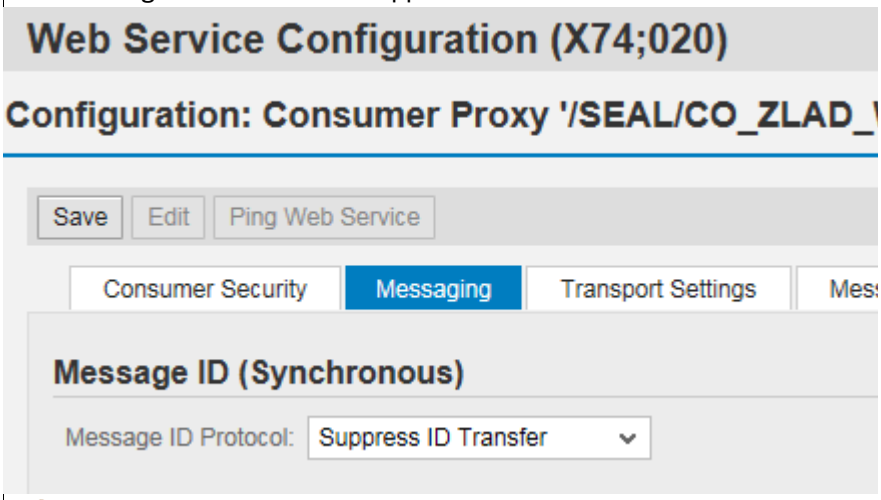
Create a Logical Port, Continuation

Step	Action
5	Enter the WSDL access settings for the logical pro and click `Finish`: • `WSDL Base: Via HTTP Access` activate • `URL for WSDL Access:` `http://server:9125/dpf4c-service-v1.3/convert?wsdl` (example) 

Suppress the Message ID Transfer

The ID transfer of the message ID protocol has to be suppressed. description

This is how you suppress the message ID transfer on the consumer system: instructions

Step	Action
1	→ *Select the ABAP Proxy*, Page 92
2	Switch to the `Configurations` tab.
3	Select the desired logical port and click `Edit`.
4	Switch to the `Messaging` tab:
5	Enter: • `Message ID Protocol`: `Suppress ID Transfer`  Caution - default Up to SAP NetWeaver Application Server 7.3, `Suppress ID Transfer` is the default. As of SAP NetWeaver Application Server 7.4, this value has to be specified explicitly:

Transfer of Large Amounts of Data

description

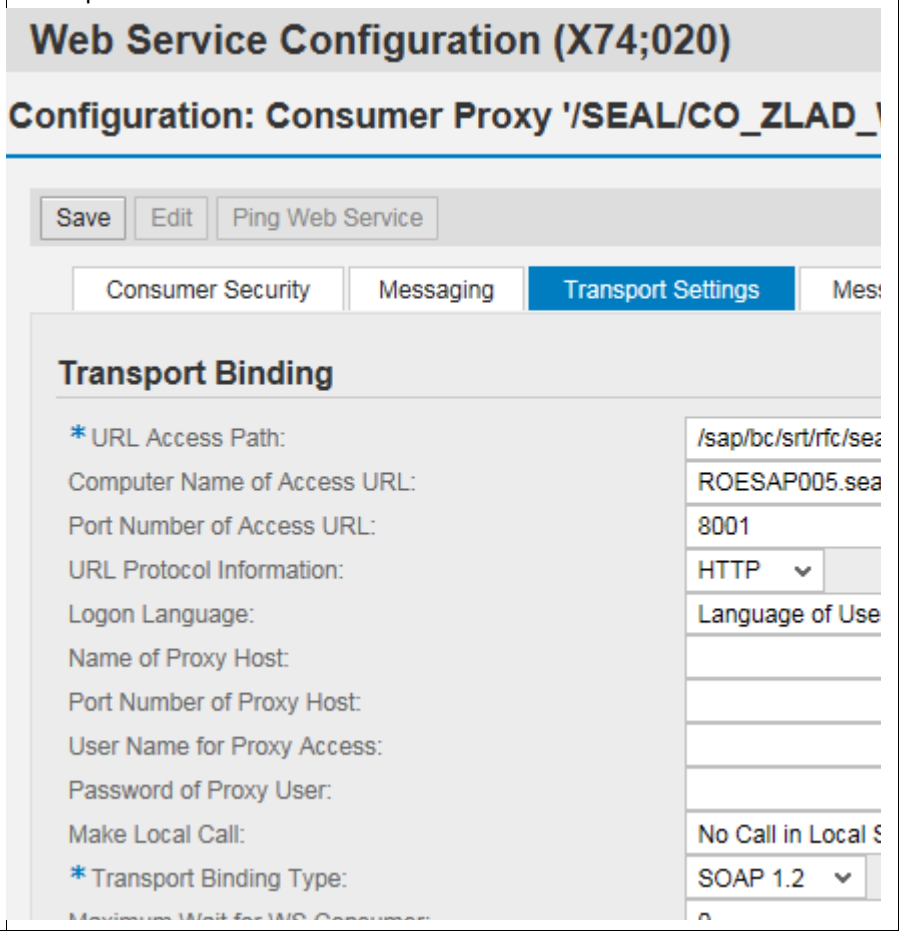
If you transfer large amounts of date in the Web service environment, you can use the SOAP Message Transmission Optimization Mechanism (MTOM).

requirement

MTOM is supported as of SAP release 7.20 or newer without restrictions, see SAP note `1582187`.

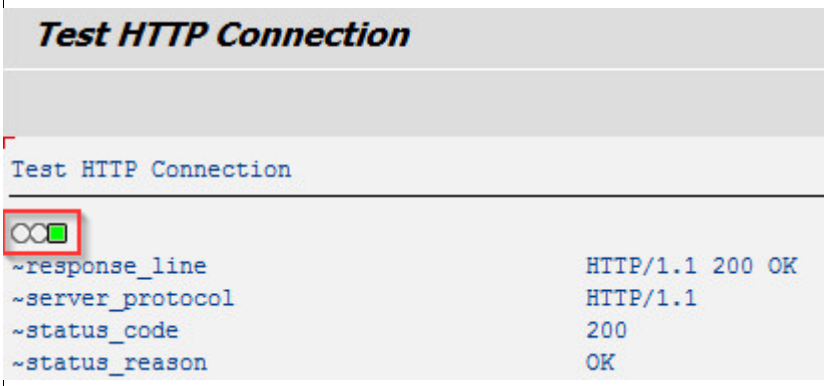
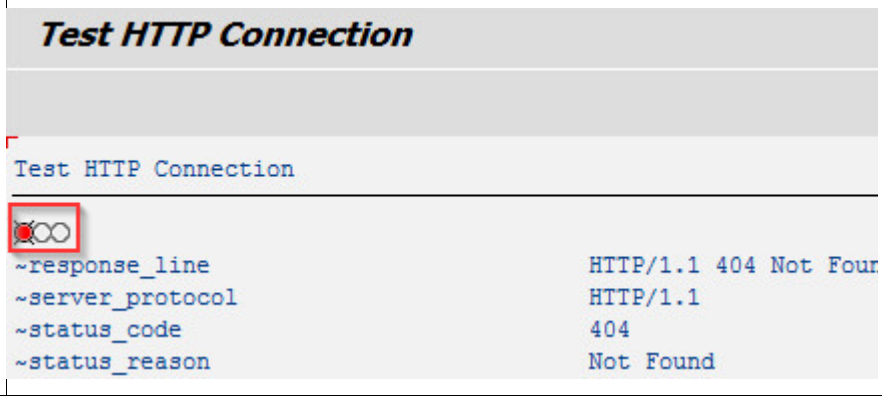
instructions

This is how you activate the optimized XML transfer for large amounts of date on the consumer system:

Step	Action
1	→ *Select the ABAP Proxy*, Page 92
2	Switch to the `Configurations` tab.
3	Select the desired logical port and click `Edit`.
4	Switch to the `Transport Settings` tab.
5	Enter: • `Optimized XML Transfer`: `MTOM` 

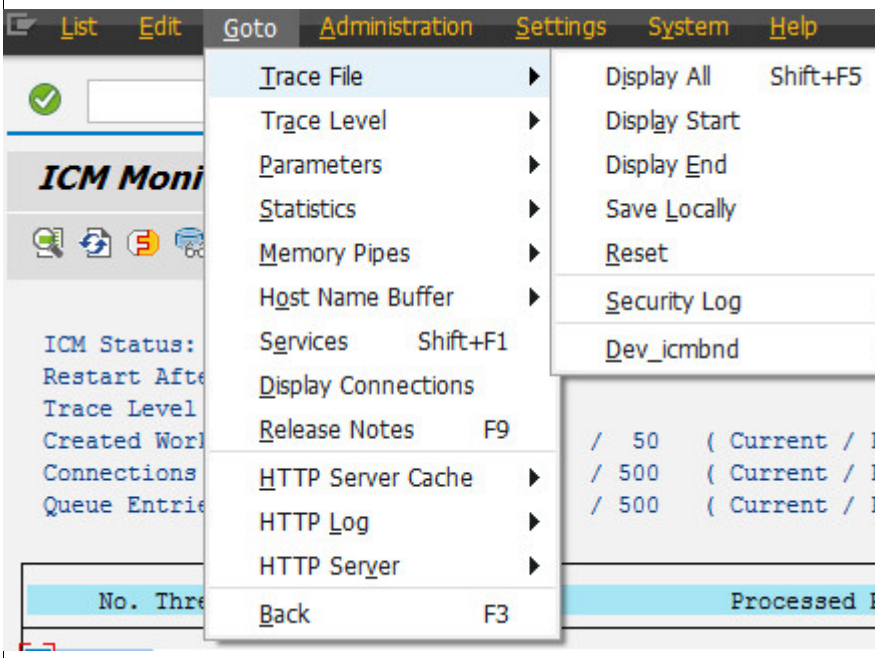
7.3 Advisable Behavior in the Case of Error

This is how you check the possible reasons in the case of error step by step: instructions

Step	Action
1	Check if the URL is accessible: • Start the rswf_test_http program (transaction: sa38). • Enter the URL to be tested and click . Example:  Result: A green status display in the first line indicates a successful connection, a red status display indicates an error. Example - HTTP connection successful:  Example - HTTP connection is faulty: 

Advisable Behavior in the Case of Error, Continuation

Step	Action

2	If an error occurs, start the ICM monitor to determine the exact cause of the error and forward this data to your Technical Project Manager at SEAL Systems if necessary:
	Transaction: `smicm`
	• Display trace file:
	`Goto→Trace File→Display All`
	• Save trace file:
	`Goto→Trace File→Save Locally`
	 
	
	Hint - reset trace file:
	For a better overview, you can delete the trace file with `Goto→Trace File→Reset` and execute the erroneous call again.
	Hint - set trace level:
	In general, the default trace level is sufficient. You can change it with `Goto→Trace Level→Set` and execute the erroneous call again.

7.4 Troubleshooting

typical problems and their solutions

The following table illustrates typical problems (P) and their approaches (A):

P:	The Web service returns an error message of the following type: `SOAP:1.023 SRT: Processing error in Internet Communication Framework: ("ICF Error when receiving the response: IC-M_HTTP_CONNECTION_FAILED"` Possible cause is that the Web service is not started.
A:	Start the Web service.

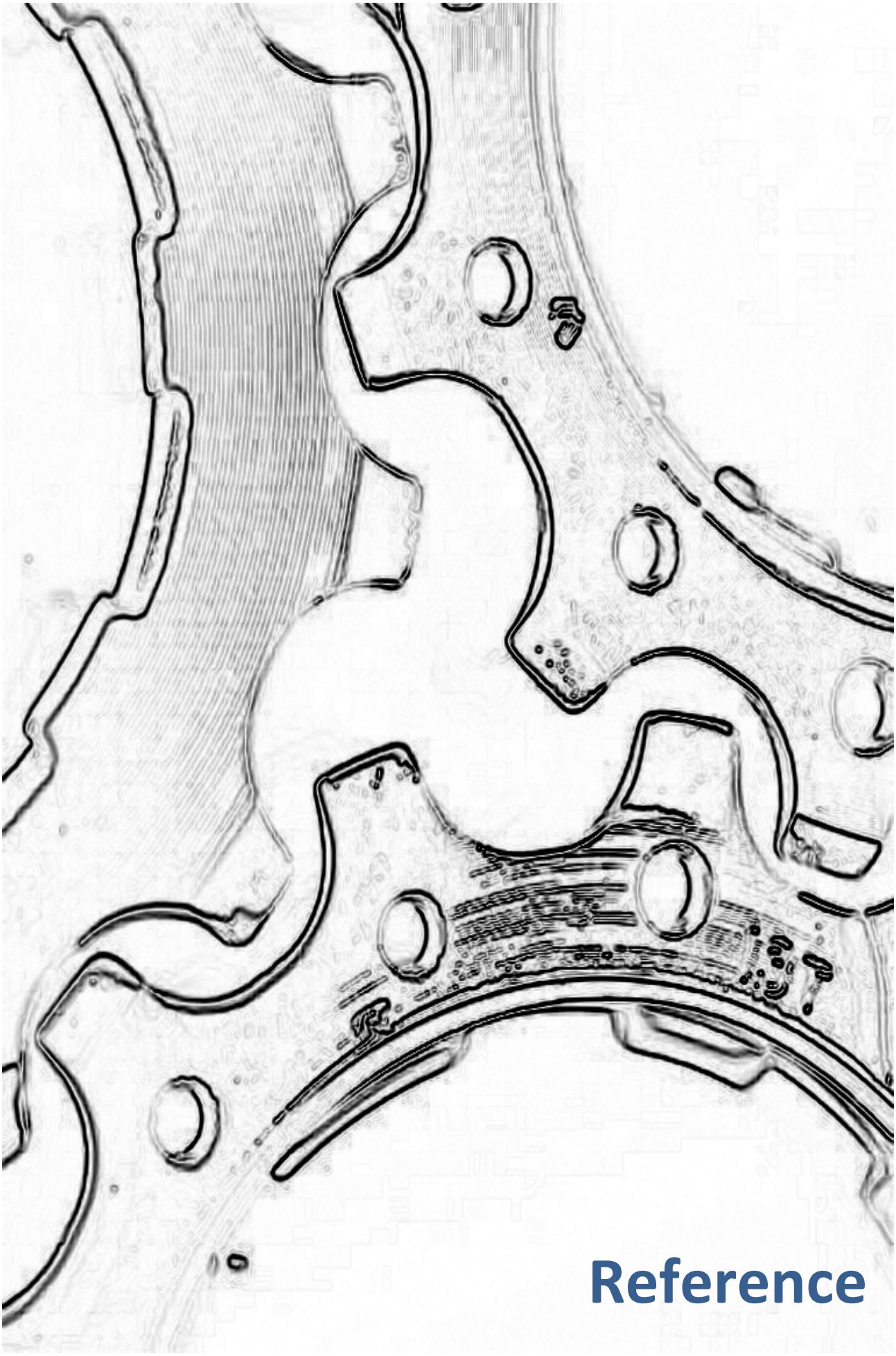
P:	The Web service returns an error message of the following type:
	`Error while calling DPF Webservice`
	`SoapFaultCode:3 MustUnderstand headers:[{http://schemas.xmlsoap.org/ws/2004/08/addressing}Action, {http://schemas.xmlsoap.org/ws/2004/08/addressing}To] are not understood`
	Possible cause is that the ID transfer of the message ID protocol is not suppressed.

A:	Suppress the message ID transfer: → *Suppress the Message ID Transfer*, Page 95
P:	The Web service returns an error message of the following type: `SOAP:1.001 CX_SXML_PARSE_ERROR: An exception was raised. Error when parsing an XML stream: '<EOF> reached` Possible cause is that the optimization of the transfer of large amounts of data is not activated.
A:	Optimize the transfer of large amounts of data: → *Transfer of Large Amounts of Data*, Page 96
P:	The Web service returns an error message of the following type: `SOAP:1.007 SRT: Unsupported xstream found: ("HTTPCode 502: Bad Gateway")` Possible cause is that the timeout is too small.
A:	Increase the timeout in `server\web\apache\conf\workers.properties`: `worker.seal-worker.socket_timeout`
P:	The Web service returns an error message of the following type: `"HTTPCode 502: Bad Gateway"` Possible cause is that the timeout is too small.

Troubleshooting, Continuation

A:	Increase the timeout for the `HTTP` protocol via the `smicm` transaction with `Goto→Services` and `Service→Change`: `Keep Alive (in Sec.)` In addition to that, check the following DPF timeouts: • `apache\conf\httpd.conf`: `TimeOut 1200` (20 min) • `apache\conf\workers.properties`: `worker.seal-workers.socket_timeout 1200` (20 min) • `tomcat\conf\server.xml`: `connectionTimeout 600000`(10 min, is generally not activated)
P:	The Web service returns an error message of the following type: `"ICM_HTTP_TIMEOUT"` Possible cause is that the timeout is too small.

A:	Check the following values for the `HTTP` log via the `smicm` transaction with `Goto→Services` and `Service→Change`:
	`Keep Alive (in Sec.)`
	`Maximum Processing Time (ProcTimeout)`



Reference

8 Configuration Files - Reference

introduction

This chapter explains the configuration files which are evaluated to establish a connection between the OM server and the SAP system in case of communication via RFC destination.

in this chapter

This chapter deals with the following topics:

→ *cadrfc.ini - Logon Information*, Page 104

→ *saprfc.ini - Connection Data*, Page 116

→ *rfcserver.cfg*, Page 146

`rfcserver.cfg` serves as an example for the following configuration files:

- `alfilechecker.cfg`
- `alviewserver.cfg`
- `convserv.cfg`
- `convservdpf.cfg`
- `dvsviewserver.cfg`
- `filecheck.cfg`
- `jrfcserver.cfg`
- `rfcserver.cfg`

8.1 cadrfc.ini - Logon Information

introduction

This chapter contains the reference information about the cadrfc.ini `configuration file.`

requirement

The `cadrfc.ini` file is evaluated for RFC client connections. Other RFC server connections ignore this file.

ASCII/Unicode

Within `cadrfc.ini` only ASCII characters are supported. Unicode characters are not supported.

location

The `cadrfc.ini` file is located in the following directory:

`applications\server\sapserv\bin_...`

 hint - gXnetplot

For gXnetplot, the file is located in:

`$GRALPLOTLIB/PDM (servermenu - c - cdl - cd PDM)`

purpose

The configuration file contains the necessary logon data which is evaluated by the BAPI and CAD interface.

The interactive RFC clients use the logon data as default values for the initial logon dialog.

The remaining RFC clients and RFC servers use the logon information for the initial logon when they start up. Later logons use the information saved in the job files (repro lists).

structure

You can specify general as well as system- and client-specific logon data:

- General logon data is specified in the configuration file without section name.
- System- and client-specific logon data is specified in separate sections subsequent to the general logon data:
  `[SAP\`*`SystemName`*`\`*`Client`*`]` or `[SAP\`*`SystemName`*`]`
  Example:
  `[SAP\W74\020]`or`[SAP\W74]`

 example

Extract of an example for `cadrfc.ini`:

```
*---------------------------------------------------
* SAP logon parameters
*---------------------------------------------------
CadRfcUser SEALCPIC
CadRfcPassword <SAPPWD>
#CadRfcPasswordCoded 0xe3f0e9c6604b14b3
#CadRfcClient 010
```

cadrfc.ini - Logon Information, Continuation

```
CadRfcLanguage EN
*-----------------------------------------------------
* Connection parameters
*-----------------------------------------------------
* Logical destination
CadRfcDestination W46
...
*-----------------------------------------------------
* Parameters specific for Dialog interface (SapConnc)
*-----------------------------------------------------
...
* important for check-in/check-out
CadDialogNetAddress DEFAULT
*-----------------------------------------------------
* TRACE parameters 0 - set trace off | 1 - on
*-----------------------------------------------------
CadRfcTrace 0
CadTraceDir
*-----------------------------------------------------
* ABAP debug parameter 0 - set debug off | 1 - on
*-----------------------------------------------------
CadRfcAbapDebug 0
...
*-----------------------------------------------------
* system/client-specific logon data
*-----------------------------------------------------
[SAP\W74]
CadRfcUser SEALPE
CadRfcPasswordCoded 0xe3f0e9c6604b14b3
CadRfcClient 010
...
```

Only a couple of parameters of the configuration file must be adjusted to the current system environment. Only these parameters are described below. The customize

cadrfc.ini - Logon Information, Continuation

	remaining parameters have reasonable defaults and do not need to be adjusted.
restart	You must restart the DMS Rlist RFC client after you have changed parameters in `cadrfc.ini`.

System-/Client-Specific Logon Data

requirement

The `rlistsap`, `sapcli`, `omscli` and `oms_server` programs must be linked with `sapini.c` version 1.14 or newer.

To check the correct version, for example of `rlistsap`, execute the following steps:

Step	Action
1	Enter: `what rlistsap | grep sapini.c`
2	As result must be displayed: `SAP-CAD $Id: sapini.c,v 1.14 2005/10/12 ...` The requirement is fulfilled if the version is 1.14 or newer. Otherwise contact your Technical Project Manager at SEAL Systems.

restriction

With regard to DMS Rlist for the initial login to the SAP system during the start of DMS Rlist, only the login data in the general (system and client independent) section of the file `cadrfc.ini` is used. The system- and client-specific login information is only evaluated during the processing of repro list jobs at runtime.

structure

System- and client-specific logon data is specified in separate sections:

`[SAP\`*`SystemName`*`\`*`Client`*`]` or `[SAP\`*`SystemName`*`]`

Example:

`[SAP\W74\020]`or`[SAP\W74]`

Caution - at the end

The sections containing the system- and client-specific logon data must be located at the end of the file after the general logon information.

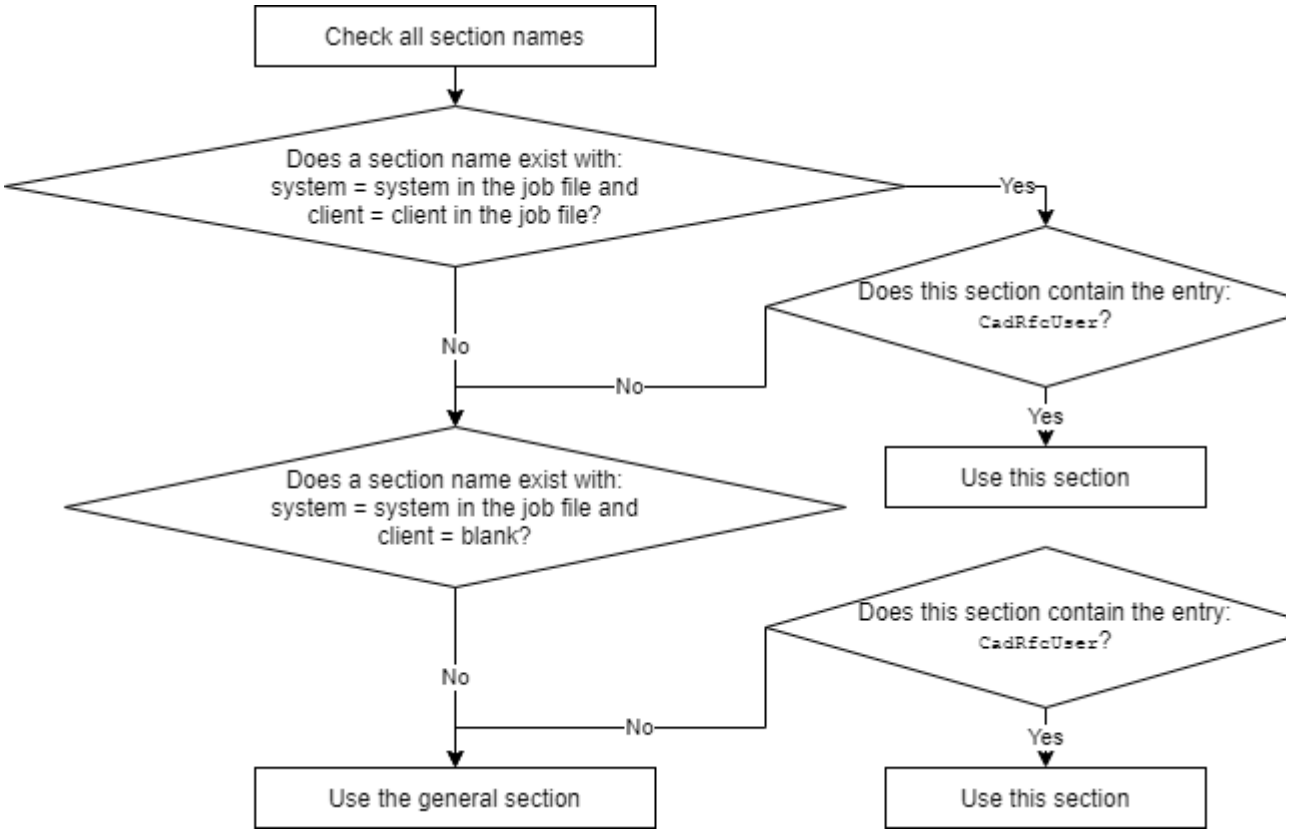
system/client comparison

The current system and the current client specified in the order file (repro list) are compared with the configuration entries in `cadrfc.ini` to determine the user to be used for logon.

System-/Client-Specific Logon Data, Continuation

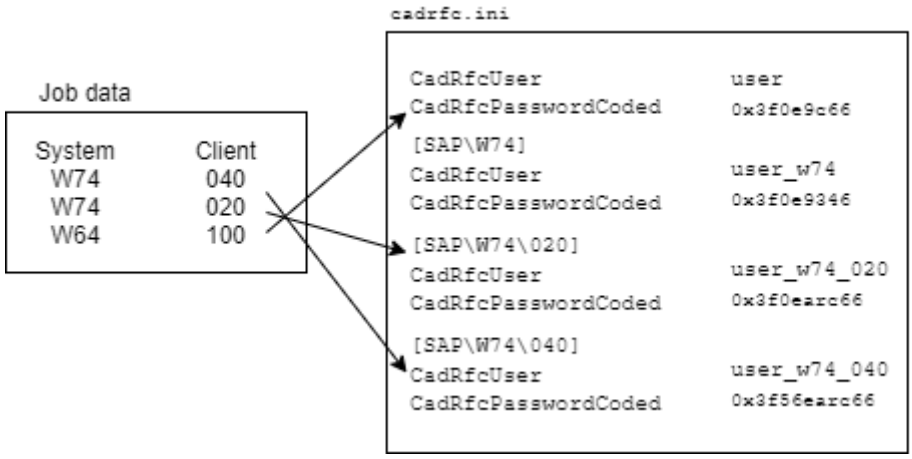
section determination

The section of `cadrfc.ini` which is to be used for the current system/client specified in the order file is determined as follows:



example

The subsequent example demonstrates which user is used for logon dependent on the system/client in the job data.



CadRfcUser Parameter

meaning

The `CadRfcUser` parameter determines the user used by the RFC client or RFC server to logon on the SAP system.

requirement

The following requirements must be fulfilled for a successful logon:

- The specified user must be identically configured on all SAP systems.
- The user name must be entered in capitals.
- The interactive RFC clients require a dialog user.
- For security reasons, a system user is generally used for logon by the remaining RFC clients, for instance DMS loader and DMS Rlist, and the RFC servers.
  Exception:
  If `USE_SAPGUI` in `saprfc.ini` is specified as `1` or `2` a dialog user must be used!

CadRfcPassword Parameter

meaning	The `CadRfcPassword` parameter specifies the uncoded password used by the RFC client and RFC server to logon on the SAP system.
alternative	The password can be specified in a coded form with the `CadRfcPasswordCoded` parameter. For security reasons, the coded passwords are to be preferred.
requirement	The `CadRfcPassword` parameter is ignored as soon as a coded password is specified with the parameter `CadRfcPasswordCoded`.
 example	The item looks as follows: `CadRfcPassword xyz`

CadRfcPasswordCoded Parameter

meaning

The `CadRfcPasswordCoded` parameter specifies the coded password used by the RFC client and RFC server to logon on the SAP system. This is the more secure variant compared to the previous `CadRfcPassword`parameter.

code the pass-word

This is how you create and save a coded password:

Step	Action
1	Execute the program to encode the password: `sealencrypt.exe config`*`password`*
2	Enter the created coded password in the `cadrfc.ini` file. 🔍 Example: `CadRfcPasswordCoded e3f0e9c6604b14b3`

CadRfcClient Parameter

meaning

The `CadRfcClient` parameter determines the client used by the RFC client and RFC server to logon on the SAP system.

CadRfcLanguage Parameter

meaning

The `CadRfcLanguage` parameter specifies the language used by the RFC client and RFC server to logon on the SAP system.

All two-character ISO-639-1 language codes installed on the SAP system can be specified as logon language. ISO-639-2B/T and ISO-639-3 are not supported.

example

The item looks as follows:

```
CadRfcLanguage EN
```

CadRfcDestination Parameter

meaning

The `CadRfcDestination` parameter specifies the SAP system on which the RFC client and RFC server log on.

 example

The item looks as follows:

```
CadRfcDestination W74
```

CadDialogNetAddress Parameter

The `CadDialogNetAddress` parameter specifies the host name of the local machine. This is used for check-in and check-out files.	meaning
If the `CadDialogNetAddress` parameter is not specified the value of the `HOSTNAME` (Windows) or `DISPLAY` (Unix) environment variable is used instead.	environment variable
The following value is used as default: `CadDialogNetAddress DEFAULT`	default

8.2 saprfc.ini - Connection Data

introduction

This chapter contains the reference information about the `saprfc.ini` configuration file.

requirement: ASCII/Unicode

Within `saprfc.ini` only ASCII characters are supported. Unicode characters are not supported.

location

The `saprfc.ini` file is located in the following directory:

`client\dvs...\bin_...` (RFC client connection)

`server\sapserv\conf` (RFC server connection)

 hint - gXnetplot

For gXnetplot, the file is located in:

`$GRALPLOTLIB/PDM (servermenu - c - cdl - cd PDM)`

contents

The configuration file contains parameter items with the system data for the individual SAP systems. This data is used to establish the RFC connections from the external server to the SAP systems.

structure

Dependent on the type of RFC destination - client or server - different parameter entries in `saprfc.ini` are required. All parameter entries concerning one type are grouped as one block without section name in the configuration file.

in this chapter

This chapter deals with the following topics:

saprfc.ini - Connection Data, Continuation

→ *USE_SAPGUI Parameter*, Page 136
→ *UNICODE Parameter*, Page 137
→ *CODEPAGE Parameter*, Page 138
→ *SNC_MODE Parameter*, Page 139
→ *SNC_MYNAME Parameter*, Page 140
→ *SNC_PARTNERNAME Parameter*, Page 141
→ *SNC_QOP Parameter*, Page 142
→ *SNC_SSO Parameter*, Page 143
→ *ASCS Parameter*, Page 144
→ *X509CERT Parameter*, Page 145

Determine System Data forsaprfc.ini

SAP Basis

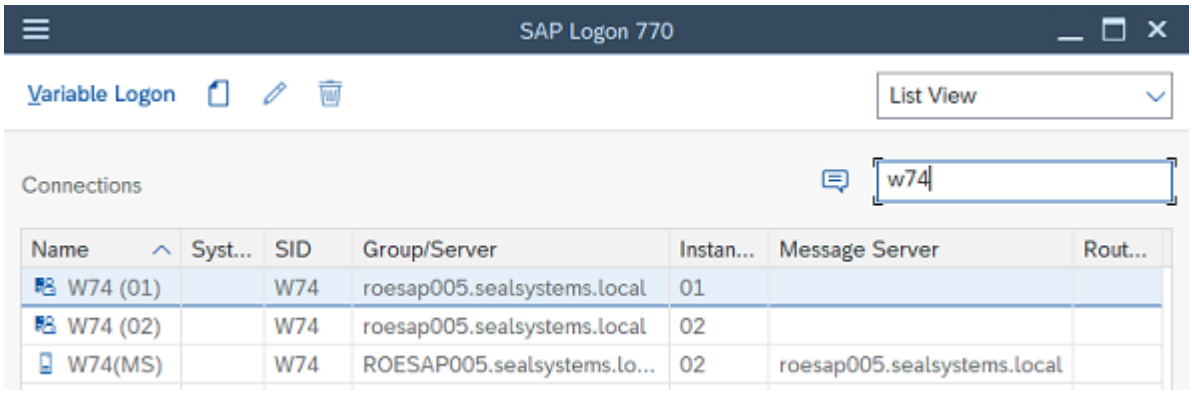
You can get the current SAP system data, such as instance number, router string and so on, from your SAP Basis.

SAP logon

If SAP GUI is installed, you can display the SAP system data also on SAP logon:

- `SID`
- `Group/Server`
- `Instance Number`
- `Message Server`
- `Router`

example



background knowledge

Further system data, like the information whether it is a Unicode SAP system, can be determined for each system via the following program execution:

`tools\bin_winnxx\sap_conn_checker`*`Parameter`*

The data is configured in the `SAPGUILandscape.xml` file.

Types of RFC Destinations and Module Classification

type of RFC destinations

The following types of RFC destinations can be distinguished:

- RFC client via application server (type A)
- RFC client via message server (type B)
- RFC server via SAP gateway (type R)

modules and types

The modules from SEAL Systems can be classified in RFC client and RFC server modules as follows:

RFC Client	RFC Server
DMS Rlist	DMS Loader
DMS Scan	Conversion Server
JSAPcli, SAPcli	RFC Server, JRFC Server, for instance for • DMS Loader/ABAP • DMS View Server • PDF Longlife Suite - SAP Integration

Parameter Overview

validity

Some parameters are evaluated for all connection types. In addition to these general parameters there are parameters which are only important for special connection types.

overview

The following table presents an overview of which parameters are evaluated for which connection type:

Parameters	RFC Client Application Server	RFC Client Message Server	RFC Server Gateway
DEST	X	X	X
TYPE	A	B	R
ASHOST	X	-	-
SYSNR	X	-	-
MSHOST	-	X	-
GROUP	-	X	-
GWHOST	-	-	X
GWSERV	-	-	X
SAPROUTER	X	X	X
RFC_TRACE	X	X	X
SEAL_TRACE	-	-	X
ABAP_DEBUG	X	X	-
USE_SAPGUI	X	X	-
UNICODE	X	X	X
CODEPAGE	-	-	X
SNC_MODE	X	X	X
SNC_MYNAME	X	X	X
SNC_PARTNER-NAME	X	X	-
SNC_QOP	X	X	X
SNC_SSO	X	X	X
ASCS	X	X	-
X509CERT	X	X	X

Parameter Overview, Continuation

The following table contains example parameter entries for the different connection types:  example

Type	Items
RFC client (application server)	DEST=W74 TYPE=A ASHOST=/H/SEALDOS3/H/sapt60-1 SYSNR=00 RFC_TRACE=0 ABAP_DEBUG=0 USE_SAPGUI=0 UNICODE=1 SNC_MODE=1 SNC_MYNAME=p:CN=SEALRFC, OU=SEALSAP, O=SEAL, C=DE SNC_PARTNERNAME=p:CN=T6B, OU=SEALSAP, O=SEAL, C=DE SNC_QOP=3 SNC_SSO=1 X509CERT=c:\seal\customer\server\sapserv\conf\sec\SEAL-RFC.crt
RFC client (message server)	DEST=W74 TYPE=B MSHOST=/H/SEALDOS3/H/sapt60-2 GROUP=Standard RFC_TRACE=0 ABAP_DEBUG=0 USE_SAPGUI=0 UNICODE=1

Parameter Overview, Continuation

Type	Items
RFC server gateway	DEST=W74RFC TYPE=R GWHOST=/H/SEALDOS3/H/sapt60-3 GWSERV=sapgw00 RFC_TRACE=0 SEAL_TRACE=1 UNICODE=1 CODEPAGE=UTF-8 SNC_MODE=1 SNC_MYNAME=p:CN=SEALRFC, OU=SEALSAP, O=SEAL, C=DE SNC_QOP=8 SNC_SSO=1

Caution - RFC client

For each RFC client, only one item is allowed, either one with TYPE=A for application server or one with TYPE=B for message server.

DEST Parameter

meaning

The `DEST` parameter contains the unique identifier for the SAP system.

It refers to the `DESTINATIONS` item in the configuration file like `rfcserver.cfg`. It is used to determine the SAP system data for the RFC destination which should be established.

naming con-ven-tion

The identifier should indicate the SAP system and the type of the RFC destination. The following naming convention is advisable where `W74` is used as example of an SAP system name:

DEST Value	Using for
`W74`	RFC client connections via application server
`W74`	RFC client connections via message server
`W74RFC`	RFC server connections via SAP gateway

length

RFC server as of 1.3.3 or newer allows identifiers with a maximum of 32 char-acters. Lower versions only support a maximum of 8 characters.

validity

The parameter `DEST` is evaluated for all destination types.

TYPE Parameter

meaning

The TYPE parameter identifies the connection type.

values

The following connection types are available:

- A
  RFC client connections via application server
  The connection via an application server is the standard connection type for RFC clients. This type is used by all interactive modules and by the modules DMS Rlist and DMS Loader.
- B
  RFC client connections via message server
  The connection via a message server is available for RFC clients as an alternative to the connection via an application server. It is rarely used. It can be used for load distribution purposes (load balancing) to distribute the registration via the message server to several application servers.
- R
  RFC server connections via SAP gateway
  RFC server or conversion server register with an intermediate SAP gateway where they then wait for queries from the SAP system or other RFC clients. Usually, an SAP gateway is running on each application server.

Caution - instance number/ port

Depending on the connection type, a port must be assigned to the instance number of the SAP system (xx) in the `winnt\system32\drivers\etc\services` system file:

- Type A, RFC client connections via application server
  `sapdp`*xx*`32xx/tcp`
- Type B, RFC client connections via message server
  `sapmsgxx 36`*xx*`/tcp`
- Type R, RFC server connections via SAP gateway
  `sapgw`*xx* `48`*xx*`/tcp` (SNC), `sapgw`*xx* `33`*xx*`/tcp` (otherwise)

hint - automatic mapping

Items in the `saprfc.ini` file starting with `sapgw` are automatically mapped to the correct port, for instance `sapgw00` to `3300` or `sapgw01` to `3301`. It is not required to map them explicitly in the system file.

Caution - message server with SAProuter

For RFC destinations via message server (TYPE=B) with SAProuter, the following requirement must be fulfilled:

- In the DNS, both the Windows host name of the message server and its SAPLOCALHOSTFULL as Fully Qualified Domain Names (FQDN) must be maintained.

validity

The TYPE parameter is evaluated for all destination types.

ASHOST Parameter

The ASHOST parameter contains the host address of the application server. — meaning

You can see the host address in: — system data

- *Group/Server*

```
ASHOST=sapserver.com
```

example

→ *Determine System Data forsaprfc.ini*, Page 118

related topics

If an SAProuter is in use (for example: SAPROUTER=/H/saprouter.com/S/3299), the full name for ASHOST is automatically composed of: — background knowledge - SAProuter

- *Router*
  Begin of string with URL specification, rest truncated
- /H/
- *Group/Server*

```
ASHOST=/H/saprouter.com/H/sapserver.com
```

The ASHOST parameter is evaluated only for the following destination types: — validity

- Type A, RFC client connections via application server

SYSNR Parameter

meaning	The SYSNR parameter contains the instance number (up to release 4.7: system number) of the application server.
system data	You can see the instance number in: • *Instance Number*
example	SYSNR=00
related topics	→ *Determine System Data forsaprfc.ini*, Page 118
validity	The SYSNR parameter is evaluated only for the following destination types: • Type A, RFC client connections via application server

MSHOST Parameter

The `MSHOST` parameter contains the host address of the message server, preferably as a Fully Qualified Domain Name (FQDN).

meaning

You can see the host address in:

- *Message Server*

system data

```
MSHOST=roemsg001.sealsystems.local
```

example

→ *Determine System Data forsaprfc.ini*, Page 118

related topics

If an SAProuter is in use (for example: `SAPROUTER=/H/saprouter.com/S/3299`), the full name for `MSHOST` is automatically composed of:

- *Router*
  Begin of string with URL specification, rest truncated
- `/H/`
- *Message Server*

background knowledge - SAProuter

```
MSHOST=/H/saprouter.com/H/roemsg001.sealsystems.local
```

The `MSHOST` parameter is evaluated only for the following destination types:

- Type `B`, RFC client connections via message server

validity

Parameter R3NAME

meaning

The R3NAME parameter contains the system ID of the SAP system..

In general, the DEST parameter corresponds to the system ID. In this case, R3NAME does not need to be specified.

Only if DEST does not correspond to the system ID is it necessary to enter R3NAME.

system data

You can see the system ID in:

- *SID*

example

```
R3NAME=W74
```

related topics

→ *Determine System Data forsaprfc.ini*, Page 118

validity

The R3NAME parameter is evaluated only for the following destination types:

- Type B, RFC client connections via message server

GROUP Parameter

The GROUP parameter contains the group identifier of the message server. meaning

You can see the group identifier of the message server in: system data

- *Group/Server*

```
GROUP=PUBLIC
```

example

→ *Determine System Data forsaprfc.ini*, Page 118 related topics

The GROUP parameter is evaluated only for the following destination types: validity

- Type B, RFC client connections via message server

GWHOST Parameter

meaning	The GWHOST parameter contains the host address of the SAP gateway, preferably as a Fully Qualified Domain Name (FQDN). The SAP gateway is generally identical to the application server.
system data	You can see the host address in: • *Group/Server*
example	GWHOST=roegw001.sealsystesm.local
related topics	→ *Determine System Data forsaprfc.ini*, Page 118
background knowledge - SAProuter	If an SAProuter is in use (for example: SAPROUTER=/H/saprouter.com/S/3299), the full name for GWHOST is automatically composed of: • *Router* Begin of string with URL specification, rest truncated • /H/ • *Group/Server* GWHOST=/H/saprouter.com/H/roegw001.sealsystems.local
validity	The GWHOST parameter is evaluated only for the following destination types: • Type R, RFC server connections via SAP gateway

GWSERV Parameter

The GWSERV parameter contains the group identifier of the SAP gateway. — meaning

The group identifier of the SAP Gateway is composed of: — system data

- sapgw
- *Instance Number*

GWSERV=sapgw00 — example

→ *Determine System Data forsaprfc.ini*, Page 118 — related topics

The GWSERV parameter is evaluated only for the following destination types: — validity

- Type R, RFC server connections via SAP gateway

Parameter SAPROUTER

meaning	The SAPROUTER parameter contains the host address of the SAProuter.
system data	You can see the host address of the SAProuter in: • *Router*
example	SAPROUTER=/H/saprouter.com SAPROUTER=/H/saprouter.com/S/sapgw01 (with optional port specification)
related topics	→ *Determine System Data forsaprfc.ini*, Page 118
hint - symbolic port	If the SAProuter string contains a symbolic port, for instance /S/sapgw01, this port has to be assigned in the system file: → *TYPE Parameter*, Page 124
validity	The SAPROUTER parameter is evaluated only for the following destination types: • Type A, RFC client connections via application server • Type B, RFC client connections via message server • Type R, RFC server connections via SAP gateway

RFC_TRACE Parameter

meaning

The RFC_TRACE parameter activates or deactivates the trace messages of the SAP components.

location and name

Trace messages are saved on the application server in the log directory in files with the following names:

`jco_rfc_*.trc`

`dev_rfc*`

If the CPIC_TRACE environment variable is set to a value >0, the following file is created additionally:

`CPIC*`

 Caution - size

Files with the trace messages can quickly become extremely large; therefore the trace messages should only be activated for test purposes. By default, it is advisable to deactivate the trace messages.

values

The following values are available:

- 0 The trace messages are deactivated.
- 1 Error messages
- 2 Additional warnings
- 3 Additional information
- 4 Additional process trace
- 5 Additional extended process trace
- 6 Additional restricted data trace
- 7 Additional data trace with metadata
- 8 Additional complete data trace with metadata

validity

The RFC_TRACE parameter is evaluated for all destination types.

SEAL_TRACE Parameter

meaning

The SEAL_TRACE parameter activates or deactivates additional information in the log file and the creation of HTML files containing the transferred data from and to SAP.

values

The following values are available:

- 0 No additional information in the log file
- 1
  Sets the size of the log file: java.util.logging.FileHandler.level=ALL
  Also writes function list, environment variables to the log file
  Sets FTP_TRACE=2, HTTP_TRACE=2
  Starts sapftp/saphttp with -t
- 2
  Also writes HTML files with the transferred data
  Also writes TLS information for URLs with https to the log file
- 3
  Also writes Java memory status to the log file

Caution - number

HTML files with the transferred data can quickly become extremely large; therefore the creation of HTML files should only be activated for trouble-shooting. By default, it is advisable to set SEAL_TRACE to 0 or 1.

default

Default is 0.

validity

The SEAL_TRACE parameter is evaluated only for the following destination types:

- Type R, RFC server connections via SAP gateway

ABAP_DEBUG Parameter

meaning

The ABAP_DEBUG parameter activates or deactivates the debugging of system functions.

requirement

The USE_SAPGUI parameter must be set to 1 or 2 to enable the debugging.

values

The following values are available:

- 0
  The debugging of the system functions is deactivated. This is the standard case.
- 1
  The debugging of the system functions is activated for test purposes.

validity

The ABAP_DEBUG parameter is evaluated only for the following destination types:

- Type A, RFC client connections via application server
- Type B, RFC client connections via message server

USE_SAPGUI Parameter

meaning

The `USE_SAPGUI` parameter determines if SAP GUI is used for communication. In addition to that it specifies if the dialogs are minimized after the end of the function.

requirement

The following requirements have to be fulfilled:

- `USE_SAPGUI` must be set to `2` for the modules DMS Repro and DMS Scan.
- If `USE_SAPGUI` is set to `1` or `2` a dialog user must be specified in `cadrfc.ini`! A system user causes error.

values

The following values are available:

- `0`
  SAP GUI is not used for communication.
- `1`
  SAP GUI is used for communication. The dialogs remain open after the end of the function.
- `2`
  SAP GUI is used for communication. The dialogs are minimized after the end of the function.

validity

The `USE_SAPGUI` parameter is evaluated only for the following destination types:

- Type `A`, RFC client connections via application server
- Type `B`, RFC client connections via message server

UNICODE Parameter

meaning

The UNICODE parameter specifies if the SAP system is a Unicode or non-Unicode system.

By means of this parameter the correct version of the programs sapftp and saphttp is determined.

values

The following values are available:

- 0: The SAP system is a non-Unicode system.
- 1: The SAP system is a Unicode system.

code page

For Unicode systems, the coding can be specified with CODEPAGE:

→ *CODEPAGE Parameter*, Page 138

validity

The UNICODE parameter is evaluated only for the following destination types:

- Type A, RFC client connections via application server
- Type B, RFC client connections via message server
- Type R, RFC server connections via SAP gateway

CODEPAGE Parameter

meaning	The `CODEPAGE` parameter specifies the coding for Unicode systems.
interrelation	The code page is only evaluated in the case `UNICODE=1`.
default	The default code page is UTF-8. These is used in the following cases: • Neither a general code page is specified in `saprfc.ini` nor a specific code page at the function start. • The specified code page is not supported by Java. • The specified code page is UTF-* (* = 7, 16, 16BE, 16LE, 32, ...), because only UTF-8 is supported.
validity	The `CODEPAGE` parameter is only evaluated for the following destination types: • Type `R`, RFC server connections via SAP gateway

SNC_MODE Parameter

The `SNC_MODE` parameter activates and deactivates SNC (activation sign).	meaning
The following values are available: • `0`: SNC is deactivated. • `1`: SNC is activated.	values
Default is `0`.	default
The `SNC_MODE` parameter is evaluated for all destination types.	validity

SNC_MYNAME Parameter

meaning	The SNC_MYNAME parameter contains the name of the user who executes remote function calls.
default	The default name is the name which is determined by the security product for the current user.
format	The name must use the following format: `p:SEAL Server - DistinguishedName from SEALRFC.pse`
example	`p:CN=UniqueName, OU=Department, O=Company, C=CountryKey` `p:CN=SEALRFC, OU=SEALSAP, O=SEAL, C=DE`
validity	The SNC_MYNAME parameter is evaluated for all destination types.

SNC_PARTNERNAME Parameter

meaning

The SNC_PARTNERNAME parameter contains the name of the communication partner (application server).

format

The name must use the following format:

`p:SAP System - DistinguishedName`

example

`p:CN=UniqueName, OU=Department, O=Company, C=CountryKey`

`p:CN=SEALRFC, OU=SEALSAP, O=SEAL, C=DE`

validity

The SNC_PARTNERNAME parameter is evaluated only for the following destination types:

- Type A, RFC client connections via application server
- Type B, RFC client connections via message server

SNC_QOP Parameter

meaning	The SNC_QOP parameter specifies the protection level (quality of protection).
values	The following values are available: • 1: Use authentication only • 2: Use integrity protection (authentication) • 3: Use confidentiality protection (integrity and authentication) • 8: Use default values • 9: Use maximum protection
default	The default security level is 3 for RFC client connections via application server (TYPE=A) or message server (TYPE=B) and 8 for RFC server connections via SAP gateway (TYPE=R).
validity	The SNC_QOP parameter is evaluated for all destination types.

SNC_SSO Parameter

The SNC_SSO parameter activates or deactivates the single sign-on mechanism (SSO) of SNC. This allows to log on on the SAP ABAP backend with a user different form the SNC identity, while SNC is used for the encryption of the network communication. The parameter must be set for logon with user/password when SNC is activated. The parameter does not need to be set when logging on via certificate/PSE (SSO, single sign-on) in the SNC context. In this case, the default applies.	meaning
A minimum kernel patch of the SAP system is required, see SAP note 1701870.	requirement
The following values are available: • 0: SSO is deactivated. Instead, user/password is used for logon, for example. • 1: SSO is activated.	values
Default is 1.	default
The SNC_SSO parameter is evaluated for all destination types.	validity

ASCS Parameter

meaning	The ASCS(ABAP SAP Central Services) parameter contains the name of the section of type R with the data of the central SAP gateway.
requirement	The ASCS parameter is only supported together with static RFC destinations.
example	Section name of type R with data from the central SAP gateway. `ASCS=W74MSRFC`
validity	The ASCS parameter is evaluated only for the following destination types: • Type A, RFC client connections via application server (only on request) • Type B, RFC client connections via message server

X509CERT Parameter

The `X509CERT` parameter contains the certificate name if the logon with certificate (SSO, single sign-on) is activated.	meaning
The certificate must be in the Based64 file format, in one line without BEGIN and END sequence.	format
The default name is: `%SEAL_CUSTOMDIR%\server\sapserv\conf\sec\SEALRFC.crt`	default
The `X509CERT` parameter is evaluated for all destination types.	validity

8.3 rfcserver.cfg

introduction

This chapter contains the reference information about the `rfcserver.cfg` configuration file.

location

The `rfcserver.cfg` file is located in the following directory:

```
server\sapserv\conf\rfcserver.cfg
```

The directory and the file can be specified differently from the standard by the `RUNTIME_CONFIG` environment variable and the `-cfgfile` parameter.

 hint - gXnetplot

For gXnetplot, the file is located in:

```
servermenu - c - cdl - cd PDM
```

structure

The configuration file is divided into separate sections.

- → *[ACTIVE] Section*, Page 147
  `[ACTIVE]` is the first section. Only section names which are specified in the section `[ACTIVE]` are regarded at the start of the RFC destinations.
- → *[SECTIONNAME] Section*, Page 148
  The remaining sections with exception of `[ACTIVE]` can be named user-defined.

 example

Example for `rfcserver.cfg`:

```
[ACTIVE] SEALSAP1 = DEV SEALSAP1 = T6A SEALSAP2 = PROD

[DEV] DESTINATIONS = “X74RFC“ PROGID = SEAL-BCXDC-001 PROGID = SEAL-BCXDC-002

[T6A] DESTINATIONS = “T6ARFC“ PROGID = SEAL-BCXDC-001 3 RLIGATE = %PLS-DATA%\io\rligate

[PROD] DESTINATIONS = “W74RFC“ PROGID = SEAL-BCXDC-001 5 PROGID = SEAL-BCXDC-002 3 PROGID = SEAL-BCXDC-003 RLIGATE = %PLSDATA%\io\stargate XMS = 128 XMX = 512
OTFCODEPAGE = ISO-8859-1
```

comment character

`#` and `!` are supported as comment characters. Within one file, only one of these comment characters has to be used. A combination of `#` and `!` causes an error at the start.

SEAL Systems provides the `update_rfc_cfg.pl` script (directory: `install\rfc`) in order to replace `!` by `#`.

restart

A restart of JRFC Server is required after modifications in `rfcserver.cfg`.

[ACTIVE] Section

contents

This section lists the names of the sections containing the RFC destinations which are to be started.

exceptions

The `[ACTIVE]` section is ignored on these conditions:

1. The `SAP_START_SYSTEM` environment variable is specified. In this case the sections names specified in the environment variable are evaluated for the start of the RFC destinations.
2. Section names are specified as parameters when directly executing `rfcserverstart`. In this case, also the `SAP_START_SYSTEM` environment variable is ignored.

parameters

The following section describes the available parameters in the `[ACTIVE]` section.

ServerName or SECTION

This name refers to a section in `rfcserver.cfg` which contains further data.

The section may contain *ServerName* items or `SECTION` items. A mix of both is not allowed.

The section may contain multiple *ServerName* or `SECTION` items. If *ServerName* items exist, only those are considered whose name matches the current server name. If `SECTION` items exist, all items are considered. For each considered item, a JRFC Server main process is started. Threads related to this main process are started for all RFC destinations which are listed as `PROGID` in the associated section.

The value is a string.

- `PROD`
  Name of the section in `rfcserver.cfg` containing further data

Default: None

[SECTIONNAME] Section

contents

This section contains the relation to the system data in `saprfc.ini` and a list of the RFC destinations which are to be started with relation to their definitions in the SAP system via the `sm59` transaction.

advantage - SAP system data only once

The SAP system data must only be defined once in `saprfc.ini` even if several RFC destinations (`PROGID`) to one SAP system are started.

advantage - separate start/stop

For all RFC destinations of one section, a JRFC Server main process is started and stopped. The subdivision into individual sections allows the start of some groups of RFC destinations while other remain stopped.

advantage - separate log files

The name of the log files are composed of:

`ModuleName``.`*`SECTIONNAME_Number`*`.log`, for instance:

`rfcserver.PROD_0.log`

Thus the log files contain only messages of a group of RFC destinations. It is easier to keep track of the status of one RFC destination.

Unicode and non-Unicode

If you want to establish RFC destinations to Unicode and non-Unicode systems, you must specify these destination in separate sections. Collect all RFC destinations to non-Unicode systems in a separate section and all RFC destinations to Unicode systems in another section. In this case, one main process is executed for the non-Unicode systems and another main process is executed for the Unicode systems.

requirement

The `[`*`SECTIONNAME`*`]` section is only evaluated on the following conditions:

1. It exits as *`ServerName`* item with the current server name or as `SECTION` item in the `[ACTIVE]` section.
2. The section name is specified via the `SAP_START_SYSTEM` environment variable. In this case the `[ACTIVE]` section is ignored.
3. It is specified as parameter when directly executing `rfcserver_start`. In this case the`SAP_START_SYSTEM` environment variable and the `[ACTIVE]` section are ignored.

parameters

The following section describes the available parameters in the `[`*`SECTIONNAME`*`]` section.

`DESTINATIONS`

This name refers to `saprfc.ini` to determine the SAP system data.

The value is a string:

- `W74RFC`
  Name of the system as specified in `saprfc.ini` at `DEST`

Default: None

[SECTIONNAME] Section, Continuation

OTFCODEPAGE

`OTFCODEPAGE` specifies the code page for downloading OTF data..

The value is a string:

- `ISO-8859-2`
  Code page supported by Java, see `https://docs.oracle.com/javase/8/docs/technotes/guides/intl/encoding.doc.html`

Default: `iso-8859-1`

PROGID

`PROGID` is the unique identifier of the RFC destination as specified at `Program ID` in the SAP system in the `sm59` transaction.

This name refers to the data in `sm59` in the SAP system. The SAP system data is determined via `DESTINATIONS`.

The section may contain multiple `PROGID` items. After each `PROGID` item the thread number, which are started from the JRFC Server main process as a maximum, can be optionally specified to allow a parallel processing.

If no `PROGID` item exists, the `PROGID` setting in `saprfc.ini` is used. If there also no item exists, no RFC destination is started.

It is advisable to use the server name as name component to get a unique identifier if multiple OM servers are to establish RFC destinations to the same SAP system.

 naming convention

The identifier can include letters, numbers, '+', '.', '-', and '_' characters!

The value is a string:

- `SEAL-ALFILECHECK-`*`ShortCutNumber`*
  ArchiveLink FileChecker, `alfileche-cker.cfg`
- `SEAL-ALVIEWS-`*`ShortCutNumber`*
  ArchiveLink View Server, `alviewserver.cfg`
- `SEAL-BCXDC-`*`ShortCutNumber`*
  BC-XDC-interface, `rfcserver.cfg`
- `SEAL-CONNC-`*`ShortCutNumber`*
  Output via JRFC Server, `rfcserver.cfg`, or RFC Server, `rfcserver.cfg`
- `SEAL-DPF4C-`*`ShortCutNumber`*
  Conversion Server and PDF Longlife Suite SAP Integration - conversion/validation of files already checked-in,`convservdpf.cfg`
- `SEAL-PDFLLS-`*`ShortCutNumber`*
  PDF Longlife Suite SAP-Integration - check-in,`filecheck.cfg`
- `SEAL-VIEWS-`*`ShortCutNumber`*
  DMS View Server,`dvsviewserver.cfg`

Default: None

RLIGATE

`RLIGATE` specifies the directory for the repro list processing. It is passed at the start of JRFC Serverwith the call parameter `-rligate`.

[SECTIONNAME] Section, Continuation

The value is a string:

- `%PLSDATA%\io\stargate`
  A final `\` or `/` is automatically added if it is missing.

Default: `%PLSDATA%\io\stargate`

`XMS`

`XMS` specifies the minimum memory at start.

The value is an integer with possible values: `0`, `64`, `128`, `256`, `512`, `1024`

- `0`
  No memory parameter is passed.
- `512`
  Minimum memory at start in MB

Default: `64`

`XMX`

`XMX` specifies the maximum memory.

 hint - recommendation:

Specify a three times larger size than the maximum file size, which is to be processed, as value.

The value is an integer with possible values: `0`, `64`, `128`, `256`, `512`, `1024`

- `0`
  No memory parameter is passed.
- `1024`
  Maximum memory in MB

Default: `256`

9 Configuration Tables - Reference

This chapter contains an alphabetically sorted list of all configuration tables relevant for the basis configuration in SAP.

introduction

Each configuration table and its fields is described in a separate section.

description

When displaying the value help for certain fields, it is possible to display the internal key of the value in addition to the description text.

internal key

This is how you display the internal keys:

display key

Step	Action
1	Click  in the icon bar of SAP GUIor press `Alt+F12` alternatively and select the `Options` item.
2	Select on the left: `Interaction Design→Visualization 1`
3	Activate in the `Controls` section: `Show keys within dropdown lists`

After modifications of the configuration tables, the end application must be restarted.

 Caution - re-start required

This chapter deals with the following topics:

in this chapter

→ */seal/bas_cr113 - Static Destination for RFC Server*, Page 152

→ */seal/bas_cr114 ´- Static Destination for RFC Client*, Page 153

→ */seal/bas_cr142 - Define Parameters*, Page 155

/seal/bas_cr113 - Static Destination for RFC Server

description

This configuration table specifies the static destinations for RFC server.

open the table

This is how you open the table:

Step	Action
1	Click  at: `Basis Configuration` →`Static Destinations` →`Create Static Destination for RFC Server` (table: `/seal/bas_cr113`)

overview

The table has the following parameters:

Static Destinations for RFC Server

Server	SAPFTP
SEAL-CONNC-OKX74-002	SEAL-SRV-SAPFTP

Server

`Server` is the RFC server for which the static sapftp destination is to be used.

The value is a string:

- `SEAL-CONNC-001`

Default: None

SAPFTP

`SAPFTP` is the name of the template for static sapftp destinations.

The value is a string:

- `SEAL-CLT-SAPFTP`
  Static sapftp destination for RFC client
- `SEAL-SRV-SAPFTP`
  Static sapftp destination for RFC server

Default: None

/seal/bas_cr114 ´- Static Destination for RFC Client

This configuration table specifies the static destinations for RFC clients. description

This is how you open the table: open the table

Step	Action
1	Click  at: `Basis Configuration` →`Static Destinations` →`Create Static Destination for RFC Client` (table: `/seal/bas_cr114`)

The table has the following parameters: overview

Static destinations for RFC Client

User Name	SAPHTTP	SAPFTP	Number
	SEAL-CLT-SAPHTTP	SEAL-CLT-SAPFTP	3

The item is only evaluated if the user currently logged-on and the specified user match. `User Name`

If no user is specified, it is called a global setting. This is valid for all users for whom no specific configurations exist.

Regarding the meaning when assigning the template for static sapftp/saphttp connections for RFC client (Table: `/seal/bas_cr114`): static sapftp/saphttp

User with which the RFC client logs on to SAP from `cadrfc.ini`

In regard to the meaning in the access table (table: `/seal/out_cr029`): access table

The following tables display only defaults of the selected user.

The value is a string:

- `SEAL1`
  User Name

Default: None

`SAPHTTP` is the name of the template for static saphttp destinations. SAPHTTP

The value is a string:

- `SEAL-CLT-SAPHTTP`
  Static saphttp destination for RFC client

Default: None

`SAPFTP` is the name of the template for static sapftp destinations. SAPFTP

/seal/bas_cr114 ´- Static Destination for RFC Client,

Continuation

The value is a string:

- `SEAL-CLT-SAPFTP`
  Static sapftp destination for RFC client
- `SEAL-SRV-SAPFTP`
  Static sapftp destination for RFC server

Default: None

`Number`

`Number` specifies the maximum number of static sapftp/saphttp destinations to be started.

The value is an integer:

- `3`

Default: `1`

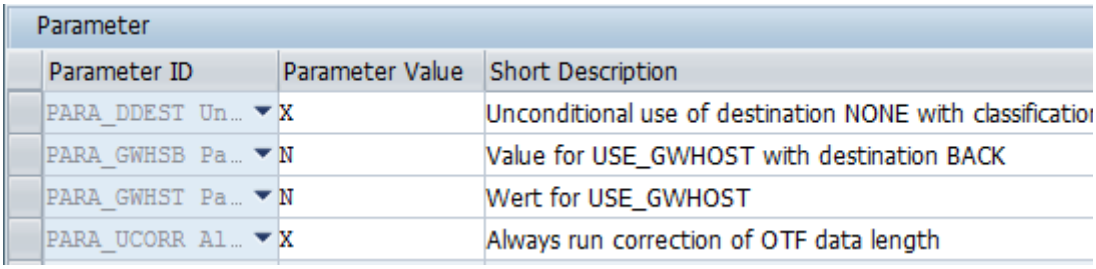
/seal/bas_cr142 - Define Parameters

General parameters are specified. description

This is how you open the table: open the table

Step	Action
1	Start the transaction: `/n/seal/img`
2	Click  at `Basis Configuration` →`Define Parameters` (table: `/seal/bas_cr142`)

The table has the following parameters: overview



`Parameter ID` defines the parameter for which a value is to be specified. The possible values at `Parameter Value` depend on the value of `Parameter ID`. `Parameter ID`

Values of the following enumeration are supported:

- `PARA_DDEST Unconditional Use of Destination NONE with Classification`
  The `DESTINATION` parameter value is specified for the characteristic evaluation.
- `PARA_GWHSB Parameter Value of USE_GWHOST at BACK`
  The parameter value of `USE_GWHOST` is specified for RFC client connections (destination `BACK`).
- `PARA_GWHST Parameter Value of USE_GWHOST`
  The parameter value of `USE_GWHOST` is specified for RFC server connections.
- `PARA_NSWAP No byte swapping in OTF data with endianness mismatch`
  Swapping bytes in bitmap data on bigendian Unicode systems is suppressed.
- `PARA_UCORR Correction in OTF Data Streams`
  The execution of the length correction in ST commands of OTF data streams is specified.

Default: None

/seal/bas_cr142 - Define Parameters, Continuation

Parameter Value

Parameter Value specifies the value of the parameter. The possible values depend on the value of Parameter ID.

Values of the following enumeration and any strings are supported depending on the value of Parameter ID:

PARA_DDEST as Parameter ID:

- empty (default)
  Destination NONE is only used for selected transactions (mm02, cv0*, Conversion Server) at the characteristic evaluation in order to accelerate the data evaluation.
- X or Y
  Destination NONEis used at all transactions at the characteristic evaluation.

PARA_GWHSB as Parameter ID:

- Y
  Set USE_GWHOST for RFC client connections to Y.
- N (default)
  Set USE_GWHOST for RFC client connections to N.

PARA_GWHST as Parameter ID:

- Y
  Set USE_GWHOST to Y.
- N (default)
  Set USE_GWHOST to N.

PARA_UCORR as Parameter ID:

- empty (default)
  Consider configuration referred to SAP note 944778 at the length correction of OTF data streams.
- X
  Always execute the length correction in ST commands of OTF data streams.

Default: See above

Short Description

Short Description contains an explanation of the parameter.

The value is a string:

- Purpose

Default: None

10 Changes

This chapter describes the most important changes for each released module version.

This chapter deals with the following topics: in this chapter

Changes with Release 1.4.2

logging

The values for `SEAL_TRACE` in the `saprfc.ini` file have been extended with the following functionality:

- `0` No additional information in the log file
- `1`
  Sets the level of the log file: `java.util.logging.FileHandler.level=ALL`
  Also writes function list, environment variables, connection status to the log file
  Sets `FTP_TRACE=2`, `HTTP_TRACE=2`
  Start sapftp/saphttp with `-t`
- `2`
  Writes additional HTML files with the transferred data
  Writes additional TLS information for URLs with https to the log file
- `3`
  Also writes Java memory status to the log file

sapftp/saphttp with SNC

In addition to the dynamic RFC destinations, static RFC destinations are also supported. Static RFC destinations offer the following advantages:

- SNC is supported for sapftp/saphttp
- Static RFC destinations can be explicitly enabled/restricted at the gateway.

central SAP gateway (ASCS)

The static RFC destination supports the use of a central SAP gateway (ASCS). This offers the following advantages:

- SNC is supported for sapftp/saphttp
- Connections via local gateways can be blocked, which increases security.

Changes with Release 1.4.1

The activation of DPF via REST additionally supports the following options: REST interface

- Logon with basic authentication
  (no support of SAP logon tickets)
- Proxy
- SSL
- HTTP 1.1
- Compression
- HTTP cookies

Changes with Release 1.4.0

SAProuter	The connection via SAProuter is supported with the `SAPROUTER` parameter (configuration file: `saprfc.ini`).

Changes with Release 1.3.5

DLL

The current version of `sapftp`/`saphttp` requires the following DLLs in the `tools\bin_xxx` directory: `sapnwrfccm.dll` (non-Unicode) and `sapnwrfc.dll` (Unicode).

Changes with Release 1.3.3

RFC destination

In the supplied configuration files for the RFC connections, for example `rfcserver.cfg`, all RFC connections are inactive. The start of the desired RFC destinations has to be activated explicitly.

Changes with Release 1.3.2

wrapper program

If Unicode SAP systems are used only (no mixed operation with non-Unicode SAP systems, determined via `UNICODE` in `saprfc.ini`), `sysinit` automatically copies the Unicode version of `sapftp` and `saphttp` from SAP to `sapftp` and `saphttp` and replaces the wrapper program from SEAL Systems.

The `sapftp`/`saphttp` wrapper programs are delivered with digital signature.

Changes with Release 1.3.1

`cadrfc.ini`	As of JRFC Server version 2.0.2 or newer, the logon information for the initial logon is no longer required for RFC server connections in `cadrfc.ini` (this information was required for JRFC Server as of version 2.0.1).
logging	The logging of additional information, for instance memory usage and the creation of HTML files containing the transferred data from and to SAP can be activated via the `SEAL_TRACE` parameter in the `saprfc.ini` file.

Changes with Release 1.3.0

The description of the integration via Web service has been inserted to the common basis documentation. Web service

Changes with Release 1.2.8

`cadrfc.ini`

As of JRFC Server 2.0.1 or newer, `cadrfc.ini` has to contain logon information for the initial logon also for RFC server connections. The data has to be located in a system-dependent section `[SAP\SystemName]` which specifies the client as `CadRfcClient` item. A client-dependent section will not be considered!

Changes with Release 1.2.6

The `sap_conn_checker` program is provided to evaluate SAP system data. `sap_conn_checker`

Changes with Release 1.2.5

start depending on the current server

The RFC destinations which are to be started can be specified within the configuration file like `rfcserver.cfg` depending on the current server.

Changes with Release 1.2.4

The `UNICODE` parameter in the `saprfc.ini` file is evaluated at the start of RFC connections. Dependent in this value the Unicode or non-Unicode variant of the `sapftp` and `saphttp` program is started. The wrapper programs `sapftp` and `saphttp` from SEAL Systems are used to determine the correct variant. It is not necessary to copy files.	Unicode
The SAP system data is specified only once in `saprfc.ini` for each SAP system. Multiple RFC connections are defined via `PROGID` items in the configuration file, for example `rfcserver.cfg`.	SAP system data
RFC destinations can be grouped within the configuration file like `rfcserver.cfg`. Each of these groups can be started and stopped individually.	start in groups
The required RFC connections for start, status request or stop can be specified via the environment variable `SAP_START_SYSTEM`.	start with environment variable
The required RFC connections for start, status request or stop can be specified via call parameters. This is not supported by `plsstart` and `dvsstart`.	start with call parameter

Bibliography

[SAP_BASECONF_SAP_TEC]
SAP Basis Configuration - SAP, System Description, SEAL Systems

[SAP_BASECONF_SNC_TEC]
SAP Basis Configuration - SNC and RFC, System Description, SEAL Systems

Terminology

The following section explains the most important terms that are used in this documentation. Terms identified by → refer to other terms in this section.

Customizing

Configuring the SAP system

cv04, cv04n

→Transaction to start the search function in SAP DMS

cv<xx>, cv<xx>n

Transaction used for document search and management within the context of document management in SAP DMS (refer to the SAP online documentation)

Document

→Document information record

Document management system

In the document management system (short: DMS), the →document information records and their assigned files are managed.

Document information record

A master record in the →DMS containing management data for a document and original files. To each document information record, a identification key is assigned, consisting of four partial keys: type, number, part and version.

Developer license

Key giving an SAP user development authorizations

Client

A unit within an SAP system that is independent with regard to action, organization, and data functions. It also has separate master records in a table within the SAP system

Naming convention

Agreement regarding name assignment for development purposes, for example all developments in a system should be given a prefix of X, Y, or Z as these prefixes have been reserved for customers by SAP.

Namespace

Protected namespaces can be reserved upon request by SAP. The namespace for SEAL Systems is `/seal/`, `/sealc/` (for customer development) or `/dvsrepro/` (old).

Transaction

Means of executing programs

Transaction code

Identifier naming a →transaction in the SAP system

Transport

Exporting/importing data between SAP systems

Abbreviations

ABAP
: Advanced Business Application Programming (SAP system programming language)

DPF®
: Digital Process Factory® from SEAL Systems

DMS
: Document management system

FTP
: File Transfer Protocol

OSS
: Online Support Service

PDF
: Adobe Portable Document Format

PDF/A
: Adobe Portable Document Format (PDF/A norm)

PLM
: Product Lifecycle Management

PLOSSYS®
: Product family from SEAL Systems

RFC
: Remote Function Call

Keywords

Symbols

A

B

C

D

F

G

H

J

K

L

M

O

P

R

S

T

U

X

Z

Index

H

J

L

M

N

O

P

R

S

T

U

V

W