Natural

SAF Security Kernel

Version 9.1.1

April 2019

ADABAS & NATURAL

Dieses Dokument gilt für Natural ab Version 9.1.1.

Hierin enthaltene Beschreibungen unterliegen Änderungen und Ergänzungen, die in nachfolgenden Release Notes oder Neuausgaben bekanntgegeben werden.

Dokument-ID: SAK-DOC-911-20211014

Table of Contents

Preface v
1 About this Documentation 1
Dokumentationskonventionen 2
Online-Informationen und Support 2
Datenschutz 4
2 Introduction 5
Architecture 6
Related Documentation 6
Password Phrases 7
3 Installation 9
Prerequisites 10
Preparing for Installation 10
Authorization 10
Modes of Operation 10
Installation Datasets 11
Installation Procedure 12
Embedded SAF Security Kernel 14
Installing the SAF Security Daemon 14
Daemon Configuration 14
4 Operator Commands 17
5 SAF* - SAF Daemon Messages 19
6 SEFM* - ADASAF SAF Interface and SAF Security Kernel Messages 23
Operator Command Messages (SEFM900+ Series) Adabas SAF Securityoperator command messages SAF Security Kerneloperator command messages 28
7 SAF Return Codes 35
8 SAF Internal Function Codes 37
9 Interpreting Trace Messages 39
10 Security Definitions 41
Defining Resources to RACF 42
Defining Resources to CA-TOP SECRET 43
Defining Resources to ACF2 45
Index 49

Preface

This document describes the SAF Security Kernel and its associated daemon. It covers installation and operation of the kernel and daemon and messages and codes issued by them. The SAF Security Kernel and Daemon are distributed on the Adabas Limited Libraries (product code WAL).

Introduction	Provides an overview of the SAF Security Kernel functionality.
Installation	Describes how to install the SAF Security Kernel.
Operator Commands	Explains the available operator commands for the SAF Security Kernel.
SAF - SAF Daemon Messages*	Describes the SAF daemon messages.
SEFM - ADASAF SAF Interface and SAF Security Kernel Messages*	Describes Adabas SAF Security (ADASAF) and SAF Security Kernel operator console and command messages.
SAF Return Codes	Describes SAF return codes.
SAF Internal Function Codes	Describes SAF internal function codes.
Interpreting Trace Messages	Describes how to interpret SAF trace messages
Security Definitions	Provides a general overview of the definition of resources to RACF, CA-Top Secret and CA-ACF2.

1 About this Documentation

- Dokumentationskonventionen 2
- Online-Informationen und Support 2
- Datenschutz 4

Dokumentationskonventionen

Konvention	Beschreibung
Fettschrift	>Kennzeichnet Elemente auf einem Bildschirm.
`Nichtproportionale Schrift`	Kennzeichnet Namen und Orte von Diensten im Format *`Ordner.Unterordner.Dienst`*, Programmierschnittstellen (APIs), Namen von Klassen, Methoden und Properties in Java.
`Kursivschrift`	Kennzeichnet: Variablen, für die Sie situations- oder umgebungsspezifische Werte angeben müssen. Neue Begriffe, wenn sie erstmals im Text auftreten. Verweise auf andere Dokumentationsquellen.
`Nichtproportionale Schrift`	Kennzeichnet: Text, den Sie eingeben müssen. Meldungen, die vom System angezeigt werden. Programmcode.
{ }	Zeigt eine Reihe von Auswahlmöglichkeiten an, von denen Sie eine auswählen müssen. Geben Sie nur die innerhalb der geschweiften Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole { } ein.
|	Trennt zwei sich gegenseitig ausschließende Auswahlmöglichkeiten in einer Syntaxzeile voneinander ab. Geben Sie eine der Auswahlmöglichkeiten ein. Geben Sie nicht das Symbol | ein.
[ ]	Zeigt eine oder mehrere Optionen an. Geben Sie nur die innerhalb der eckigen Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole [ ] ein.
...	Zeigt an, dass Sie mehrere Auswahlmöglichkeiten desselben Typs eingeben können. Geben Sie nur die Informationen ein. Geben Sie nicht die drei Auslassungspunkte (...) ein.

Online-Informationen und Support

Dokumentationswebsite der Software AG

Sie finden die Dokumentation zu den Produkten der Software AG auf der Dokumentationswebsite der Software AG unter **https://documentation.softwareag.com**.

Empower, die Produktsupportwebsite der Software AG

Falls Sie noch kein Benutzerkonto für Empower haben, können Sie eine E-Mail an empower@softwareag.com senden. Geben Sie darin Ihren Namen, den Namen Ihrer Firma und deren E-Mail-Adresse an und beantragen Sie die Einrichtung eines Benutzerkontos.

Wenn Sie ein Benutzerkonto erhalten haben, können Sie den eService-Bereich von Empower unter **https://empower.softwareag.com/** aufrufen und dort Support-Fälle online öffnen.

Informationen zu Software AG-Produkten finden Sie auf der Empower-Produktsupportwebsite unter **https://empower.softwareag.com**.

Unter **Products & Documentation** können Sie Anträge bezüglich Produktmerkmalen und Produktverbesserungen einreichen, Informationen über die Verfügbarkeit von Produkten abrufen und Produkte herunterladen.

Im **Knowledge Center** finden Sie Informationen zu Programmkorrekturen (Fixes) und frühzeitige Warnungen, technische Abhandlungen (Papers) und Artikel aus der Wissensdatenbank.

Wenn Sie noch Fragen haben und telefonisch mit uns Kontakt aufnehmen möchten, können Sie im Kontaktverzeichnis des Globalen Supports unter **https://empower.softwareag.com/public_directory.aspx** eine der dort für Ihr Land angegebenen örtlichen oder gebührenfreien Telefonnummern auswählen.

Software AG TECHcommunity

Auf der Website der Software AG TECHcommunity unter **http://techcommunity.softwareag.com** finden Sie Dokumentationen und andere technische Informationen.

- Sie können auf Produktdokumentationen zugreifen, wenn Sie die erforderlichen Authentifizierungsdaten für die TECHcommunity haben. Andernfalls müssen Sie sich registrieren und "Documentation" als Interessengebiet angeben.
- Sie erhalten Zugang zu Artikeln, Code-Beispielen, Demos und Lernprogrammen.
- Sie können an von Software AG-Experten moderierten Online-Diskussionsforen teilnehmen, um Fragen zu stellen, über bewährte Methoden und Prozesse (Best Practices) zu diskutieren und zu erfahren, wie andere Kunden die Technologien der Software AG nutzen.
- Sie können Links auf externe Websites benutzen, die sich mit offenen Standards und Web-Technologien befassen.

Datenschutz

Die Produkte der Software AG stellen Funktionen zur Verarbeitung von personenbezogenen Daten gemäß der Datenschutz-Grundverordnung (DSGVO) der Europäischen Union zur Verfügung. Gegebenenfalls sind in der betreffenden Systemverwaltungsdokumentation entsprechende Schritte dokumentiert.

2 Introduction

- Architecture 6
- Related Documentation 6
- Password Phrases 7

The System Authorization Facility (SAF) is used by z/OS and compatible sites to provide rigorous control of the resources available to a user or group of users. Security packages such as RACF, CA-ACF2, and CA-Top Secret allow the system administrator to

- maintain user identification credentials such as user ID and password; and
- establish profiles determining the datasets, storage volumes, transactions, and reports available to a user.

The resulting security repository and the infrastructure to administer it represent a significant investment. At the same time, the volume of critical information held by a business is constantly growing, as is the number of users referencing the data. The challenge of controlling these ever-increasing accesses requires a solution that is flexible, easy to implement and, above all, one that safeguards the company's investment.

The SAF Security Kernel acts as an agent for other Software AG products such as Adabas, Natural, and Entire Net-Work. It allows them to secure resources via a SAF-compliant security system, thus enhancing the scope of the security system to enable:

- a single control and audit system for all resources
- a single definition of userids and passwords
- industry standard protection of resources such as Adabas data and Natural libraries
- maximized return on investment in the security repository

Architecture

A SAF security solution comprises two separate components:

- a product-specific component which is distributed and installed with the product being protected (Adabas, Natural, Entire Net-Work or EntireX)
- a product-independent SAF Security Kernel (the subject of this document) which may be embedded in an authorized product or operate as a separate authorized daemon

Related Documentation

For details on securing specific products such as the following, refer to the relevant product documentation:

- Adabas SAF Security
- Natural SAF Security
- Entire Net-Work

- EntireX Security

Some of these products are distributed with a copy of the SAF kernel. The individual product documentation indicates if this is the case.

Password Phrases

The SAF Security Kernel provides password phrase support with Adabas Limited (WAL) Library Version 8.3.4 or above.

For information on which specific products can take advantage of this feature, refer to the relevant product documentation:

- Adabas SAF Security
- Natural SAF Security
- Entire Net-Work
- EntireX Security

3 Installation

- Prerequisites 10
- Preparing for Installation 10
- Authorization 10
- Modes of Operation 10
- Installation Datasets 11
- Installation Procedure 12
- Embedded SAF Security Kernel 14
- Installing the SAF Security Daemon 14
- Daemon Configuration 14

This section describes how to install the SAF Security Kernel.

Prerequisites

The following are prerequisites:

- z/OS
- SAF-compliant security system
- For Adabas SAF Security (AAF) installations, the SAF Security Kernel supplied in the shared Adabas 8.2 SP4 library (WAL 8.2 SP4 patch level 2, or WAL824P002) requires Adabas SAF Security 8.2 SP2 or later; it is not compatible with previous versions of Adabas SAF Security.

Preparing for Installation

Before installing the SAF Security Kernel, review all possible configuration options for the kernel itself and for the product(s) it will secure.

If the kernel will execute as a daemon, in its own address space, allocate a unique node number to it.

Authorization

The kernel load library and any other step libraries in the kernel's loading environment must be APF authorized.

Modes of Operation

The kernel may be embedded with a product (that is, it may run in the same address space). This is the case for Adabas and Entire Net-Work. To implement this mode of operation, you simply need to add the kernel load library (and any load libraries used as the target of installation assembly and link jobs) to the step library concatenation, ensuring that they are APF authorized.

For products other than Adabas and Entire Net-Work, the kernel operates under a daemon, in its own address space as a target in the Software AG network. This mode of operation is described in more detail below.

For both modes of operation, the SAF Security Kernel must run under a defined user ID. This user ID must have sufficient authority to invoke the AUTH, VERIFY, and EXTRACT functions of RACROUTE and to issue third-party checks on behalf of all users.

Installation Datasets

The SAF Security Kernel is supplied as a component of the Adabas Limited Libraries

WALvrs.LOAD

WAL*vrs*.LOAD is a standard load library containing modules needed to operate the SAF Security Kernel.

This library must be APF-authorized and available on the loading environment of any job that uses the SAF Security Kernel. Jobs that include the SAF Security Kernel are:

- The SAF Security Daemon, used by Natural SAF Security and EntireX Security
- Adabas nuclei protected by Adabas SAF Security
- Entire Net-Work nodes protected by Entire Net-Work SAF Security

The WAL*vrs*.LOAD modules for SAF Security all have names beginning with SAF.

WALvrs.SRCE

WAL*vrs*.SRCE is a standard source library containing Assembler macros (names beginning NA2M) and source books (SAFCFG, SAFPOS and SAFPSEC) which must be assembled as part of the SAF Security Kernel installation. There are also several example members:

SAFAEXT	CA-ACF2 extract for Natural RPC and program protection
SAFRCLSN	RACF class definitions for Natural SAF Security
SAFRCLSX	RACF class definitions for EntireX Security
SAFTEXT	CA-Top Secret extract for Natural RPC and program protection
SAFDDCAR	Daemon DDCARD input
SAFPARMS	Sample SAFCFG

WALvrs.JOBS

WAL*vrs*.JOBS is a standard source library containing example jobs for installing the SAF Security Kernel. These examples have names beginning SAF.

Installation Procedure

This section describes how to install the SAF Security Kernel.

Step 1 Assemble the Configuration Mode

The configuration module defines the required installation options. Only general options are described here. For information about product-specific options, see the relevant product documentation. A sample job is provided in SAFI010 in the jobs library.

The configuration module is created by assembling a source member similar to the SAFPARMS member supplied on the source library. This source member invokes the SAFCFG macro, (also supplied on the source library), specifying your site-specific options and requirements. The SAF Security Kernel uses the settings in SAFCFG to determine:

- Which resources are protected for which products
- Security classes to be used for resource checking
- How resource profile names are constructed
- Caching requirements

The resulting load module, SAFCFG, must be available to any job that includes the SAF Security Kernel and, in the case of EntireX, to the jobs being secured. You may decide to maintain different SAFCFG modules for different secured products. However, it is critical that the daemon use exactly the same configuration module as EntireX jobs secured by that daemon.

Set the following parameters to the appropriate values:

GWDBID=nnnnn	Node ID of SAF server
GWSIZE=nnnnn	Buffer size in K (approximately 512 bytes per user)
GWMSGL={0, 1 ,2,3}	Message level
GWSTYP={ 1 ,2,3}	Security repository type
SAFPRINT={ N ,Y}	Write trace messages to DDPRINT (N) or SAFPRINT (Y)

Message level indicates which diagnostic messages will be written to DDPRINT or SAFPRINT:

1 (the default)	only security violations are traced
2	only successful checks are traced
3	all checks are traced
0	tracing is suppressed

Security repository type identifies the SAF security system in use:

1 (the default)	RACF
2	CA-Top Secret
3	CA-ACF2

SAFPRINT specifies where security check trace messages should be written:

N (the default)	DDPRINT
Y	SAFPRINT

If you specify Y, but do not provide a SAFPRINT dataset, the trace messages will be written to DDPRINT. The SAFPRINT dataset must be defined in the JCL and may refer to a SYSOUT dataset or to a file defined with `RECFM=F` (or FB) and `LRECL=121`.

Step 2 Assemble the RACROUTE Macros

The SAF Security Kernel requires the same version of the RACROUTE macros as used at the customer site. Sample job SAFI020 is provided to assemble the module containing these macros.

Before running SAFI020, set the parameter `STY` to RACF, TSS or ACF2 as appropriate. The `REL` parameter specifies the RACROUTE macro `RELEASE` parameter used by SAFPSEC. Unless advised otherwise, specify `REL=2.1` (the default).

Note: For password phrase support, the SAFPSEC from WAL8.3.4 or above must be used and the REL= parameter must be set to 7730 or above. This is a pre-requisite for password phrase support in the RACROUTE macro.

The resulting load module, SAFPSEC, must be available to any job that includes the SAF Security Kernel.

Step 3 Assemble the Operating System Services Module

Sample job SAFI021 is provided to assemble the operating system services module, SAFPOS. The resulting load module, SAFPMAC, must be available to any job that includes the SAF Security Kernel.

Embedded SAF Security Kernel

For those products (Adabas and Entire Net-Work) that use an embedded SAF Security Kernel, you need only add the load library containing the kernel (SAFKRN) and the three load modules created above to the step library concatenation.

Installing the SAF Security Daemon

For those products (Natural and EntireX) that need a SAF Security Kernel running in a separate, authorized address space, you must install a SAF Security Daemon.

The SAF Security Daemon runs in its own address space, using Adabas modules to establish inter-process communication. It signs on to the Adabas SVC as a target and is therefore accessible in the same way as an Adabas database. Consequently, the SAF Security Daemon (and its Kernel) can be accessed remotely, via Entire Net-Work.

Software AG recommends that you run the SAF Security Daemon as a started task, although it may be run as a batch job. The SAF Security Daemon must run APF-authorized, therefore all step libraries must be APF-authorized.

Additionally, the SAF Security Daemon must run under a userid with sufficient authority to invoke the RACROUTE AUTH, EXTRACT and VERIFY functions and to make third-party checks on behalf of other users.

Sample JCL to execute the daemon is provided in SAFI024 in the jobs library.

Daemon Configuration

The daemon is configured by parameter input. The parameters are read from the DDCARD dataset at startup. An example dataset is provided in SAFDDCAR in the source library. Following is a description of valid parameters, with default value and meaning.

Parameter	Default	Meaning
NODE	None	Identifies this SAF Security Daemon. Must be a number between 1 and 65535 and must be unique among all targets.
PRODUCT	None	Defines which products are available in this server. Specify SAF.
FORCE	NO	Defines whether or not an existing ID table entry for the same node should be overwritten. Valid values are YES and NO. Specify YES only when advised to by Software AG.
LOCAL	NO	Defines whether or not this server is to be accessible from remote users, via Entire Net-Work. Valid values are YES (the server is not accessible) and NO (the server is accessible).
NC	20	Defines the maximum number of concurrent requests that can be processed by the server. Specify a number between 1 and 32767. If a request to the server fails with response code 151 (ADARSP151), increase NC.
NABS	16	Defines the number of 4K storage blocks to be used for transmitting information between clients and the server. Specify a number between 1 and 32767. If a request to the server fails with response code 255 (ADARSP255), increase NABS.
LU	65535	Defines the maximum total length of data for a request to the server. Do not change this parameter value unless advised to by Software AG.
TIMER	10	Defines how often the server is to wake up and look for work (note that the server wakes up anyway whenever it receives a request or operator command). Specify a value in seconds.
CT	60	Defines how many seconds the server will allow for a client to accept a completed request. If the client fails to acknowledge receipt of the request within this time, the server issues an ADAM93 USER GONE message and the client receives response code 254 (ADARSP254). If you get response code 254 (ADARSP254) frequently, increase the value of CT (the maximum is 32767) and also of NC and NABS.
SVC	0	Defines which SVC number is to be used. Specify your Adabas SVC.
MPMWTO	NO	Defines whether the server should send informational messages to the operator console or not. You should specify YES until you are satisfied that the server is operating correctly.
DEFAULT	None	Defines the default product to which requests will be passed. Specify SAF.
SAF PARM	SAFCFG	If you need to change the name of the configuration module (for example, you have different configuration modules with different settings), you can specify the name of the configuration module the daemon is to use. For example: SAF PARM=CFGDAEM .

4 Operator Commands

MVS operator communication with the daemon is achieved using the z/OS `Modify (F)` command. All operator commands for the SAF Security Kernel are prefixed with SAF. For example:

```
F jobname,SAF SSTAT
```

The available operator commands are:

Command	Description
`SHELP`	Display all possible SAF Kernel operator commands.
`SLOGOFF userid`	Log an individual SAF User ID off from the security system. Any cached security checks for this user are discarded.
`SNEWCOPY`	Restart the SAF Kernel, ensuring that all data held in its cache is flushed (the same as `SREST`). Additionally, reload SAFKRN and its dynamically loaded modules.
`SREST`	Restart the SAF Kernel, ensuring that all data held in its cache is flushed. Any data held by the security system itself in the SAF Kernel address space is also flushed. The operation is transparent to all online and batch users.
`SSHUT`	Perform an orderly shutdown of the SAF Kernel started task. This command should always be used to request an orderly termination. You may also use ADAEND, for example: `F jobname,ADAEND`
`SSNAP hhhhhhhh`	Display a selected portion of the SAF Kernel's memory. Operation is not terminated.
`SSTAT`	Display general statistics on the operator console for the SAF Kernel.
`STRACE {0|1|2|3}`	Dynamically activate or deactivate SAF Kernel diagnostic tracing: ■ 0 – tracing is suppressed ■ 1 – only security violations are traced ■ 2 – only successful checks are traced ■ 3 – all checks are traced

Command	Description
SUSERS	Display a list of active users.
SUSTAT userid	Display statistics for a specified user.

5 SAF* - SAF Daemon Messages

The following messages may be issued by the SAF daemon.

SAF001I **Unable to load required module: {xxxxxx}**

Explanation The SAF kernel or one of the modules it needs could not be loaded. Ensure that all SAF modules (including those created by installation assembly jobs – SAFCFG, SAFPSEC and SAFPMAC) are available.

SAF002I **Unable to obtain anchor storage**

Explanation A memory allocation failed during initialization. Increase the region size.

SAFD01S **SAF CANNOT INITIALISE, GETMAIN ERROR**

Explanation There is insufficient memory for the SAF daemon to initialize. Increase the region size.

SAFD02S **SAF CANNOT INITIALISE, KERNEL LOAD ERROR**

Explanation Installation error (SAFCKN load module not available). Ensure that all required load modules are available.

SAFD03E **DDCARD open error: ## - terminating**

Explanation The SAF daemon was enable to open DDCARD. Ensure that the DDCARD DD statement is specified correctly.

SAFD04I **Input parameter: {xxx}**

Explanation The daemon echoes the values of the supplied DDCARD parameters.

SAFD05E **Invalid parameter: {###########}**

Explanation DDCARD contained an invalid parameter. The SAF daemon terminates. Correct the parameter in error.

SAFD06E **Product parameter not specified**

Explanation DDCARD did not contain `PRODUCT=SAF`. Ensure that `PRODUCT=SAF` and `DEFAULT=SAF` are both specified.

SAFD07E **MPM failure - function: ## error ##**

Explanation The SAF daemon received an error from ADAMPM. This message will usually be preceded by an explanatory message. If in doubt, contact your Software AG technical support representative for assistance.

SAFD08E **IOR failure - function: ## error ##**

Explanation An error occurred during an ADAIOR service call. Contact your Software AG technical support representative for assistance.

SAFD09I **Shutdown requested**

Explanation The SAF daemon has been instructed to shut down.

SAFD10E **Getmain for command queue failed**

Explanation The SAF daemon failed to allocate its command queue. Increase the region size.

SAFD11I **SAF Kernel is active on node {nnnnn sss }CIB={aaaaaaaa}**

Explanation The daemon is now active and ready to receive security requests; nnnnn is the node ID, sss is the SVC number, and aaaaaaaa is the address of the daemon's main storage area.

SAFD12I **Oper type in: SAF {xxxxx}**

Explanation Message 12I is issued before processing of an operator command.

SAFD13E **Invalid operator command**

Explanation An invalid operator command was entered.

SAFD14I **Target {nnnnn} termination in progress**

Explanation Message 14I is issued during daemon termination (nnnnn is the daemon's node ID).

SAFD15I **Target {nnnnn} ended normally**

Explanation Message 15I is issued during daemon termination (nnnnn is the daemon's node ID).

SAFD22E **Load for module: {########} failed**

Explanation The indicated module could not be loaded. Ensure that it is available.

SAFD25E **{###} is an invalid product name**

Explanation An invalid `PRODUCT= parameter` was specified in DDCARD.

SAFD26E **Proxy module SAFPXY was not found, product SAF will not be called**

Explanation Ensure that SAFPXY and all other required load modules are available.

SAFD30E **Getmain for product parm block failed**

Explanation A memory allocation failed during initialization. Increase the region size.

SAFD31E **Cab allocation error in module syscoru**

Explanation A memory allocation failed during initialization. Increase the region size.

SAFD34E **UAB allocation error in module syscoru**

Explanation A memory allocation failed during initialization. Increase the region size.

SAFD40S **Abend {code} Psw {pppppppp pppppppp}**

Explanation Message 40S is issued during abnormal termination. It shows the abend code, Program Status Word, module that abended and register contents.

In the event of an abend, please ensure you collect the messages, the dump and any trace messages or snaps that have been generated.

SAFD42S **Module {module} entry {entry-point} offset {offset}**

Explanation Message 42S is issued during abnormal termination. It shows the abend code, Program Status Word, module that abended and register contents.

In the event of an abend, please ensure you collect the message, the dump and any trace messages or snaps that have been generated.

SAFD43S **Regs 00-03 {register contents}**
Regs 04-07 {register contents}
Regs 08-11 {register contents}
Regs 12-15 {register contents}

Explanation Message 43S is issued during abnormal termination. It shows the abend code, Program Status Word, module that abended and register contents.

In the event of an abend, please ensure you collect the message, the dump and any trace messages or snaps that have been generated.

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

- Operator Command Messages (SEFM900+ Series) Adabas SAF Securityoperator command messages SAF Security Kerneloperator command messages 28

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.

7 SAF Return Codes

ADASAF and the SAF Security Kernel display an eight-byte code containing various return and reason codes from SAF. This information is shown in a number of messages denoted "SSSSSSSS".

The SAF and ADASAF return codes contains the following structure:

Position Within Message Code	Information Content
Byte: 1	SAF return code (R15 after RACROUTE)
Byte: 2	Function code (see section *SAF Internal Function Codes*)
Byte: 3	RACROUTE return code
Byte: 4	RACROUTE reason code
Byte: 5-8	Internal reason code

The SAF trace messages written to DDPRINT, when GWMSGL is not 0, include the first four bytes of the following information, printed as eight hexadecimal digits. The ADASAF trace messages include the first four bytes of the following information, also printed as eight hexadecimal digits:

Position Within Trace Message	Information Content
Digits 1 and 2	SAF return code (R15 after RACROUTE)
Digits 3 and 4	Function code (see section *SAF Internal Function Codes*)
Digits 5 and 6	RACROUTE return code
Digits 7 and 8	RACROUTE reason code

Refer to the *IBM Security Server RACROUTE Macro Reference* manual 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.

8 SAF Internal Function Codes

SAF Security Kernel and ADASAF internal function codes include:

Function Code (Hex)	Description
00	Authorize Natural Library
04	Authorize Adabas access
08	Authorize SYSMAIN function
0C	Authorize Natural system files
10	Authorize Natural program execution
14	Authorize Broker service
18	Authorize Entire Net-Work access (Net-Work SAF Security) or Adabas cross-level access (Adabas SAF Security) or RPC execution (Natural SAF Security).
1C	Authorize SQL Server access
44 or 6C	Authenticate user

9 Interpreting Trace Messages

The SAF Kernel may optionally write trace messages to DDPRINT (or SAFPRINT). These trace messages have the following format:

```
Time   Jobname  Result     Return Code Type SAF Userid Level Resource Name
13:19:19 DAEFCODE SEF DENIED    08040800 RQ 02 :USERA   : (02) CMD00153.FIL00005
```

Field	Explanation
Time	Time the security check was made.
Jobname	Job that requested the security check. For Adabas and Net-Work SAF Security this is the job that issued the Adabas call being checked.
Result	SEF DENIED: the security system rejected the access attempt. SEF PERMITTED: the security system allowed the access.
Return Code	The return code consists of 4 hexadecimal bytes which contain the following information. The numbers in brackets refer to the values in the example trace message above. ■ Byte 1 (08) - R15 after RACROUTE ■ Byte 2 (04) – internal function code (see table above) ■ Byte 3 (08) – RACROUTE return code ■ Byte 4 (00) – RACROUTE reason code The return code can be interpreted by checking the RACROUTE manual referred to above for the appropriate RACROUTE function (AUTH for an authorize function; VERIFY for authenticate). For a RACROUTE AUTH, R15 of 8 with return code 8 and reason code 0 means the user is not authorized to use the requested resource. This is a normal security violation. For PERMITTED security checks, the return code contains 00000000 or 00000001. 00000001 indicates that the security check was satisfied from the SAF Kernel's cache (that is, the same user had previously requested the same resource access and the SAF Kernel had cached the security system's successful response).

Field	Explanation
Type	The internal SAF Kernel request type. This may be: ■ 01 – authorize Natural library ■ 02 – authorize Adabas access ■ 03 – authorize SYSMAIN function ■ 04 – authorize Natural system files ■ 05 – authorize Natural program execution ■ 06 – authorize Broker service ■ 07- authorize Net-Work (or Adabas cross-level) access ■ 08 – authorize SQL server access ■ 13 – authenticate user ■ 23 – authorize Natural RPC execution
SAF Userid	The SAF User ID for which access was requested.
Level	The access level requested: ■ 02 – read ■ 04 – update ■ 08 – control ■ 80 – alter
Resource Name	The name of the resource for which access was requested. For successful user authentications, resource name contains: ■ XXNEWU – user successfully authenticated or ■ XX - user already logged on

In the example trace message shown above: at 13:19:19, SAF user USERA in job DAEFCODE attempted to read Adabas file 5 in database 153 but did not have the necessary security access.

10 Security Definitions

- Defining Resources to RACF 42
- Defining Resources to CA-TOP SECRET 43
- Defining Resources to ACF2 45

SAF Security is implemented by defining resource classes and profiles and permitting users the necessary access to those profiles. Specific requirements for class and profile definitions and access levels are described in the individual product documentation.

This section describes in general how to define resources to RACF, CA-Top Secret and CA-ACF2.

Defining Resources to RACF

This section describes how the resources are defined to RACF. For exact details of the procedures to be followed for the installed RACF version, consult the relevant IBM manuals.

Overview of tasks

- Add classes to Class Descriptor Table
- Update z/OS Router Table
- Activate new classes
- Assign user ID for the SAF Security Started Task
- Permit user access to resource profiles

To add classes to Class Descriptor Table

1. Add the resource classes to the RACF Class descriptor table. Refer to the *IBM SPL RACF* manual. For an example, see IBM SYS1.SAMPLIB, member RACINSTL.
2. For flexibility, allocate maximum length for the classes (80).
3. Define the classes to enable discrete and generic profile use.
4. Check further attributes controlling the level of RACF messages generated when performing RACROUTE calls, as well as the required level of SMF recording. Sample definitions are provided in source members SAFRCLSN and SAFRCLSX.

To update the z/OS Router Table

- Update the z/OS router table as described in the *IBM SPL RACF* manual. For an example, see the IBM SYS1.SAMPLIB, member RACINSTL, section RFTABLE.

To activate new classes

- Activate new resource classes with SETROPTS (see *IBM RACF Command Language Reference* manual). For an example, activate class NBKSAG:

```
SETROPTS CLASSACT(NBKSAG)
SETROPTS GENCMD(NBKSAG)
SETROPTS GENERIC(NBKSAG)
```

To assign user ID for the SAF Security Started Task

- The SAF Security Kernel runs either in its own Started Task or in an Adabas or Entire Net-Work started task. Assign a user ID to these jobs with the relevant RACF authorizations, including the ability to perform `RACROUTE, TYPE=EXTRACT, TYPE=AUTH` and `TYPE=VERIFY` calls on profiles belonging to the defined classes.

To permit user access to resource profiles

- After adding profiles to protect the different resources, permit users the required level of access, using the relevant RACF Commands. The following example adds resource profile ETB.POLICY.QUOTE1 and grants READ access to user USER2 and CONTROL access to USER3. USER2 represents a client and requires READ access to execute while USER3 represents a server component that needs CONTROL access to register:

```
RDEFINE NBKSAG ETB.POLICY.QUOTE1 UACC(NONE)
PERMIT ETB.POLICY.QUOTE1 CLASS(NBKSAG) ACCESS(READ) ID(USER2)
PERMIT ETB.POLICY.QUOTE1 CLASS(NBKSAG) ACCESS(CONTROL) ID(USER3)
```

Defining Resources to CA-TOP SECRET

This section describes how the resources are defined to TOP SECRET. For exact details of the procedures to be followed for the installed version of TOP SECRET, consult the relevant CA-TOP SECRET manual.

Overview of tasks

- Add CA-TOP SECRET Facility
- Assign user ID for the SAF Security Started Task
- Add procedure name for the Started Task
- Add resource type to Resource Definition Table
- Assign ownership of resources
- Permit defined resources to Users

To add CA-TOP SECRET facility

- CA-TOP SECRET enables a set of authorization checks to be made against a certain facility. For example, this can be used to secure the development environment SAGDEV separately from the production environment SAGPROD. Alternatively, a default facility of batch can be used.

  When adding additional facilities, use the following attributes:

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

To assign a user ID for the SAF Security Started Task

- Add one user ID for each instance of the SAF Security 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(user-id) DEPT(dept) MASTFAC(fac)
```

To add a procedure name for the SAF Security Started Task

- The procedure name under which the SAF Security 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(user-id)
```

To add resource types to Resource Definition Table

- Add the resource types to the CA-TOP SECRET Resource Definition Table (RDT). Below is an example for resource type NBKSAG. Refer to the CA-TOP SECRET Reference Guide for a detailed explanation of the following commands and arguments:

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

To assign ownership of resources

- Assign ownership to a particular resource as shown in the following example. This must be done before permitting access to defined resource profiles:

```
TSS ADD(user1) NBKSAG(ETB.POLICY.QUOTE1)
```

  This makes user user1 the owner of the Broker service etb.policy.quote1.

To permit defined resource to users

- Permit access to a resource profile as in the following example. In the example, user user2 is permitted READ access to the Broker service etb.policy.quote1. This enables the user to execute as a client and issue requests to this Broker service:

```
TSS PER(user2) NBKSAG(ETB.POLICY.QUOTE1) FAC(fac) ACCESS(READ)
```

Defining Resources to ACF2

This section describes the definition of resources to ACF2 versions 5 and 6. For details of the procedures required for the current software version, please consult the relevant ACF2 manual.

Note: ACF2 provides insufficient return codes to determine whether a resource profile does not exist or simply the user does not have access to it. Therefore, if access is denied by ACF2, the SAF Security Kernel will always report "Access denied resource not allowed" in the error message.

Consequently the SAF Security configuration options such as BKUNI=Y to allow access to undefined resources are not applicable where ACF2 is used.

To define resources to ACF2 version 5

1 The SAF Security Kernel executes as a normal started task in z/OS. Define the user ID of the server task to ACF2 with the following attributes:

```
MUSASS,NON-CNCL,STC
```

To avoid the NON-CNCL attribute, APAR TW95626 must be applied.

2 Activate the SAF Interface using the command:

```
GSO OPTS -  SAF
```

3 Switch off all SAF checks by inserting the SAFSAVE record as follows:

```
SAFSAVE CLASSES(-) CNTLPTS(-) SUBSYS(-)
```

4 Switch on the SAF security checks for the SAF Security Kernel by inserting the SAFPROT record as follows:

```
CLASSES(-) CNTLPTS(-) SUBSYS(ADARUN)
```

5 For the general resource class name used by SAF Security product options, define a 3-character ACF2 resource type code by inserting a SAFMAPS record as follows:

```
SAFMAPS MAPS(NBK/NBKSAG)
```

6 Define the required resource profiles to ACF2 using the new type code.

The following example shows the addition of a Broker service etb.policy.quote1, allowing READ access for USER2:

```
$KEY(ETB.POLICY.QUOTE1) TYPE(NBK) UID(user2) ALLOW SERVICE(READ)
```

To define resources to ACF2 version 6 and above

1 The SAF Security Kernel executes as a normal started task in z/OS. Define the user ID of the server task to ACF2 with the following attributes:

```
MUSASS,STC
```

ACF2 version 6.1 and 6.2 no longer require TW95626,as these versions are more SAF-compliant.

2 Insert SAFDEF records as follows:

```
SAFDEF.EXS1
FUNCRET(4) FUNCRSN(0) ID(ENTIREX) MODE(GLOBAL)
RACROUTE(REQUEST=VERIFY SUBSYS=ADARUN REQSTOR=-)
RETCODE(4)
```

```
SAFDEF.EXS2
FUNCRET(4) FUNCRSN(0) ID(ENTIREX) MODE(GLOBAL)
RACROUTE(REQUEST=AUTH SUBSYS=ADARUN REQSTOR=-)
RETCODE(4)
```

```
SAFDEF.EXS3
FUNCRET(4) FUNCRSN(0) ID(ENTIREX) MODE(GLOBAL)
RACROUTE(REQUEST=EXTRACT SUBSYS=ADARUN REQSTOR=-)
RETCODE(4)
```

3 For the general resource class names used by SAF Security product options, define a 3-character ACF2 resource type code by inserting a CLASMAP record as follows:

```
CLASMAP
ENTITYLN(0) MUSID() RESOURCE(NBKSAG) RSRCTYPE(NBK)
```

4 Define the required security profiles to ACF2 using the new type code. The following example shows the addition of a Broker service etb.policy.quote1, allowing READ access only for user ID user2:

```
$KEY(ETB) TYPE(NBK)
POLICY.QUOTE1 UID(user2) SERVICE(READ)   ALLOW
POLICY.QUOTE1 UID(-)                     PREVENT
```

Index

A

Adabas SAF Security
- console and system data set messages,
- internal function codes, 37
- messages, 23
- operator command messages,
- return codes, 35

I

internal function codes, 24

M

messages
- SAF daemon, 19

R

return codes
- internal function codes, 24
- structure, 24

S

SAF daemon messages, 19
SAF Security Kernel
- console and system data set messages,
- internal function codes, 37
- messages, 19, 23
- operator command messages,
- return codes, 35

SAF* messages, 19
SEFM* messages, 23