Net-Work SAF Security Administration Guide

Administration

Version 6.4.1

October 2018

ADABAS & NATURAL

This document applies to Net-Work SAF Security Version 6.4.1 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: WAF-DOC-641-20181011

Table of Contents

Preface

This document provides information for administrators responsible for configuring and running Entire Net-Work SAF Security once Entire Net-Work is installed.

Note: Entire Net-Work SAF Security is provided in the Software AG product option called the Entire Net-Work SAF Security (product code WAF), which is an add-on to the Entire Net-Work product and must be ordered separately.

The Entire Net-Work SAF Security documentation is organized as follows:

SAF Security Overview	Provides an overview of Entire Net-Work SAF Security and its features.
SAF Security Prerequisites	Lists the prerequisites to using Entire Net-Work SAF Security.
Installing NETSAF	Describes the steps to install Entire Net-Work SAF Security (NETSAF).
Securing Entire Net-Work	Describes how to secure Entire Net-Work using Entire Net-Work SAF Security. Command and user ID security as well as resource profile definitions are described. In addition, steps are provided for securing Entire Net-Work with RACF, CA-ACF2, and CA-Top Secret.
SAF Security Operator Commands	Lists and describes the operator commands available with Entire Net-Work SAF Security.
SAF Security Messages and Codes	Lists and describes the messages associated with Entire Net-Work SAF Security.

1 Conventions

Notation "*vr* SP *s*", *vrs*, or *vr*: When used in this documentation, the notation "*vr* SP *s*", *vrs*, or *vr* stands for the relevant version, release, and system maintenance level numbers. For further information on product versions, see *version* in the *Glossary*.

This document covers the following topics:

Syntax Conventions

The following table describes the conventions used in syntax diagrams of Entire Net-Work statements.

Convention	Description	Example
uppercase, bold	Syntax elements appearing in uppercase and bold font are keywords. When specified, these keywords must be entered exactly as shown.	**DRIVER TCPI [DRVCHAR =** *driver-char* **| #]** The syntax elements DRIVER, TCPI, and DRVCHAR are Entire Net-Work keywords.
lowercase, italic, normal font	Syntax elements appearing in lowercase and normal, italic font identify items that you must supply.	**DRIVER TCPI [DRVCHAR =** *driver-char* **| #]** The syntax element *driver-char* identifies and describes the kind of value you must supply. In this instance, you must supply the special character used to designate that an operator command is directed to the TCP/IP line driver, rather than to a specific link.
underlining	Underlining is used for two purposes: 1. To identify default values, wherever appropriate. Otherwise, the defaults are explained in the accompanying parameter descriptions. 2. To identify the short form of a keyword.	**DRIVER TCPI [DRVCHAR =** *driver-char* **| #]** In the example above, # is the default that will be used for the DRVCHAR parameter if no other record buffer length is specified. Also in the example above, the short version of the DRVCHAR parameter is D.

Convention	Description	Example
vertical bars (|)	Vertical bars are used to separate mutually exclusive choices. **Note:** In more complex syntax involving the use of large brackets or braces, mutually exclusive choices are stacked instead.	DRIVER TCPI API = { BS2 | CNS | EZA | HPS | OES } In the example above, you must select BS2, CNS, EZA, HPS, or OES for the API parameter. There are no defaults.
brackets ([ ])	Brackets are used to identify optional elements. When multiple elements are stacked or separated by vertical bars within brackets, only one of the elements may be supplied.	DRIVER TCPI [DRVCHAR = *driver-char* | #] In this example, the DRVCHAR parameter is optional.
braces ({ })	Braces are used to identify required elements. When multiple elements are stacked or separated by vertical bars within braces, one and only one of the elements must be supplied.	DRIVER TCPI API = { BS2 | CNS | EZA | HPS | OES } In this example, one of the following values is required for the API parameter: BS2, CNS, EZA, HPS, or OES.
other punctuation and symbols	All other punctuation and symbols must be entered exactly as shown.	LINK *linkname* TCPI [NETADDR = *n1.n2.n3.n4*] [,][-] In this example, the periods must be specified in the IP address. In addition, options must be separated by commas and dashes should be used as needed to indicate that parameter settings continue on the next line.

Syntax Rules

The following rules apply when specifying Entire Net-Work parameter statements:

- Each Entire Net-Work parameter statement occupies positions 1 - 72 of at least one line.
- The statement type (NODE, LINK, TRANSDEF, or DRIVER) must be specified as the first non-blank item on the statement.
- The node name, driver name, translation definition function, or link name follows the statement type, separated by at least one blank (space).

- Keyword parameters may be specified following either the node name on NODE statements or the driver name on DRIVER and LINK statements. Keyword parameters are separated from their arguments by an equal (=) sign, and from other keyword parameters by at least one blank (space) or a comma (,).
- When the acceptable values for a parameter are Y and N (yes and no), any other value is treated as an N, unless there is a documented default, and processing continues without any warning.
- When the acceptable values for a parameter fall within a range (e.g., 1 - 2147483647) and a value outside the range is specified, the value is automatically reset to the maximum value within the range, unless documented otherwise for the parameter. Processing continues without any warning.
- A statement can be continued beginning in any column of the next line by specifying a dash (-) as the last nonblank character in any column of the current line, before column 73.
- Comment lines begin with an asterisk (*) in position 1 and can be inserted anywhere in the statement sequence.
- Some keywords may require a list of subparameters separated by commas; the list must be enclosed in parentheses ( ) unless only the first subparameter is to be entered. Omitted ("defaulted") subparameters must be represented by placeholder commas if subsequent parameters are to be entered. The following are examples of correct subparameter strings:

```
KEYWORD=(value1,value2,value3)
KEYWORD=(value1,,value3)
KEYWORD=(,,value3)
KEYWORD=(,value2)
KEYWORD=value1
```

- Hexadecimal keyword values can be entered by prefixing the value with an "X". For example:

```
LINK . . . ADJID=X0064, . . .
```

2 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *`folder.subfolder.service`*, APIs, Java classes, methods, properties.
`Italic`	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at **http://documentation.softwareag.com**. The site requires credentials for Software AG's Product Support site Empower. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at **https://empower.softwareag.com/**.

You can find product information on the Software AG Empower Product Support website at **https://empower.softwareag.com**.

To submit feature/enhancement requests, get information about product availability, and download products, go to **Products**.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the **Knowledge Center**.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at **https://empower.softwareag.com/public_directory.asp** and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at **http://techcommunity.softwareag.com**. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

3 SAF Security Overview

As the trend towards distributed computing increases, both across platforms within a company and between partners, security of data has become ever more critical. The Entire Net-Work SAF Security Interface (NETSAF) addresses this issue by providing point-of-access verification of incoming requests.

NETSAF (product code WAF) is a separate, optional product for z/OS environments. It allows Entire Net-Work clients to access SAF data sources. Validation is carried out against the SAF repository, thus maximizing the investment that most z/OS sites have made in their mainframe security repository.

The following features minimize the overhead required for the administration, operation, and execution of a mainframe security system:

- Categorized Access

  For example, all access from mainframe clients can be verified against the same security profile.

- Locality of Security Checking

  NETSAF can be activated on a link by link basis. For example, an installation may have several Entire Net-Work nodes, of which only one communicates externally. Security checking can be activated for that node alone and only for the external links.

4 SAF Security Prerequisites

The prerequisites for the Entire Net-Work SAF Security Interface (NETSAF) are as follows:

- z/OS, running the SAF security system
- the current version of Entire Net-Work for Mainframes

  In order for NETSAF to function correctly, the Entire Net-Work started task must run from APF authorized libraries. In addition, it must run under a user ID defined with sufficient authority to invoke the RACROUTE functions TYPE=AUTH and TYPE=VERIFY and to make third-party checks on behalf of other users.

5 Installing NETSAF

This chapter describes installation topics for NETSAF.

Installation Media

The Entire Net-Work SAF Security Interface (NETSAF) installation media contains the following libraries:

Note: The complete list of libraries provided with Entire Net-Work can be found in the full Entire Net-Work documentation.

Library	Contents
WAF*vrs*.JOBS	The sample z/OS job library for Entire Net-Work SAF Security. The *vrs* in the library name represents the *version* of Entire Net-Work SAF Security. Jobs SAGI010, SAGI020, and SAGI021, which are used to assemble the NA2PPRM, NA2PSEC, and NA2POS modules, can be found here.
WAF*vrs*.LOAD	The z/OS load library for Entire Net-Work SAF Security. The *vrs* in the library name represents the *version* of Entire Net-Work SAF Security. The NETSAF and WAFNUC load modules, which will be copied into one of the Entire Net-Work STEP libraries, can be found here.
WAF*vrs*.SRCE	The z/OS source library for Entire Net-Work SAF Security. The *vrs* in the library name represents the *version* of Entire Net-Work SAF Security. Modules NA2PPRM, NA2PSEC, and NA2POS, with the associated macros and the sample parameter module WAFPARM can be found here.

Installation Procedure

A sample job is provided for each step in the installation procedure. Sample jobs are contained in the WAF*vrs*.JOBS library.

Step 1. Unload the Installation Libraries

Use IEBCOPY to unload the libraries from the NETSAF installation media.

Step 2. Assemble the System Parameter Module

Assemble the system parameter module NA2PPRM to define the required installation options. You may use the sample job SAGI010. The different parameters are summarized below:

General Parameters

The following general parameters influence the operation of the Entire Net-Work SAF Security Interface:

Parameter	Description
GWSIZE = *nnn*	Buffer size K (512 bytes per user)
GWMSGL = {1,2,3}	Message level: 1 = security violations only 2 = successful security checks only 3 = complete trace of authorization activity
GWSTYP = {1,2,3,4}	Security repository type: 1 = RACF 2 = TOP SECRET 3 = ACF2 4 = MSP RACF

NA2PPRM Parameters

The following parameters are contained in the NA2PPRM macro that is supplied with NETSAF. These parameters are used to secure requests received by Entire Net-Work.

Parameter	Description
NWUNI = {Y|N}	Allow access to undefined resources
NWCLASS = ADASEC	Name of general resource class or type
NWFLEN = {0|1|2}	DBID/Fnr character string: 0 = 3 digit resource profiles with leading zeros 1 = 5 digit resource profiles with leading zeros 2 = 3 and 5 digit resource profiles, no leading zeros
NWCPUID = *xxxxxx*	CPU ID of the machine considered local
NWSUPER = {N|Y}	LPARs are considered local access
NWUIDH = *x*	User ID determination (mainframe)
NWUIDU = *x*	User ID determination (UNIX)

Parameter	Description
NWUIDW = *x*	User ID determination (Windows)
FAILMODE = {F|W}	Reaction to violation: F = Failure W = Warning

Step 3. Assemble the RACROUTE Module

The SAF (RACROUTE) macros used by Entire Net-Work SAF Security Interface must be at the same version as those used at your site. You may use the sample job SAGI020 to assemble NA2PSEC, the module containing these macros.

The parameter STY should be assigned one of the following values: RACF, TSS, or ACF2

Ensure that the REL parameter is set correctly:

- For RACF, it should be set to the correct RACF version number.
- For CA-TOP SECRET and ACF2, it should be set to the corresponding value for the equivalent level of RACF and not the version of ACF2 or TOP SECRET itself. For example: REL=2.2

Step 4. Assemble the Operating System Services Module

You may use the sample job SAGI021 to assemble operating system services module NA2POS.

Installation Verification

You have now installed the Entire Net-Work SAF Security Interface (NETSAF).

To verify the installation, activate Entire Net-Work SAF Security Interface on a test node link with FAILMODE initially set to 'W'. When a database access call is received on that link, the following message will be displayed in the console output of the Entire Net-Work job:

SEFM210* SAF GATEWAY IS ACTIVE FOR ENTIRE NET-WORK

Various diagnostic messages will follow, depending on the message level specified for the GWMSGL parameter when the parameter module was assembled in **Step 2** of the NETSAF installation procedure.

6 Securing Entire Net-Work

This section describes the facilities for securing Entire Net-Work calls to data sources. The targets that can be selectively restricted to include Adabas SQL Server, EntireX Communicator, and Entire System Server.

Command Protection

Targets on the host are secured by defining resource profiles representing each Adabas database, Adabas SQL Server, or EntireX Communicator target. For Adabas, resource profiles can be defined at the file level. The command type determines which access level is required for successful authorization. All read and find commands are considered as READ access. Commands requiring UPDATE access are AMEND, ERASE, and INSERT.

NETSAF recognizes three categories of Adabas direct call commands:

1. Data access commands (Lx, Sx and HI)
2. Data update commands (Ax, Ex and Nx)
3. The following Transaction data commands (Commands Which Access or Create ET Data):
   - RE commands with Option 1 set to A or I need read access;
   - OP commands with Command Option 2 set to E need read access;
   - ET, CL, and C3 commands with Command Option 2 set to E need update access.

Note: NETSAF authorization calls are not performed when running BT Transaction Data Commands and the Special Commands C1, C5, HI, RC and RI.

Derivation of User ID

User ID derivation is of primary importance. Because security checks are based on trusted user IDs, no password verification is normally performed. However, the user ID must exist in the security repository. In some cases, the user ID is previously authenticated in the caller's home environment or the user ID is fixed by, for example, the Entire Net-Work configuration.

This section explains the choices for user ID derivation. A user's identity can be lost if calls are routed through an intermediate gateway node.

Windows Clients

The NWUIDW system parameter is used to specify the source of the user ID for database calls originating from a Windows client:

```
NWUIDW = { 2 | 3 | 5 }
```

Value	Description
2	The user ID is derived from a defined Entire Net-Work Link name.
3	The user ID is derived from a defined Entire Net-Work Node name. This is the default value.
5	All accesses use the common user ID (defined NWUSRW=*name*, where "WINUSER" is the default value).

UNIX Clients

The NWUIDU system parameter is used to specify the source of the user ID for database calls originating from a UNIX client:

```
NWUIDU = { 2 | 3 | 5 }
```

Value	Description
2	The user ID is derived from a defined Entire Net-Work Link name.
3	The user ID is derived from a defined Entire Net-Work Node name. This is the default value.
5	All accesses use the common user ID (defined NWUSRU=*name*, where "UNIXUSR" is the default value).

Mainframe Clients

The NWUIDH system parameter is used to specify the source of the user ID for database calls originating from a mainframe client:

```
NWUIDH = { 2 | 5 | 6 | 7 }
```

Value	Description
2	The user ID is derived from the originating job name.
5	All accesses use the common user ID (defined NWUSRH=*name*, where "HOSTUSR" is the default value). This is the default parameter value.
6	The user ID is the CPU ID of the calling machine.
7	The user ID is derived from the local security system if ADASAF or ADAESI is installed.

Defining Resource Profiles

In order to secure Entire Net-Work, it is necessary to define resource profiles denoting DBID and FNR in the SAF repository. Valid access levels are READ, UPDATE, and CONTROL. CONTROL applies to SYSAOS commands, for example.

The options defined in the system parameter module during the NETSAF installation procedure determine the actual look of the resource profiles. For example, the NWFLEN parameter determines whether leading zeros are included or not. Resources are defined using upper case characters only.

The following table contains examples of resource profile definitions:

Example	Explanation
CMD195.FIL002	This three digit resource profile protects file 2 of database 195 and includes leading zeros.
CMD00019.FIL01053	This five digit resource profile protects file 1053 of database 19 and includes leading zeros.
CMD19.FIL1053	This is the same five digit resource profile with leading zeros omitted.
CMD01202	This five digit resource profile protects ESQ node 1202 and includes leading zeros.

Note: If fixed-length Database IDs and file numbers are used in the resource profile names (that is, the NWFLEN parameter specifies 0 or 1). File number 00000 (NWFLEN=1) or 000 (NWFLEN=0) is checked for the relevant database. RE commands need read access; OP commands with Command Option 2 set to E need read access; ET, CL, and C3 commands with Command Option 2 set to E need update access.

- NWFLEN Parameter
- NWUNI Parameter
- NWSUPER Parameter
- NWCPUID Parameter

- NWCLASS Parameter

NWFLEN Parameter

```
NWFLEN = { 0 | 1 | 2 }
```

Resource profile definitions are based on a character string containing the DBID and, optionally, the FNR associated with a command. The character string may be constructed from digits that include leading zeros.

The NWFLEN system parameter is used to specify whether leading zeros are included or not. If zeros are included, either three- or five-digit numbers are used, depending on whether the DBID or the FNR can exceed 255.

Value	Description
0	3-digit resource profiles with leading zeros; this is the default value.
1	5-digit resource profiles with leading zeros.
2	3- and 5-digit resource profiles with no leading zeros.

Commands routed to the Adabas SQL Server automatically generate resource checks based on the five digit DBID. Security definitions protecting Adabas SQL Server itself must always be supplied in this format. When leading zeros are suppressed, commands with no file number are not validated. These commands include OP, CL, RC, RE, ET and BT.

NWUNI Parameter

```
NWUNI = { Y | N }
```

The NWUNI system parameter is used to allow access to resources that are not defined to the security system. Normally, access to undefined resources is prevented. Profiles representing Entire Net-Work targets are added to the security repository with either a default access or by granting access to specific users and groups. NWUNI=N is the default value.

Note: This option does not allow access to resources that are defined with universal access "none".

NWSUPER Parameter

```
NWSUPER = { Y | N }
```

The NWSUPER system parameter is used to cause LPARs to be regarded as local access. This option is useful when the host machine is partitioned into two or more LPARs. Commands originating from a different LPAR on the same physical machine can be considered as coming from the host and therefore not subject to authorization checks. NWSUPER=N is the default value.

NWCPUID Parameter

```
NWCPUID = cpu-id
```

It is sometimes advantageous to regard a particular mainframe as "trusted". Authorization checks are not performed for commands originating from the trusted computer. The NWCPUID system parameter is used to specify the CPU ID of the trusted computer.

NWCLASS Parameter

```
NWCLASS = { resource-name | ADASEC }
```

Each SAF based security system provides the facilities required for maintaining resource profiles. RACF enables the grouping of similar resource profiles into a resource class. CA-Top Secret and CA-ACF2 provide resource types which give equivalent functionality.

The NWCLASS system parameter is used to specify the name of the resource class or type used when performing authorization checks for Entire Net-Work. ADASEC is the default value. The maximum length of a resource name is 17 characters.

Defining Resources to RACF

This section explains how to add resource definitions to RACF. For details about the procedures to be followed, refer to the relevant IBM manuals.

Step 1. Add Classes to Class Descriptor Table

Add resource classes to the RACF Class descriptor table. Refer to the IBM SPL RACF manual. An example is given in IBM SYS1.SAMPLIB, member RACTABLE.

Classes must be allocated the maximum lengths as described above and be defined for discrete and generic profile use. Other attributes control the level of RACF logging and SMF recording when executing RACROUTE calls. Sample definitions are provided in member RACTABLE.

Step 2. Update z/OS Router Table

Update the z/OS router table. Refer to the IBM SPL RACF manual. An example is given in IBM SYS1.SAMPLIB, member RACTABLE, section RFTABLE.

Step 3. Activate New Classes

Activate new resource classes with SETROPTS. Refer to the *IBM RACF Command Language Reference*.

For example, activate class ADASEC:

```
SETROPTS CLASSACT(ADASEC)
SETROPTS GENCMD(ADASEC)
SETROPTS GENERIC(ADASEC)
```

Step 4. Assign a User ID for the Started Task

NETSAF is run as a started task or batch job. It requires a user ID that has the relevant RACF authorizations including the ability to perform RACROUTE, TYPE=EXTRACT calls on profiles belonging to the resource classes activated in Step 3 above.

Step 5. Permit Users Access to Resource Profiles

Add profiles to protect the different resources and permit users the required level of access. The following RACF commands add resource profile CMD00001.FIL01234 and grant READ access to the userid "DAN":

```
RDEFINE ADASEC CMD00001.FIL01234 UACC(NONE)
PERMIT CMD00001.FIL01234 CLASS(ADASEC) ACCESS(READ) ID(DAN)
```

Defining Resources to CA-ACF2

This section explains how to define resources to CA-ACF2. For details about the procedures to be followed, refer to the relevant CA-ACF2 manuals.

Step 1. Add Task User ID to CA-ACF2

NETSAF normally executes as a z/OS started task. Define the user ID of the started task to CA-ACF2 with the following attributes:

```
MUSASS,STC
```

Step 2. Insert the SAFDEF Record

The SAFDEF record must be inserted as follows:

```
SAFDEF FUNCRET(4) FUNCRSN(0) ID(SAFGWAY) MODE(GLOBAL)
RACROUTE(REQUEST=AUTH SUBSYS=ADARUN- REQSTOR=- )
RETCODE(4) ↩
```

Step 3. Insert the CLASMAP Record

Define a three-character CA-ACF2 resource type code for the general resource class name used by Entire Net-Work SAF Security Interface:

```
CLASMAP
ENTITYLN(0) MUSID() RESOURCE(ADASEC) RSRCTYPE(ADA) ↩
```

Step 4. Define Resource Rules

Define the required security profiles in CA-ACF2 using the new type code. In the following example, the resource CMD00001.FIL01234 is added and user ID "DAN" is given READ access:

```
$KEY(CMD00001.FIL01234) TYPE(ADA) UID(DAN) ALLOW SERVICE(READ)
```

Defining Resources to CA-Top Secret

This section explains how to define resources to CA-Top Secret. For details about the procedures to be followed, refer to the relevant CA-Top Secret documentation.

Step 1. Assign Additional CA-Top Secret Facilities

Entire Net-Work SAF Security Interface issues authorization checks against specific CA-Top Secret facilities. By default, these facilities are batch and STC.

Additional facilities can be defined by modifying pre-defined models. For example, a facility can be defined that enables development and production environments to be secured separately.

The following attributes are important and should be assigned when modifying facilities for Entire Net-Work SAF Security Interface:

```
NAME=fac,AUTHINIT,MULTIUSER,NONPWR,PGM=ADA,NOABEND
```

Step 2. Assign a User ID for the Started Task

Add one user ID for each instance of the Entire Net-Work started task. If required, different facilities can be assigned to development and production tasks.

The designated facility is assigned to the started task user ID:

```
TSS CRE(userid) DEPT(dept) MASTFAC(fac)
```

Step 3. Add a Procedure Name for the Started Task

The procedure name under which the Entire Net-Work started task executes must be defined to CA-Top Secret. Different procedure names are suggested when securing different environments separately with the use of non-default CA-Top Secret facilities:

```
TSS ADD(STC) PROC(proc) USER(userid)
```

Step 4. Add Resource Type to Resource Definition Table

Resource types must be added to the CA-Top Secret resource definition table (RDT). Resource definitions relating to Entire Net-Work SAF Security Interface are kept in resource type ADASEC. For a detailed explanation of the following command and arguments, refer to the *CA-Top Secret Reference Guide*:

```
TSS ADD(RDT) RESCLASS(ADASEC)
RESCODE(HEXCODE)
ATTR(LONG)
ACLST(NONE,READ,CONTROL)
DEFACC(NONE)
```

Step 5. Assign Ownership of Resources

Assign ownership to each resource. This must be done before granting access to defined resource profiles. In the following example, ownership of resource CMD00001.FIL01234 is assigned to user ID "DAN".

```
TSS ADD(DAN) ADASEC(CMD00001.FIL01234)
```

Step 6. Grant Access to Defined Resources

Grant users access to resource profiles. In the following example, user ID "ELLEN" is granted READ access to an EntireX Communicator service. This enables the user to execute as a client, issuing requests to the EntireX Communicator service:

```
TSS PER(ELLEN) ADASEC(CMD00001.FIL01234) FAC(fac)
ACCESS(READ)
```

The FAC (facility) argument can be omitted if Entire Net-Work SAF Security Interface operates under the Master facility Batch or STC.

7 SAF Security Operator Commands

The z/OS Modify (F) operator command can be used to communicate with the Entire Net-Work SAF Security Interface. The format is as follows:

```
F NETWORK,SAF command
```

The following table describes the available commands:

Command	Description
* SREST	Perform a general restart of the Entire Net-Work SAF Security Interface component. This ensures that all data held in the Entire Net-Work SAF Security Interface buffer and all data held by the security system itself in the Entire Net-Work SAF Security Interface address space is flushed. The operation is transparent to all online and batch users.
* SSTAT	Display general statistics on the operator console for the Entire Net-Work SAF Security Interface. These statistics are the same as those available using the Online Services from Natural.
* SUSERS	Display a list of all active users known by the Entire Net-Work SAF Security Interface.
* SUSTAT *userid*	Display statistics for a specified user. These statistics are the same as those available from the Natural Online Services.
* SSNAP *hhhhhhhh*	Display a selected portion of the Entire Net-Work SAF Security Interface storage. Operation is not terminated.
* SHELP	Display all possible operator commands.
* SNEWCOPY	Like the SREST command, perform a general restart. In addition, reload the parameter and server modules. This is useful if a parameter needs to be modified without stopping Entire Net-Work.

8 SEFM* - ADASAF SAF Interface and SAF Security Kernel Messages

ADASAF displays an eight-byte code containing various return codes from SAF. This information is shown in a number of messages denoted *ssssssss*.

The ADASAF return code "ssssssss" contains the following structure:

Position	Information Content	
Byte: 1	SAF return code	
Byte: 2	Function code. ADASAF internal function codes (hex) include:	
	04	Authorize Adabas access
	44 or 6C	Authenticate user
Byte: 3	Return code from security system, for example RACF	
Byte: 4	Reason code from security system, for example RACF	
Bytes: 5 - 8	SAF reason code	

Refer to the IBM manual External Security Interface (RACROUTE) Macro Reference manual for z/OS for a thorough explanation of all possible return/reason codes. CA-Top Secret and CA-ACF2 can provide different return code values in some circumstances.

BLS0334 **SYMBOL 'NETSAF' CANNOT BE FOUND. LOADING ABORTED**

Explanation This message should be ignored.

SEFM001 **{ssssssss} : {user} : {resource}**

Explanation The security system determined that the user identified in the message (*user*) does not have authorization for the resource listed in the message (*resource*). System return and reason codes are given in the hexadecimal string *ssssssss*. This message is displayed when access has been denied to a particular resource.

SEFM002 ***{XX} to request FF : {user} : {resource}**

Explanation An unexpected response code (*XX*) was received from the SAF Security Kernel for the user identified in the message (*user*) when requesting function *FF* to be performed on the resource specified in the message (*resource*).

SEFM004 ***Natural programs not extracted**

Explanation The SAF Security Kernel was not able to extract a list of protected program objects from the security system on behalf of Natural users.

Action Obtain a trace of SAF call RACROUTE EXTRACT from the security system and contact your Software AG technical support representative. ACF2 and Top Secret users should ensure that the protected programs have been extracted from the security system and supplied to the SAF Security Kernel via the SEFEXT DD statement in the daemon started task JCL.

SEFM006 ***ADARSP {XX}({xx}) to request FF : {user}**

Explanation The SAF Security Kernel returned the Adabas response code (*XX*) and subcode (*xx*) shown in the message to request *FF* for the user shown in the message (*user*).

Action Ensure that the SAF Kernel started task is active. Check its output for error messages. Take the necessary remedial action indicated by the Adabas response code.

SEFM008 ***SAF Gateway (V{v.r}) started**
*** SAF Security Kernel (V{x.x.x} - BUILD {xxxx}) started**

Explanation Entire Net-Work SAF Security Interface (ADASAF) startup completed or the SAF Security Kernel initialized successfully.

Action No action is required for this informational message.

SEFM009 **Module {module-name} not loaded**

Explanation Entire Net-Work SAF Security Interface could not load the module listed in the message (*module-name*).

Action Ensure that the module is in the STEPLIB and that the region size is sufficient.

SEFM013 ***Less {memory | storage} acquired than specified**

Explanation The SAF Security Kernel or the Entire Net-Work SAF Security Interface (ADASAF) were not able to allocate all the memory or storage required to satisfy the buffer size specified in its parameters. Operation continues.

Action Ensure that the region size is sufficient and the parameters are appropriate.

SEFM014 ***No {memory | storage}could be acquired**

Explanation Entire Net-Work SAF Security Kernel or the SAF Security Interface (ADASAF) could obtain no storag or memory at system startup. Operation has terminated.

Operation has terminated.

Action Ensure that the region size is sufficient and system parameters are appropriate.

SEFM015 ***Logic error - {XXXX} for request FF : {user}**

Explanation The SAF Security Kernel suffered an internal error. A general restart is performed and the operation continues.

Action Keep all information written to DDPRINT and contact your Software AG technical support representative.

SEFM016	***SAF logoff failed {ssssssss} ACEE AAAA : {user}**
Explanation	The SAF Security Kernel was unable to logoff *user* from the security system. The SAF error code is *ssssssss*.
Action	Contact your Software AG technical support representative.

SEFM017	***Insufficient space to initialize - make Natural buffer {XX}**
Explanation	The Natural SAF interface requires a larger value to be specidied for NSFSIZE.
Action	Increase the Natural NSFSIZE parameter.

SEFM020	***GETMAIN failed / IDSIZE error**
Explanation	The Natural SAF interface could not acquire storage from the designated IDMSBUF.
Action	Increase Natural region and/or thread size.

SEFM021	***Illegal storage use / relocation problem**
Explanation	Internal problem in Natural SAF storage use.
Action	Contact your Software AG technical support representative.

SEFM025	***Natural IDMSBUF parameter is not defined**
Explanation	The Natural NSFSIZE parameter has not been specified.
Action	Ensure NSFSIZE is set correctly in the Natural parameters.

SEFM026	***Natural protected programs not extracted code: {XX}**
Explanation	The list of protected programs could not be returned from the SAF Security Kernel to Natural.
Action	Ensure the same copy of the configuration module SAFCFG is used by all system components. Check that the GWSTYP parameter defined in SAFI010 and STY parameter in SAFI020 are both correctly set for the installed security system and that all installation requirements have been met.

SEFM028	***System files not found in environment table**
Explanation	The current Natural system files were not matched in the table defining all possible system file sets.
Action	Ensure that the environment definitions in Natural Security are correct.

SEFM029 ***Error in communications layer - check installation procedure**

Explanation Possible reasons for error: Adabas link module installed into this component is not reentrant.

SEFM030 ***SQL table / VIEview could not be identified for file ({XX},{YY})**

Explanation Interface could not identify table name for DBID/FNR of an SQL request.

Action Ensure interface is correctly installed, then contact your Software AG technical support representative.

SEFM031 ***DBID / FNR identified with SQL request not recognized {XXXX}**

Explanation Interface component could not determine the DBID/FNR associated with this SQL request.

Action Contact your Software AG technical support representative.

SEFM041 ***Interface installed for Net-work**

Explanation The interface is installed for operation with Entire Net-Work.

Action No action is required for this informational message.

SEFM049 ***User type T not permitted by installed options**

Explanation The SAF Kernel will not permit user type *T* to operate using the currently installed options.

SEFM050 ***Error writing SMF record : {XX}**

Explanation The stated error occurred when an SMF record was being written.

SEFM051 ***SAFPRINT dataset not defined, DDPRINT will be used**

Explanation `SAFPRINT=Y` is set in SAFCFG, but no SAFPRINT dataset is defined.

SEFM205 ***CPU identity : {cpuid}**

Explanation The interface component linked to Entire Net-Work displays the CPU ID of the host machine.

Action No action is required for this informational message.

SEFM210	***SAF Gateway is active for Entire Net-Work**
Explanation	The Entire Net-Work SAF Security Interface is active.
Action	No action is required for this informational message.

SEFM255	***Unauthorized use of request**
Explanation	Attempted illegal use of security request.
Action	Contact your Software AG technical support representative.

Operator Command Messages (SEFM900+ Series) Adabas SAF Securityoperator command messages SAF Security Kerneloperator command messages

The following messages are displayed in response to operator commands:

SEFM900	*** Operator issued command: {command}**
Explanation	Entire Net-Work SAF Security Interface (ADASAF) or the SAF Security Kernel received the operator command identified in the message.
Action	No action is required for this informational message.

SEFM901	*** SAF server - General statistics (at {hhhhhhhh})** *** SAF Security Kernel - General statistics (at {hhhhhhhh})**
Explanation	The operator command for general statistics was issued. Here is an example of the statistics messages produced for the SAF server:

```
SEFM901 * SAF SERVER - GENERAL STATISTICS (AT hhhhhhhh)
SEFM902 * RESOURCE   CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES ↩
LEN
SEFM903 * APPLICATION         1          0          0           ↩
0    8
SEFM903 * ADABAS              0          0          0           ↩
0   32
SEFM903 * SYSMAIN             0          0          0           ↩
0   13
SEFM903 * SYSTEM FILE         2          0          0           ↩
0   24
SEFM903 * PROGRAM             0          0          0           ↩
0   17
SEFM903 * BROKER              0          0          0           ↩
0   32
SEFM903 * NET-WORK            0          0          0           ↩
0    0
SEFM903 * SQL SERVER          0          0          0           ↩
0    0
SEFM904 * USERS - ACTIVE: 1 FREE: 55 OVEWRITES: 0
```

Here is an example of the statistics messages produced for the SAF Security Kernel:. The address in the first line is the address of the SAF Kernel's storage cache.

```
SEFM901 * SAF SECURITY KERNEL - SERVER STATISTICS (AT 12C47000)
SEFM902 * RESOURCE   CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES ↩
LEN
SEFM903 * APPLICATION        10          0          0           ↩
0    8
SEFM903 * DBMS CHECK          0          0          0           ↩
0   17
SEFM903 * SYSMAIN             0          0          0           ↩
0   21
SEFM903 * SYSTEM FILE         2          0          0           ↩
0   40
SEFM903 * PROGRAM             0          0          0           ↩
0   17
SEFM903 * BROKER              0          0          0           ↩
0   68
SEFM903 * NET-WORK            0          0          0           ↩
0   17
SEFM903 * SQL SERVER          0          0          0           ↩
0   32
SEFM904 * CACHED USERS:       1 HIGH WATERMARK:     1 MAX USERS: ↩
 5545
SEFM905 * OVERWRITES:         0 AUTHENTICATED:      0 DENIED:    ↩
    0
```

Action No action is required for this informational message.

SEFM902 - 905 **{statistics}**

Explanation Various statistics for the SAF server and the SAF Security Kernel are displayed. See message **SEFM901**.

Action No action is required for this informational message.

SEFM909 *** {SAF Gateway | SAF Security Kernel} - shutdown initiated**

Explanation The operator issued a command to shut down Entire Net-Work SAF Security Interface or the daemon started task (SAF Security Kernel). This message is also issued when a secure Adabas nucleus, Net-Work node or Adabas SQL server terminates.

Action No action is required for this informational message.

SEFM910 ***{SAF Server | SAF Security Kernel} - list all active users**

Explanation The operator issued a command to display a list of currently active users.

The following is a sample of the output produced for the SAF server:

```
SEFM910 * SAF SERVER - LIST ALL ACTIVE USERS
SEFM911 * USERID     CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES ↩
BUFF
SEFM912 * K11079              3          0          0           ↩
0   0
```

The following is a sample of the output produced for the SAF Security Kernel:

```
SEFM910 * SAF GATEWAY - LIST ALL ACTIVE USERS
SEFM911 * USERID CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES BUFF
SEFM912 * K11079         3          0          0           0   0
```

Action No action is required for this informational message.

SEFM911 ***{userid} . . .**

Explanation The operator issued a command to display statistics specific to a currently active user.

The following is a sample of the output produced for the SAF server:

```
SEFM911 * NXB        CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES ↩
 BUFF
SEFM912 * APPLICATION         1          0          0           ↩
0    0
SEFM912 * DBMS CHECK          0          0          0           ↩
0    0
SEFM912 * SYSMAIN             0          0          0           ↩
0    0
SEFM912 * SYSTEM FILE         2          0          0           ↩
0    0
SEFM912 * PROGRAM             0          0          0           ↩
0    0
SEFM912 * BROKER              0          0          0           ↩
0    0
SEFM912 * NET-WORK            0          0          0           ↩
0    0
SEFM912 * SQL SERVER          0          0          0           ↩
0    0
```

The following is a sample of the output produced for the SAF Security Kernel:

```
SEFM911 * SJU        CHECK(+VE) CHECH(-VE) CHECK SAVED OVERWRITES ↩
 BUFF
SEFM912 * APPLICATION        10          0           0          ↩
 0   10
SEFM912 * DBMS CHECK          0          0           0          ↩
 0    0
SEFM912 * SYSMAIN             0          0           0          ↩
 0    0
SEFM912 * SYSTEM FILE         2          0           0          ↩
 0    2
SEFM912 * PROGRAM             0          0           0          ↩
 0    0
SEFM912 * BROKER              0          0           0          ↩
 0    0
SEFM912 * NET-WORK            0          0           0          ↩
 0    0
SEFM912 * SQL SERVER          0          0           0          ↩
 0    0
```

Action No action is required for this informational message.

SEFM913	*** No active users found in SAF {Server | Gateway | Security Kernel}**
Explanation	No active users were found in Entire Net-Work SAF Security Interface (ADASAF) or in the SAF Security Kernel.
Action	No action is required for this informational message.

SEFM914	*** Requested user {userid} not found in SAF {Server | Gateway | Security Kernel}**
Explanation	The requested user was not found in the Entire Net-Work SAF Security Interface (ADASAF) or in the SAF Security Kernel.
Action	No action is required for this informational message.

SEFM915	**SEFM915 * SAF Security Kernel - snap of server memory**
Explanation	This message is issued in response to an `SSNAP` operator command and is followed by a sequence of SEFM916 messages.
Action	No action is required for this informational message.

SEFM916	*** {hhhhhhhh hhhhhhhh hhhhhhh hhhhhhhh hhhhhhhh.x..X.Y}/**
Explanation	This message contains the results of an `SSNAP` command. Each SSNAP snaps up to 256 bytes and shows the address, the hexadecimal storage contents, and the interpretation.
Action	No action is required for this informational message.

SEFM918	*** Supplied address is outside of legal range**
Explanation	An attempt was made to snap storage outside the bounds of the SAF Kernel's cache.

SEFM919	***Operator command did not contain required arguments**
Explanation	A required parameter was omitted from an operator command. For example, `SUSTAT` with no userid specified.
Action	Correct the operator command and try again.

SEFM920	**SSNAP, SSTAT, SUSERS, SUSTAT, SREST, SNEWCOPY, SLOGOFF, STRACE**
Explanation	This message is issued in response to an `SHELP` operator command and lists the valid SAF Kernel operator commands.
Action	No action is required for this informational message.

SEFM921 *** Memory allocation failure - users cannot be logged off**

Explanation The SAF Kernel was unable to obtain temporary storage (approximately 16Kb) to log users off in response to an `SREST`, `SNEWCOPY` or `SLOGOFF` operator command.

Action Increase the region size.

SEFM922 *** User {userid} logged off**

Explanation This message is issued in response to an `SLOGOFF` operator command. The indicated user has been logged off from the security system.

Action No action is required for this informational message.

SEFM923 *** User {userid} not logged off - user not found**

Explanation This message is issued in response to an `SLOGOFF` operator command. The requested user could not be found in the SAF Kernel's cache.

Action Verify the correct user ID was specified.

SEFM924 *** User {userid} not logged off - return code {ZZ}**

Explanation This message is issued in response to an `SLOGOFF` operator command. An internal error (indicated by ZZ) occurred while attempting to log the reqeusted user off.

Action Evaluate the return code to determine the cause of the error.

SEFM928 *** Invalid trace setting - must be 0, 1, 2 or 3**

Explanation The `STRACE` operator command was issued with an invalid trace setting.

Action Correct the trace setting and try again.

SEFM929 *** Invalid SAF Security Kernel operator command**

Explanation An invalid SAF Security Kernel operator command was entered.

Action Specify a valid SAF Security Kernel operator command.

Index

R

S

U

V

W