

Security Definitions

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. It covers the following topics:

- Defining Resources to RACF
 - Defining Resources to CA-Top Secret
 - Defining Resources to CA-ACF2
-

Defining Resources to RACF

This section describes how the resources are defined to RACF. It covers the following topics:

- Adding Classes to a Class Descriptor Table
- Updating the OS/390 Router Table
- Activating New Classes
- Assigning a User ID for the SAF Security Started Task
- Permitting User Access to Resource Profiles

For detailed procedures, refer to the IBM manual for the installed version of RACF.

Adding Classes to a 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 member RACFCLSX.

Updating the OS/390 Router Table

Update the OS/390 router table as described in the IBM SPL RACF manual. For an example, see the IBM SYS1.SAMPLIB, member RACINSTL, section RFTABLE.

Activating New Classes

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

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

Assigning a 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

Permitting 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 ID "user2" and CONTROL access to "user3". "user2" represents a client and requires READ access to execute while "user3" represents a server component that requires CONTROL access in order 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 resources are defined to CA-Top Secret. It covers the following topics:

- Adding a CA-Top Secret Facility
- Assigning a User ID for the SAF Security Started Task
- Adding a Procedure Name for the SAF Security Started Task
-

Adding Resource Types to Resource Definition Table

-

Assigning Ownership of Resources

-

Permitting Defined Resources to Users

For detailed procedures, refer to the Computer Associates manual for the installed version of CA-Top Secret.

Adding a 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
```

Assigning 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, as follows:

```
TSS CRE(user-id) DEPT(dept) MASTFAC(fac)
```

Adding 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, as follows:

```
TSS ADD(STC) PROC(proc) USER(user-id)
```

Different procedure names are suggested when securing different environments separately with the use of non-default CA-Top Secret facilities.

Adding 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)
```

Assigning 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".

Permitting Defined Resources to Users

Permit access to a resource profile as shown in the following example.

```
TSS PER(user2) NBKSAG(etb.policy.quote1) FAC(fac) ACCESS (READ)
```

This permits user "user2" 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:

Defining Resources to CA-ACF2

This section describes the definition of resources to CA-ACF2 versions 5 and 6. For detailed procedures, refer to the Computer Associates manual for the installed version of CA-ACF2.

Note:

CA-ACF2 provides insufficient return codes to determine whether a resource profile does not exist or is simply not accessible to the user. Therefore, if access is denied by CA-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 CA-ACF2 is used.

CA-ACF2 version 5

1.

The SAF Security Kernel executes as a normal started task in OS/390. Define the user ID of the task to CA-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 following 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 names used by SAF Security product options, define a 3-character CA-ACF2 resource type code by inserting a SAFMAPS record as follows:

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

6.

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

```
$KEY(etb.policy.quote1) TYPE(NBK) UID(user2) ALLOW SERVICE(READ)
```

CA-ACF2 version 6

1.

The SAF Security Kernel executes as a normal started task in OS/390. Define the user ID of the server task to CA-ACF2 with the following attributes:

```
MUSASS,STC
```

Note:

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

2.

Insert the 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 CA-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 CA-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
```