Enhanced Access Control for SCLM for z/OS

User's Guide

Release 1

SC27-1591-00

Enhanced Access Control for SCLM for z/OS

User's Guide

Release 1

SC27-1591-00

Note!
Before using this information and the product it supports, be sure to read the general information under "Notices" on page 117.

First Edition (October 2002)

This edition applies to IBM Enhanced Access Control for SCLM for z/OS, Release 1, Program Number 5697-H59, and to any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address below.

This publication is available on the Web at:

http://www.ibm.com/software/ad/sclmsuite/accesscontrol/

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

IBM Corporation
H150/090
555 Bailey Avenue
San Jose, CA
95141-1003
U.S.A.

or fax your comments from within the U.S., to: 800-426-7773 or, from outside the U.S., to: 408-463-2629

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

About this document

This document describes how to use the IBM® Enhanced Access Control for Software Configuration Library Manager for z/OS™ licensed program, from here on called Enhanced Access Control for SCLM.

Enhanced Access Control for SCLM is an access control program for Software Configuration Library Manager (SCLM). Enhanced Access Control for SCLM works in conjunction with Resource Access Control Facility (RACF®). This document describes Enhanced Access Control for SCLM—how to implement, customize and use it.

Enhanced Access Control for SCLM is supported with SCLM releases corresponding to OS/390 V2R10 or later.

Enhanced Access Control for SCLM is supported with RACF releases corresponding to OS/390 V2R10 or later.

Who should read this document

This document is intended for security administrators responsible for maintaining and monitoring access controls over the SCLM environment. It assumes that you understand RACF concepts and your installation's implementation of access controls. If you are new to RACF or SCLM, you may want to review the information in "Related publications" on page viii before using this document.

What you need to know to understand this document

Before you read this document, you need to have a good understanding of access control issues and how RACF works. You should also have an understanding of SCLM concepts. This assumes familiarity with documents in the OS/390® Security Server and OS/390 ISPF libraries, together with practical experience in maintaining RACF access control definitions.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully.

You can perform most tasks required to set up and run Enhanced Access Control for SCLM using a 3270 emulator logged on to TSO.

IBM Personal Communications (Version 5.0.1 for Windows 95®, Windows 98®, Windows NT®, and Windows 2000®; version 4.3 for OS/2) provides 3270 emulation with accessibility features for people with disabilities. You can use this product to provide the accessibility features you need.

People with limited vision who use screen reader software might find this item requires particular attention:

Pop-up windows
Enhanced Access Control for SCLM uses the ISPF function that produces pop-up windows for some tasks. The pop-up and its frame are just text

that overlays the underlying information on the displayed panel. The frame of such a pop-up is not usually recognized as such by screen reader software, so you may need to gain some familiarity with reading such panels before the information becomes meaningful. ISPF pop-up windows can be displayed on a full screen by using the RESIZE (F4) command.

Conventions and terminology used in this document

Chapter 1, "Introduction" on page 1 introduces the concepts and terminology relevant to Enhanced Access Control for SCLM. The Web Site at http://www.ibm.com/ibm/terminology/ consolidates several of the main glossaries created for IBM products in one convenient location, including the Glossary of Computing Terms.

How this document is organized

This document has these chapters:

Chapter 1, "Introduction" on page 1 introduces Enhanced Access Control for SCLM and describes the purpose, benefits, concepts and operation.

Chapter 2, "Using the Enhanced Access Control for SCLM Dialog" on page 15 discusses how to use the Enhanced Access Control for SCLM dialog to administer access control definitions.

Chapter 3, "Enhanced Access Control for SCLM Definitions" on page 37 describes the definitions used by Enhanced Access Control for SCLM, how they work, and administration considerations.

Chapter 4, "Utilities and Sample Library" on page 57 describes the Enhanced Access Control for SCLM utility programs, along with sample execution and definition JCL.

Chapter 5, "Operator Commands" on page 69 describes the Enhanced Access Control for SCLM operator commands.

Chapter 6, "Problem Determination" on page 73 describes the Enhanced Access Control for SCLM Violation Reports, and how to resolve access control violations.

Chapter 7, "Installation" on page 95 describes the steps you need to follow to install Enhanced Access Control for SCLM.

Chapter 8, "Messages" on page 103 describes the messages issued by Enhanced Access Control for SCLM.

Related publications

You can find more information in these publications:

RACF

- z/OS SecureWay® Security Server RACF Security Administrator's Guide SA22-7638
- z/OS SecureWay Security Server RACF System Programmer's Guide SA22-7681
- z/OS SecureWay Security Server RACF General User's Guide SA22-7685

Software Configuration and Library Manager (SCLM)

- z/OS ISPF SCLM Reference SC34-4818
- z/OS ISPF SCLM Project Manager's and Developer's Guide SC34-4817

Related publications

Chapter 1. Introduction

What is Enhanced Access Control for SCLM?

Enhanced Access Control for SCLM improves access control of your SCLM resources. It prevents accidental damage to SCLM-managed data sets, and offers additional levels of access control granularity over SCLM functions. Enhanced Access Control for SCLM augments the services of RACF, and should be administered by the RACF administrator.

The central concept of Enhanced Access Control for SCLM is that access to SCLM resources is provided when SCLM programs are used. This avoids accidental damage to SCLM data sets resulting from updates using non-SCLM programs. The SCLM programs are described using Applications. The data sets to be controlled and their access rules are described using Profiles.

When Enhanced Access Control for SCLM is active, it monitors RACF data set violations. If a violation occurs for a data set managed according to the Enhanced Access Control for SCLM profiles, then the defined access rules are used to assign access privileges. If sufficient access privilege is not defined, then a RACF data set violation occurs.

Benefits

Without Enhanced Access Control for SCLM, SCLM users operating in a RACF environment must be granted UPDATE access to manipulate SCLM-managed data sets. Otherwise, they would receive RACF data set violations when performing various SCLM functions. However, the UPDATE access applies even if the data set is accessed using facilities other than SCLM. As a result, accidental or malicious damage may occur, and RACF controls will not prevent this.

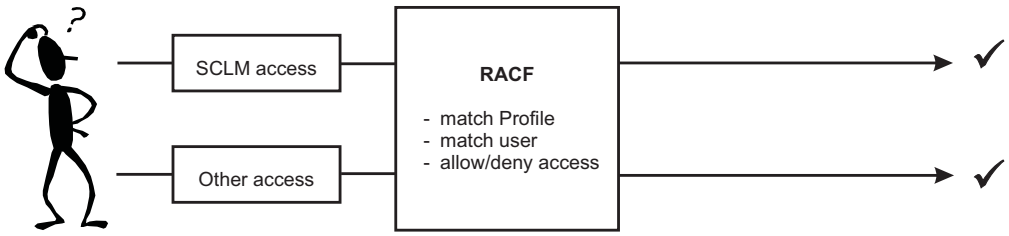


Figure 1. RACF data set validation

Enhanced Access Control for SCLM augments RACF controls. After normal RACF access controls are applied, Enhanced Access Control for SCLM can be used to grant access when a specific set of applications like SCLM are used. If access is attempted using other Applications, then the RACF rules denying access apply.

Benefits

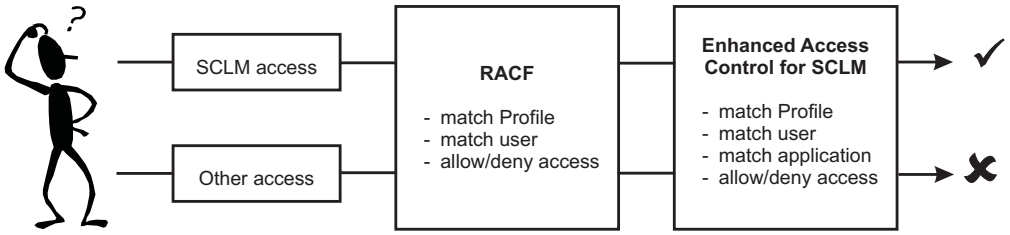


Figure 2. RACF with Enhanced Access Control for SCLM data set validation

The Enhanced Access Control for SCLM Applications can define various sub functions of SCLM; therefore an SCLM Promote may be allowed access whereas an SCLM Edit may be denied access. This provides additional levels of control over SCLM functions.

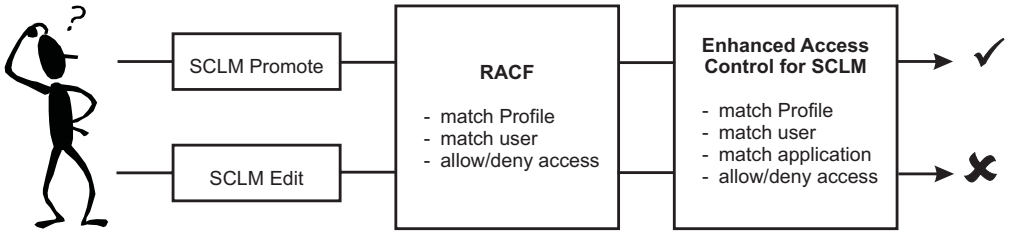


Figure 3. Enhanced Access Control for SCLM limits SCLM functions

Definitions

Enhanced Access Control for SCLM includes two types of definitions to provide access control. Profiles identify the data sets to be controlled, along with the access rules applicable for those data sets. Applications define the SCLM program environments that must be used in order to gain access to the data sets.

The Profile and Application definitions are maintained via online panels within the ISPF Dialog. These are then saved in the Enhanced Access Control for SCLM Rule File. A utility program loads the Profile and Application definitions into memory, and the Enhanced Access Control for SCLM Validation Routine uses these definitions to determine how access controls should be applied to data sets under its management.

Profiles

A Profile identifies a data set or RACF generic data set Profile to be validated by Enhanced Access Control for SCLM. Discrete Profiles like SCLM.DEVT.SOURCE describe a specific data set, and Enhanced Access Control for SCLM will always validate against the discrete data set Profile if one has been defined. Generic Profiles like SCLM.DEVT.* describe multiple data sets, and match the coding rules for RACF generic data set Profiles. Enhanced Access Control for SCLM will validate against a generic Profile if a discrete data set Profile has not been defined **and** RACF performed validation against a generic Profile of the same name.

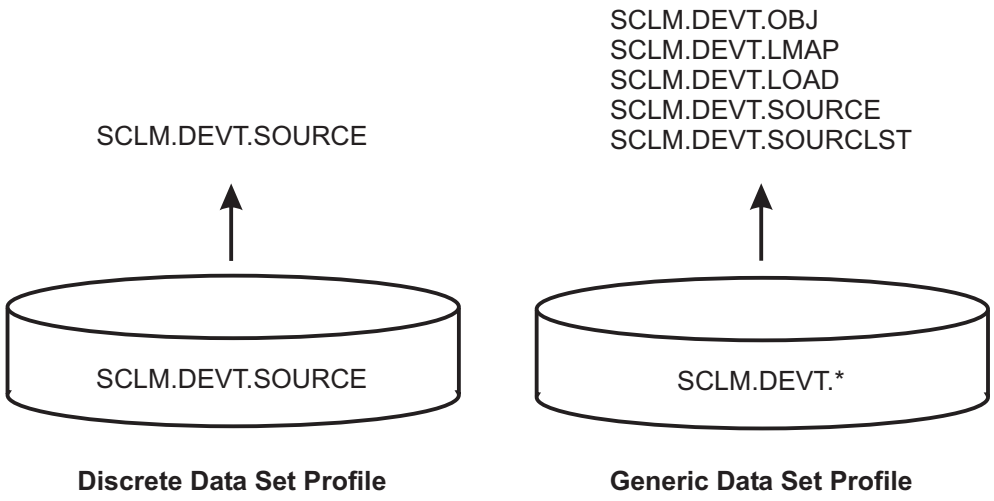


Figure 4. Discrete data set profiles and generic data set profiles

The Profiles also define the access rules validated by Enhanced Access Control for SCLM. These access rules determine who can have access privileges, and the controlling Applications or program environment required in order for those access privileges to apply. The access rule has three parts:

Application	Describes the SCLM programs as Applications/Functions
User	Describes the RACF user IDs or Groups, or * for all users
Access	Describes the access privilege, such as NONE, READ, UPDATE, CONTROL, and ALTER.

Here is an example of a Profile definition and its access rules. This is a discrete data set Profile as it controls only the one data set called SCLM.DEVT.SOURCE. This Profile has four access rules.

Profile: SCLM.DEVT.SOURCE

	Application	Function	User/Group	Access
1	SCLM	EDIT	FRED	READ
2	SCLM	EDIT	PGMRS	UPDATE
3	SCLM	PROMOTE	*	NONE
4	SCLM	PROMOTE	MNGRS	UPDATE

Access rule 1 shows that the user ID FRED is assigned the READ access privilege when the SCLM.DEVT.SOURCE data set is accessed via the SCLM EDIT Application and Function program list.

Access rule 2 is for the RACF Group called PGMRS. They will be assigned the UPDATE privilege if they access the data set by performing an SCLM EDIT.

Access rule 3 applies when the Application and Function of SCLM PROMOTE is used. In this case, the user value of * applies to all users, and they are assigned a privilege of NONE.

Access rule 4 also applies to the Application and Function of SCLM PROMOTE, but this case is specifically for the RACF Group called MNGRS who will be assigned the access privilege of UPDATE.

Profiles

Enhanced Access Control for SCLM matches the data set access request against its definitions for the Profile, Application and User/Group to determine the appropriate access privilege. Most-to-least specific matching is performed, therefore:

- Discrete data set Profiles take precedence over generic data set Profiles.
- RACF user IDs take precedence over RACF Groups, which take precedence over * for all users.

Chapter 3, "Enhanced Access Control for SCLM Definitions" on page 37 provides more information regarding Profile definitions, including how they work, examples and administration considerations.

Applications

The principal feature of Enhanced Access Control for SCLM is its ability to limit data set access via specific Applications. These Applications describe the programs that must be used to obtain access: a control program called the High Program and a service program called the Low Program. For example, ISRSCLM is an SCLM High Program, and FLMP may be used as the Low Program for the SCLM Promote function. By defining both the High and Low Programs, SCLM and its various sub functions can be secured.

Defining the SCLM programs as Applications simplifies access rule writing, as multiple combinations of SCLM programs may be grouped into one or a few applications. The Applications may also be assigned Function names to distinguish various SCLM functions or services. Whereas the Application name is mandatory, the Function name is optional. It provides flexibility in the way Applications are defined.

Here is an example definition for the Application and Function name of SCLM Promote. While these names are arbitrary, they should be assigned values that readily distinguish their use. You could equally have assigned the Application name of Promote and not used a Function name. This Application contains three High and Low Program pairs.

	Application:	SCLM		**Application:**	PROMOTE
	Function:	PROMOTE	or	**Function:**	
	High Program	**Low Program**		**High Program**	**Low Program**
1	FLMCMD	FLMP	1	FLMCMD	FLMP
2	FLMS$SRV	FLMP	2	FLMS$SRV	FLMP
3	ISRSCLM	FLMP	3	ISRSCLM	FLMP

Program pair 1
: specifies the High Program of FLMCMD. This is the controlling program when the SCLM Command Interface is executed. The Low Program is FLMP, because this is the program that controls the in the SCLM Promote service.

Program pair 2
: specifies the High Program of FLMS$SRV. This is the controlling program when the SCLM FLMLNK Subroutine or Call Interface is executed.

Program pair 3
: specifies the High Program of ISRSCLM. This is the controlling program when SCLM is executed online via TSO.

Chapter 3, "Enhanced Access Control for SCLM Definitions" on page 37 provides more information regarding Application definitions, including how they work, examples, suggested definitions and administration considerations.

Components

Here is a summary of the five Enhanced Access Control for SCLM components.

The **ISPF Dialog**	is an online interface that is used to define the SCLM Applications and the rules that govern user access to data set Profiles via those Applications.
The **Rule File**	is a VSAM KSDS data set that stores the Enhanced Access Control for SCLM definitions entered from the online panels.
The **Rule Load utility**	loads into memory a reformatted image of the Rule File, and activates the Enhanced Access Control for SCLM MVS Subsystem, and loads into memory the Validation Routine.
The **MVS Subsystem**	acts as an anchoring point for the in-memory rule addresses, and processes the Enhanced Access Control for SCLM operator commands.
The **Validation Routine**	is invoked via a RACF exit during access request validation. It assigns access privileges according to the in-memory rules as loaded from the Rule File.

The ISPF Dialog

The Enhanced Access Control for SCLM dialog is an ISPF based menu-driven dialog that is used to define the access controls for SCLM. These items describe how to use the dialog, and these equate to the options shown on the Enhanced Access Control for SCLM primary option menu. From these menu options, you define access controls for your SCLM resources.

1. Define Settings that govern the ISPF dialog displays. These settings include the name of the rule file that stores the access control rules.
2. The Status Information panel provides information regarding the current operation of Enhanced Access Control for SCLM. For example, it can tell you if Enhanced Access Control for SCLM is currently active on your system, and the name of the rule file that was last used to load the access control rules.
3. Define Profiles of the data set resources to be controlled, and the access rules for those data sets. The access rules describe combinations of users, access privileges, and Applications or SCLM programs whereby access is allowed.
4. Define Applications that describe the SCLM programs used for access control.
5. Report Violations that have occurred, thereby assisting your refinement of Enhanced Access Control for SCLM access control rules.

The Rule File

Enhanced Access Control for SCLM has its own rules database to store the rules that describe the conditions under which access is granted. The Rule File is a VSAM KSDS that is administered via online panels. The rule file contains the Enhanced Access Control for SCLM Profile and Application definitions.

The Rule File

Although multiple Rule Files may be defined, only one of these may be active on your system at any time. A central copy of the Rule File is loaded into memory via the Rule Load utility program HSSSSINT. Therefore changes to the rule file Profile or Application definitions do not take effect until they have been reloaded into memory by the Rule Load utility. The previous in-memory rules will then be replaced.

From the ISPF dialog, the Status Information panel shows you the Rule File last used to load the in-memory copy of the rules. The Settings panel determines the Rule File to be updated when changing Profile or Application definitions.

The Rule Load Utility

To enhance performance, Enhanced Access Control for SCLM loads an image of its Rule File definitions into memory. The HSSSSINT Rule Load Utility performs this task. The in-memory rules are formatted to simplify and speed processing for the Validation Routine.

If Enhanced Access Control for SCLM has not been previously started, the Rule Load Utility dynamically installs its MVS Subsystem and loads into memory the Validation Routine. If the Enhanced Access Control for SCLM administrator updates the Rule File via the online panels, then the Rule Load Utility may be rerun to refresh the in-memory copy of the rules.

Enhanced Access Control for SCLM will not be active on your system until the Rule Load Utility has been run. After each system IPL, the Rule Load Utility needs to be run again, and this may be automated by the use of a JCL procedure executed during the system IPL.

The MVS Subsystem

Enhanced Access Control for SCLM uses an MVS™ subsystem to anchor the storage addresses of the in-memory copy of the rules. The Rule Load Utility dynamically installs and initializes this subsystem, assigning its name based on a JCL parameter. The Validation Routine uses this MVS subsystem to locate the in-memory rules for data set access validation.

Once installed, the subsystem recognizes these MVS console operator commands, where <ssid> is the MVS subsystem-id:

<ssid> DISABLE	disables or stops Enhanced Access Control for SCLM validation processing
<ssid> ENABLE	enables or resumes Enhanced Access Control for SCLM validation processing
<ssid> INSTALL	dynamically installs the Validation Routine into the RACF environment
<ssid> UNINSTALL	dynamically uninstalls the Validation Routine from the RACF environment

The Validation Routine

The Validation Routine assigns access privileges to a data set, based on the Profile access rules. The Validation Routine receives control after RACF has completed its security validation. The RACF RACHECK post-processing exit ICHRCX02 is used to invoke the Validation Routine.

If the user has been granted data set access via RACF, then Enhanced Access Control for SCLM honors the RACF privilege and performs no further access checking.

However, if RACF denies access, then Enhanced Access Control for SCLM may grant access, as determined by the Profile access rules and the execution conditions of the original access request. In this event, the Validation Routine:

- Determines the Profile for validation based on the data set name referenced in the access request, or the RACF generic data set Profile used for validation
- Matches the program execution environment to the Applications defined within the Profile access rules
- Checks and withholds access if the SCLM program environment appears compromised
- Matches the user to the list of users or RACF Groups defined for the matching Application within the Profile
- Assigns the access privilege according to the matched access rule
- If access is not granted, access details are collected into an in-memory buffer to simplify online violation reporting
- Returns to RACF, signaling whether the access request has been granted.

The Validation Routine uses an in-memory copy of the Profile and Application definitions for validation purposes. These in-memory rules are formatted by the Rule Load utility (HSSSSINT) to simplify and speed validation checking.

Interaction with RACF

The Validation Routine interfaces to RACF via the RACHECK Post Processing exit. Although two installation methods are provided, dynamic or static, the technique for interaction with RACF is essentially the same.

The diagram shows how Enhanced Access Control for SCLM interacts with RACF for a data set access request when the Validation Routine is statically installed.

Interaction with RACF

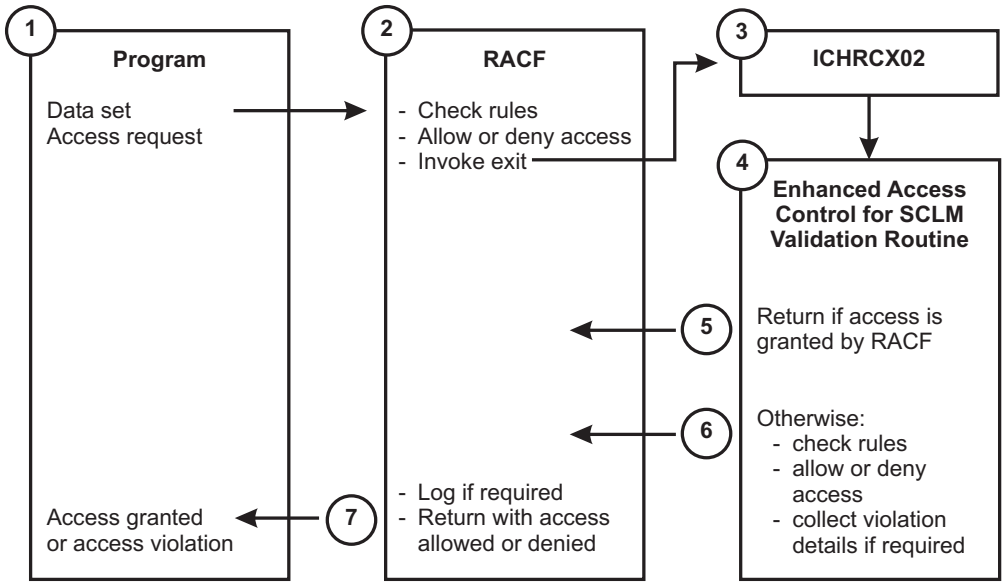


Figure 5. Interaction with RACF

1. A program issues a data set access request. This request is processed by RACF.
2. RACF checks its security rules and determines if the user should be allowed or denied access.
3. RACF invokes the ICHRCX02, RACHECK post-processing validation exit.
4. The RACF exit invokes the Enhanced Access Control for SCLM Validation Routine Interface (HSSRCX02), which in turns invokes the Validation Routine.
5. If RACF has allowed data set access, then Enhanced Access Control for SCLM ignores further checking and immediately relinquishes control.
6. However, if RACF has denied access, then Enhanced Access Control for SCLM checks its own rules. Profile, Application, and user matching are performed. Access is allowed or denied as dictated by the rules. A violation trace record may be collected to simplify rule refinement.
7. Control is returned to RACF. If required, RACF logging occurs.
8. RACF relinquishes control. The access request is granted or the program receives a RACF data set violation, which ever is appropriate.

Security and Administration Considerations

Enhanced Access Control for SCLM operates in conjunction with RACF to extend access control privileges. As Enhanced Access Control for SCLM can be used to grant access to data sets, it is imperative that administration of its definitions be as tightly controlled as the definitions for RACF. Therefore, the central RACF security officer should administer Enhanced Access Control for SCLM. The Enhanced Access Control for SCLM data sets themselves should also be secured, in particular:

- The Enhanced Access Control for SCLM execution load library (<smpedlib>.SHSSLINK) containing the HSSSSINT Rule Load Utility should be restricted to the use of the Enhanced Access Control for SCLM administrator.

  Note: If the SHSSLINK data set is secured as recommended above, then this data set should not be added to the MVS Linklist. Data sets in the Linklist are opened by a system function, therefore access is not attributed to the user.

- The Enhanced Access Control for SCLM ISPF panel library (<smpedlib>.SHSSPENU) should be restricted to the use of the Enhanced Access Control for SCLM administrator.
- The Rule File(s) used for storing the Enhanced Access Control for SCLM Profile and Application definitions should be restricted to the use of the Enhanced Access Control for SCLM administrator.

Back up of the Rule File should be performed regularly, in order to effect quick and timely recovery. Updated definitions saved in the Rule File cannot be automatically backed out. Therefore, it may be advisable to create two Rule Files: one for the currently active rules, and one for making rule alterations. These Rule Files may be synchronized after any new rules have been implemented and deemed as acceptable.

As the Rule File is a standard VSAM KSDS data set, the IDCAMS utility may be used to delete or define the Rule File, or copy records (REPRO) between Rule Files. This utility may also be used to resize or reorganize the Rule File.

Getting Started

This section is intended for a first-time Enhanced Access Control for SCLM administrator, perhaps initially setting up Enhanced Access Control for SCLM for use after the software installation has been completed. To get started quickly, work through these steps and learn about Enhanced Access Control for SCLM as you go.

1. **Software Installation**

   Chapter 7, "Installation" on page 95 describes the steps that need to be followed to install Enhanced Access Control for SCLM. Your system programmers may have completed this product installation already.

   However, you should confirm with them that the ICHRCX02 RACF validation exit has been tailored to invoke the Enhanced Access Control for SCLM HSSRCX02 module and that the ICHRCX02 exit is active, or whether the INSTALL command is to be used to dynamically install the Validation Routine after the Enhanced Access Control for SCLM definitions have been created and loaded into memory.

   In addition, ensure that the Enhanced Access Control for SCLM software and panel libraries are secured, so that access to these is restricted to Enhanced Access Control for SCLM administrators only.

2. **Invoke the Enhanced Access Control for SCLM ISPF Dialog**

   You will need to use the Enhanced Access Control for SCLM online panels to administer Enhanced Access Control for SCLM definitions.

   Check with your system programmers to see if the Enhanced Access Control for SCLM ISPF Dialog has been set up as a menu option within your ISPF environment. If not, the Enhanced Access Control for SCLM ISPF Dialog can be invoked from TSO option 6 (ISPF Command Shell) by entering the following command, where <smpetlib> represents the data set name high level qualifiers assigned to the Enhanced Access Control for SCLM software libraries:

   ```
   EX '<smpetlib>.SHSSEXEC(HSSOREXX)'  '<smpetlib>'
   ```

3. **Check if a Rule File is in use**

   Go to the Status Information panel in the ISPF Dialog. It will display information about the current status of Enhanced Access Control for SCLM. If the display indicates that the product is active, it also shows the name of the

Rule File used to load the definitions into memory. Rather than create a new Rule File in the next step, you may prefer to use this one.

You can learn more by referring to these topics:
- "The Status Information panel" on page 20
- "The ISPF Dialog" on page 5

4. **Create the Rule File**

   Create the Rule File that is used to store Profile and Application definitions. Sample JCL can be found in the SHSSSAMP product data set, member HSSRDEFN, or you can review these topics:
   - "HSSRDEFN - Rule File definition JCL" on page 57
   - "The Rule File" on page 5

5. **Set the Rule File for your ISPF session**

   Go to the Settings panel in this ISPF Dialog, and enter the name of your Rule File in the list area provided. Then select (S) this file as the one you wish to edit. You can learn more by referring to these topics:
   - "The Settings panel" on page 17
   - "The ISPF Dialog" on page 5

6. **Create Application definitions**

   Create Application definitions that describe the SCLM program environment. Suggested definitions have been provided, and sample JCL to load these into your Rule File was contained in the SHSSSAMP product data set, member HSSRDEFN that was used to allocate the Rule File in step 4.

   If you have already loaded the Application definitions, then you should review these. If you have not loaded the Application definitions you may do so now, or manually enter definitions via the Application Selection and Application Maintenance panels.

   Consideration should be given to the way SCLM functions are grouped into Applications, as this may simplify Profile access rule writing. Refer to "Application considerations for writing Profile access rules" on page 46 for more information.

   In the test example in the following steps, only the SCLM EDIT Application and Function are required. However, if you are loading the definitions via JCL, then do the whole lot now.

   You can obtain more information from the topics below:
   - How to load predefined Applications is described in "HSSRDEFN - Rule File definition JCL" on page 57.
   - "The Application Selection panel" on page 26
   - "The Application Maintenance panel" on page 28
   - "Applications definitions" on page 37
   - Appendix A, "Suggestions for Application Definitions" on page 111

   You will need to be familiar with the names of the Applications and Functions so that you can write the Profile access rules in the next step.

7. **Create Profile definitions**

   Create Profile definitions that describe the data sets and RACF generic Profiles to be controlled. The Profile definitions also contain the access rules for Enhanced Access Control for SCLM.

   Initially, just create one Profile to become familiar with Enhanced Access Control for SCLM. Go to the Profile Selection panel in the ISPF Dialog, and enter I (to insert) a new Profile. When the Profile Maintenance panel is

displayed, enter values for the Profile Name (the data set whose access will be controlled) and Data field (descriptive text). Leave all the prefix value fields set to *.

For testing purposes, make the Profile name match an SCLM-managed source library that you would normally be allowed to edit, even outside of SCLM.

At the bottom of the panel are input fields for the access rules. Enter the name of an Application and Function defined in the previous step, then the user ID or RACF Group, and lastly the level of access. You may use the PROMPT key (default is F4) to display a pop-up list of values for the fields Application, Function and Access. The Access field allows the prefix characters N (NONE), R (READ), U (UPDATE), C (CONTROL) and A (ALTER) to be entered.

For testing purposes, use the Application/Function that controls the editing of an SCLM source data set. The suggested definitions assign this the Application and Function name of SCLM and EDIT. Set your user ID, in the User/Group field, and give yourself UPDATE access.

Later you can repeat, insert or delete access rules from the list. You may overtype list values at any time. The columns may be sorted by placing your cursor onto a column heading and pressing ENTER. SORT from the VIEW Action Bar allows you to change the default sort sequence.

SAVE the updated Profile definitions and exit the panel.

You can learn more by referring to these topics:
- "The Profile Selection panel" on page 22
- "The Profile Maintenance panel" on page 23
- "Profiles" on page 48

8. **Load the definitions into memory**

   The definitions stored within the Rule File are loaded into memory via the HSSSSINT utility. All access validation performed by Enhanced Access Control for SCLM is made against the in-memory rules. Rule reloading replaces the previous in-memory rules.

   Sample JCL to execute the HSSSSINT utility can be found in the SHSSSAMP product data set, member HSSRLOAD, or you can learn how to run the utility via the topics below:
   - "HSSRLOAD - Rule Load Utility JCL" on page 60
   - "The Rule File" on page 5

   The execution job for HSSSSINT should complete with RC=00. The SYSPRINT file will contain information messages for your inspection.

9. **Verify via the Status Information panel that the load is complete**

   Go to the Status Information panel in the ISPF Dialog, as described in step 3 on page 9. The information display should show that Enhanced Access Control for SCLM is ACTIVE, and that you loaded the rules from your Rule File.

   You can learn more by referring to these topics:
   - "The Status Information panel" on page 20

10. **Dynamically INSTALL the Validation Routine**

    If the Enhanced Access Control for SCLM Validation Routine is installed and active on your system, you may ignore this step. However, if your system programmers indicated in step 1 on page 9 that the Validation Routine would be dynamically installed, this can be done now. Your system programmers should do this for you.

    You can learn more by referring to these topics:
    - "Dynamic installation using the INSTALL command" on page 98

- "INSTALL Command" on page 70

11. **Adjust the RACF rules**

    Adjust the RACF access rules to restrict access to READ for the data set and user ID that you have now defined to Enhanced Access Control for SCLM.

    As a general rule, use RACF to grant READ access to the SCLM related data sets, as this greatly simplifies rule writing. It avoids preparing Application definitions for the many program combinations that may read a data set, such as compare utilities, search-for-string utilities, copy utilities, and so forth.

12. **Test data set access outside of SCLM**

    Attempt to update the data set outside of SCLM, and in a manner not defined to Enhanced Access Control for SCLM. In our test example, try to edit or update the data set using the native TSO utilities, like the ISPF editor (TSO option 2) or IEBCOPY (TSO option 3.3). A RACF data set violation is expected. If this does not occur, then:
    - The RACF rules were not adjusted correctly in step 11 above; or
    - Access is being granted by Enhanced Access Control for SCLM because you have edited the file via SCLM, or the Application/Function definition in use is not restrictive to SCLM.

13. **Examine Violations**

    Go to the Violation Selection panel in the ISPF Dialog. The display should show a violation entry for the data set access attempt that failed in the previous step. If this does not appear, this indicates that:
    - The data set or generic Profile used by RACF has not been defined correctly to Enhanced Access Control for SCLM. Check the spelling of the Profile names.
    - The Rule File used to store the Profile and Application definitions has not been loaded into the in-memory rules. This should have been confirmed previously in step 9 on page 11.
    - The Enhanced Access Control for SCLM product is not correctly installed on your system. Reconfirm the installation issues described previously in step 1 on page 9.

    You can learn more from these topics:
    - "The Violation Selection panel" on page 30
    - "The Violation Detail panel" on page 32
    - "The Violation Programs panel" on page 35

14. **Test data set access via SCLM**

    Attempt to update the data set in a manner controlled by the Enhanced Access Control for SCLM definitions. In our example, try to edit the data set via SCLM. Data set access should be granted. If this does not occur, then:
    - Examine the violation log records as described in the previous step. The violation report will assist you in identifying the reason for the failure.

15. **Quick start is completed**

    By this point you have successfully created the Rule File, created an Application and Profile definition, loaded these into the in-memory rules and verified that the Enhanced Access Control for SCLM controls are operational. However, there is more to do to plan and implement Enhanced Access Control for SCLM as the next section will show.

Planning your Enhanced Access Control for SCLM implementation

Select an SCLM project

If your installation has multiple SCLM projects under management, then a staged approach to Enhanced Access Control for SCLM implementation is recommended. This will allow you to become familiar with how Enhanced Access Control for SCLM operates. Pick a candidate SCLM project. The SCLM project should be representative of your other projects, and be willing to accept some disruptions during Profile access rule refinement.

Arrange participant involvement

Implementing the Enhanced Access Control for SCLM controls for a subset of users will minimize disruption while you verify the effectiveness of your Profile access rules and Application definitions. Arrange for an SCLM developer to work with you on testing the Enhanced Access Control for SCLM access controls. Seek the cooperation of the RACF and SCLM administrators, as you may need their help when changing definitions.

Review SCLM translators

The SCLM translators execute programs to parse, build, or copy SCLM-managed objects. These programs must be loaded from an APF library or from an assigned task library.

The SCLM translator FLMTRNSL macro TASKLIB parameter defines the task library for program execution. If the TASKLIB is not assigned, the program must be loaded from an APF library. Programs loaded from common storage (LPA) already have APF library status. Programs loaded from the MVS Linklist where the MVS parameter LINKAUTH=LINKLIST is set, also have APF library status. In other cases, you must either:

- Alter the SCLM translator to include a TASKLIB parameter to define the program execution library; or
- Place the program into an authorized library

Review of your SCLM translators to determine task library assignment or if programs are APF loaded may become time consuming. An alternative is to test access using Enhanced Access Control for SCLM and review access violations to determine the cases where SCLM translators require refinement.

Identify the SCLM resources to be controlled

Identify the SCLM-managed data sets that will be controlled - by their data set name and by their RACF Profile for validation. Also determine the types of SCLM services that will be allowed against the data set, and which user IDs or RACF Groups will have access, and the access privileges for those users. Prepare a table to summarize these relationships:

Profile	Application-Function	User or Group	Access
SCLM.DEV1.SOURCE	SCLM Edit	PGMRS	UPDATE
	SCLM Build	MNGRS	UPDATE
		PGMRS	UPDATE

Identify the SCLM resources to be controlled

Profile	Application-Function	User or Group	Access
SCLM.DEV1.LOAD	SCLM Build	MNGRS	UPDATE
		PGMRS	UPDATE

Typically READ access is controlled via RACF and not Enhanced Access Control for SCLM, as reading using a specific execution program pathway is not required. This greatly simplifies rule writing. It avoids having to prepare Application definitions for the many program combinations that may read a data set, such as compare utilities, search-for-string utilities, and copy utilities, along with the APF load restrictions applied to these.

Initially limit this table to a subset of SCLM-managed libraries. The set should allow the exercise of the common SCLM services, like editing, browsing and building.

Prepare the Application definitions

Profile access rule writing can be simplified by the way SCLM functions are grouped into Applications. Review "Application considerations for writing Profile access rules" on page 46. Consider the table prepared in the previous section, and plan how the SCLM functions will be defined into Applications. Add the Application names to be used into the table. Using the Enhanced Access Control for SCLM ISPF dialog, prepare the Application definitions.

Prepare the Profile definitions

Profile definitions can be written from the table prepared in "Identify the SCLM resources to be controlled" on page 13. Prepare all the access rules required for each Profile. Even though one or two SCLM developers may conduct testing, use the appropriate RACF Groups in preference to specific user IDs. Run the Rule Load Utility and correct any reported errors.

Adjust RACF definitions

Adjust the RACF Profiles for the SCLM-managed data sets. Deny access for the specific user IDs that will conduct testing. In this way, only the testers will be impacted if the Enhanced Access Control for SCLM definitions require refinement.

Test access and refine definitions

Test user access to the data sets. Review the violation reports and refine the access rules where appropriate. "Data set access validation errors" on page 74 provides useful information for resolving access errors.

When access is correct, progress to other SCLM-managed data sets as described in "Identify the SCLM resources to be controlled" on page 13. When access for the entire SCLM project is correct, progressively implement the remaining users for that SCLM project by removing their access via RACF.

Chapter 2. Using the Enhanced Access Control for SCLM Dialog

The Enhanced Access Control for SCLM dialog is an ISPF-based menu-driven dialog that is used to define the access controls for SCLM.

The dialog requires no special customization or setup. Once a Rule File has been defined to store the Enhanced Access Control for SCLM definitions, you can start using the dialog panels to create and maintain definitions.

This chapter describes how to use the Enhanced Access Control for SCLM ISPF dialog:

- "ISPF Environment Considerations" describes how you can use features developed into the dialog.
- The rest of the chapter describes in detail the Enhanced Access Control for SCLM dialog. Information is presented in the same order as the primary menu options.

ISPF Environment Considerations

Enhanced Access Control for SCLM has been designed to follow CUA conventions, while also accommodating established ISPF conventions. For example:

- Possible actions are presented in action bar pull-down menus; those available from the File or View pull-down menus can also be requested from the command line.
- A menu or selection list item can be selected either by positioning the cursor over it (point-and-shoot) or by specifying its corresponding number, and then pressing Enter.
- For some entry fields you can select from a list of available choices by positioning the cursor on the field and pressing Prompt (F4). A+ (plus sign) to the right of the field or column heading indicates that Prompt is available.
- Short-cut navigation to the primary Enhanced Access Control for SCLM functions is available. For example, to invoke Violations reporting where you may view recent access violations, you can select option 5 from the primary menu, or enter =5 on the command line from anywhere in the Enhanced Access Control for SCLM dialog. The Menu action pull down bar may also be used for navigation.
- Help is available throughout the dialog. Context-sensitive help is available for each panel and input field, and there is an online tutorial.

CUA Attribute Settings

The Enhanced Access Control for SCLM dialog is designed to use the default CUA attributes. Changing CUA attributes, while allowed, may reduce your understanding of some panels in the dialog. Therefore it is recommended that these Panel Element settings be honored:

- Data entry fields (Choice Entry, List Entry, and Normal Entry) have the USCORE (underscore) Highlight attribute set. This will allow you to easily identify the input fields on each panel.

- Reference Phrases, List Items, List Item Description and Normal Text fields have their attributes set differently. This will allow you to easily distinguish between headings, field descriptions, choices, and instructions.
- Point-and-Shoot and Normal Entry Field have their Color attributes set differently. While this is not the CUA default, identifying Point-and-Shoot fields will be easier and enable you to use the Enhanced Access Control for SCLM dialog more effectively.

Function Key Settings

Enhanced Access Control for SCLM follows standard conventions for function keys, for example, F1=Help, F3=Exit, F4=Prompt, F7=Backward, F8=Forward, F12=Cancel.

Note: ISPF facilities accessed using the KEYS and KEYLIST commands allow the user to assign alternative functions to the keys.

Enhanced Access Control for SCLM may also display the function key settings at the bottom of each panel. The ISPF command PFSHOW ON/OFF allows display of key settings to be turned on and off. The default settings for the function keys can be displayed using the KEYSHELP command, also available from Help in the action bar.

Prompt (F4)

Prompt is available on some data entry fields in the Enhanced Access Control for SCLM dialog to help you specify valid values. To use this facility, position the cursor on the field and press Prompt (F4). A list is displayed from which you can select an acceptable value.

Point-and-Shoot Fields

Enhanced Access Control for SCLM employs point-and-shoot fields, such as menu items. To ensure point-and-shoot works for the standard ISPF operations, use the ISPF SETTINGS command and select **Tab to point-and-shoot fields**. This option does not affect the Enhanced Access Control for SCLM functions.

Display Size

Enhanced Access Control for SCLM panels are designed to accommodate 24 lines, with some panels incorporating scrollable areas for tabular lists. Screen display sizes of 32 or 43 lines reduce the amount of scrolling required for these panels.

Displaying Messages

Enhanced Access Control for SCLM uses both long and short messages. Short messages display at the top right, on the same line as the panel title. Long messages are designed to display in a pop-up window. However, long messages of less than the panel width can be customized to display just below or above the command line rather than in a window. To do this, use the ISPF SETTINGS command, and check that **Long message in pop-up** is not selected.

Messages displayed in a window can be moved to another location on the panel by:

1. Positioning the cursor on the top or bottom border of the message window, and pressing Enter.
2. Positioning the cursor at the location on the panel to which you wish to move the message, and pressing Enter.

The Primary Option Menu

The Primary Option Menu lets you navigate to other panels within the ISPF dialog. These panels allow you to maintain definitions, or view access control violation reports.

```
  Menu  Settings  Help
 ──────────────────────────────────────────────────────────────────────────
                Enhanced Access Control for SCLM - Primary Menu
Option ===> ____________________________________________________________


      1  Settings        Terminal and user parameters
      2  Status          Display current status information
      3  Profiles        Display or amend data set Profile rules
      4  Applications    Display or amend Application definitions
      5  Violations      Display log of recent access violations
      X  Exit            Exit Enhanced Access Control for SCLM




   IBM Enhanced Access Control for SCLM for z/OS
   Licensed Materials - Property of Fundi Software
   5697-H59 (C) Copyright Fundi Software 2002.
   All Rights Reserved. Use, duplication or disclosure restricted.

 F1=Help    F3=Exit   F12=Cancel
```

Figure 6. Primary Option Menu

The Settings panel

The Settings panel lets you customize the way the Enhanced Access Control for SCLM ISPF Dialog operates. It lets you choose processing options, and select the Rule File data set to be used for maintaining Profile and Application definitions.

```
 File  Menu  Function keys  Help
──────────────────────────────────────────────────────────────────────────────
                                 Settings                        Row 1 from 2
Command ===> ____________________________________________ Scroll ===> CSR

Options
    Enter '/' to select option
      / Confirm delete of Profile definition
      / Confirm delete of Application Function definition
      / Confirm autosave of Profile Rules updates
      / Confirm autosave of Application updates

Current Rule File  : 'YOUR.RULE.FILE'

Specify Rule File
   Enter '/' to select option

      _   'ANOTHER.RULE.FILE'
      _   'YOUR.RULE.FILE'
      **End**



 F1=Help      F3=Exit      F7=Backward  F8=Forward  F12=Cancel
```

Figure 7. The Settings panel

Panel Layout

The Settings panel has three parts:

Options
: Turn on and turn off processing options where / turns the option on, and blank turns it off.

Current Rule File
: Displays the Rule File currently selected for editing. This is the Rule File data set that is updated if you change Profile or Application definitions.

Specify Rule File
: Lists the Rule Files that may be selected for edit processing. You may insert (I), delete (D), update by overtyping, or select (S) the Rule File entries. This is a scrollable list. The number of entries in the list is shown by the row counts in the top right of the panel.

Confirm Profile Delete

This option only applies to requests to delete a Profile definition. It applies to the deletion of the whole Profile, not just an access rule within the Profile.

This field determines whether a pop-up confirmation window should be displayed when you attempt to delete the Profile. The pop-up window allows you to accept or cancel the deletion request, and change the delete Profile confirmation setting either permanently or temporarily for this Enhanced Access Control for SCLM session.

The delete Profile confirmation setting has these values:

/ Requests Enhanced Access Control for SCLM to prompt you for confirmation of a Delete request.

blank Indicates you do not want to confirm a Delete.

Confirm Application Delete

This option only applies to requests to delete an Application definition. It applies to the deletion of the whole Application, not just a High-Low Program pair within the Application.

This field determines whether a pop-up confirmation window should be displayed when you attempt to delete the Application. The pop-up window allows you to accept or cancel the deletion request, and change the delete Application confirmation setting either permanently or temporarily for this Enhanced Access Control for SCLM session.

The delete Application confirmation setting has these values:

/ Requests Enhanced Access Control for SCLM to prompt you for confirmation of a Delete request.

blank Indicates you do not want to confirm a Delete.

Confirm Profile Autosave

This option only applies when a Profile definition has been updated, and exiting the panel implies automatic saving of the updates. This automatic saving of updates is called AUTOSAVE.

This field determines whether a pop-up confirmation window should be displayed when you exit the panel with unsaved updates. The pop-up window allows you to accept or cancel the updates, and change the AUTOSAVE Profile confirmation setting either permanently or temporarily for this Enhanced Access Control for SCLM session.

The AUTOSAVE Profile confirmation setting has these values:

/ Requests Enhanced Access Control for SCLM to prompt you to confirm the automatic saving of updates.

blank Indicates that you do not want to confirm an AUTOSAVE. Any updates are automatically saved on panel exit.

Confirm Application Autosave

This option only applies when an Application definition has been updated, and exiting the panel implies automatic saving of the updates. This automatic saving of updates is called AUTOSAVE.

This field determines whether a pop-up confirmation window should be displayed when you exit the panel with unsaved updates. The pop-up window allows you to accept or cancel the updates, and change the AUTOSAVE Application confirmation setting either permanently or temporarily for this Enhanced Access Control for SCLM session.

The AUTOSAVE Application confirmation setting has these values:

/ Requests Enhanced Access Control for SCLM to prompt you to confirm the automatic saving of updates.

blank Indicates that you do not want to confirm an AUTOSAVE. Any updates are automatically saved on panel exit.

Current Rule File

This field displays the Rule File that is currently selected for editing. Definitions are retrieved from or saved to this Rule File. The Rule File is displayed as a fully qualified data set name, enclosed in quotes.

Current Rule File

You may select another Rule File for processing from the list of data sets in the Settings panel area labeled Specify Rule File.

Specify Rule File

This is a tabular display of Rule Files that may be selected for processing. You may scroll backward (default is F7) and forwards (default is F8) through the list. The fields include:

Option — The left-most input field is the processing option for the associated Rule File. The options are:

- / Display a pop-up window of options
- I Insert a Rule File list entry line
- D Delete this Rule File list entry
- S Select this as the current Rule File

Rule File — The Rule File data set name. Overtype this area to specify a new data set name. If you enclose the name in quotes, it is treated as a fully qualified data set name, otherwise Enhanced Access Control for SCLM adds the user's TSO PREFIX to resolve the fully qualified data set name.

The Status Information panel

The Status Information panel provides information about the current state of operation for Enhanced Access Control for SCLM. A REFRESH command can be invoked by function key (default is F5), line command, or the Functions Action Bar option to refresh the display with current data.

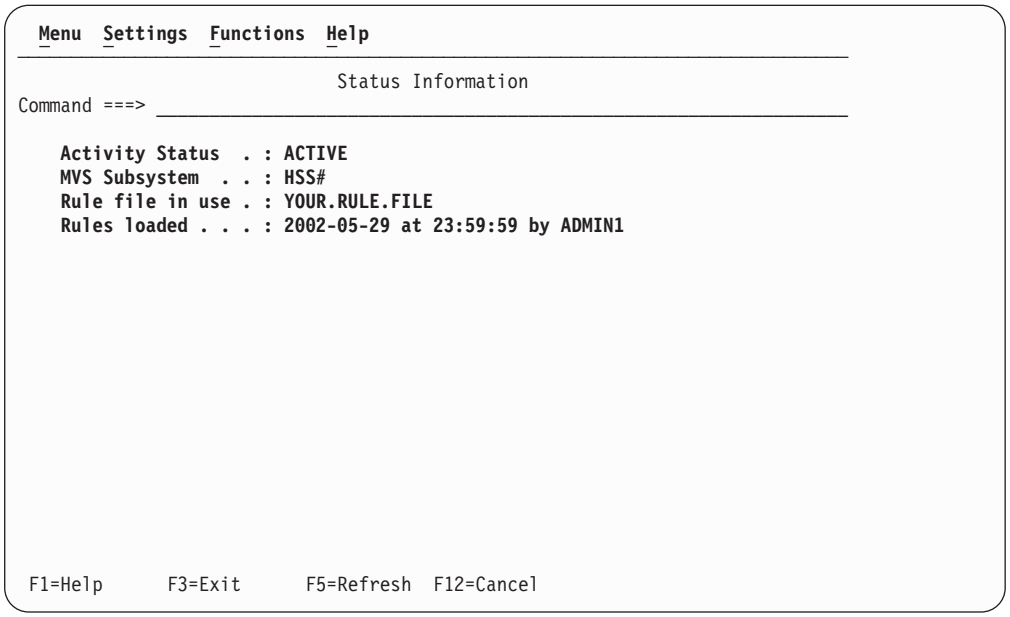


Figure 8. The Status Information panel

Command Line

The Status Information panel recognizes this special command:

REFRESH — Refresh the display by obtaining current status information. You can also enter the REFRESH command using a function key (the default is F5), or from the Functions Action Bar option.

The Activity Status field

The Activity Status field indicates the state of operation for Enhanced Access Control for SCLM. The status values are:

Active
: Enhanced Access Control for SCLM is in normal operational mode. It performs access control services as RACF data set violations occur. You may stop Enhanced Access Control for SCLM processing, thereby making it inactive, by issuing this MVS operator command:

```
<MVS subsystem> DISABLE
```

Inactive
: Enhanced Access Control for SCLM is not operational, because it has been stopped by the DISABLE operator command. In this state Enhanced Access Control for SCLM does not perform access control services. Therefore RACF data set violations occur in cases where Enhanced Access Control for SCLM previously granted access. You may resume Enhanced Access Control for SCLM processing, thereby making it active, by issuing this MVS operator command:

```
<MVS subsystem> ENABLE
```

Not Initiated
: Enhanced Access Control for SCLM is not operational, because it has not been started. The Rule Load Utility has not been successfully run to activate the MVS subsystem, load the in-memory rules, and load the Validation Routine.

  In this state Enhanced Access Control for SCLM does not perform access control services. Therefore RACF data set violations occur in cases where you might expect Enhanced Access Control for SCLM to grant access. Executing the Rule Load Utility starts Enhanced Access Control for SCLM.

Reloading
: Enhanced Access Control for SCLM is reloading the in-memory rules. When this is complete, Enhanced Access Control for SCLM resumes the previous ACTIVE or INACTIVE status, or becomes ACTIVE if it was previously NOT INITIATED. During the RELOADING Enhanced Access Control for SCLM performs access control services if it was previously ACTIVE.

The MVS subsystem field

The MVS subsystem field displays the name of the MVS subsystem used by Enhanced Access Control for SCLM. This field is blank if Enhanced Access Control for SCLM has not been started, and is therefore in a NOT INITIATED state.

The Rule file in use field

The Rule File In Use field displays the name of the Rule File last used to load the in-memory copy of the rules. This field is left blank if Enhanced Access Control for SCLM has not been started, and is therefore in a NOT INITIATED state. The Rule File is shown as a fully qualified data set name, without quotes.

Note: This Rule File may differ from that used for editing purposes using the ISPF dialog. Choose the current Rule File for editing using the Settings panel.

The Rules Loaded field

The Rules Loaded field displays when and by whom the Rule File was last used to load the in-memory copy of the rules. The field has this layout:

The Rules Loaded field

```
Rules Loaded: YYYY-MM-DD at HH:MM:SS by USER ID
```

This field is left blank if Enhanced Access Control for SCLM has not been started, and is therefore in a NOT INITIATED state.

The Profile Selection panel

The Profile Selection panel lets you maintain the Profile definitions that describe to Enhanced Access Control for SCLM the data sets to be validated for access control.

```
 Menu  Settings  Help
──────────────────────────────────────────────────────────────────────────────
                              Profile Selection                 Row 1 from 6
Command ===> ____________________________________________ Scroll ===> CSR

Profile Prefix (Use * for all Profiles)
       *_____________________________________

Profile Processing
   Enter '/' for selection list

        Profile
     s  SCLM.DEVT.*
     _  SCLM.DEVT.SOURCE
     _  SCLM.EXPORT.FILE
     _  SCLM.PROJDEFS.LOAD
     _  SCLM.RELEASE.*
     _  SCLM.TEST.*
****************************** Bottom of data ********************************



F1=Help      F3=Exit      F7=Backward  F8=Forward  F12=Cancel
```

Figure 9. The Profile Selection panel

The panel displays a list of Profile definitions. From this list you can select an entry for updating, delete an entry, or insert new entries. Upon entry, the list is displayed is ascending sort sequence.

The panel has two parts:

Prefix filter	Limits the Profile List display
Profile List	Is the list of Profiles validated by Enhanced Access Control for SCLM

Prefix filter

The prefix filter limits the Profiles display. Only Profiles matching the prefix will be displayed. These examples show how the prefix filter works:

*	displays all Profiles
SC*	displays Profiles starting with SC
SCLM*	displays Profiles starting with SCLM
SCLM.*	displays Profiles starting with SCLM.
SCLM.DEVT*	displays Profiles starting with SCLM.DEVT

Profile List

This is a tabular display of Profiles that may be selected for processing. The list is scrollable. The number of entries in the list is shown by the row counts in the top right of the panel. If the prefix filter is active, then FILTERING ACTIVE replaces the row counts in the top right of the panel.

The Profile List fields include:

Option — The processing option for the associated Profile. The options are:

- / Display a pop-up window of options
- D Delete this Profile list entry
- I Insert a new Profile definition
- S Select this Profile for updating

Profile — The Profile name. This is a display field that cannot be overtyped. The Profiles identify discrete data sets, or RACF generic data set Profiles to be validated by Enhanced Access Control for SCLM.

The Profile names are fully qualified and displayed without quotes. Only Profile names matching the Profile Prefix field appear in the display.

The Profile Maintenance panel

The Profile Maintenance panel lets you maintain the access rules for a Profile. These access rules determine the user access privileges assigned by Enhanced Access Control for SCLM, and the conditions under which they apply.

```
  File  Menu  Settings  View  Help
 ──────────────────────────────────────────────────────────────────────────────
                            Profile Maintenance                  Row 1 from 4
Command ===> ____________________________________________________ Scroll ===> CSR

  Profile:  SCLM.DEVT.*
  Data:
  SCLM Developer datasets
  ____________________________________________________________________________
  ____________________________________________________________________________
  ____________________________________________________________________________

    Enter a prefix value to limit the display of each column
        Application  Function     User/Group   Access
        *            *            *            *

    Enter '/' to select option or overtype rules

       Application +  Function +  User/Group  Access +
    _  SCLM           BROWSE       *           READ
    _  SCLM           BUILD        PGMRS       UPDATE
    _  SCLM           EDIT         PGMRS       UPDATE
    _  SCLM           PROMOTE      MNGRS       UPDATE
****************************** Bottom of data ********************************
 F1=Help      F3=Exit      F4=Prompt    F7=Backward  F8=Forward  F12=Cancel
```

Figure 10. Profile maintenance

The panel has four parts:

Profile — Identifies the data set or generic Profile
Data — A textual description for information purposes
Prefix filters — Limits the Access Rules list display
Access Rules — Tabular data defining the access rules

Command Line

The Profile Maintenance panel recognizes these special commands:

Command Line

SORT — Display a pop-up window to control the sort sequencing of the Access Rules list. This pop-up window may also be presented via the View Action Bar option.

SORT <column name> — Sort the Access Rules list based on the specified column. You can specify multiple column names, in which case these are sorted in the specified sequence order. Valid column names are Application, Function, User or Group, and Access. If an invalid column name is entered, then the sort pop-up window is displayed.

Profile

The Profile is the name of a discrete data set or RACF generic data set Profile to be validated by Enhanced Access Control for SCLM.

If you are inserting a new Profile definition, then a value for this field must be entered. The value may be changed until the Profile definition is saved, after which the field may not be overtyped. The Profile name is entered fully qualified and without quotes. The Profile name conforms to the conventions for data set names or RACF generic data set Profiles.

If you are maintaining an existing Profile definition, the Profile name cannot be overtyped.

If a Profile has been assigned an incorrect name value, the incorrect Profile definition must be deleted and the correct Profile definition inserted.

Data

The DATA field allows descriptive text to be added to the Profile definition. This text is stored along with the Profile definitions in the Rule File, however Enhanced Access Control for SCLM does not use it in any way.

This area is provided for the administrator's use. The area may contain any data; its contents are not validated. You may leave this area blank.

Prefix filters

The Prefix filters limit the Access Rule List display. Only Access Rules matching the prefix filter values will be displayed.

In the first example, all prefix filters are set to * (anything); therefore, all Access Rule List entries are displayed.

```
Prefix Values:
Application     Function        User/Group      Access
*               *               *               *

Access Rule List:
Application     Function        User/Group      Access
SCLM            BROWSE          *               READ
SCLM            BROWSE          PETER           READ
SCLM            BROWSE          SIMON           READ
SCLM            EDIT            STEVE           READ
SCLM            EDIT            PETER           UPDATE
SCLM            EDIT            PAUL            UPDATE
```

In the next example, only one filter is used. It is for User/Group and the filter value P* means only users or RACF groups commencing with P are displayed. The display appears as:

```
Prefix Values:
Application     Function        User/Group      Access
*               *               P*              *

Access Rule List:
Application     Function        User/Group      Access
SCLM            BROWSE          PETER           READ
SCLM            EDIT            PETER           UPDATE
SCLM            EDIT            PAUL            UPDATE
```

In the next example, two filters are used. Both must be satisfied for the access rule to appear in the display.

```
Prefix Values:
Application     Function        User/Group      Access
*               ED*             P*              *

Access Rule List:
Application     Function        User/Group      Access
SCLM            EDIT            PETER           UPDATE
SCLM            EDIT            PAUL            UPDATE
```

If no access rules satisfy the prefix filters, then the display appears as follows:

```
Prefix Values:
Application     Function        User/Group      Access
*               *               JOHN*           *

Access Rule List:
Application     Function        User/Group      Access
```

Access Rules

This is a tabular display of Access Rules. The list is scrollable. The number of entries in the list is shown by the row counts in the top right of the panel. If prefix filters are active, then FILTERING ACTIVE replaces the row counts in the top right of the panel. To simplify viewing, the Access Rules list may be sorted as described in the section that follows.

The Access Rules values may be overtyped, or the PROMPT key (default is F4) may be used to display a list of values for the Application, Function and Access columns. The table fields include:

Option
: The processing option for the Access Rule. The options are:
  - / Display a pop-up window of options
  - D Delete this Access Rule
  - I Insert a new Access Rule line, or I2 for two lines and so on
  - R Repeat this Access Rule line, or R2 to repeat twice and so on

Application
: The Application name, which must be previously defined to Enhanced Access Control for SCLM. The Application and Function define SCLM services by means of the SCLM programs that provide those services.

  Enhanced Access Control for SCLM validates that data set access is performed using the program combinations defined in the Applications/Functions definitions.

Function — The Function name, which must be previously defined to Enhanced Access Control for SCLM for the specified Application. The Function can be left blank for Applications defined without a Function value.

User/Group — The user ID or RACF Group. The access rule will only be applicable if this field value matches:

1. The RACF user ID in use when access was attempted
2. A RACF Group to which the user was currently connected at the time the access was attempted
3. The value *, which is used to designate all users

Sorting the Access Rules

By default, the Access Rules list is sorted in ascending order based on the column sequence Application, Function, User/Group and Access. However this can be changed. Your default sort sequence can be controlled via the action bar View pull down option, or by entering SORT on the command line. This will present a pop-up window to control the sort sequencing. The pop-up window appears as follows:

```
──────────── Profile Rule Sort ────────────

To sort the display, enter the sort sequence
number for each column and A or D for ascending
or descending order


Sequence    Column        A or D
   1     APPLICATION        A
   2     FUNCTION           A
   3     USER/GROUP         A
   4     ACCESS             A


Press ENTER to confirm and perform the sort.
Press CANCEL or EXIT to cancel sort.
```

Figure 11. The Profile Rule Sort pop-up

From this pop-up window, you may assign each column a unique sort sequence number, from 1 to 4. You also indicate whether each column should be sorted in ascending or descending sequence by entering an A or D adjacent to the column name.

The sorting sequence may be changed temporarily by placing your cursor onto one of the column headings and pressing Enter, for by entering SORT <column name> on the command line.

The Application Selection panel

The Application Selection panel lets you maintain the Application definitions that describe to Enhanced Access Control for SCLM the SCLM program environment.

```
  Menu  Settings  Help
 ──────────────────────────────────────────────────────────────────────────────
                            Application Selection                Row 1 from 11
Command ===> ____________________________________________________ Scroll ===> CSR

Application Prefix (Use * for all Applications)
         *_______

Application Processing
   Enter '/' for selection list

         Application  Function
      _  SCLM         ALL
      _  SCLM         ARCHREP
      _  SCLM         BROWSE
      _  SCLM         BUILD
      s  SCLM         EDIT
      _  SCLM         EXPORT
      _  SCLM         IMPORT
      _  SCLM         MIGRATE
      _  SCLM         PROMOTE
      _  SCLM         SAMPLE
      _  SCLM         UTILITY
 F1=Help      F3=Exit      F7=Backward  F8=Forward  F12=Cancel
```

Figure 12. The Application Selection panel

The panel displays a list of Application/Function definitions. From this list you may select an entry for updating, delete an entry, or insert new entries. Upon entry, the list is displayed is ascending sort sequence.

The panel has two parts:

Prefix filters	Limits the list of Applications displayed
Application List	The list of defined Applications/Functions

Prefix filters

The prefix filters limit the Application List display. Only Applications matching the prefix value are displayed.

These examples show how the prefix filter works:

*	Displays all Applications
SC*	Displays Applications starting with SC
SCLM*	Displays Applications starting with SCLM

Application List

This is a tabular display of Applications that may be selected for processing. The list is a scrollable. The number of entries in the list is shown by the row counts in the top right of the panel. If the prefix filter is active, then FILTERING ACTIVE replaces the row counts in the top right of the panel. The Application List fields include:

Option — The processing option for the associated Application. The options are:

/	Display a pop-up window of options
D	Delete this Application list entry
I	Insert a new Application definition
S	Select this Application for updating

Application — The Application name. This is a display field that cannot be

overtyped. The combination of Application and Function identify an SCLM service, and will define the SCLM programs that provide that service.

Function — The Function name. This is a display field that cannot be overtyped. The Function field allows the Application to be classified to add clarity when preparing Application Definitions.

The Application Maintenance panel

The Application Maintenance panel lets you define the SCLM programs that govern Enhanced Access Control for SCLM access control.

```
 File  Menu  Settings  Help
──────────────────────────────────────────────────────────────────────────────
                           Application Maintenance            Row 1 to 8 of 22
Command ===> ____________________________________________     Scroll ===> CSR

  Application  SCLM
  Function . . EDIT
  Data:
  Controls SCLM member editing. High Programs are: ISRSCLM (online edit);
  FLMCMD (FLMCMD Edit); FLMS$SRV (FLMLNK Edit). Cloud9 requires High-Low
  Programs CLZRSDRV-CLZRSDRV, FLMCMD-FLMCMD, and FLMCMD-FLMCXUDI.

    Enter '/' to select option or overtype pairs

        High Program  Low Program
    _   CLZRSDRV      CLZRSDRV
    _   FLMCMD        FLMCMD
    _   FLMCMD        FLMCXUDI
    _   FLMCMD        FLME$CRT
    _   FLMCMD        FLME$END
    _   FLMCMD        FLME$IM
    _   FLMCMD        FLME$SAV
    _   FLMCMD        FLME$SMO
 F1=Help      F3=Exit      F7=Backward  F8=Forward  F12=Cancel
```

Figure 13. The Application Maintenance panel

The panel has four parts:

Application — The Application name
Function — The Function within the Application
Data — A text description for information purposes
Program List — Tabular data of High-to-Low program environments

Application

The Enhanced Access Control for SCLM administrator assigns the Application name. It identifies a set of program conditions, like an SCLM function, and would typically reflect the purpose or use of those programs.

Enhanced Access Control for SCLM lets you classify Applications into Functions. This provides flexibility in the way the program combinations are defined and referenced when writing the Profile Access Rules.

If you are inserting a new Application/Function definition, then you must enter a value for this field. The value may be changed until the Application definition is

saved, after which the field may not be overtyped. The Application name is entered without quotes. The combination of Application and Function must be unique.

If you are maintaining an existing Application definition, the Application name cannot be overtyped.

Function

Enhanced Access Control for SCLM lets you classify Applications into Functions. This provides flexibility in the way the program combinations are defined and referenced when writing Profile access rules.

The Application and Function are assigned by the Enhanced Access Control for SCLM administrator. Each combination of Application and Function must be unique. The Application/Function identifies a set of program conditions referenced within Profile access rules. Therefore, the names given to Application and Function would typically reflect the purpose or use of those programs.

If you are inserting a new Application/Function definition, then a value for Function is optional. Any entered value may be changed until the Application definition is saved, after which the field may not be overtyped. The Function name is entered without quotes. The combination of Application and Function must be unique.

If you are maintaining an existing Application definition, the Function name cannot be overtyped.

Data

The DATA field allows descriptive text to be added to the Application definition. This text is stored along with the Application definitions in the Rule File, however Enhanced Access Control for SCLM does not use it in any way.

This area is provided for the administrator's use. The area may contain any data; its contents are not validated. You may leave this area blank.

Program List

This is a tabular display of program pairs that describe an SCLM program environment, and hence an SCLM service. The list is a scrollable. The number of entries in the list is shown by the row counts in the top right of the panel.

The Program List entries may be overtyped to change values, or lines may be inserted or repeated and overtyped. Changes will not be stored in the Rule File until a SAVE command is issued, or you exit from the panel. CANCEL will discard changes on exit.

The Program List is sorted in ascending sequence based on the column sequence High Program, Low Program.

The table fields include:

Option — The processing option for the Program List entry. The options are:

- / Display a pop-up window of options
- D Delete this Program List entry
- I Insert a new Program List entry
- R Repeat this Program List entry

Program List

High Program
: The High Program is the controlling SCLM program. Typically this is FLMCMD for the SCLM Command interface services; FLMS$SRV for the SCLM Call or Subroutine interface services via FLMLNK; or ISRSCLM for the SCLM online services via ISPF. Refer to "What is the High Program?" on page 40 for information regarding candidate High Program values.

Low Program
: The Low Program is an SCLM program that performs a specific service. This typically is the SCLM program that opens the data sets to be controlled, invokes SCLM translators, or executes non-SCLM utilities.

  Defining the Low Program requires knowledge of the SCLM programs and execution environment. Refer to "What is the Low Program?" on page 42 for an in-depth description of the SCLM Low Programs.

The Violation Selection panel

The Violation Selection panel displays recent Enhanced Access Control for SCLM validation requests where access was denied, resulting in a violation. You may view these violations to assist with Profile access rule refinement.

```
 Menu  Settings  Functions  Help
──────────────────────────────────────────────────────────────────────────────
                            Violation Selection                Row 1 from 20
Command ===> ____________________________________________ Scroll ===> CSR

Enter a prefix value to limit the display of each column
    Date        Time      User      Data Set
    *           *         *         *
    ──────────  ────────  ────────  ──────────────────────────────────────────

Enter '/' to view Violation details

    Date        Time      User      Data Set
/   2002-06-04  09:05:49  MNGR3     SCLM.DEVT.SOURCE
_   2002-06-04  08:42:18  MNGR3     SCLM.DEVT.SOURCE
_   2002-06-04  08:39:40  MNGR3     SCLM.DEVT.SOURCE
_   2002-05-31  13:33:27  PGMR3     SCLM.TEST.SOURCE
_   2002-05-31  12:45:33  PGMR2     SCLM.TEST.SOURCE
_   2002-05-31  12:38:16  PGMR2     SCLM.TEST.SOURCE
_   2002-05-31  12:08:04  PGMR1     SCLM.TEST.SOURCE
_   2002-05-31  12:07:44  PGMR1     SCLM.TEST.SOURCE
_   2002-05-31  12:07:16  PGMR1     SCLM.TEST.SOURCE
_   2002-05-31  12:02:16  MNGR2     SCLM.TEST.SOURCE
_   2002-05-30  14:34:26  MNGR1     SCLM.DEVT.SOURCE
_ F1=Help      F3=Exit      F5=Refresh   F9=SWAP     F12=Cancel
```

Figure 14. The Violation Selection panel

As data set violations occur, RACF violation records are logged. However, Enhanced Access Control for SCLM also keeps an in-memory buffer of its most recent violations to simplify reporting and assist access rule refinement. When the Violation Selection panel is invoked or refreshed, these details are collected and presented for display.

The list of violations remains constant until you exit the panel or issue the REFRESH command. The violation list is displayed in reverse chronological sequence, and prefix filters may be used to limit the display. Violation details are displayed by selecting an entry from the list.

The panel has two parts:

Prefix filters	Limit the Violations List display
Violations List	The list of violations

Command Line

The Violation Selection panel recognizes this command:

REFRESH — Refresh the display by obtaining the current violation records from the in-memory buffer. The REFRESH command may also be entered via function key (default is F5), or from the Functions Action Bar option.

Prefix filters

The Prefix filters limit the Violations List display. Only violations matching the prefix filter values are displayed.

In this example, all prefix filters are set to * (anything) therefore all Violation List entries are displayed.

```
Prefix Values:
Date            Time            User            Data Set
*               *               *               *

Violations List:
Date            Time            User            Data Set
2002-06-04      08:42:18        MNGR3           SCLM.DEVT.SOURCE
2002-06-04      08:39:40        MNGR3           SCLM.DEVT.SOURCE
2002-05-31      13:33:27        PGMR3           SCLM.TEST.SOURCE
2002-05-31      12:45:33        PGMR2           SCLM.TEST.SOURCE
2002-05-31      12:38:16        PGMR2           SCLM.TEST.SOURCE
2002-05-31      12:08:04        PGMR1           SCLM.TEST.SOURCE
2002-05-31      12:07:44        PGMR1           SCLM.TEST.SOURCE
2002-05-31      12:07:16        PGMR1           SCLM.TEST.SOURCE
2002-05-31      12:02:16        MNGR2           SCLM.TEST.SOURCE
2002-05-30      14:34:26        MNGR1           SCLM.DEVT.SOURCE
```

In the next example, only one filter is used. It is for User and the filter value MNGR* means user IDs commencing with MNGR are displayed. The display appears as:

```
Prefix Values:
Date            Time            User            Data Set
*               *               MNGR*           *

Violations List:
Date            Time            User            Data Set
2002-06-04      08:42:18        MNGR3           SCLM.DEVT.SOURCE
2002-06-04      08:39:40        MNGR3           SCLM.DEVT.SOURCE
2002-05-31      12:02:16        MNGR2           SCLM.TEST.SOURCE
2002-05-30      14:34:26        MNGR1           SCLM.DEVT.SOURCE
```

In the next example, two filters are used. Both must be satisfied for the violation to appear in the display.

Prefix filters

```
Prefix Values:
Date            Time            User            Data Set
*               *               MNGR*           SCLM.TEST*

Violations List:
Date            Time            User            Data Set
2002-05-31      12:02:16        MNGR2           SCLM.TEST.SOURCE
```

If no access rules satisfy the prefix filters, then the display will look like this:

```
Prefix Values:
Date            Time            User            Data Set
2001*           *               MNGR*           SCLM.TEST*

Violations List:
Date            Time            User            Data Set
```

Violations List

The Violations List is a tabular display of Enhanced Access Control for SCLM access violations that may be selected for reporting. The list is displayed in reverse chronological sequence, and prefix filters may be used to limit the display. The fields include:

Option
: Any option value entered indicates that the violation entry details should be displayed.

Date
: The date the access violation occurred. This is a display field that cannot be overtyped. The date has the format YYYY-MM-DD.

Time
: The time the access violation occurred. This is a display field that cannot be overtyped. The time is based on a 24 hour clock in the format HH:MM:SS.

User
: The RACF user ID of the person that obtained the access violation. This is a display field that cannot be overtyped. The user ID is always shown, even if Enhanced Access Control for SCLM performed its access control validation against the user's RACF group.

Data Set
: The access violation data set. The fully qualified data set name, without quotes, will appear, even if a generic Profile was used to determine access privileges. This is a display field that cannot be overtyped.

  The displayed value is the entity against which RACF performed access validation. It most cases this will be a data set name as access requests usually relate to data set open requests. However in some circumstances, RACF may be validating access to a Profile definition itself (not a data set access), therefore the entity name will appear as a Profile name and not a data set name.

The Violation Detail panel

The Violation Detail panel displays information regarding a specific access control violation.

```
 Menu  Settings  Help
 ──────────────────────────────────────────────────────────────────────────
                                Violation Detail
Command ===> ____________________________________________________________

Validation Details
  Profile used  . . : SCLM.DEVT.SOURCE
  Application . . . : SCLM
  Function  . . . . : EDIT
  User or Group . . : MNGR3
  Violation reason  : Reason code 09: The user's assigned privilege within
  the matching Application was less than that required to perform the data
  set open.

Violation Details
  Data set  . . . . : SCLM.DEV1.SOURCE
  Date  . . . . . . : 2002-06-04          Time  . . . . . . : 09:05:49
  User  . . . . . . : MNGR3               Group . . . . . . : MNGRS
  Access required . : UPDATE              Access granted  . : NONE

Enter '/' to select option
  / Display program chain details

 F1=Help    F3=Exit   F12=Cancel
```

Figure 15. The Violation Detail panel

The panel has three parts:

Validation Details	Information on how Enhanced Access Control for SCLM validated the access request.
Violation Details	Information on the access request, including RACF user ID and Group information.
Display Programs	Requests that further information be displayed about the programs that were in use when the violation occurred.

Profile used

Profile Used identifies the Profile definition selected by Enhanced Access Control for SCLM for the access control validation. This Profile matches either the data set name against which the access request was made, or the generic Profile definition matching that used by RACF for its validation purposes. The name is displayed fully qualified, without quotes.

Application

Application identifies the Application definition selected by Enhanced Access Control for SCLM for the access control validation.

If Application is blank, then an Application match was not found, or the validation failed before Application matching was performed.

Function

Function identifies the Function of the Application definition selected by Enhanced Access Control for SCLM for the access control validation.

If the Function Used is blank, then one of the following conditions is true:
- An Application/Function match was not found

- The validation failed before Application matching was performed
- The matching Application did not have a value for the Function field

User or Group

Identifies the user ID selected by Enhanced Access Control for SCLM for the access control validation, or whether a RACF current-connect Group list entry was selected.

The User or Group field displays one of these values:

<user ID> The RACF user ID of the user associated with the data set access request. This user ID was matched within the Profile access rules.

GRPLIST One or more of the user's RACF current-connect Groups were matched in the Profile access rules. The Group field in the Violation Detail panel displays the user's first RACF current-connect Group. Use RACF services to determine other user-to-Group connections.

***** The Profile access rule * (all users) definition was matched.

blank A match of the user ID or a RACF current-connect Group was not found in the Profile access rules, or the validation failed before user ID matching was performed.

Violation Reason

The reason why Enhanced Access Control for SCLM has failed the data set access control request. Chapter 6, "Problem Determination" on page 73 contains descriptions for each violation reason code.

Data set

The name of the data set validated by RACF in the access control request. The displayed value is the entity against which RACF performed access validation. In most cases this is a data set name, as access requests usually relate to data set open requests. However in some circumstances, RACF may be validating access to a Profile definition itself (not a data set access), therefore the entity name may appear as a Profile name and not a data set name.

Date

The date of the data set access request. Date is shown in the format YYYY-MM-DD.

Time

The time of the data set access request. Time is shown as a 24 hour clock in the format HH:MM:SS.

User

The user ID validated by RACF for the data set access control request.

Group

The RACF current-connect Group as validated by RACF for the data set access control request. If the user ID was connected to multiple RACF Groups, only the first of these is shown.

Access required

The access privilege required by RACF in order to satisfy the data set access request.

Access granted

The access privilege returned by Enhanced Access Control for SCLM to RACF for the data set access request. If RACF had already assigned an access privilege superior than those defined to Enhanced Access Control for SCLM, then the superior RACF privilege will be honored.

The Access Granted privilege is expected to be less than the Access Required privilege, otherwise the data set access would have been granted and a violation would not have occurred.

Display program chain details

Requests that another ISPF panel be displayed. This panel describes the execution program environment at the time of the data set access request.

You may enter any value in the option field area, and press ENTER to view the Violations Program panel.

The Violation Programs panel

The Violation Programs panel displays the execution program chain at the time of the data set access request. These were the programs in control when the data set access occurred.

```
  Menu  Settings  Help
 ------------------------------------------------------------------------------
                              Violation Programs               Row 1 to 5 of 5
Command ===> ____________________________________________________ Scroll ===> CSR


                           Program   Library Notes
                           ISPTASK   APF, TASKLIB
          High program --> ISRSCLM   APF, TASKLIB
                           FLMDDL    APF, TASKLIB
                           FLMED$    APF, TASKLIB
           Low program --> FLME$SAV  APF, TASKLIB
****************************** Bottom of data ********************************









  F1=Help      F3=Exit      F7=Backward  F8=Forward  F12=Cancel
```

Figure 16. The Violation Programs panel

The Execution Program List

The list shows the programs executing at the time of the data set access request. The list is displayed in top-down sequence; therefore the last program entry shown is the program that performed the data set access request. The program list is a chain of programs, for example:

```
   Program
1. ISPTASK      the TSO control program executed first
2. ISRSCLM      program 2 was invoked by program 1
3. FLMDDL       program 3 was invoked by program 2
4. FLMED$       program 4 was invoked by program 3
5. FLME$SAV     program 5 requested the data set access
```

High and Low Program indicators

If an Application match occurred, then the corresponding High and Low Programs are indicated to the left of the program list. The absence of these notes indicates that the Application was not matched, or the validation request failed before Application matching was performed.

Library Notes

Library Notes appear to the right of the program list. These notes provide additional information about how the execution program was invoked. The notes may have these values:

APF — The program was loaded from an APF library or common storage.

TASKLIB — A task library was active in order to locate and load the program. Task libraries are similar in concept to the STEPLIB libraries used in JCL. Note that the task library assignment passes (or cascades) to other programs executed under the same task.

To avoid tampering or compromise of the SCLM execution environment, Enhanced Access Control for SCLM performs various checks to ensure that the SCLM programs are invoked from an APF library, and that any utility programs executed by SCLM are themselves either APF or loaded from a task library assigned by SCLM. If the access request violation has occurred owing to failure of these environment checks, these Library Notes may be useful in understanding the cause of the failure.

Chapter 3. Enhanced Access Control for SCLM Definitions

Enhanced Access Control for SCLM supports two types of definitions: Applications and Profiles. These definitions combine to describe the data set resources to be controlled, and the way in which access may be gained.

The chapter has three parts:
- "Applications definitions" describes the Application definitions and suggests definitions for SCLM.
- "Profiles" on page 48 describes the Profile definitions and their associated access rules.
- "Validation Routine matching of Profile access rules" on page 51 describes how the Validation Routine uses the definitions to assign access privileges.

Applications definitions

The application definitions identify an SCLM program environment through which data set access may be granted. The applications are defined using the Enhanced Access Control for SCLM ISPF dialog. A list of suggested application definitions has been provided and may be loaded using a SHSSSAMP sample library member as described in "HSSRDEFN - Rule File definition JCL" on page 57. These applications are described in Appendix A, "Suggestions for Application Definitions" on page 111.

Here is an example of an application definition:

Application name: SCLM

Function name: PROMOTE

Data: This is the program environment for the SCLM Promote function.

High Program	Low Program
FLMCMD	FLMP
FLMS$SRV	FLMP
ISRSCLM	FLMP

Figure 17. An example application definition

The application definition has four parts:
- The Application Name identifies the Application definition.
- The Function Name allows the Application to be further distinguished. It is optional.
- The Data field allows the administrator to add textual information to the definition.
- Each High and Low Program pair describes an SCLM program environment.

Application name

The application name combined with the function name uniquely identifies the application definition. The administrator assigns the application name. It should reflect the purpose or nature of the SCLM program environment that it describes.

Application name

The Application name is a maximum of 8 characters in length and conforms to the naming conventions used for programs: it may consist of the National characters A to Z, 1 to 9, @, # and $, and the first character may not be numeric. An application name must be supplied—it cannot be left blank.

Applications may be classified into functions. This provides flexibility in the way the application definitions are named. The combination of application and function name is referenced in the profile access rules.

"Application considerations for writing Profile access rules" on page 46 considers further the use of Application names.

Function name

The function name combines with the application name to uniquely identify an application definition. The function name is optional, and may be left blank. The administrator assigns the function name. It should reflect the purpose or nature of the SCLM program environment that it describes.

If entered, the function name is a maximum of 8 characters in length and conforms to the naming conventions used for programs: it may consist of the National characters A to Z, 1 to 9, @, # and $, and the first character may not be numeric.

The application and function names are referenced in the profile access rules.

Data

The Data field lets the administrator store information text along with the application definition. The content of the Data field is not validated or used by Enhanced Access Control for SCLM in any way. You may use this field to contain notes or explanations regarding the use of the application definition. Use of the Data field is optional and it may be left blank.

If entered, the Data field allows three lines of 75 characters to be entered. As the Data field is not validated, it may contain any value.

High and Low Program

The High and Low Programs are defined as pairs. Each pair describes an SCLM program environment, where the High Program indicates the SCLM controlling program (and therefore the method of SCLM invocation) and the Low Program indicates the SCLM service through which access will be controlled.

The High Program is optional. If left blank, the High Program defaults to ISPTASK for validation purposes. As SCLM executes as a TSO/ISPF application, all of the SCLM high-level programs execute under the control of ISPTASK.

The Low Program is mandatory and must be specified.

The High and Low Programs are paired. Therefore in the following example, there are only two pairs. The combination of ISRSCLM and FLMB is invalid, and has not been defined as a High and Low Program pair.

Pair	High Program	Low Program
1	FLMCMD	FLMB
2	ISRSCLM	FLMP

Both the High Programs and the Low Programs conform to the standard naming conventions for programs. The program name is a maximum of 8 characters in length, it may consist of the National characters A to Z, 1 to 9, @, # and $, and the first character may not be numeric.

Understanding applications

The application is a definition of a program environment. It describes a sequence of execution programs or program pathway that must be used in order to gain access to a data set.

Figure 18 illustrates two alternative methods of editing a data set: on the left, editing via SCLM; on the right, editing via the native ISPF editor.

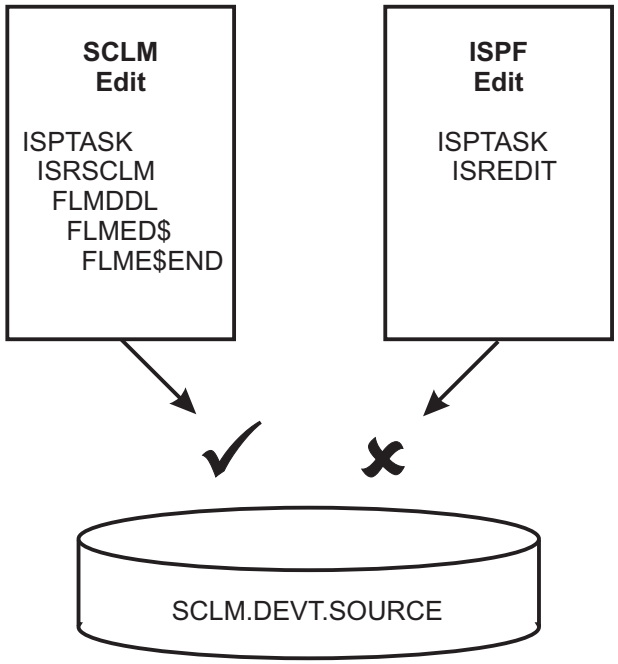


Figure 18. SCLM Edit and ISPF Edit Programs

You can see in Figure 18 that SCLM requires a series of programs to be executed in sequence in order to perform the edit service. The first program is ISPTASK. This program works in conjunction with TSO to provide an ISPF environment. Next, the SCLM controlling program ISRSCLM executes under the control of ISPTASK. The ISRSCLM program in turn uses other SCLM programs which eventually lead to FLME$END. This is the SCLM program that opens the data set for output and saves the changes when you END or EXIT from the edit session.

On the other hand, the ISPF Editor uses a different sequence of programs. The program ISREDIT performs the data set access. The SCLM programs do not appear in the program execution path at all as the ISPF editor program does not use them.

By defining the execution program pathway to Enhanced Access Control for SCLM, you may grant access via approved methods and conversely limit access through other methods. The purpose of the Application definition is to provide flexibility in the way these program pathways are specified.

Enhanced Access Control for SCLM does not require you to define all the programs in the execution pathway. Instead, the Application defines only the start and end of the path. The start is called the High Program and the end is called the Low Program.

The pairing of High and Low Programs defines a specific program environment. This program environment is called an Application, because it is by means of this Application that data access is granted.

What is the High Program?

In order to limit data access via a program pathway, you need to define the start and end of the path. Note that the application definitions need only define the SCLM portion of the program pathway.

Figure 19 illustrates an SCLM Promote running as a batch job. It shows a program pathway that includes the TSO startup programs under which SCLM runs, the SCLM programs themselves, and finally any utilities that SCLM itself may invoke.

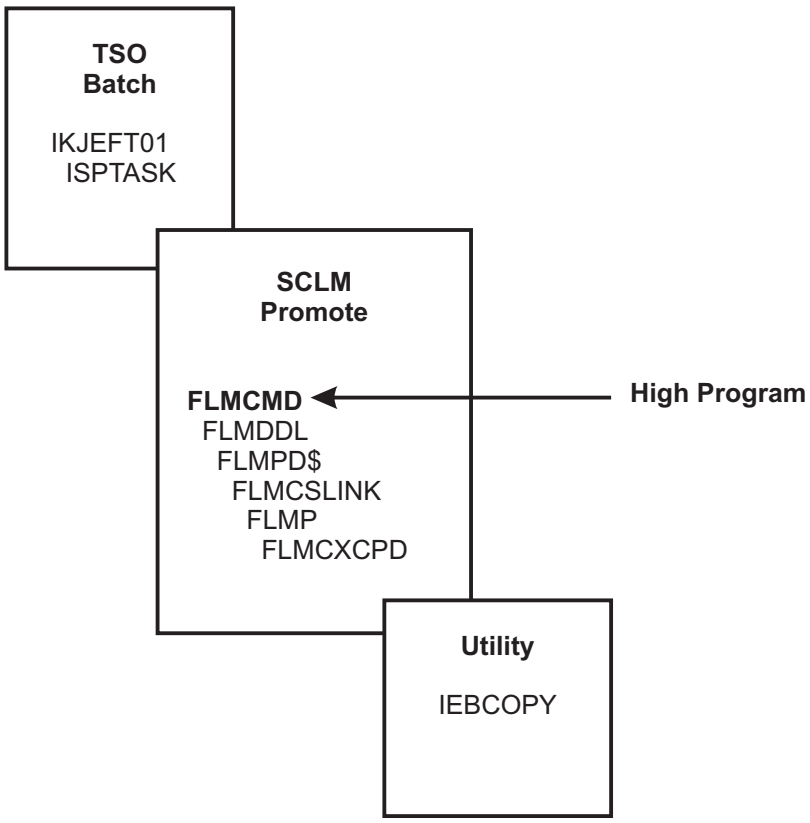


Figure 19. SCLM batch Promote execution program pathway

You may have noted that in Figure 19 the controlling SCLM program is FLMCMD, whereas in Figure 18 for an SCLM Edit the controlling SCLM program was ISRSCLM. SCLM employs different controlling programs depending upon how it is invoked:

SCLM Control Program	Method of Invocation
FLMCMD	Command interface, invoked via batch TSO, TSO command, CLIST, or REXX.

FLMS$SRV	Call interface, invoked via SCLM subroutine interface program FLMLNK.
ISRSCLM	Online panel interface, invoked via online TSO/ISPF.

The purpose of the Application High Program definition is to ensure that the program issuing the data set access request was invoked in the correct manner. The High Program defines the SCLM controlling program, whereas the Low Program defines the SCLM service program.

Enhanced Access Control for SCLM is flexible in the way High Programs are defined. FLMCMD, FLMS$SRV and ISRSCLM may all be defined as High Programs within the one application definition. However, you may separate these into different applications to achieve greater control over the way SCLM is used. For example, you may define applications for the each of the SCLM interfaces: the FLMCMD command interface; the FLMLNK subroutine or 'called' interface; and ISRSCLM, the ISPF online interface. Consider the following examples of four applications and their associated High Program definitions.

Application:	SCLM	Command	Called	Online
Function:	Promote	Promote	Promote	Promote
	High Programs	**High Program**	**High Program**	**High Program**
	FLMCMD	FLMCMD	FLMS$SRV	ISRSCLM
	FLMS$SRV			
	ISRSCLM			

Based on these Applications, if a user was granted data set access via SCLM Promote, then the access would be granted using any of the SCLM invocation interfaces. If the user was granted data set access via Called Promote only, then promotions could be limited to cases where an application program invokes SCLM via the FLMLNK subroutine interface.

The Enhanced Access Control for SCLM Validation Routine ensures that all of the SCLM programs from the Application High Program to the Low Program have been loaded from APF libraries, and in the correct manner.

These checks avoid incorrect copies of the SCLM programs from being used. Therefore, the High Program definition serves another purpose: it determines the starting point for verifying the program execution environment.

To ensure maximum protection, the High Program should specify the first or highest SCLM program that receives control. In the case of the FLMLNK subroutine interface program, the FLMS$SRV program is used because FLMLNK may be link edited into the calling application program and therefore not appear in the execution program pathway.

The High Program may be omitted (or left blank) in the application definition. If this occurs, Enhanced Access Control for SCLM validates that all programs above the Low Program up to and including the ISPTASK program are loaded from APF libraries or common storage. However, the use of ISPTASK as the High Program is not recommended, as this may extend the validation checks for APF loading onto other programs that call SCLM services via the FLMLNK interface.

What is the High Program?

Here are the programs that should be considered for the SCLM Application High Program definitions.

High Programs	Comments
FLMCMD	the SCLM Command Interface control program
FLMS$SRV	the SCLM FLMLNK Call or Subroutine Interface control program
ISRSCLM	the SCLM online TSO/ISPF control program

Figure 20. SCLM High Programs

In summary, these points of significance apply to the High Program definition:

1. All programs from the High Program to the Low Program must be loaded from APF libraries or common storage.
2. The High Program should specify the first or highest SCLM program that receives control.
3. The High Program may be used to control how SCLM gained access, for example via online TSO/ISPF or by an application program calling SCLM services via FLMLNK.
4. The High Program may be left undefined, in which case all programs above the Low Program up to and including ISPTASK must be loaded from APF libraries or common storage.

What is the Low Program?

The Low Program defines the SCLM program providing a service or function. Just as the High Program marks the start of an execution program pathway, the Low Program marks the end of the pathway. The Low Program must be specified, and may not be left blank.

Figure 21 illustrates the execution program pathway for an SCLM Promote executed in batch.

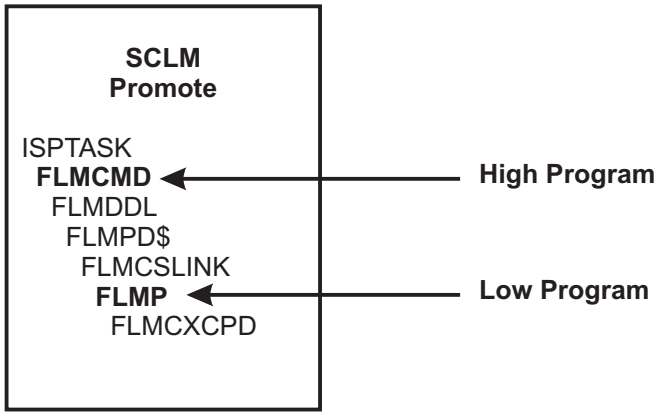


Figure 21. SCLM batch Promote High and Low Programs

The last program to be executed is IEBCOPY. This is an MVS utility that copies PDS members. It is used by SCLM, and is not part of the SCLM program suite. The IEBCOPY program was invoked by the SCLM program FLMCXCPD, which is itself an SCLM utility program used by a variety of SCLM services. The lowest SCLM program in the execution pathway that is specific to the SCLM Promote function is FLMP. The program FLMP is the SCLM Promote program.

By defining a high and low program pair of FLMCMD and FLMP, you are saying that the SCLM batch Promote service must execute a chain of programs that encompass FLMCMD through to FLMP. It is the chain of programs from FLMCMD to FLMP that uniquely defines this batch SCLM Promote function.

The Low Program should define the lowest program in the execution pathway that uniquely identifies the function or service being controlled. Programs higher than this may not identify sufficiently the SCLM process to be controlled.

Figure 22 illustrates the execution program pathways for member editing via SCLM. It shows that the SCLM Edit program FLMED$ uses other programs to update the data set when certain editor commands are performed.

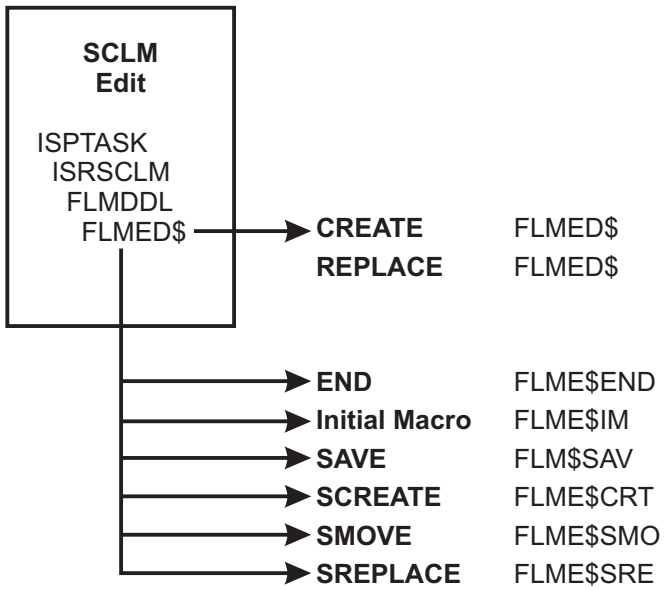


Figure 22. SCLM Edit execution program pathways

Although the program FLMED$ is common to all the edit commands it is an unsuitable candidate for the Low Program. FLMED$ permits CREATE and REPLACE command processing, whereas updates to SCLM-managed data sets should use their SCLM counterparts SCREATE and SREPLACE.

In Figure 23, the same concept is shown for the SCLM utilities. In this case, the SCLM utilities program FLMUDU$ appears in the execution program pathway for the three illustrated functions. However, the Migrate, Export, and Import functions each have their own SCLM program to provide the service.

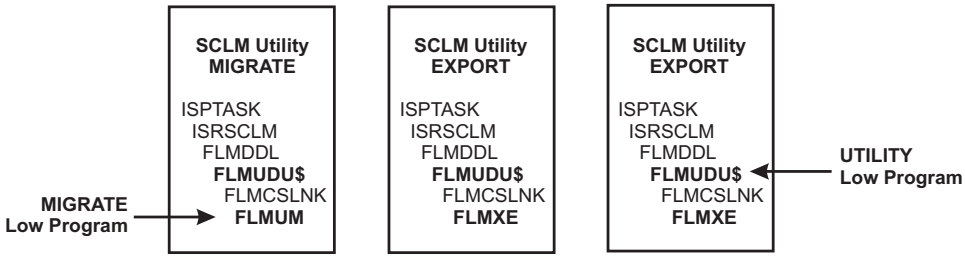


Figure 23. SCLM Utilities and Migrate, Export and Import functions

In this case, an Application with a Low Program definition of FLMUDU$ includes by extension Migrate, Export, and Import. However, other Applications, each with its own unique Low Program value, could be created specifically for Migrate, Export, or Import.

Consider a case where two applications with differing Low Programs both satisfy the execution program pathway. This is illustrated in Figure 24. In this case, an Application of SCLM UTILITY is defined with a Low Program of FLMUDU$. Another Application called SCLM IMPORT is defined with a Low Program of FLMXI.

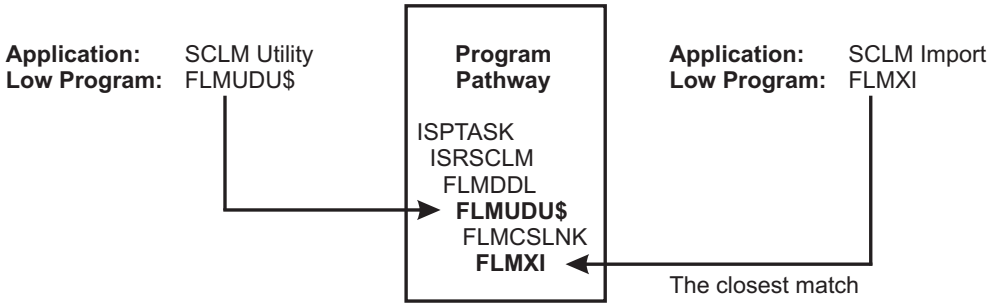


Figure 24. Execution Program Pathway matching two Low Programs

In the case illustrated in Figure 24, the closest Application match is for SCLM Import. The matching technique compares the execution programs from the bottom-up with the Low Programs defined in the Applications. Therefore, whichever Application has a Low Program either matching or nearest in the chain to the program that requested the data set access will be matched first. The technique for matching Applications is described more fully in "Matching the Application for validation" on page 53.

Figure 25 illustrates the case where the High Program and the Low Program are the same.

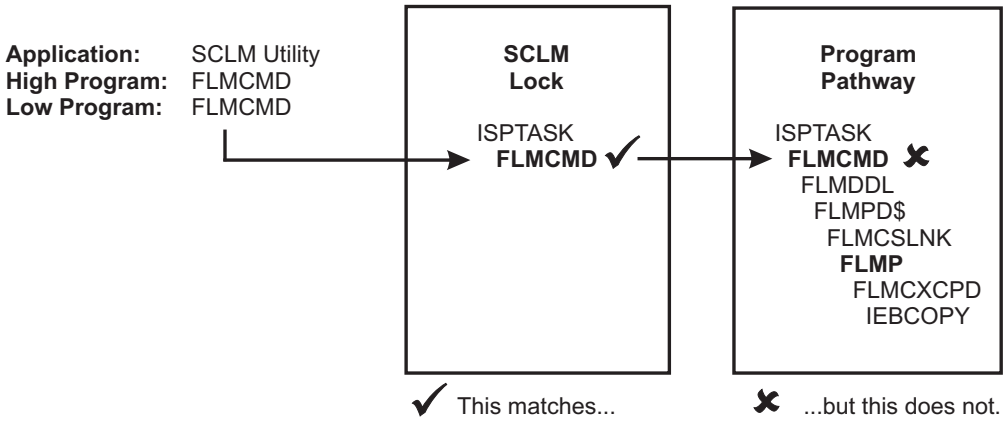


Figure 25. Where High and Low Programs are the same

If the Low Program and the High Program are the same, then only one program is defined as the SCLM execution program pathway. If this occurs, then the Low Program and only the Low Program may issue the data set request. If other programs appear following the Low Program, then the execution program pathway does not match the Low Program.

Figure 26 shows how the Low Program affects some of the program environment validation checks applied by the Enhanced Access Control for SCLM Validation Routine.

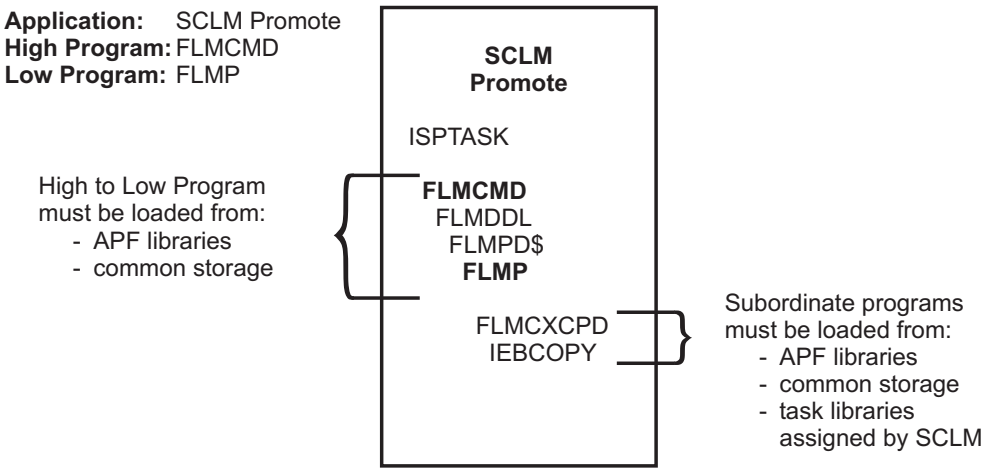


Figure 26. Environment checks triggered by the Low Program

The Validation Routine ensures that all of the programs from the application High Program to the Low Program have been loaded from APF libraries or common storage, and in the correct manner. It also verifies that any programs subordinate to the Low Program have been loaded from APF libraries, common storage or task libraries assigned by an approved higher level program. These checks avoid incorrect copies of SCLM and subordinate programs from being used.

Appendix B, “Summary of SCLM Services and High-Low Programs” on page 113 provides recommendations for Low Programs.

In summary, these points of significance apply to the Low Program definition:

1. The Low Program should define the lowest program in the pathway that uniquely identifies the function or service being controlled.
2. The Low Program is mandatory and must be specified.
3. Access via the Low Program includes by extension subordinate programs that it may execute, because the Low Program marks the end point in the execution program pathway.
4. The Low Program either matching or nearest in the pathway to the program that requested the data set access will be matched first.
5. If the Low Program has the same value as the High Program, then the Low Program must be the last program in the execution program pathway. Rule 3, which includes by extension subordinate programs, does not apply to this Low Program. Rule 4, which allows the Low Program to be near to but not exactly at the end of the program pathway, does not apply to this Low Program.
6. All programs from the High Program to the Low Program must be loaded from APF libraries or common storage.
7. All programs subordinate to the Low Program must be loaded from APF libraries, common storage, or a task library assigned by an approved higher-level program.

Application considerations for writing Profile access rules

The Enhanced Access Control for SCLM administrator determines how the SCLM High and Low Programs will be grouped into the Application definitions. As access to the Profile data sets are granted via the Applications, consideration must be given to the names of the Applications and the High and Low Programs they contain.

The names given to the Application and Function should reflect the purpose of the program combinations. This will add clarity to the Profile access rules that reference them. The Function name is optional. It provides flexibility in the way Applications are named.

The suggested Application/Functions referenced herein use SCLM for the Application name and assign a Function name to identify the SCLM service. You may prefer to use the Application name as the SCLM service and not use a Function name, in order to reduce effort when writing Profile access rules. Therefore:

The examples are like this...		but you could equally do this...	
Application	**Function**	**Application**	**Function**
SCLM	BROWSE	BROWSE	
SCLM	BUILD	BUILD	
SCLM	EDIT	EDIT	

The suggested Application/Functions definitions group both batch and online SCLM services into the one definition. If you want to limit SCLM access to either batch or online usage, then you may split the suggested Applications into multiple definitions. Therefore:

The examples are like this...		but you could equally do this...	
Application	**Function**	**Application**	**Function**
SCLM	BUILD	BATCH	BUILD
		ONLINE	BUILD

You may decide to create Application/Function definitions for specific types of SCLM users, like SCLM administrators or programmers. In this case, all the SCLM services performed by that type of user could be grouped into the one Application definition.

The examples are like this...		but you could equally do this...	
Application	**Function**	**Application**	**Function**
SCLM	IMPORT	SCLM	ADMIN
SCLM	MIGRATE		
SCLM	UTILITY		
SCLM	EDIT	SCLM	PGMRS
SCLM	BUILD		

If you have multiple SCLM projects that require differing Application definitions, you may categorize the Applications by project:

The examples are like this...

Application	Function
SCLM	IMPORT
SCLM	MIGRATE
SCLM	UTILITY

but you could equally do this...

Application	Function
PROJECT1	ADMIN
PROJECT2	ADMIN

Application definitions should be of sufficient granularity to distinguish the SCLM services to be controlled. If many SCLM services are defined within the one Application definition, then access to all these services will be granted when that Application is granted access within the Profile access rule. To limit access to a specific SCLM service, your Profile access rules need to reference an Application that defines that SCLM service uniquely.

Consider this Application definition. It contains High and Low Program pairs for online SCLM editing, and the FLMCMD Build and Promote functions. If a Profile access rule granted access to a user via this Application, then the access applies regardless of which of the three SCLM services was used.

Application: SCLM

Function: Example

Data: This is an example of an Application that defines multiple High-Low program pairs.

High Programs	Low Programs	
FLMCMD	FLMB	This is the SCLM batch Build function
FLMCMD	FLMP	This is the SCLM batch Promote function
ISRSCLM	FLME$CRT	This is the SCLM online Edit (SCREATE) function
ISRSCLM	FLME$END	This is the SCLM online Edit (END) function
ISRSCLM	FLME$IM	This is the SCLM online Edit (Initial Macro) function
ISRSCLM	FLME$SAV	This is the SCLM online Edit (SAVE) function
ISRSCLM	FLME$SRE	This is the SCLM online Edit (SREPLACE) function
ISRSCLM	FLMB	This is the SCLM online Build function
ISRSCLM	FLMP	This is the SCLM online Promote function

Avoid multiple Applications that share the same High and Low Program definitions. If a data set Profile has access rules that references two Applications that share the same High and Low Program definitions, then both Applications equally apply. If this occurs, then Enhanced Access Control for SCLM applies the highest privilege that is required to satisfy the data set access request.

Consider this Profile access rule example, where the Applications SCLM-ALL and SCLM-EDIT both include the High-Low program combinations for the SCLM edit service.

Case	Application	Function	User/Group	Access
1	SCLM	ALL	*	UPDATE
2	SCLM	EDIT	*	NONE

As the Applications for both Case 1 and Case 2 are satisfied, Enhanced Access Control for SCLM assigns the higher privilege of UPDATE from Case 1. You can avoid confusion by ensuring that your Applications definitions do not share the same High and Low Program combinations; or by ensuring that the access rules for a Profile do reference two or more Applications with the same High and Low Program pairs.

As a general rule, use RACF to grant READ access to the SCLM related data sets, as this greatly simplifies rule writing. It avoids preparing Application definitions for the many program combinations that may read a data set, such as compare utilities, search-for-string utilities, copy utilities, and so forth. The requirement to authorize the read program's load library is also removed. Reading using a specific execution program pathway is not required usually, as the data cannot be damaged.

Refer to Appendix A, "Suggestions for Application Definitions" on page 111 to view the list of suggested applications.

Profiles

The Profile definition identifies a data set or a RACF generic data set Profile to be to be validated by Enhanced Access Control for SCLM. The Profiles are defined via the Enhanced Access Control for SCLM ISPF Dialog.

Here is an example of a profile definition:

Profile name: SCLM.DEVT.SOURCE
Data: This is a discrete Profile for the data set SCLM.DEVT.SOURCE.
Access Rules:

Application	Function	User/Group	Access
SCLM	BUILD	PGMRS	UPDATE
SCLM	EDIT	PGMRS	UPDATE
SCLM	PROMOTE	MNGRS	UPDATE

Figure 27. Example Profile definition

The Profile definition has three parts:

- The Profile Name identifies the data set or RACF generic data set Profile to be validated.
- The Data field allows the administrator to add textual information to the definition.
- Each access rule line identifies a set of conditions by which data set access is granted.

Profile name

The profile name matches the name of a data set or a RACF generic data set profile that is to be validated by Enhanced Access Control for SCLM. The profile name is a maximum of 44 characters and is entered as a fully qualified data set name without quotes.

Discrete Profiles

Describe a single data set, and the Profile name must match exactly the data set name. A discrete Profile may be defined and validated by Enhanced Access Control for SCLM, even though a corresponding RACF discrete Profile is not defined. Enhanced Access Control for SCLM validates against a matching discrete Profile, even though RACF performed its validation against a generic data set Profile.

Generic Profiles

Describe a RACF generic data set Profile that will also be validated by Enhanced Access Control for SCLM. The Profile name must match exactly the RACF generic data set Profile name.

RACF access request validation may select a generic data set Profile for validation purposes. If this occurs, then Enhanced Access Control for SCLM validates against its own Profile definitions in this sequence:

1. If defined, a discrete data set Profile takes precedence over generic Profiles.
2. If defined, the generic data set Profile exactly matching that used by RACF will be used.

Data

The Data field allows the administrator to store information text along with the Profile definition. The content of the Data field is not validated or used by Enhanced Access Control for SCLM in any way. You may use this field to contain notes or explanations regarding the use of the Profile definition. Use of the Data field is optional and it may be left blank.

If entered, the Data field allows three lines of 75 characters to be entered. As the Data field is not validated, it may contain any value.

Access Rules

The access rules are a series of definitions that govern access to the data sets controlled by the Profile. The rules contain three elements:

- The Application and Function determine conditions under which access is granted
- The User/Group determines who should be given access
- The Access determines the level of access privilege to be assigned.

Values must be entered for Application, User/Group and Access. The combination of Application and Function must be previously defined to Enhanced Access Control for SCLM, therefore the Function value must be valid for the chosen Application.

Enhanced Access Control for SCLM matches the Profile access rules with the execution conditions at the time of data set access request. This matching technique is described in “Validation Routine matching of Profile access rules” on page 51. However, Enhanced Access Control for SCLM uses a most-specific to least-specific matching technique as follows:

1. Match the Profile, where discrete data set Profiles takes precedence over generic Profiles.
2. Match the Application; where Low Programs matching or nearest in the execution chain to the program requesting the data set access take precedence.
3. Match the user, where user IDs take precedence over RACF Groups which take precedence over * for all users.

4. Assign the access privilege of the matched access rule, or the higher privilege if multiple access rules are matched.

How access rules work may be best understood by an example. Consider Figure 28 to see how the access rules might apply to FRED, a programmer in the RACF Group PGMRS.

Case	Application	Function	User/Group	Access
1	SCLM	BROWSE	*	READ
2	SCLM	BROWSE	MNGRS	NONE
3	SCLM	BROWSE	PGMRS	READ
4	SCLM	EDIT	FRED	NONE
5	SCLM	EDIT	PGMRS	UPDATE

Figure 28. Example of Profile access rules

If FRED in RACF Group PGMRS tries to perform an SCLM edit of the data set, then case 4 applies so NONE is the assigned privilege. Others in the RACF Group PGMRS would satisfy case 5 and receive the UPDATE privilege. If FRED tried an SCLM browse of the data set, then case 3 applies and he would receive the READ privilege.

Reconsider the example access rules in Figure 28 for how they might apply to JOHN a manager in RACF Group MNGR.

If JOHN in RACF Group MNGRS tries an SCLM edit of the data set, then neither his user ID, RACF Group or * (all users) can be matched within the SCLM EDIT Application and Function. Therefore, as no matching case is found JOHN receives no privileges from Enhanced Access Control for SCLM. A RACF data set violation occurs.

If JOHN tries an SCLM browse of the data set, then case 2 is matched, and JOHN receives the NONE privilege. Note that case 1 for user * or all users, will not override the more specific access rule of case 2.

Enhanced Access Control for SCLM matches the execution conditions of data set access request with the Profile access rules. This matching technique is described in “Matching the Application for validation” on page 53. When the access rule match is selected, the user will be assigned the appropriate access privilege.

When entering access rules via the ISPF Dialog, rule duplicates discarded. Rule conflicts are reported for immediate resolution. A rule conflict occurs when the combination of Application, Function and User/Group are identical for two or more rules, but the Access field privileges differ.

Application

This field combines with the Function name to identify an Application definition. The Application, by means of its High and Low Program definitions, describes a program environment through which access to the data set is controlled.

The Application field is a maximum of 8 characters, and it may not be left blank. The combination of Application and Function name must be previously defined to Enhanced Access Control for SCLM as an Application definition. If the Profile

access rules reference an Application definition that is subsequently deleted, then the Rule Load Utility will issue warning messages and discard that access rule when the in-memory rules are loaded.

Function

The field combines with the Application name to identify an Application definition. Only Function names valid for the Application may be entered. If this field is left blank, then this indicates that its corresponding Application definition also has a blank value for the Function field. The Function field is a maximum of 8 characters.

User/Group

The User/Group field identifies the users to whom the access rules apply. This is an 8 character field that must be specified, it may not be left blank. The access rule applies when the field value matches:

1. The RACF User ID in use when the access was attempted.
2. A RACF Group to which the user was currently connected at the time the access was attempted.
3. The value * to designate all users.

Access

The Access field identifies the access privilege associated with the rule. This is an 8 character field the value of which corresponds to the RACF data set access privileges as shown below:

None	Allows no data set access
Read	Allows data set read access
Update	Allows data set read and write access
Control	Allows data set read and write access, including VSAM improved control interval processing
Alter	Allows data set read, write, delete, rename, move and scratch access

Validation Routine matching of Profile access rules

The Validation Routine assigns access privileges to a data set, based on the Profile access rules. After RACF validation is complete, it invokes the RACHECK post-processing exit ICHRCX02. Enhanced Access Control for SCLM uses this exit to invoke its own validation routine.

If RACF denies access, then Enhanced Access Control for SCLM may grant access, as determined by the access rules and the execution conditions of the original access request. The Validation Routine:

1. Matches the Profile for validation based on the data set name referenced in the access request, or the RACF generic data set Profile used for validation.
2. Matches the program execution environment to the Applications defined within the Profile access rules.
3. Verifies that the program execution environment has not been corrupted or tampered, thereby compromising access controls.
4. Matches the user to the list of user IDs or RACF Groups defined for the matching Application within the Profile.
5. Assigns the access privilege according to the matched access rule.

Validation Routine matching of Profile access rules

The Validation Routine uses an in-memory copy of the Rule File for validation purposes. These in-memory rules are formatted by the Rule Load utility (HSSSSINT) to simplify and speed validation checking.

Matching the Profile for validation

When a RACF data set violation occurs, Enhanced Access Control for SCLM will check its in-memory rules firstly to see if the violation data set is controlled via one of its discrete data set Profile definitions. Enhanced Access Control for SCLM will always perform its validation using the discrete data set Profile if one is defined.

If a discrete data set Profile is not found, then Enhanced Access Control for SCLM examines the Profile name used during RACF validation. If RACF validated against a generic data set Profile, then Enhanced Access Control for SCLM checks its in-memory rules for a generic data set Profile of the same name. If such a Profile is not found, then Enhanced Access Control for SCLM has not matched on either a discrete or a generic Profile, so control is relinquished to RACF and a data set access violation will occur. If a matching generic data set Profile is found, then Enhanced Access Control for SCLM will use this Profile for validation purposes.

This list summarizes the rules governing Profile selection by the Enhanced Access Control for SCLM Validation Routine:

1. Enhanced Access Control for SCLM performs access validation only in cases where RACF has denied access.
2. A discrete data set Profile defined to Enhanced Access Control for SCLM will be used, even though a similar Profile is not defined to RACF.
3. Enhanced Access Control for SCLM will match the data set access against a discrete data set Profile before attempting to match a generic Profile.
4. The generic Profile matched by Enhanced Access Control for SCLM will be the same as that used by RACF when it performed data set validation.
5. If RACF validated the access request against a discrete data set Profile and not a generic Profile, then Enhanced Access Control for SCLM likewise will only match against a discrete Profile.
6. If Enhanced Access Control for SCLM cannot match the access request with either a discrete data set Profile or the same generic data set Profile as used by RACF, then the data set is considered to be uncontrolled. Further access validation is bypassed and a RACF data set violation occurs.

Consider the examples in Figure 29. An access request for the data set SCLM.TEST.SOURCE occurs, and RACF selects the generic data set Profile of SCLM.TEST.* for validation purposes. See how Enhanced Access Control for SCLM matches this against its Profiles in each of the three cases:

Access to data set: SCLM.TEST.SOURCE

RACF selected Profile: SCLM.TEST.*

Case 1		**Profiles**	**Notes**
	(1)	SCLM.TEST.SOURCE	(1) matched, as the discrete data set Profile
	(2)	SCLM.TEST.S*	
	(3)	SCLM.TEST.*	
	(4)	SCLM.*	
Case 2		**Profiles**	**Notes**
	(1)	SCLM.TEST.S*	
	(2)	SCLM.TEST.*	(2) matched, as the generic data set Profile

	(3)	SCLM.*
Case 3	**Profiles**	**Notes**
	(1) SCLM.TEST.S*	No matching Profile found
	(2) SCLM.	

Figure 29. Example of Profile matching

Discrete data set Profiles must be defined to Enhanced Access Control for SCLM as fully qualified data set names, otherwise the Profile names will not match the data set name. Likewise, generic data set Profiles must be defined to Enhanced Access Control for SCLM exactly as they have been defined to RACF, otherwise these also will not be matched.

Matching the Application for validation

The Validation Routine determines if the programs defined by the Application are active at the time of the data set access request. To accomplish this, the Validation Routine builds a list of the execution programs in reverse sequence to simplify matching. This is illustrated below for an SCLM Build, run in TSO foreground from the SCLM TSO/ISPF interface.

Execution Programs	**Notes**
FLMCXCPD	Check first the last program to be executed
FLMB	
FLMCSLNK	
FLMUDU$	
FLMDDL	
ISRSCLM	
ISPTASK	Check last the first program in the execution order.

Figure 30. Execution Program List for SCLM editing

This execution program list is then matched to the High and Low Program pairs as referenced by the Applications within the Profile access rules. Consider Figure 31. It illustrates the Low Program matching sequence in a case where the Profile access rules resolve to only two High and Low Program pairs: one with a Low Program of FLMP (SCLM Promote); and the other with a Low Program of FLMUDU$ (SCLM Utility functions).

	Low Programs		**Execution Programs**	**Low Program Match Sequence**	**Match Notes**
A.	FLMP	1	FLMCXCPD	1 to A, then 1 to B	
B.	FLMUDU$	2	FLMB	2 to A, then 2 to B	
		3	FLMCSLNK	3 to A, then 3 to B	
		4	FLMUDU$	4 to A, then **4 to B**	match found as **4 to B**
		5	FLMDDL	5 to A, then 5 to B	
		6	ISRSCLM	6 to A, then 6 to B	
		7	ISPTASK	7 to A, then 7 to B	

Figure 31. Example of Low Program matching

Matching the Application for validation

Low Program matching continues until a match is found. In Figure 31 on page 53, the fourth lowest execution program FLMUDU$ is matched with one of the Low Programs. When a Low Program match is found, its paired High Program is then checked against the remaining part of the execution program list, starting from the Low Program match point. This is illustrated in Figure 32.

	Low Programs	High Programs		Execution Programs	High Program Match Sequence	Match Notes
A.	FLMP	ISRSCLM	1	FLMCXCPD		
B.	FLMUDU$	ISRSCLM	2	FLMB		
			3	FLMCSLNK		
			4	FLMUDU$	ISRSCLM to 4	matching starts here
			5	FLMDDL	ISRSCLM to 5	
			6	ISRSCLM	ISRSCLM to 6	match found
			7	ISPTASK	ISRSCLM to 5	

Figure 32. Example of High Program matching

As illustrated in Figure 32, the High Program is matched. Therefore, the matching Application has been discovered. It contains the High and Low Program pair of ISRSCLM and FLMUDU$. The Validation Routine now progresses onto user matching for access to that Profile via the identified Application.

If the High Program was not matched, then this High and Low Program pair do not match. In addition, all other High and Low Program pairs sharing the same High Program can be ignored, as these also cannot be matched. The Low Program matching resumes. If the Execution Program chain is exhausted without finding a match, then no Applications satisfy the execution conditions, and Enhanced Access Control for SCLM will not grant access privileges.

If multiple Applications share the same Low Program value and these Applications are referenced within a Profile's access rules, then two or more Applications could match the program execution environment for the data set access request. This situation should be avoided as discussed in "Application considerations for writing Profile access rules" on page 46. However, if this occurs, then Enhanced Access Control for SCLM applies the highest privilege that is required to satisfy the data set access request.

Matching the User for validation

Before user matching is performed, the Validation Routine determines within the selected Profile the subset of access rules applicable for the matched Application. The user is then matched to this subset of access rules in this order of precedence:

User ID Use the access privilege assigned to the user ID first.

Group Use the access privilege assigned to any RACF current-connect Groups second.

***** Use the access privilege assigned to * (all users) third.

None If none of the above applies, then no match is found and the user is not granted any privilege.

Consider the example of access rules for the Application and Function of SCLM Edit, and how they might apply to user IDs FRED and JANE, both programmers

in RACF Group PGMRS.

Case	Application	Function	User/Group	Access
1	SCLM	EDIT	FRED	NONE
2	SCLM	EDIT	PGMRS	UPDATE
3	SCLM	EDIT	*	READ

If FRED tries to perform an SCLM edit of the data set then the user ID is matched with case 1, so NONE is the assigned access privilege. The user ID JANE is not explicitly defined, so this and other user IDs in the RACF Group PGMRS would satisfy case 2 and receive the UPDATE access privilege. For anyone else, case 3 would apply and they would receive the READ access privilege.

When matching is performed for RACF current-connect Groups, the user may be connected to multiple Groups at the same time. Therefore it is possible for two or more access rule cases to apply for the same user. If this occurs, Enhanced Access Control for SCLM honors the higher access privilege.

Consider the example of access rules for the Application and Function of SCLM Edit, and how they might apply to user ID JANE who is connected to two RACF Groups: MNGRS for managers; and PGMRS for programmers.

Case	Application	Function	User/Group	Access
1	SCLM	EDIT	FRED	NONE
2	SCLM	EDIT	MNGRS	READ
3	SCLM	EDIT	PGMRS	UPDATE

As the user ID JANE is not explicitly defined, matching will be performed using the RACF current-connect Groups. Both cases 2 and 3 apply, as JANE is connected to RACF Groups MNGRS and PGMRS. However, the SCLM Edit function requires UPDATE access, and case 2 for RACF Group MNGRS only assigns the READ privilege. Therefore, case 3 for RACF Group PGMRS becomes the matched Group because the higher of the access privileges is honored.

The matching of user ID, RACF current-connect Group, or * (all users) applies within the subset of access rules for the matched Application. Consider the next example of access rules for varying Applications, and how they might apply to user IDs FRED and JANE, both programmers in RACF Group PGMRS, and user ID BOB a manager in RACF Group MNGRS.

Case	Application	Function	User/Group	Access
1	SCLM	BROWSE	*	READ
2	SCLM	EDIT	FRED	NONE
3	SCLM	EDIT	PGMRS	UPDATE
4	SCLM	PROMOTE	MNGRS	UPDATE

If an SCLM Browse was performed, then only the access rule for case 1 applies. User matching is performed against the access rule User/Group value * (all users). Therefore FRED, JANE and BOB receive the READ privilege.

If an SCLM Edit was performed, then access rule cases 2 and 3 apply. User matching is performed against the access rule User/Group values of FRED (case 2) and PGMRS (case 3). User ID FRED would match case 2 and receive NONE as the access privilege. The user ID JANE is not defined in case 2 or 3, so RACF Group

matching occurs and PGMRS is matched with case 3. JANE is assigned the UPDATE privilege. User ID BOB is also not defined in case 2 or 3, nor is his connected RACF Group MNGRS, nor is * (all users). Therefore, BOB does not match any of the SCLM Edit access rules and receives no privileges.

The rules for matching for user ID, RACF Group, or * still apply, even when multiple Applications match the execution condition. In the next example, assume that the Applications definitions for SCLM ALL and SCLM EDIT have a common High and Low Program pair that has been matched, and therefore both Applications match. Consider how the access rules might apply to user IDs FRED and JANE, both programmers in RACF Group PGMRS.

Case	Application	Function	User/Group	Access
1	SCLM	ALL	PGMRS	READ
2	SCLM	EDIT	FRED	NONE
3	SCLM	EDIT	PGMRS	UPDATE

If FRED attempts an SCLM edit of the data set, then the user ID of FRED in case 2 takes matching precedence over his RACF current-connect Group of PGMRS in cases 1 and 3. Therefore, case 2 applies and FRED is assigned the access privilege of NONE.

However, user ID JANE is not explicitly referenced in the access rules, and her RACF Group PGMRS satisfies both cases 1 and 3. In these cases the higher access privilege is assigned, so JANE matches case 3 and she will receive UPDATE access.

Assigning the access privilege

The access privilege assigned to a user is determined by the Profile and access rule matched by the Validation Routine. Consider the example of access rules and how they might apply to user ID FRED a programmer in RACF current-connect Group PGMRS.

Case	Application	Function	User/Group	Access
1	SCLM	BROWSE	*	READ
2	SCLM	EDIT	FRED	NONE
3	SCLM	EDIT	PGMRS	UPDATE

If user ID FRED attempts an SCLM browse of the data set, the Validation Routine matches on case 1, so the access privilege of READ is assigned. If FRED attempts an SCLM edit of the data set then case 2 matches, so the access privilege of NONE is assigned.

When writing Profile access rules via the ISPF dialog, duplicated entries with conflicting access privileges are prohibited.

Application	Function	User/Group	Access	Notes
SCLM	EDIT	FRED	NONE	Conflicts prohibited
SCLM	EDIT	FRED	UPDATE	Conflicts prohibited

If multiple Applications match the execution programs for the data set access request, then it is possible for two or more access rule cases to apply for the same user. If this occurs, Enhanced Access Control for SCLM honors the higher access privilege. This situation is discussed in “Matching the Application for validation” on page 53.

Chapter 4. Utilities and Sample Library

The Enhanced Access Control for SCLM Sample Library (SHSSSAMP) contains sample JCL members to allocate, initialize and load the Rule File. A sample SMP/E ++USERMOD to install the HSSRCX02 Validation Routine Interface into the RACF RACHECK Post Processing exit ICHRCX02 is also provided.

The members provided in the Sample Library (SHSSSAMP) are:

HSSRDEFN JCL to define and initialize the Rule File

HSSRLOAD JCL to run the HSSSSINT Rule Load Utility

HSSUMOD1 Sample SMP/E ++USERMOD for the installation of the HSSRCX02 Validation Routine Interface into the RACF RACHECK Post Processing exit.

The Enhanced Access Control for SCLM Rule Load Utility, HSSSSINT, is also described in this chapter. It loads the Rule File into memory for use by the Validation Routine.

HSSRDEFN - Rule File definition JCL

This JCL defines the Rule File, and then initializes the file by loading suggested Application definitions from the AHSSDATA file. This JCL is found in the SHSSSAMP data set member HSSRDEFN.

HSSRDEFN - Rule File definition JCL

```
//HSSJOB   JOB
//* ********************************************************************
//* **     Enhanced Access Control for SCLM FOR Z/OS AND OS/390      **
//* **           VERSION 1, RELEASE 1, MODIFICATION 0                **
//* ********************************************************************
//* **               H278110 - V1R1M0 5697-H59                       **
//* **           (C) COPYRIGHT FUNDI SOFTWARE. 2002                  **
//* **      LICENSED MATERIALS - PROPERTY OF FUNDI SOFTWARE.         **
//* **                   ALL RIGHTS RESERVED.                        **
//* **        USE, DUPLICATION OR DISCLOSURE RESTRICTED.             **
//********************************************************************
//*                                                                  *
//*  NAME = HSSRDEFN                                                 *
//*                                                                  *
//*  DESCRIPTIVE NAME = Rule File Allocation JCL                     *
//*                                                                  *
//*  FUNCTION = This JCL allocates and initializes a Rule File.      *
//*                                                                  *
//*  NOTE    1. Customize the job by adjusting the IDCAMS volume     *
//*             parameters to reflect your installation standards.   *
//*             If SMS is in used, this will require replacement     *
//*             of the VOLUME parameter with the SMS equivalents.    *
//*                                                                  *
//*          2. Customize the job by setting the following symbols:  *
//*                                                                  *
//*             rule.file : The Rule File, a VSAM KSDS data set that *
//*                         is used to store Enhanced Access Control *
//*                         for SCLM definitions.                    *
//*                                                                  *
//*             volser    : The volume serial number onto which the  *
//*                         Rule File will be allocated.             *
//*                                                                  *
//*             smpetarg  : The SMP/E target library high level      *
//*                         qualifier.                               *
//*                                                                  *
//*          3. Customize the job to either initialize the Rule File, *
//*             or to also load the suggested Application definitions.*
//*                                                                  *
//*             The AHSSDATA member used for the IDCAMS REPRO in the *
//*             INIT job step determines the method of initialization.*
//*             Choose one of the following members for the AHSSDATA *
//*             data set:                                            *
//*                                                                  *
//*             HSSRULES  : This is the default within the JCL. It   *
//*                         will both initialize the Rule File and   *
//*                         load the suggested Application           *
//*                         definitions.                             *
//*                                                                  *
//*             HSSRINIT  : This will initialize the Rule File only. *
//*                         Suggested Application definitions will   *
//*                         not be loaded into the Rule File.        *
//*                                                                  *
//********************************************************************
```

Figure 33. HSSRDEFN Sample JCL to define the Rule File (Part 1 of 2)

```
//*
//DEFINE   EXEC PGM=IDCAMS
//*
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
 DEFINE CLUSTER(NAME(<rule.file>)              -
          INDEXED                              -
          KEYS(45 0)                           -
          RECORDSIZE(20000 32654)              -
          CYLINDERS(2 1)                       -
          SHAREOPTIONS(3,3)                    -
          FREESPACE(10 10)                     -
          REUSE                                -
          VOLUMES(<volser>))                   -
        DATA(NAME(<rule.file>.DATA)            -
          CONTROLINTERVALSIZE(32768))          -
       INDEX(NAME(<rule.file>.INDEX))
/*
//*
//INIT     EXEC PGM=IDCAMS,COND=(0,NE)
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  *
 REPRO INFILE(INDD) OUTFILE(OUTDD)
/*
//OUTDD    DD  DISP=SHR,DSN=<rule.file>
//INDD     DD  DISP=SHR,DSN=<smpetarg>.AHSSDATA(HSSRULES)
/*
```

Figure 33. HSSRDEFN Sample JCL to define the Rule File (Part 2 of 2)

The Rule File is a VSAM KSDS data set. It is created using the IDCAMS utility, and must include an initialization record that has the character Z in column 1. The sample JCL both allocates and initializes the file for you. Comments heading the JCL provide tailoring instructions. Before using the JCL you should:

1. Adjust the JOB card.
2. Replace <rule.file> with the name of your Rule File.
3. Replace the IDCAMS volume related parameters to reflect your installation standards. If SMS data set allocation is in use, then omit the VOLUMES parameter and use the appropriate SMS parameters. If SMS is not in use, then replace <volser> with your volume serial number.
4. Replace <smpetarg> with the data set name prefix for the Enhanced Access Control for SCLM product installation libraries.
5. Adjust the AHSSDATA member name for the INDD file in the INIT job step to specify one of these members:

   HSSRINIT This member initializes the Rule File, however none of the suggested Application definitions are loaded. The resultant rule file is empty apart from the initialization record.

   HSSRULES This member both initializes the Rule File and loads the suggested Application definitions. Delete these Application definitions using the ISPF dialog, if you do not want them.

The sample JCL should complete with return code zero. If the JCL is to be run to redefine an existing Rule File, then the Rule File should be deleted prior to the execution of the job steps in the sample JCL.

HSSRLOAD - Rule Load Utility JCL

This JCL runs the HSSSSINT Rule Load Utility. This utility loads a copy of the Rule File into memory for use by the Validation Routine. This JCL is found in the SHSSSAMP data set member HSSRLOAD.

```
//HSSJOB   JOB
//* ********************************************************************
//* **     Enhanced Access Control for SCLM FOR Z/OS AND OS/390      **
//* **            VERSION 1, RELEASE 1, MODIFICATION 0               **
//* ********************************************************************
//* **               H278110 - V1R1M0 5697-H59                       **
//* **           (C) COPYRIGHT FUNDI SOFTWARE. 2002                  **
//* **      LICENSED MATERIALS - PROPERTY OF FUNDI SOFTWARE.         **
//* **                   ALL RIGHTS RESERVED.                        **
//* **        USE, DUPLICATION OR DISCLOSURE RESTRICTED.             **
//********************************************************************
//*                                                                  *
//*  NAME = HSSRLOAD                                                 *
//*                                                                  *
//*  DESCRIPTIVE NAME = Rule File load utility JCL                   *
//*                                                                  *
//*  FUNCTION = This JCL executes the HSSSSINT Rule Load Utility to:  *
//*                                                                  *
//*               1. Establish the MVS subsystem if required.        *
//*               2. Load the Validation Routine if required.        *
//*               3. Load the Rule File rules into memory.           *
//*                                                                  *
//*  NOTE       Customize the job by setting the following symbols:  *
//*                                                                  *
//*             hss#      : The MVS subsystem name used to anchor the *
//*                         addresses of the in-memory rules. If you  *
//*                         remove the SSID parameter, then the       *
//*                         default subsytem name of HSS# is assigned.*
//*                                                                  *
//*             rule.file : The Rule File. See sample HSSRDEFN for   *
//*                         allocation JCL for the Rule File.        *
//*                                                                  *
//*             smpetarg  : The SMP/E target library high level      *
//*                         qualifier.                               *
//*                                                                  *
//********************************************************************
//*
//RULELOAD PROC HSSLINK=<smpetarg>.SHSSLINK,
//             RULEFILE=<rule.file>,
//             SSID=<hss#>
//*
//RULELOAD EXEC PGM=HSSSSINT,PARM='SSID=&SSID'
//STEPLIB  DD DISP=SHR,DSN=&HSSLINK
//HSSLINK  DD DISP=SHR,DSN=&HSSLINK
//HSSRULES DD DISP=SHR,DSN=&RULEFILE
//SYSPRINT DD SYSOUT=*
//         PEND
//*
//LOAD     EXEC RULELOAD
```

Figure 34. HSSRLOAD Sample JCL to run the Rule Load Utility

The sample includes an in-stream JCL procedure to execute the HSSSSINT utility program. To simplify the job JCL, this in-stream procedure may be copied into a member within a JCL procedure library defined to your JES system.

The Profile definitions contained in the Rule File referenced in the JCL will be copied into memory and subsequently used for validating data set access requests.

Comments heading the JCL provide tailoring instructions. Before using the JCL you should:

1. Adjust the JOB card.
2. Replace <hss#> with the MVS subsystem-id Enhanced Access Control for SCLM will use. If you omit the SSID parameter from the JCL, then the subsystem name defaults to HSS#.
3. Replace <smpetarg> with the data set name prefix for the Enhanced Access Control for SCLM product installation libraries.

Refer to "HSSSSINT - Rule Load Utility" for further information regarding expected return codes and run notes.

HSSSSINT - Rule Load Utility

The Rule Load Utility copies the Profile and Application definitions into memory to speed access for the Validation Routine. It also activates the Enhanced Access Control for SCLM MVS subsystem and loads the Validation Routine into memory. The HSSSSINT utility:

1. Validates the contents of the Rule File Profile and Application definitions, issuing warning messages where appropriate.
2. Dynamically installs and initializes the MVS subsystem that Enhanced Access Control for SCLM will use.
3. Prepares reformatted Profile and Application definitions to speed access for the Validation Routine, and loads these into ECSA memory.
4. Loads the Validation Routine into memory.

HSSSSINT may be rerun at any time to refresh the in-memory rules used by the Validation Routine. The utility may be run while Enhanced Access Control for SCLM is in either an enabled or disabled state.

Following a system IPL, the Rule Load Utility must be run before Enhanced Access Control for SCLM becomes operative. Data access validations cannot occur until the Enhanced Access Control for SCLM subsystem is initialized; and the in-memory rules and Validation Routine are loaded.

"HSSRLOAD - Rule Load Utility JCL" on page 60 provides an example of the execution JCL.

Job Control Statements

Here are the job control statements for HSSSSINT:

JOB Starts the job.

EXEC Specifies the program name (PGM=HSSSSINT) or, if the job control statements reside in a procedure library, the procedure name. The EXEC statement may include an optional parameter, SSID.

HSSLINK DD
Defines the input load library that contains the Validation Routine. The DDname is required. Data set concatenation is not supported.

HSSRULES DD
Defines the input Rule File that contains the Profile and Application definitions. The DDname is required. Data set concatenation is not supported.

STEPLIB Defines the stepped load library from which the HSSSSINT program is loaded. HSSSSINT must be loaded from an APF library.

"Security and Administration Considerations" on page 8 recommends that access to the SHSSLINK library be restricted to the RACF or Enhanced Access Control for SCLM administrator, and that the library not be added to the MVS Linklist. This will necessitate the use of a STEPLIB or JOBLIB in the HSSSSINT execution JCL.

SYSPRINT DD
Defines an output sequential data set for messages. The data set can be written to a system output device, a tape or DASD volume, or dummy (DD DUMMY).

The EXEC statement can include an optional PARM parameter to specify the MVS subsystem-id to be used by Enhanced Access Control for SCLM. The syntax of the EXEC statement is:

```
//[stepname]   EXEC   PGM=HSSSSINT[,PARM='SSID=hss#]
```

where:

PGM=HSSSSINT
Specifies that you want to run the HSSSSINT program.

PARM='SSID=hss#'
Specifies the MVS subsystem-id to be used by Enhanced Access Control for SCLM. The subsystem-id is a maximum of 4 characters. It may consist of the National characters A to Z, 1 to 9, @, # and $; and the first character may not be numeric.

If the SSID parameter is omitted, the HSSSSINT utility defaults the MVS subsystem-id to HSS#.

Examples

In this example, an in-stream JCL procedure has been used to run the HSSSSINT program.

```
//RULELOAD PROC HSSLINK=HSS.SHSSLINK,
//             RULEFILE=ENHANCED.ACCESS.CONTROL.FOR.SCLM.RULE.FILE,
//             SSID=HSS#
//*
//RULELOAD EXEC PGM=HSSSSINT,REGION=32M,PARM='SSID=&SSID'
//STEPLIB  DD DISP=SHR,DSN=&HSSLINK
//HSSLINK  DD DISP=SHR,DSN=&HSSLINK
//HSSRULES DD DISP=SHR,DSN=&RULEFILE
//SYSPRINT DD SYSOUT=*
//         PEND
//*
//LOAD     EXEC RULELOAD
```

Here is more information about the job control statements:

- An in-stream JCL procedure has been used. A procedure library member may be used to simplify the job JCL.
- The EXEC statement includes a PARM for the SSID parameter. This parameter specifies an MVS subsystem-id to be used by Enhanced Access Control for SCLM. This 4 character name must be unique to Enhanced Access Control for SCLM.

- STEPLIB DD defines the stepped load library from which the HSSSSINT program is loaded. HSSSSINT must be loaded from an APF library.
- HSSLINK DD defines the load library that contains the HSSRVALD validation routine program.
- HSSRULES DD defines the Rule File. This is a VSAM KSDS data set that contains the Enhanced Access Control for SCLM Application and Profile definitions.
- SYSPRINT DD defines the output messages data set.

In this example, the PARM for the MVS subsystem-id has been omitted.

```
//RULELOAD EXEC PGM=HSSSSINT
//STEPLIB  DD DISP=SHR,DSN=HSS.SHSSLINK
//HSSLINK  DD DISP=SHR,DSN=HSS.SHSSLINK
//HSSRULES DD DISP=SHR,DSN=YOUR.RULE.FILE
//SYSPRINT DD SYSOUT=*
```

Here is more information about the job control statements:

- The EXEC statement omits a PARM for the SSID parameter. The default MVS subsystem-id value of HSS# is assumed.

Return Codes

HSSSSINT returns a code in register 15 to indicate the results of program execution. The return codes and their meanings are:

Codes	Meaning
00 (X'00')	Successful completion.
04 (X'04')	Probable successful completion. A warning message is written.
08 (X'08')	Processing failed. An error message is written. The loading of the in-memory rules has not completed. Previously loaded rules remain in effect.

HSSUMOD1 - SMP/E ++USERMOD to install the Validation Routine Interface program

The sample HSSUMOD1 illustrates how the RACF RACHECK Post Processing exit ICHRCX02 can be modified to invoke the Enhanced Access Control for SCLM Validation Routine Interface program, HSSRCX02. Refer to "Install the Validation Routine into your RACF system" on page 97 for information regarding the Enhanced Access Control for SCLM installation procedure.

HSSUMOD1 - SMP/E ++USERMOD

```
++USERMOD(ZZZZZZZ) /*

 Copyright Notice:
   This program is Copyright (c) (2002) with all rights
   reserved and is supplied subject to the condition
   that it shall not, by way of trade or otherwise, wholly
   or in part, be lent, resold, hired out, or otherwise
   circulated without the prior written consent of Fundi
   Software.

 Description:
   Usermod for the RACF RACHECK post processing exit, ICHRCX02.

   The purpose of this Usermod is to extend the dataset validation
   of the RACHECK macro by invoking the Validation Routine
   of ENHANCED ACCESS CONTROL FOR SCLM.

   Please refer to the product documentation for ENHANCED ACCESS
   CONTROL FOR SCLM to obtain more information.
                                                                     */
++VER(Z038) FMID(racf-fmid) .
++JCLIN .
//HSSUMOD1 JOB
//*
//*******************************************************************/
//* JOB NAME = HSSUMOD1                                             */
//*                                                                 */
//* DESCRIPTIVE NAME = SMP/E USERMOD FOR RACF EXIT ICHRCX02         */
//*                    FOR Enhanced Access Control for SCLM         */
//*                                                                 */
//* STATUS = VERSION 01  RELEASE 01  MODIFICATION LEVEL 00          */
//*                                                                 */
//* FUNCTION = SAMPLE USERMOD APPLIED TO RACF EXIT ICHRCX02         */
//*            TO INVOKE THE Enhanced Access Control for SCLM       */
//*            VALIDATION ROUTINE.                                  */
//*                                                                 */
//*       LICENSED MATERIALS - PROPERTY OF FUNDI SOFTWARE           */
//*       5697-H59 (C) COPYRIGHT FUNDI SOFTWARE                     */
//*       ALL RIGHTS RESERVED.                                      */
//*       USE, DUPLICATION OR DISCLOSURE RESTRICTED.                */
//*                                                                 */
//* NOTES =                                                         */
//* 1) REVIEW THE SMP CONTROL STATEMENTS BEFORE APPLYING THIS       */
//*    SAMPLE SMP/E USERMOD.                                        */
//* 2) CHANGE THE racf-fmid FIELD IN THE SMP/E USERMOD              */
//*    STATEMENTS TO COMPLY WITH THE FMID FOR YOUR SITE'S           */
//*    INSTALLED VERSION OF RACF.                                   */
//* 3) TAILOR THE SAMPLE ICHRXC02 EXIT CODE TO INSERT ANY           */
//*    INSTALLATION DEFINED CODE.                                   */
//*******************************************************************/
```

Figure 35. HSSUMOD1 Sample SMP/E ++USERMOD for the RACHECK Post Processing exit (Part 1 of 3)

```
//*
//LKED EXEC PGM=IEWL,PARM='RENT'
//SYSUT1   DD UNIT=SYSALLDA,SPACE=(TRK,(1,1))
//SYSPRINT DD SYSOUT=*
//SYSLMOD  DD DSN=SYS1.LPALIB,DISP=SHR
//SYSLIN   DD *
 ENTRY ICHRCX02
 ORDER ICHRCX02
 INCLUDE SYSPUNCH(ICHRCX02)
 INCLUDE SYSPUNCH(HSSRCX02)
 NAME ICHRCX02(R)
/*
++SRC(ICHRCX02) DISTLIB(DUMMY).
ICHRCX02 TITLE 'ICHRCX02 - RACHECK POST PROCESSING EXIT'
ICHRCX02 CSECT                          .RACHECK POST PROCESSING EXIT
ICHRCX02 AMODE 31
ICHRCX02 RMODE ANY
         STM   R14,R12,12(R13)          .save callers reg's
         LR    R12,R15                  .program base
         USING ICHRCX02,R12             .program addressability
         GETMAIN R,LV=DATALEN,LOC=ANY   .get a work area
         ST    R1,8(R13)                .standard
         ST    R13,4(R1)                .register linkage
         LR    R13,R1                   .new save area base
         USING DATA,R13                 .data addressability

*--------------------------------------------------------------------
* 1.    Installation ICHRCX02 code is inserted here
*--------------------------------------------------------------------

*        Insert here the ICHRCX02 installation defined exit code.


         EJECT

*--------------------------------------------------------------------
* 2.    Call Enhanced Access Control for SCLM RACF Exit
*--------------------------------------------------------------------

         L     R1,4(,R13)               .previous save area
         L     R1,24(,R1)               .restore R1 as at initial entry
         L     R15,=V(HSSRCX02)         .HSS exit entry point
         BASR  R14,R15                  .invoke it

*--------------------------------------------------------------------
* 3.    Exit
*--------------------------------------------------------------------

EXIT     DS    0H
         LR    R1,R13                   .data area to free
         L     R13,4(R13)               .old save area address
         FREEMAIN R,A=(1),LV=DATALEN    .free the data area
         SR    R15,R15                  .always return code zero
         RETURN (14,12),RC=(15)         .return to caller
```

Figure 35. HSSUMOD1 Sample SMP/E ++USERMOD for the RACHECK Post Processing exit (Part 2 of 3)

```
*--------------------------------------------------------------------
* CONSTANT AREA
*--------------------------------------------------------------------
         LTORG

*--------------------------------------------------------------------
* DATA AREA
*--------------------------------------------------------------------
DATA     DSECT
SAVE           DS  18F                .save area
DATALEN  EQU   *-DATA
         EJECT
*--------------------------------------------------------------------
* EQUATES AND DSECTS
*--------------------------------------------------------------------
R0       EQU   0                      .REGISTER EQUATES
R1       EQU   1
R2       EQU   2
R3       EQU   3
R4       EQU   4
R5       EQU   5
R6       EQU   6
R7       EQU   7
R8       EQU   8
R9       EQU   9
R10      EQU   10
R11      EQU   11
R12      EQU   12
R13      EQU   13
R14      EQU   14
R15      EQU   15
*
         END
++MOD(HSSRCX02).
```

Note: The HSSRCX02 object code is embedded within the HSSUMOD1 sample member following the ++MOD(HSSRCX02) statement.

Figure 35. HSSUMOD1 Sample SMP/E ++USERMOD for the RACHECK Post Processing exit (Part 3 of 3)

The sample requires review and tailoring by the systems programmer for your installation. If the RACF RACHECK Post Processing exit ICHRCX02 is already active on your system, then apply the tailoring to the existing SMP/E ++USERMOD for ICHRCX02. Comments within the sample provide tailoring instructions.

These notes assume knowledge of SMP/E procedures:

1. Assign the SMP/E ++USERMOD number, replacing ++USERMOD(ZZZZZZZ) statement.
2. Replace the character string RACF-FMID with the SMP/E FMID belonging to the RACF release in use at your installation.
3. If you are creating or replacing an SMP/E ++USERMOD for ICHRCX02 based on the supplied sample, then:
   - Insert any installation-required code into the sample source code, following the comments area marked **1. Installation ICHRCX02 code is inserted here**. Resolve within the inserted code, any entry or exit coding that duplicates or conflicts with that already provided in sample.
   - If you are modifying an existing SMP/E ++USERMOD for ICHRCX02 to invoke the HSSRCX02 Validation Routine Interface, then:
     – Copy into your ICHRCX02 exit the code appearing in the sample following the comments area marked **2. Call Enhanced Access Control for**

SCLM RACF Exit. Note that R1 (register 1) must contain the same address as that provided to the ICHRCX02 on its invocation.
  - The SYSLIN input cards require the INCLUDE SYSPUNCH(HSSRCX02) statement.
  - Following the source code of ICHRCX02, insert an SMP/E ++MOD statement for HSSRCX02 and its object code. This ++MOD(HSSRCX02) statement and the associated object code are contained within the HSSUMOD1 member in the AHSSSAMP sample library.
- The SMP/E ++USERMOD should be applied and tested in conjunction with Enhanced Access Control for SCLM prior to installation onto your production system.

HSSUMOD1 - SMP/E ++USERMOD

Chapter 5. Operator Commands

This chapter describes the functions, syntax, and parameters of the Enhanced Access Control for SCLM operator commands. You can use these commands to control the operation of Enhanced Access Control for SCLM.

The RACF user ID and current-connect Group authorities determine whether the user has the privilege to issue operator commands.

Command Syntax Notation

The Enhanced Access Control for SCLM operator commands conform to these syntactical rules.

Valid characters
: The commands are restricted to use of these characters:
  - A to Z
  - 0 to 9
  - # $ @
  - equal (=), comma (,), quotes (') or brackets '()' may only be used where explicitly shown in the command syntax, or in trailing comments.

Case translation
: Commands may be entered in upper or lower case. The commands are translated into upper case automatically before processing.

Blanks
: A blank or space separator must be entered between the MVS subsystem-id (shown as <ssid>) and the command verb. Multiple blanks may be entered in place of a single blank. Blanks may not be used elsewhere within the command. Blanks may precede or follow the command.

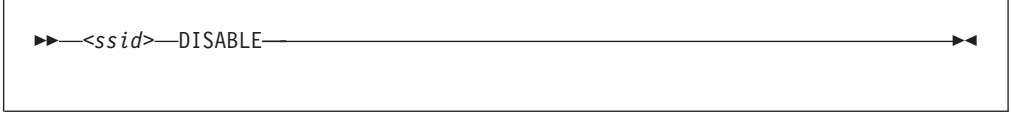
<ssid>
: This token represents the name of MVS subsystem-id used by Enhanced Access Control for SCLM. Enhanced Access Control for SCLM will only process the command if the correct MVS subsystem-id is used.

Comments
: Comments may be appended to the command, following a blank or space separator.

DISABLE Command

Use the DISABLE command to stop Enhanced Access Control for SCLM processing.

DISABLE command



<ssid>
: This token represents the MVS subsystem-id used by Enhanced Access Control for SCLM. This MVS subsystem processes the DISABLE command.

DISABLE Command

DISABLE The DISABLE command requests Enhanced Access Control for SCLM to cease data set access validation, thereby making it inactive. If this command is issued, Enhanced Access Control for SCLM will not perform access control services. Therefore RACF data set violations will occur in cases where Enhanced Access Control for SCLM previously granted access. You may reverse the disabled state and resume Enhanced Access Control for SCLM processing by issuing the ENABLE command.

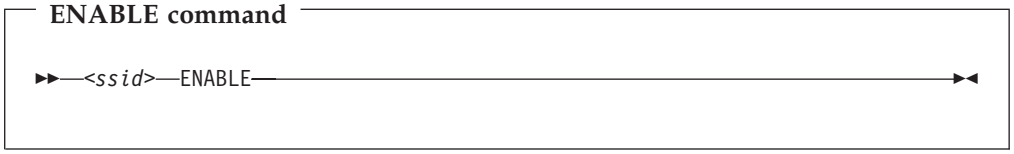
When the DISABLE command is executed, the Activity Status field on the Status Information panel is updated to reflect the inactive status. Refer to "The Status Information panel" on page 20 for more information.

Note that the Rule Load Utility may be run after the DISABLE command has been issued and Enhanced Access Control for SCLM is in an inactive state.

ENABLE Command

Use the ENABLE command to resume Enhanced Access Control for SCLM processing following a previous DISABLE command.

ENABLE command



<ssid> This token represents the MVS subsystem-id used by Enhanced Access Control for SCLM. This MVS subsystem processes the ENABLE command.

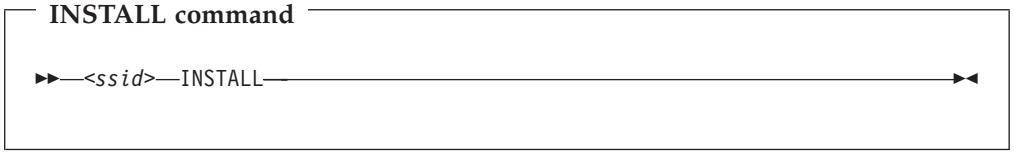
ENABLE The ENABLE command requests Enhanced Access Control for SCLM to resume processing that was previously suspended owing to a DISABLE command. If this command is issued, Enhanced Access Control for SCLM resumes data set access control validation.

When the ENABLE command is executed, the Activity Status field on the Status Information panel is updated to reflect the active status. Refer to "The Status Information panel" on page 20 for more information.

INSTALL Command

Use the INSTALL command to dynamically install the Enhanced Access Control for SCLM Validation Routine into your RACF security environment on the currently executing system.

INSTALL command



<ssid> This token represents the MVS subsystem-id used by Enhanced Access Control for SCLM. This MVS subsystem processes the INSTALL command.

INSTALL The INSTALL command requests Enhanced Access Control for SCLM to dynamically install its Validation Routine (HSSRVALD) into the RACF environment, thereby allowing Enhanced Access Control for SCLM to become operational.

INSTALL is performed immediately, without disruption to RACF services. The install remains in effect until the MVS system is shut down, or the UNINSTALL command is issued. The dynamic installation does not alter the contents of RACF or Enhanced Access Control for SCLM software or definition data sets. The INSTALL command works as follows:

1. If the RACF RACHECK Post Processing exit is not active, INSTALL sets the Validation Routine as the exit and then activates the exit to RACF.
2. If the RACF RACHECK Post Processing exit is already active, INSTALL hooks the Validation Routine onto the exit. The existing installation exit code is preserved and is executed first, the HSSRVALD validation occurs last.

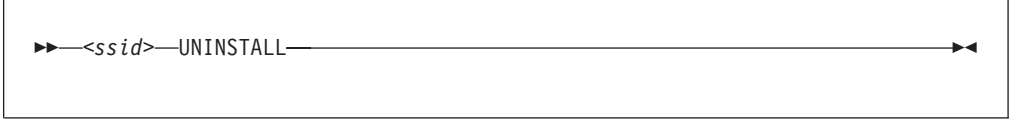
Permanent installation of the Validation Routine may be achieved by applying an SMP/E ++USERMOD onto your RACF system. See "Static installation using an SMP/E ++USERMOD" on page 98 for more information. Do not use the INSTALL command if Enhanced Access Control for SCLM is already installed and active on your system by means of the SMP/E ++USERMOD.

The temporary, dynamic installation of the Validation Routine performed by the INSTALL command can be reversed using the UNINSTALL command.

UNINSTALL Command

Use the UNINSTALL command to back out the dynamically installed Enhanced Access Control for SCLM Validation Routine from your RACF security environment.

UNINSTALL command



<ssid> This token represents the MVS subsystem-id used by Enhanced Access Control for SCLM. This MVS subsystem processes the UNINSTALL command.

UNINSTALL The UNINSTALL command requests Enhanced Access Control for SCLM to back out the dynamically installed Validation Routine (HSSRVALD) from the RACF environment. The UNINSTALL effectively reverses the INSTALL command process, thereby undoing the previous dynamic installation. The UNINSTALL will make Enhanced Access Control for SCLM become non-operational.

UNINSTALL Command

You should not use UNINSTALL to temporarily suspend Enhanced Access Control for SCLM processing. The DISABLE command may be used for this purpose.

UNINSTALL only applies when the INSTALL command has been used to dynamically install the Validation Routine. It cannot be used to back out the permanent installation of Enhanced Access Control for SCLM into the RACF environment as described in "Static installation using an SMP/E ++USERMOD" on page 98.

UNINSTALL is performed immediately, without disruption to RACF services. The back out remains in effect until another INSTALL command is issued. UNINSTALL does not alter the contents of RACF or Enhanced Access Control for SCLM software or definition data sets. The UNINSTALL command works as follows:

1. If the Validation Routine has been activated as the RACF RACHECK Post Processing Exit (ICHRCX02), then UNINSTALL removes the Validation Routine as the exit thereby deactivating the exit.
2. If the Validation Routine has been hooked onto an existing RACF RACHECK Post Processing exit, UNINSTALL removes those hooks leaving the normal installation code operating as the RACF exit.

Chapter 6. Problem Determination

This chapter holds information about Enhanced Access Control for SCLM problem determination.

- "Eliminating User Errors" on page 74

  This section helps you resolve common problems using Enhanced Access Control for SCLM. It explains Enhanced Access Control for SCLM violation records, and helps you refine your Profile access rules to improve access controls.

- "Diagnosis" on page 90

  This section describes the steps you need to follow to gather the information needed to work with IBM support.

For the list and explanation of Enhanced Access Control for SCLM messages, see Chapter 8, "Messages" on page 103.

Collecting Helpful Diagnostic Information

Before resolving problems, perform these steps to determine the source of a problem:

1. Describe the symptoms.
2. List these items:
   - Error message data
   - Program termination message data
3. Consider the problem circumstances:
   - Is the correct Rule File loaded into memory?
   - Has Enhanced Access Control for SCLM been disabled?
   - Is this a new problem, or a recurrence of a previous problem?
   - Are recent changes contributing to the problem?
   - Are other functions and services working normally?
   - Can the problem be reproduced?
4. Analyze the failure to identify the type of problem as described in the following section.

Identifying Types of Problems

After collecting the information described in the preceding paragraph, determine the type of problem you have found. Problems may be caused by:

- The way you are using Enhanced Access Control for SCLM
  - The access privilege granted to a data set was not what was expected, or as reported.
  - A component of Enhanced Access Control for SCLM has issued an error message.
  - Enhanced Access Control for SCLM is inoperative because the software installation did not complete successfully.
- Enhanced Access Control for SCLM program errors
  - A component has failed without issuing an Enhanced Access Control for SCLM error message.
  - The Enhanced Access Control for SCLM error message action has told you to contact IBM support.

Identifying Types of Problems

If the problem is an Enhanced Access Control for SCLM program error, then you should contact IBM support for help. “Diagnosis” on page 90 will provide instructions on how to gather the data required to work with IBM support.

If the problem relates to the way you are using Enhanced Access Control for SCLM, then the following sections will help you to resolve the problem yourself.

Eliminating User Errors

This section explains how to diagnose and rectify common causes of Enhanced Access Control for SCLM problems. It discusses:

Data set access validation errors

These errors occur when the access privilege granted to a data set was not what was expected, or for a reason inconsistent with what was expected. The sections that follow will help you to understand why the data set access request outcome has occurred. Some rule refinement may be necessary to achieve the access controls you desire.

If the access granted was not what was expected or inconsistent with the violation trace records, the following topics will assist you with problem resolution:

Access Received	Access Expected	Violation Record	Page	Section
Granted	Granted	No	75	Access granted correctly
Granted	Granted	Yes	75	Access granted, but a violation is reported
Granted	None	No	76	Access should not have been granted
Granted	None	Yes	75	Access granted, but a violation is reported
None	Granted	No	77	Access failed, yet no violation is reported
None	Granted	Yes		Refer to the following table of Validation Routine return codes
None	None	No	77	Access failed, yet no violation is reported
None	None	Yes		Refer to the following table of Validation Routine return codes

If an Enhanced Access Control for SCLM violation trace record is reported, use its return code (RC) to select the appropriate topic from the list below:

RC	Page	Section
01	79	Violation RC=01, Opening program not APF or attached as a subtask
02	80	Violation RC=02, Opening program not APF or assigned a subtask task library
03	80	Violation RC=03, Opening program not APF and task libraries have defaulted
04	81	Violation RC=04, Opening program not APF - an APF program must be used
05	82	Violation RC=05, Low Program not APF
06	82	Violation RC=06, A non-APF program found in the High to Low Program chain
07	83	Violation RC=07, User-id, RACF Group or * not found in Profile
08	84	Violation RC=08, Profile has no Applications matching the execution environment
09	85	Violation RC=09, User’s assigned privilege was less than that required

10	86	Violation RC=10, Environment compromised by unauthorized asynchronous task
11	87	Violation RC=11, Invalid TSO/ISPF environment detected during APF program checks
12	87	Violation RC=12, Invalid TSO/ISPF environment for an APF program execution
13	87	Violation RC=13, ISPF subroutines module is invalid
14	87	Violation RC=14, Invalid TSO/ISPF environment detected
15	88	Violation RC=15, High/Low Programs equal, but opening program not Low Program
16	88	Violation RC=16, TSO invoked by an unauthorized control program

Access granted correctly

This case occurs when the user has been granted access to the data set as expected. No violation record appears. This is a normal outcome for data access validation. No problem exists. The data set access may have been granted by:

- RACF
- Enhanced Access Control for SCLM
- Installation code or another product installed into the RACF RACHECK Post Processing exit.

Access granted, but a violation is reported

This case occurs when the user has been granted access to the data set, but a violation record is found in Enhanced Access Control for SCLM. There appears to be an inconsistency between the outcome of the access request and the reported violation record. There are two explanations for this problem, summarized as:

- The violation record relates to a different access request.
- Installation code or another product installed into the RACF RACHECK Post Processing exit granted access after Enhanced Access Control for SCLM processing completed.

Make sure that the violation record exactly matches the circumstances under which the user gained access to the data set. The violation record may relate to a different access request, which was denied. The date, time, data set or Profile, and user should exactly match. Try to reproduce the problem to confirm that the violation record matches the data set access request. If the violation record does not match the successful access request, then this problem is resolved.

The presence of a valid violation record indicates that Enhanced Access Control for SCLM has denied data set access. This denial must have been overruled after Enhanced Access Control for SCLM has completed validation processing. This can occur when installation code or other access control products share the RACF RACHECK Post Processing exit. There are two explanations:

- Installation code within the RACF RACHECK Post Processing exit has granted data set access, overruling Enhanced Access Control for SCLM. Confirm with your system programmer whether any installation code within the exit overrides or ignores Enhanced Access Control for SCLM validations.

  This situation does not apply if Enhanced Access Control for SCLM has been dynamically installed into your RACF environment by means of the INSTALL operator command. Refer to “Dynamic installation using the INSTALL command” on page 98 for more information regarding this type of installation.
- Another access control product invoked via the RACF RACHECK Post Processing exit has granted the data set access, overruling Enhanced Access

Control for SCLM. Confirm with your systems programmer whether other access control products are invoked via the RACF exit.

When this situation exists, the identified problem may occur intermittently. The execution sequence of Enhanced Access Control for SCLM and the other access control product may vary depending if installation into the RACF environment occurs dynamically. Additionally, access rule changes in either product may affect the other.

Access should not have been granted

This case occurs when the user has been granted access to the data set, however an access violation was expected. No violation record is found in Enhanced Access Control for SCLM. There are four explanations for this problem:

- RACF has granted access.
- Installation code or another product installed into the RACF RACHECK Post Processing exit granted access.
- Enhanced Access Control for SCLM granted access using in-memory rules loaded from a different Rule File.
- Enhanced Access Control for SCLM granted access, possibly matching on a Profile access rule different than that expected.

The RACF definitions may have granted the user access to the data set. If you are implementing access controls via Enhanced Access Control for SCLM, then ensure the RACF access rules deny the user data set access. Enhanced Access Control for SCLM access validations only occur in cases where RACF has denied access.

Installation code or other access control products installed into the RACF RACHECK Post Processing exit may have granted access to the data set. If this occurs before Enhanced Access Control for SCLM validation, then validation is bypassed because the user already has sufficient data set access privilege. If this occurs after Enhanced Access Control for SCLM validation, then a violation record appears for Profiles controlled by Enhanced Access Control for SCLM where the access rules do not grant sufficient privilege.

Enhanced Access Control for SCLM may have validated the data set access request using in-memory rules loaded from a different Rule File, or from an older version of the Rule File. The "Status Information" panel as described on page 20 displays the name of the Rule File that was loaded into memory, and the date and time of the load. If the wrong Rule File is in use, or if the Profile access rules have been updated since the last rule refresh, run the Rule Load Utility to refresh the in-memory rules.

Enhanced Access Control for SCLM may have granted the data set access request by matching on a Profile access rule other than the one expected. Consider your Profile access rules giving attention to the Profile selected for validation, the Applications in use, and the user IDs and RACF Groups defined in those rules as follows:

- Determine the correct Profile for validation. "Matching the Profile for validation" on page 52 describes how the Profile is selected. If a discrete data set Profile defined to Enhanced Access Control for SCLM exactly matches the data set name, then this Profile will be used. Otherwise, a generic Profile is used.

  The Enhanced Access Control for SCLM generic Profile name must match exactly the name of the generic Profile used by RACF for its validation. In the example below, RACF has selected the generic Profile SCLM.TEST.* for

validation. Therefore, Enhanced Access Control for SCLM matches on the same Profile, even though other Profiles may appear to be a closer match to the data set name.

Access to data set:	SCLM.TEST.SOURCE
RACF selected Profile:	SCLM.TEST.*

Profile	**Outcome**
SCLM.TEST.S*	Rejected, because it was not used by RACF
SCLM.TEST.*	Matched, as the generic data set Profile
SCLM.*	Rejected, because it was not used by RACF

If Enhanced Access Control for SCLM did not match on the Profile you expected, you will need to adjust the access rules for the correct Profile.

- Determine the Application matched for validation purposes. "Matching the Application for validation" on page 53 describes how the Application is selected. Matching is more complex if Applications share the same Low Program values, because multiple Applications may apply. This situation may cause confusion.

  If you are unsure of which Application was matched, adjust the Profile access rules to assign NONE as the user ID access privilege for each combination of Application and Function referenced within the Profile. Carefully note any access rule changes, so that these can be reversed later.

  Run the Rule Load Utility to refresh the in-memory rules, and then test the user access. An access violation record should now be collected. If an access violation does not occur, this indicates that access is being granted by Enhanced Access Control for SCLM through some other Profile definition; or by RACF, installation code or another access control product.

  The violation record indicates the Application matched by Enhanced Access Control for SCLM. The Profile access rules for this Application need to be reconsidered. The user ID or one of the RACF current-connect Groups for that user ID may have a higher access privilege than was expected.

  Adjust the Profile access rules, and reverse the changes employed to discover the matching Application. Run the Rule Load Utility to refresh the in-memory rules. The user's access should again be tested to see if the problem persists. If so, this indicates that multiple Applications match the data set access program execution environment, and the steps described above may need to be repeated to discover other matching Applications and correct the Profile access rules accordingly.

- Determine the user ID or RACF current-connect Groups matched for validation purposes. "Matching the User for validation" on page 54 describes how this matching is performed. When matching is performed against the RACF current-connect Groups, any one of these Groups may be used to satisfy the access request.

  Use RACF services to check the user-to-Group connections. Check each connected RACF Group against the Profile access rules for the matching Application or Applications. Adjust the rules to limit access. You may need to add a Profile access rule for the user ID if changing the access privilege for the RACF Group is inappropriate.

Access failed, yet no violation is reported

This case occurs when the user has been denied access to the data set, however a violation record is not found in Enhanced Access Control for SCLM. There appears to be an inconsistency between the outcome of the access request and the absence of a violation record. There are four explanations for this problem, summarized as:

- The violation record no longer appears in the violations buffer.

Data set access validation errors

- The discrete or generic RACF Profile used for validation was not defined to Enhanced Access Control for SCLM.
- Enhanced Access Control for SCLM was disabled, not installed, or otherwise inactive at the time of the access request.
- Installation code or another product installed into the RACF RACHECK Post Processing exit denied access after Enhanced Access Control for SCLM processing completed.

The violation records reside within an in-memory buffer. This buffer keeps only the most recent violation records, with older records automatically purged. The "Violation Selection" panel as described on page 30 displays the date and time of the violations. If the violation occurred before the oldest shown on the violations list, then the violation record may have already been purged. Reproduce the data access violation, and then check the violations list.

Violation records only appear for data sets that match the discrete or generic Profiles defined to Enhanced Access Control for SCLM. Other data sets are not validated, and violation records are not collected for them. Ensure that the correct Profiles are defined to Enhanced Access Control for SCLM. Consider these three items:

- If RACF performed its validation using a discrete Profile, then the same discrete Profile must be defined to Enhanced Access Control for SCLM. Failure to do so will result in Enhanced Access Control for SCLM bypassing validation for that data set; therefore no violation records will be collected. Add this discrete Profile to your definitions if it is to be controlled by Enhanced Access Control for SCLM.
- If RACF performed its validation using a generic data set Profile, then the identical Profile should be defined to Enhanced Access Control for SCLM. Refer to "Matching the Profile for validation" on page 52 for more information. Add this generic Profile to your definitions if it is to be controlled by Enhanced Access Control for SCLM.
- If the Profiles are defined in your Rule File, verify that the correct Rule File was used to load the in-memory rules. If the Profile definitions were recently changed, ensure that the Rule Load Utility was successfully run after those definitions were updated. The Status Information panel displays the name of the active Rule File and when this was loaded into memory. Run the Rule Load Utility for the correct Rule File to resolve this problem.

If Enhanced Access Control for SCLM was inactive at the time of the data set access request, then violation records cannot be collected. Access failures will occur in cases where Enhanced Access Control for SCLM normally grants access. Enhanced Access Control for SCLM may be inactive owing to a number of reasons:

- Enhanced Access Control for SCLM may not be installed correctly. Ensure that Enhanced Access Control for SCLM has been correctly installed into the RACF environment. Refer to "Install the Validation Routine into your RACF system" on page 97 for more information.

  If the dynamic installation technique is used, the INSTALL operator command may have failed or not been issued. If the static installation technique is used, the correct RACF RACHECK Post Processing exit may not have been implemented at the last MVS system IPL. You will need to complete the Enhanced Access Control for SCLM installation.
- The Rule Load Utility may not have completed successfully since the last MVS system IPL. Therefore, the Enhanced Access Control for SCLM subsystem and in-memory rules will not be loaded. A not-initiated state displayed on the Status

Information panel confirms this condition. Run the Rule Load Utility now and resolve any problems that it may encounter.

- The DISABLE operator command may have been issued to stop access validation. The Status Information panel displays an inactive state. Try to determine who issued the DISABLE command and why. You may resume validation by issuing the ENABLE command.
- The UNINSTALL operator command may have been issued to remove Enhanced Access Control for SCLM from the RACF environment. Try to determine who issued the UNINSTALL command and why. You may use the INSTALL command to install Enhanced Access Control for SCLM into the RACF environment.

Enhanced Access Control for SCLM may have granted data set access, but this was overruled after Enhanced Access Control for SCLM has completed validation processing. This can occur when installation code or other access control products share the RACF RACHECK Post Processing exit. There are two explanations:

- Installation code within the RACF RACHECK Post Processing exit has denied data set access, overruling Enhanced Access Control for SCLM. Confirm with your system programmer whether any installation code within the exit overrides or ignores Enhanced Access Control for SCLM validations.

  This situation does not apply if Enhanced Access Control for SCLM has been dynamically installed into your RACF environment by means of the INSTALL operator command. Refer "Dynamic installation using the INSTALL command" on page 98 for more information regarding this type of installation.
- Another access control product invoked via the RACF RACHECK Post Processing exit has denied the data set access, overruling Enhanced Access Control for SCLM. Confirm with your system programmer whether other access control products are invoked via the RACF exit.

  When this situation exists, the identified problem may occur intermittently. The execution sequence of Enhanced Access Control for SCLM and the other access control product may vary depending if installation into the RACF environment occurs dynamically. Additionally, access rule changes in either product may affect the other.

Violation RC=01, Opening program not APF or attached as a subtask

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM validates that the Low Program is loaded from an APF library. Any programs subordinate to the Low Program also must be loaded from APF libraries, or attached as a subtask by the Low Program or an approved subordinate program. This check ensures that the method of subordinate program attachment is consistent with SCLM's operation, and avoids the use of incorrect program copies. The program that attempted the data set access failed this validation check. It was not loaded from an APF library, nor was it attached as a subtask. As a result, access was denied.

The violation record execution program details will provide more information. Refer to "The Violation Programs panel" on page 35. Listed after the Low Program are the subordinate programs. The Library Notes indicate whether programs are

loaded from APF libraries or if a task library is active. The Low Program or a subordinate program is at fault. The offending programs will have their APF note omitted.

This problem may be resolved in one of two ways:

- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.
- Adjust the SCLM environment to assign a subtask task library to locate the program. This resolution method only applies if the violation occurs when running an SCLM translator. Adjust the SCLM translator FLMTRNSL macro to use CALLMETH=ATTACH (call method of ATTACH) and include a TASKLIB parameter to define the program execution libraries.

Violation RC=02, Opening program not APF or assigned a subtask task library

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM validates that the Low Program is loaded from an APF library, and that any programs subordinate to the Low Program are either loaded from APF libraries or from a subtask task library assigned by the Low Program or an approved subordinate program. This avoids the use of incorrect program copies. The program that attempted the data set access, failed this validation check. It was not loaded from an APF library, nor was a subtask task library assigned to locate the program. As a result, access was denied.

The violation record execution program details will provide more information. Refer to "The Violation Programs panel" on page 35. Listed after the Low Program are the subordinate programs. The Library Notes indicate whether programs are loaded from APF libraries or if a task library is active. The Low Program or a subordinate program is at fault. The offending programs will have their APF note and TASKLIB note omitted.

This problem may be resolved in one of two ways:

- Adjust the SCLM environment to assign a subtask task library to locate the program. This option applies if the violation occurs when running an SCLM translator. Adjust the SCLM translator FLMTRNSL macro to include a TASKLIB parameter to define the program execution libraries.
- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.

Violation RC=03, Opening program not APF and task libraries have defaulted

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot

be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM validates that the Low Program is loaded from an APF library, and that any programs subordinate to the Low Program are either loaded from APF libraries or from a subtask task library assigned by the Low Program or an approved subordinate program. This avoids the use of incorrect program copies. The program that attempted the data set access, failed this validation check. It was not loaded from an APF library, and specific task libraries were not the assigned to the subtask. The task libraries have defaulted or cascaded from the calling task. As a result, access was denied.

The violation record execution program details will provide more information. Refer to "The Violation Programs panel" on page 35 Listed after the Low Program are the subordinate programs. The Library Notes indicate whether programs are loaded from APF libraries or if a task library is active. The Low Program or a subordinate program is at fault. The offending programs will have their APF note omitted.

This problem may be resolved in one of two ways:

- Adjust the SCLM environment to assign a subtask task library to locate the program. This option applies if the violation occurs when running an SCLM translator. Adjust the SCLM translator FLMTRNSL macro to include a TASKLIB parameter to define the program execution libraries.
- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.

Violation RC=04, Opening program not APF - an APF program must be used

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM validates that the Low Program is loaded from an APF library. Depending upon the execution circumstances, programs subordinate to the Low Program must also be loaded from APF libraries. This avoids the use of incorrect program copies. The program that attempted the data set access, failed this validation check. It was not loaded from an APF library as required by the execution circumstances. As a result, access was denied.

The violation record execution program details provide more information. Refer to "The Violation Programs panel" on page 35. Listed after the Low Program are the subordinate programs. The Library Notes indicate whether programs are loaded from APF libraries. The Low Program or a subordinate program is at fault. The offending programs will have its APF note omitted.

This problem may be resolved in only one way:

- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If

the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.

Violation RC=05, Low Program not APF

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM validates that the Low Program is loaded from an APF library. All programs in the execution pathway from the High Program to the Low Program must be loaded from an APF library. This avoids the use of incorrect program copies. The Low Program failed this validation check. It was not loaded from an APF library. As a result, access was denied.

The violation record execution program details provide more information. Refer to "The Violation Programs panel" on page 35. The Library Notes indicate whether programs are loaded from APF libraries. The Low Program is at fault. It will have its APF note omitted.

This problem may be resolved in one of two ways:

- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.
- If the Low Program is a program executed via an SCLM translator, then adjust your Application Low Program definition to be the lowest SCLM program (these have an FLM prefix). Run the Rule Load Utility. It may also be necessary to adjust the SCLM translator FLMTRNSL macro to use CALLMETH=ATTACH (call method of ATTACH) and include a TASKLIB parameter to define the program execution libraries. The SCLM administrator will need to make these changes. Once these changes are complete, test the access.

  Enhanced Access Control for SCLM validates that all programs in the execution chain from High Program to the Low Program are loaded from APF libraries. However, programs subordinate to the Low Program may be APF loaded or task library loaded. By changing the Application Low Program definition, the APF load restriction is applied onto the lowest SCLM program module. The change to the SCLM translator assigns a task library, thereby satisfying the validation checks applied to programs subordinate to the Low Program.

Violation RC=06, A non-APF program found in the High to Low Program chain

The Validation Routine has detected an error in the execution program environment at the time of the data set access request. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software.

Enhanced Access Control for SCLM requires that all High to Low Programs in the execution pathway are loaded from an APF library. This avoids the use of incorrect program copies. A program in the High to Low Program execution pathway failed this validation check. It was not loaded from an APF library. As a result, access was denied.

The violation record includes execution program details that provide more information. Refer to "The Violation Programs panel" on page 35. The Library Notes indicate whether programs are loaded from APF libraries. One or more of the programs from the High Program to the Low Program is at fault. It will have its APF note omitted.

This problem may be resolved in one of two ways:

- Load the offending program from an APF library. This may require that the program be copied into an existing APF library, or the program library be defined as APF. First verify that the correct version of the program was in use. If the program was loaded from an APF library, ensure that APF status was not lost owing to the presence of non-APF libraries in the program execution library search order.
- Consider the Application High and Low Program definitions. These should identify SCLM programs, which are loaded from APF libraries. However if these include non-SCLM programs (for example ISPTASK), then the validation for APF loading may have been applied to other programs that are not APF loaded.

  Examination of the program execution list from the "Violation Programs" panel should quickly identify if non-SCLM programs appear in the High to Low Program chain.

  "What is the High Program?" on page 40 describes the High Program definition. If the High Program value is something other than ISRSCLM or an FLM prefixed program name, then non-SCLM programs may be included for validation of APF loading. This will likely apply if installation programs are being used to invoke SCLM services via the FLMLNK interface routine.

  If the High Program in use is too restrictive, change the Application High Program definition. Run the Rule Load Utility, and then test the access.

  "What is the Low Program?" on page 42 describes the Low Program definition. If the Low Program is not an SCLM module but a subordinate program invoked by SCLM, then the validation for APF loading is also applied to this subordinate program. This may be too restrictive.

  If the Low Program is too restrictive, change the Application Low Program definition, and run the Rule Load Utility. Also adjust the SCLM translator used to invoke the programs subordinate to the Low Program. The FLMTRNSL macro should specify CALLMETH=ATTACH (call method of ATTACH) and include a TASKLIB parameter to define the program execution libraries. The SCLM administrator will need to make these changes. Once these changes are complete, test the access.

Violation RC=07, User-id, RACF Group or * not found in Profile

The Validation Routine has selected the Profile for validation. However, this Profile contains no access rules that apply to the user. Therefore the data set access request is denied. This problem can be resolved by adjusting the Enhanced Access Control for SCLM Profile access rules.

"Matching the User for validation" on page 54 describes how the Validation Routine matches the user to the Profile access rules. In this case, neither the user ID, a RACF current-connect Group, or * (all users) were defined within the Profile. Therefore the user could not be granted access. Note that the Profile is checked for all RACF current-connect Groups if the user is connected to more than one Group at the time of the data set access request.

This problem may be resolved in one of two ways:

- The Profile access rules can be altered to include new rules for the user. This could be applied to the user ID, a RACF current-connect Group, or * (all users) - whichever is appropriate. Run the Rule Load Utility, and test the data set access.
- The user's RACF definitions may be altered to connect the user to a RACF Group that has access privileges within the Enhanced Access Control for SCLM Profile definitions. The RACF administrator will need to make this change. The user may then test the data set access.

If the Profile assigns access privileges for the user, then there appears to be an inconsistency with the violation record and the Profile access rules. There are four explanations:

- The in-memory rules were loaded from a different Rule File.
- The in-memory rules were loaded prior to Profile access rule changes that granted access to the user.
- The Rule Load Utility issued warning messages and discarded the Profile access rules for the user.
- The user was not connected to the referenced RACF Group at the time of the data set access request.

Use the Status Information panel to check the name of the Rule File used to load the in-memory rules, and the date and time the last rule refresh was performed.

If the wrong Rule File was used to load the in-memory rules, run the Rule Load Utility to refresh the in-memory rules. If the Rule File has been updated since the last rule load refresh, then run the Rule Load Utility now to refresh the in-memory rules.

If warning messages were issued during the last execution of the Rule Load Utility, review and resolve any identified problems. The Profile or Application definitions likely require refinement. Run the Rule Load Utility to refresh the rules, correcting any errors reported.

If the user was not connected to the RACF Group referenced in the Profile access rule, then the user should connect to the correct RACF Group to obtain access. Alternatively, the Profile access rules may be altered to add new rules for the user ID or other RACF Groups. If the Profile access rules are changed, then run the Rule Load Utility and test the access.

Violation RC=08, Profile has no Applications matching the execution environment

The Validation Routine has selected the Profile for validation. However, none of the Application definitions referenced in the Profile have High and Low Programs that match the program execution environment. Therefore the data set access request is denied. This problem can be resolved by adjusting the Enhanced Access Control for SCLM Profile and Application definitions.

Consider the execution programs within the violation record. These can be viewed from the Violation Programs panel. You may observe that:

- The program environment matches an Application definition, however this Application is not referenced in the Profile rules. In this case, add new Profile access rules for that Application. Run the Rule Load Utility, and test the access.
- The program environment is a near match to an Application definition. Perhaps another High and Low Program combination needs to be added to the Application definition. Caution should be taken when changing an existing

Application, as new High and Low Program definitions may impact access to Profiles referencing that Application. If the Application is changed, run the Rule Load Utility, and test the access.

- The program environment is unlike existing Applications. A new Application definition may be required. In this case, define the new Application and add Profile access rules for that Application. Run the Rule Load Utility, and test the access.

If the Profile references an Application that has High and Low Programs matching the program execution environment, then there appears to be an inconsistency with the violation record and the Rule File definitions. There are three explanations:

- The in-memory rules were loaded from a different Rule File.
- The in-memory rules were loaded prior to Profile or Application changes that effected user access.
- The Rule Load Utility issued warning messages and discarded the Profile access rules for the Application.

Use the Status Information panel to check the name of the Rule File used to load the in-memory rules, and the date and time the last rule refresh was performed.

If the wrong Rule File was used to load the in-memory rules, run the Rule Load Utility to refresh the in-memory rules. If the Rule File has been updated since the last rule load refresh, then run the Rule Load Utility to refresh the in-memory rules. Test the access.

If warning messages were issued during the last execution of the Rule Load Utility, review and resolve any identified problems. The Profile or Application definitions likely require refinement. Run the Rule Load Utility to refresh the rules, correcting any errors reported. Test the access.

Violation RC=09, User's assigned privilege was less than that required

The Validation Routine has selected the Profile and access rule for validation. However, the access privilege assigned to the user was less than that required to satisfy the data set access request. For example, the access request required UPDATE privilege, but the user privilege was only READ. Therefore the data set access request is denied. This problem can be resolved by adjusting the Profile access rules.

The violation record includes details that provide more information. Refer to "The Violation Detail panel" on page 32. Verify that the Profile and Application selected for validation are those expected, as these determine the access rules used to obtain user privilege. The violation record will also show if the access rule was matched on the user ID, * (for all users), or GRPLIST for a RACF current-connect Group. If GRPLIST is displayed, then the highest access privilege of any RACF Group to which the user was connected will be assigned. Use RACF services to determine user-to-Group connections.

Refer to the following topics if more information is required on the Profile rule matching techniques used by the Validation Routine:

- "Matching the Profile for validation" on page 52
- "Matching the Application for validation" on page 53
- "Matching the User for validation" on page 54

• "Assigning the access privilege" on page 56

The problem may be resolved in two ways:
- Change the Profile access rules to grant the user the appropriate privilege. Run the Rule Load Utility, and test the access.
- Change the RACF user-to Group connections to associate the user with another Group that has the appropriate level of access. Test the access.

If the user privilege in the Profile access rules is correct, then there appears to be an inconsistency with the violation record and the Rule File definitions. There are four explanations:
- The in-memory rules were loaded from a different Rule File.
- The in-memory rules were loaded prior to Profile or Application changes that effected user access.
- The Rule Load Utility issued warning messages and discarded the Profile access rules for the Application.
- The user was not connected to the referenced RACF Group at the time of the data set access request.

Use the Status Information panel to check the name of the Rule File used to load the in-memory rules, and the date and time the last rule refresh was performed.

If the wrong Rule File was used to load the in-memory rules, run the Rule Load Utility to refresh the in-memory rules. If the Rule File has been updated since the last rule load refresh, then run the Rule Load Utility to refresh the in-memory rules. Test the access.

If warning messages were issued during the last execution of the Rule Load Utility, review and resolve any identified problems. The Profile or Application definitions likely require refinement. Run the Rule Load Utility to refresh the rules, correcting any errors reported. Test the access.

If the user was not connected to the RACF Group referenced in the Profile access rule, then the user should connect to the correct RACF Group to obtain access. Alternatively, the Profile access rules may be altered to add new rules for the user ID or other RACF Groups. If the Profile access rules are changed, then run the Rule Load Utility. Test the access.

Violation RC=10, Environment compromised by unauthorized asynchronous task

The Validation Routine checks and withholds access if the SCLM program environment appears compromised. During this checking, an unauthorized asynchronous task was found in the program environment. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

This problem may occur in an online TSO/ISPF environment, where other tasks (possibly in split sessions) are present. One of these tasks has the capability of interrupting or compromising the SCLM environment. Closing the split session or logging off then onto TSO may resolve the problem. The problem may be intermittent, depending upon the mix of tasks in the TSO/ISPF environment. If the problem persists, contact your IBM representative for help.

This problem should not occur in the batch job environment. If it does occur, this indicates that SCLM has been invoked in an abnormal way. Correct the method of SCLM execution, and test the access. If the SCLM method of invocation is correct or required, contact your IBM representative for help.

Violation RC=11, Invalid TSO/ISPF environment detected during APF program checks

The Validation Routine checks and withholds access if the SCLM program environment appears compromised. During this checking, the structure of the TSO/ISPF environment was found to be invalid. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

This problem can only occur in an online TSO/ISPF environment. It indicates that the TSO/ISPF environment has been tampered, resulting in an abnormal TSO/ISPF structure. Logging off then onto TSO may resolve the problem. If the problem persists, contact your IBM representative for help.

Violation RC=12, Invalid TSO/ISPF environment for an APF program execution

The Validation Routine checks and withholds access if the SCLM program environment appears compromised. During this checking, the structure of the TSO/ISPF environment was found to be invalid. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

This problem can only occur in an online TSO/ISPF environment. It indicates that the TSO/ISPF program structure for the execution of APF programs is invalid, possibly owing to environment tampering. Logging off then onto TSO may resolve the problem. If the problem persists, contact your IBM representative for help.

Violation RC=13, ISPF subroutines module is invalid

The Validation Routine checks and withholds access if the SCLM program environment appears compromised. During this checking, the ISPF subroutines module was found to be invalid. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

This problem can only occur in an online TSO/ISPF environment. It indicates that the TSO/ISPF environment has been compromised, possibly owing to environment tampering. Logging off then onto TSO may resolve the problem. If the problem persists, contact your IBM representative for help.

Violation RC=14, Invalid TSO/ISPF environment detected

The Validation Routine checks and withholds access if the SCLM program environment appears compromised. During this checking, the structure of the TSO/ISPF environment was found to be invalid. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

Data set access validation errors

This problem can only occur in an online TSO/ISPF environment. It indicates that the TSO/ISPF environment has been tampered, resulting in an abnormal TSO/ISPF structure. Logging off then onto TSO may resolve the problem. If the problem persists, contact your IBM representative for help.

Violation RC=15, High/Low Programs equal, but opening program not Low Program

The Validation Routine has matched the Profile and Application for validation. The Application's matching High and Low Program have the same value, therefore the data set access request must be issued by the Low Program itself, and not by a subordinate program. As the request was issued by a subordinate program (one invoked after the Low Program), data set access is denied. This problem can be resolved by adjusting the Enhanced Access Control for SCLM Profile and Application definitions.

Refer to "What is the Low Program?" on page 42 for a description of the Low Program processing rules.

Consider the execution programs within the violation record. These can be viewed from the Violation Programs panel. You may observe that:

- Listed above (before) the High-and-Low program may be other SCLM programs (these have an FLM prefix or have the name ISRSCLM).
- Listed below (after) the High-and-Low program are the subordinate programs, some of which may be other SCLM programs (these have an FLM prefix).

This problem may be resolved in one of two ways:

- If other SCLM programs are invoked prior to the Low Program, then change the High Program value for the Application's matched High-Low Program pair. Set the High Program value to be the first SCLM program in the execution program pathway. By setting the High and Low Programs to different values, programs subordinate to the Low Program may be used to issue the data set access request. Run the Rule Load Utility, and then test the access.
- If other SCLM programs are invoked after the Low Program, then change the Low Program value for the Application's matched High-Low Program pair. Set the Low Program value to be the last SCLM program in the execution program pathway. By setting the High and Low Programs to different values, programs subordinate to the Low Program may be used to issue the data set access request. Run the Rule Load Utility, and then test the access.

Violation RC=16, TSO invoked by an unauthorized control program

The Validation Routine determined that a control program invoked the TSO program (IKJEFT01, IKJEFT1A, IKJEFT1B or IKJEFT1I). However, this control program is not authorized - it was not loaded from common storage or an APF library. Therefore the data set access request was denied. This type of problem cannot be resolved by changes to the Profile access rules; nor is it caused by the installation of the Enhanced Access Control for SCLM software. The problem is specific to the program execution conditions at the time of the data set access request.

This problem should never occur, as TSO fails during start up if executed by an unauthorized program. If the problem does occur, the TSO/ISPF environment may be compromised owing to tampering. If the program execution environment is correct and the problem persists, contact your IBM representative for help.

Operator command and MVS subsystem errors

These errors occur when an operator command has been issued to the Enhanced Access Control for SCLM MVS subsystem. Three types of problems may occur:

- An HSS message indicates that an invalid command is received

  The command entered has incorrect syntax. Review the command syntax as referenced in Chapter 5, "Operator Commands" on page 69. Check the command for spelling and syntax correctness. Ensure that the command was issued for the correct MVS subsystem. If the syntax appears correct, remove any comments following the command and issue the command again. If the problem persists, contact IBM support for help.
- A non-HSS invalid command message is returned

  The message indicates that the MVS subsystem referenced in the operator command is either not active, or for an MVS subsystem other than used by Enhanced Access Control for SCLM. The Status Information panel displays the name of the Enhanced Access Control for SCLM MVS subsystem. If the Enhanced Access Control for SCLM MVS subsystem has not been started, run the Rule Load Utility.
- Operator command failure

  If the operator command fails, collect details as described in "Diagnosis" on page 90 and report the problem to IBM support.

Utility and batch job executions

These errors occur when the execution of an Enhanced Access Control for SCLM sample batch job or utility program fails. Error messages will describe the error. Three types of problems may occur:

- Job control statements are in error

  Correct any JCL errors for the job. Resubmit the job for processing. If the error cause also appears in the SHSSSAMP sample members, collect details as described in "Diagnosis" on page 90 and report the problem to IBM support.
- HSS error messages logged

  Correct the errors as instructed by the error message. Resubmit the job for processing. "HSS Messages" on page 104 describes the error messages and corrective action. If the error messages are not documented or if the recommended action fails to resolve the problem, collect details as described in "Diagnosis" on page 90 and report the problem to IBM support.
- Non-HSS error messages logged

  Correct the errors as instructed by the error message. Resubmit the job for processing. If the error cause also appears in the SHSSSAMP sample members, collect details as described in "Diagnosis" on page 90 and report the problem to IBM support.

Product installation errors

These problems occur when sample installation jobs fail, or the documented installation steps are incomplete, inaccurate or misleading. Four types of problems may occur:

- Job control statements are in error

  Correct any JCL errors for the job. Resubmit the job for processing. If the error cause also appears in the SHSSSAMP sample members or in a sample from the Program Directory on the product installation media, collect details as described in "Diagnosis" on page 90 and report the problem to IBM support.
- HSS error messages logged

Product installation errors

Correct the errors as instructed by the error message. Resubmit the job for processing. "HSS Messages" on page 104 describes the error messages and corrective action. If the error messages are not documented or if the recommended action fails to resolve the problem, collect details as described in "Diagnosis" and report the problem to IBM support.

- Non-HSS error messages logged

  Correct the errors as instructed by the error message. Resubmit the job for processing. If the error cause also appears in the SHSSSAMP sample members, collect details as described in "Diagnosis" and report the problem to IBM support.
- Installation documentation incorrect or inadequate

  Collect details as described in "Diagnosis" and report the problem to IBM support.

Diagnosis

If you are experiencing difficulty using Enhanced Access Control for SCLM, your first step should be to make sure the problem is not due to the way you are using the product. Before going through the procedures described here, you should review "Eliminating User Errors" on page 74.

If you have determined that Enhanced Access Control for SCLM is the cause of your problem, you need to gather information to help isolate the problem and find a solution. The information required is:

- Type of failure
- Function that failed
- Release level
- Maintenance level

Some of the information, for example program number or service level, is independent of the particular problem and does not require you to make a judgment. For other information, you must choose one of several possibilities. Your choice depends on the specific symptoms of the problem.

For reporting the problem to IBM, you need to be prepared to provide supporting materials and evidence such as sample inputs and outputs, and a description of the circumstances in which the problem occurred.

Types of Failure

The following descriptions should help you determine which condition best describes the type of failure that has occurred. If you do not know which condition to select, choose one that best describes the failure.

Abend

This type of failure occurs when a program terminates prematurely. This condition almost always produces a dump. When an abend occurs, collect this information before calling IBM:

- The abend code of the dump
- A brief description of what was entered to cause the abend to occur
- If the abend was a program interrupt:
  - The program that abended
  - The displacement within the program where the abend occurred
  - The data that was being referenced when the abend occurred

Documentation

This problem involves online and hard copy documentation. Report a documentation problem when it falls into the categories listed below:

- Documented descriptions of the Enhanced Access Control for SCLM organization or operation do not match the actual organization or operation.
- Information that is essential to the installation, operation, or service of Enhanced Access Control for SCLM is missing from or incorrect in the documentation.
- Information in the documentation is unclear and prevents the effective use of Enhanced Access Control for SCLM.

Note: If you have suggestions, comments, or questions concerning an Enhanced Access Control for SCLM manual, use the Reader's Comment Form at the back of the manual.

IBM requires this information in order to resolve a documentation problem:

- The complete manual number including the revision number, or the message number or function in error if the error is in the online help text
- The section and page number of the error
- The sentence or sentences in error
- A brief description of what you think is correct

Error An error condition is normally detected by the presence of an error message. Information required resolving this type of problem is:

- The message number
- The program that issued the message, if known
- The data that caused the message to appear

Incorrect Outcome

This type of problem involves the incorrect processing by an Enhanced Access Control for SCLM component, or for the validation of a data set access request. Enhanced Access Control for SCLM is not likely to recognize that a problem exists; therefore, an error message may not appear. IBM needs this information to resolve this type of problem:

- The component in error
- The Rule File definitions in error
- Some indication of why you feel the outcome is incorrect

Loop A loop condition generally causes an abend to occur. MVS has specific abend codes to indicate loop conditions. When a loop occurs, this information is required:

- The program causing the loop
- The abend code and dump information
- As many instructions as can be reasonably determined within the loop
- A brief description of what caused the loop to occur

Message

A message error occurs when a message:

- Contains incorrect data
- Is not documented, or is not documented correctly

- Is generated when it should not be
- Is not generated when it should be
- Is not the message that should occur

The information required to resolve this type of error is:

- The message number
- A brief description of what is wrong with the message
- A brief description of what the message should be.

Performance

A performance problem is generally one of the hardest problems to resolve. Typically, it does not occur in a batch job. If you feel you are having a performance problem with Enhanced Access Control for SCLM, supply IBM with this information:

- Your operating environment, that is, the processor, the operating system, and any other factor that you feel might be contributing to the problem.
- The Enhanced Access Control for SCLM function or component
- The Enhanced Access Control for SCLM module(s), if it can be reasonably determined
- Whether or not the problem always occurs, or only occurs at certain times
- If the problem occurs occasionally, a description of what else was running in the system when the problem occurred.

Wait This type of error normally occurs under these conditions:

- Enhanced Access Control for SCLM is waiting for some condition to be satisfied.
- Enhanced Access Control for SCLM appears to be waiting for some event that is unlikely to occur.
- Enhanced Access Control for SCLM has not recognized the occurrence of an event for which it has suspended processing.

Sometimes a wait error condition generates a dump. You should refer to the appropriate operating system manual to determine the abend code associated with this type of error condition. The information necessary to resolve this type of problem is:

- The component involved
- A dump, if one was generated.

Release Level (VRM)

The release level (Version, Release, Modification) of Enhanced Access Control for SCLM should be stated in all communications with IBM. In addition, you should know the release level of any of these products that are relevant to the problem:

- OS/390 or z/OS
- RACF
- SCLM

Maintenance Level

The maintenance level of Enhanced Access Control for SCLM corresponds to the latest PTF tape installed on Enhanced Access Control for SCLM, plus any Authorized Program Analysis Reports (APARs) installed on top of the Program Temporary Fix (PTF) tape. If no maintenance has been installed on Enhanced

Access Control for SCLM, tell the IBM support representative the date when Enhanced Access Control for SCLM was installed on your system. It is also necessary to know the maintenance level of the products described in the previous section "Release Level (VRM)" on page 92.

Problem Materials and Evidence

If a problem occurs while using Enhanced Access Control for SCLM, this information is required:

- A copy of the Rule File used for the job or for validation processing
- A copy of the job stream used for the job, including the Job Control Language (JCL) and commands
- A listing of the output generated including the messages or violation trace records issued
- A written scenario describing the problem and the conditions applicable at that time.

Problem Materials and Evidence

Chapter 7. Installation

This chapter describes the procedure for installing Enhanced Access Control for SCLM. Before proceeding with this installation, follow the installation instructions in the Program Directory supplied with Enhanced Access Control for SCLM.

To install the product, you need to:
1. Secure the product libraries.
2. Authorize the SHSSLINK library.
3. Authorize the SCLM execution library.
4. Install the Validation Routine into your RACF system.
5. Select the Enhanced Access Control for SCLM MVS subsystem-id.
6. Install the ISPF dialog.
7. Start using Enhanced Access Control for SCLM.

System Requirements

Hardware Requirements

If your OS/390 or z/OS operating system, RACF, and ISPF/PDF were installed in compliance with their documented minimum hardware requirements, you have only these additional requirements to consider in installing Enhanced Access Control for SCLM:

- DASD storage required for the Enhanced Access Control for SCLM product. For information on DASD requirements, refer to the Program Directory that is shipped with Enhanced Access Control for SCLM.

Software Requirements

Enhanced Access Control for SCLM operates in all supported versions of the OS/390 and z/OS environments. Enhanced Access Control for SCLM requires these products:

- The Resource Access Control Facility (RACF) component of the SecureWay Security Server
- ISPF/PDF Version 4 Release 1 (5655-042) or later

These products are components of the OS/390 and z/OS environments.

In addition, SMP/E Release 8.1 (5668-949) or later is required for installation and maintenance.

Storage Requirements

Enhanced Access Control for SCLM utilities execute in virtual storage regions. The online panels execute via TSO/ISPF sessions. Typical storage use begins at 1024K. Here are the recommended minimum region sizes:

- TSO/ISPF session storage size of 2048K for the Enhanced Access Control for SCLM ISPF dialog
- Batch region size of 2048K for the HSSSSINT Rule Load Utility

ECSA storage is required for the Validation Routine, the in-memory rules, and the diagnostic violation trace buffer. The ECSA storage requirements will vary based

on the number of Applications, Profiles and Profile access rules contained within the Rule File, however, ECSA requirements are expected to be in the range of 100K to 300K.

- ECSA storage of 16K is required for the central copy of the Validation Routine
- ECSA storage in the range of 50K to 250K is expected for the in-memory rules
- ECSA storage in the range of 20K to 30K is expected for the violations trace buffer

The ECSA storage requirements may be more accurately predicted using this formula:

ECSA in K = 16 for the Validation Routine
+ 25 for the violations trace buffer
+ Number of Profiles × 0.055
+ Number of Applications × 0.025
+ Number of Profiles × average number of access rules per Profile × 0.015
+ Number of Applications × average number of High-Low Program pairs × 0.025

Here are some example calculations based on this formula:

Example 1

50 Profiles with an average of 30 access rules each
15 Applications with an average of 5 High-Low Program pairs each
70K ECSA

Example 2

100 Profiles with an average of 40 access rules each
20 Applications with an average of 5 High-Low Program pairs each
110K ECSA

Example 3

200 Profiles with an average of 50 access rules each
20 Applications with an average of 5 High-Low Program pairs each
205K ECSA

Example 4

300 Profiles with an average of 80 access rules each
40 Applications with an average of 5 High-Low Program pairs each
425K ECSA

Secure the product libraries

Access to the Enhanced Access Control for SCLM definitions and software should be limited to the central RACF security officer. Use RACF to secure these Enhanced Access Control for SCLM software libraries. Failure to limit access to these libraries results in access control exposures.

- SHSSLINK — the Enhanced Access Control for SCLM execution load library
- SHSSPENU — the Enhanced Access Control for SCLM ISPF panel library

Note: The SHSSLINK data set should not be added to the MVS Linklist. The loading and execution of programs from Linklist data sets is an MVS system function not attributed to the user. Therefore access controls applied to SHSSLINK may be ineffective if the data set is added to the MVS Linklist.

The Enhanced Access Control for SCLM definitions Rule File should be secured in a similar manner when it is created.

Authorize the SHSSLINK library

The Enhanced Access Control for SCLM programs must be executed from an authorized (APF) library. Therefore, authorize the SHSSLINK Enhanced Access Control for SCLM execution library. Do not copy the Enhanced Access Control for SCLM programs into an existing authorized library, as this may circumvent the security controls applied to the Enhanced Access Control for SCLM execution library described in "Secure the product libraries" on page 96.

Authorize the ISPF and SCLM execution libraries

The SCLM programs also need to be executed from APF libraries or common storage. This may already be the case at your installation. Authorize the ISPF/SCLM SISPLPA and SISPLOAD program libraries. The required SCLM programs are considered APF loaded if:

- The programs have been loaded into the MVS link pack area (LPA).
- The program libraries are defined in the MVS Linklist and your installation has the MVS parameter setting LINKAUTH=LINKLIST
- The program libraries are defined in the MVS authorized library list.

Note: Other programs executed via SCLM translators may also need to reside in APF libraries. This is discussed in "Review SCLM translators" on page 24.

Install the Validation Routine into your RACF system

The Enhanced Access Control for SCLM Validation Routine must be installed into your RACF environment. You may do this:

Statically Apply an SMP/E ++USERMOD to invoke the HSSRCX02 Validation Routine Interface program from the RACF RACHECK Post Processing exit ICHRCX02. This method will require a system IPL for the exit to take effect. However, once this ++USERMOD is applied, the Validation Routine Interface program will remain in effect on successive IPLs.

Dynamically The Enhanced Access Control for SCLM INSTALL operator command may be used to dynamically install the Validation Routine into your RACF environment. This dynamic method does not require an IPL. However, the dynamic install remains in effect only for the current MVS session. A subsequent IPL requires the Validation Routine to be installed again.

The dynamic installation method does not require the SMP/E ++USERMOD to be applied to the RACF RACHECK Post Processing exit, nor does it use the HSSRCX02 Validation Routine Interface program.

You may combine the methods. For example, you may apply SMP/E ++USERMOD as a permanent solution, and use the INSTALL command as a temporary measure for the current MVS session while you are waiting for the next system IPL and the SMP/E ++USERMOD to take effect.

Static installation using an SMP/E ++USERMOD

"HSSUMOD1 - SMP/E ++USERMOD to install the Validation Routine Interface program" on page 63 describes the sample SMP/E ++USERMOD and provides tailoring instructions. The SHSSSAMP sample library member HSSUMOD1 contains the SMP/E ++USERMOD source code.

This SMP/E ++USERMOD defines an installation exit for the RACF RACHECK Post Processing Exit ICHRCX02. This exit is invoked by RACF following its normal data set access validation. Enhanced Access Control for SCLM uses this exit to invoke its Validation Routine Interface program (HSSRCX02), which in turn invokes the Validation Routine (HSSRVALD). Refer to "Interaction with RACF" on page 7 for a description of how the RACF ICHRCX02 exit interacts with the Enhanced Access Control for SCLM Validation Routine.

If the ICHRCX02 RACF exit is already in use at your installation, you may alter your exit code to invoke the Enhanced Access Control for SCLM Validation Routine Interface program. The sample SMP/E ++USERMOD code contains a section starting with the comment header **2. Call Enhanced Access Control for SCLM RACF Exit** that may be inserted into your exit code to do this. Note that R1 (register 1) must contain the same value as that provided to the ICHRCX02 on its invocation.

Alternatively, you may copy your installation exit code into the sample SMP/E ++USERMOD in the area starting with the comment header **1. Installation ICHRCX02 code is inserted here**.

If the ICHRCX02 RACF exit is not in use at your installation, you may use the sample SMP/E ++USERMOD code as supplied. Follow the tailoring notes as described in the section "HSSUMOD1 - SMP/E ++USERMOD to install the Validation Routine Interface program" on page 63.

The ICHRCX02 exit code should be prepared as an SMP/E ++USERMOD, and not as a module prepared outside the control of SMP/E. Using SMP/E ensures that product maintenance applied to the Enhanced Access Control for SCLM Validation Routine Interface program will automatically result in SMP/E JCLIN processing to refresh the ICHRCX02 exit code.

The SMP/E ++USERMOD places the ICHRCX02 executable module into SYS1.LPALIB. This makes the exit accessible to RACF at the next MVS system IPL. At start-up, RACF automatically detects the presence of the ICHRCX02 exit, and begins to use the exit. There is no need to change RACF definitions for the exit to take effect. The ICHRCX02 exit will remain in effect for successive IPLs. The exit may be removed by its deletion from the SYS1.LPALIB data set. An SMP/E RESTORE should be used if the exit is to be removed.

Dynamic installation using the INSTALL command

Refer to "INSTALL Command" on page 70 for a description of the INSTALL command. This command dynamically installs the Enhanced Access Control for SCLM Validation Routine into your RACF environment.

The dynamic installation takes effect immediately without disruption to RACF services, and remains in effect until the next MVS system IPL. The dynamic install method may be used if the RACF RACHECK Post Processing exit ICHRCX02 is

already active or not. If already active, the installation exit code is preserved and executed first followed by the Enhanced Access Control for SCLM Validation Routine processing.

The INSTALL command may be issued after the MVS subsystem used by Enhanced Access Control for SCLM is active. This subsystem is activated on the first successful execution of the HSSSSINT Rule Load Utility, which loads the Enhanced Access Control for SCLM definitions into memory. Therefore, the dynamic installation of the Validation Routine requires this sequence of events:

1. A Rule File must be created and populated with Application and Profiles definitions.
2. Execution of the HSSSSINT Rule Load Utility activates the MVS subsystem.
3. The INSTALL operator command may be used to dynamically install the Validation Routine into RACF.

When installing Enhanced Access Control for SCLM for the first time, the Rule File and its associated definitions may not be prepared. Therefore, use of the INSTALL operator command will need to be deferred until such time as the HSSSSINT utility has been successfully executed.

Select the Enhanced Access Control for SCLM MVS subsystem-id

An MVS subsystem-id must be assigned for use by Enhanced Access Control for SCLM. "The MVS Subsystem" on page 6 describes the subsystem.

The MVS subsystem-id is specified as a JCL parameter for the execution of the Rule Load Utility. The default value is HSS#, however this may be changed. "HSSSSINT - Rule Load Utility" on page 61 describes the utility along with the parameter specification.

You need to determine the MVS subsystem-id that will be used in the utility's JCL. You may also want to create a JCL procedure for the execution of the utility. Refer to "HSSRLOAD - Rule Load Utility JCL" on page 60 for sample JCL.

Install the ISPF dialog

You can install the Enhanced Access Control for SCLM libraries into your ISPF environment statically or dynamically. The two installation methods are described in the following sections.

Statically Adds the Enhanced Access Control for SCLM libraries to your TSO/ISPF logon procedure. This method is not recommended. If the libraries are secured as described in "Secure the product libraries" on page 96, this may necessitate the creation of a new TSO/ISPF logon procedure specifically for the use of the Enhanced Access Control for SCLM administrator, otherwise data set access violations may occur during TSO logon.

Dynamically Adds the Enhanced Access Control for SCLM libraries to your TSO/ISPF session dynamically when the Enhanced Access Control for SCLM invocation REXX procedure is executed. This is the preferred method, as the libraries are only assigned when needed using LIBDEF and ALTLIB services. This method may be used for existing TSO/ISPF logon procedures.

Install the ISPF dialog

The components of the Enhanced Access Control for SCLM dialog are delivered in these libraries:

SHSSEXEC REXX EXECs

SHSSLINK Link/Load modules

SHSSM*xxx* ISPF messages

SHSSP*xxx* ISPF panels

SHSST*xxx* ISPF input tables

where *xxx* identifies the national language. For example, SHSSPENU is the ISPF panel library for U.S. English.

The Enhanced Access Control for SCLM ISPF dialog is invoked using the HSSOREXX REXX EXEC. This EXEC may be executed from the TSO command processor panel, or added to an ISPF menu panel. Some examples of these two formats appears as:

TSO command

```
EX '<hlq>.SHSSEXEC(HSSOREXX)' '<hlq> <language>'
%HSSOREXX' 'NODYNAM <language>'
```

ISPF menu

```
CMD(EX ''<hlq>.SHSSEXEC(HSSOREXX)'' ''<hlq> <language>'') NOCHECK
CMD(EX %HSSOREXX ''NODYNAM <language>'') NOCHECK
```

The Enhanced Access Control for SCLM ISPF command HSSOREXX accepts four parameters:

HLQ The data set name high level qualifiers for Enhanced Access Control for SCLM data sets, or NODYNAM. NODYNAM specifies that dynamic allocation will not be used for the Enhanced Access Control for SCLM libraries.

Language Optional. Identifies the national language. Currently, Enhanced Access Control for SCLM only supports U.S. English (language value of ENU), which is also the default if language is omitted.

PASSAPPL Optional. Overrides the enforcement of the default Enhanced Access Control for SCLM application NEWAPPL(HSSO).

LLQs Optional. Overrides the default data set name low-level qualifiers for the five Enhanced Access Control for SCLM data sets. If specified, all five qualifiers must be entered in the correct order, enclosed in brackets and separated by commas. The default low level qualifiers in sequence are:

```
(SHSSEXEC,SHSSLINK,SHSSMENU,SHSSPENU,SHSSTENU)
```

but you could replace these with different low level qualifiers like:

```
(EXEC,LINKLIB,MESSAGES,PANELS,TABLES)
```

Dynamic Setup

To enable the Enhanced Access Control for SCLM libraries to be dynamically set up when the Enhanced Access Control for SCLM dialog is used:

1. On the TSO command processor panel, enter
   ```
   EX '<hlq>.SHSSEXEC(HSSOREXX)' '<hlq> <language>'
   ```
2. To add Enhanced Access Control for SCLM to an ISPF menu, set &ZSEL to:

```
CMD(EX ''<hlq>.SHSSEXEC(HSSOREXX)'' '<hlq> <language>'') NOCHECK
```

NOCHECK is specified to support entry of concatenated commands via the direct option (trail). Also specify on the calling panel:

```
&ZTRAIL=.TRAIL
```

Static Setup

To install the Enhanced Access Control for SCLM libraries statically within your ISPF library setup:

1. Include the library <hlq>.SHSSEXEC in your SYSEXEC or SYSPROC concatenation. This library contains the required EXECs. It is allocated with fixed-block 80 record format during installation.

   You should put these libraries in the SYSEXEC concatenation. However, if you want to put them in SYSPROC, it must have a record length of 80 bytes.

   Ensure that all libraries contained in your concatenations are either in the same format (F, FB, V, VB) and have the same block size, or are in order of decreasing block sizes. Otherwise, you may experience problems using Enhanced Access Control for SCLM.
2. Add the remaining libraries to your ISPF library setup:
   - Include the panel library <hlq>.SHSSPENU (or other language) in the ISPPLIB concatenation.
   - Include the link/load module library <hlq>.SHSSLINK in the ISPLLIB concatenation.
   - Include the table library <hlq>.SHSSTENU (or other language) in the ISPTLIB concatenation.
   - Include the message library <hlq>.SHSSMENU (or other language) in the ISPMLIB concatenation.
3. On the TSO command processor panel, enter:

   ```
   %HSSOREXX 'NODYNAM <language>'
   ```
4. To add Enhanced Access Control for SCLM to an ISPF menu, set &ZSEL to:

   ```
   CMD(%HSSOREXX ''NODYNAM <language>'') NOCHECK
   ```

Overriding the Default Application

To override the default Enhanced Access Control for SCLM application, use the PASSAPPL parameter in the ISPF menu &ZSEL setting:

```
CMD(EX ''<hlq>.SHSSEXEC(HSSOREXX)''' ''<hlq> <language> PASSAPPL'') NOCHECK NEWAPPL(HSSP)
```

Enhanced Access Control for SCLM will then use HSSP as the application, rather than the default of HSSO.

Overriding the Data Set Low Level Qualifiers (LLQs)

To override the default Enhanced Access Control for SCLM data set low level qualifiers, specify the five data set low level qualifiers as the last parameter in the ISPF menu &ZSEL setting. For example:

```
%HSSOREXX ''<hlq> <language> (EXEC,LINKLIB,MESSAGES,PANELS,TABLES)")
```

where Enhanced Access Control for SCLM uses:

'<hlq>.EXEC'	as the REXX EXEC library
'<hlq>.LINKLIB'	as the Link/Load library
'<hlq>.MESSAGES'	as the ISPF messages library

Overriding the Data Set Low Level Qualifiers (LLQs)

'<hlq>.PANELS'	as the ISPF panels library
'<hlq>.TABLES'	as the ISPF input tables library

Start using Enhanced Access Control for SCLM

The basic installation tasks are now complete, and the Enhanced Access Control for SCLM administrator may now start using the product. "Getting Started" on page 9 provides instructions for the administrator. These include:

- Defining the Rule File
- Creating Application and Profile definitions
- Loading the definitions into memory via the HSSSSINT Rule Load Utility
- Execution of the INSTALL command if dynamic installation of the Validation Routine is required
- Testing and refinement of the Profile access rules

Chapter 8. Messages

This chapter lists the messages issued by Enhanced Access Control for SCLM, a message description, the action the system takes when the message is issued, and the action you should take when you get the message. The messages are listed in numerical order.

The return codes set at the completion of batch processing are also described, along with an explanation of the messages format.

Return Codes

Enhanced Access Control for SCLM programs return a code in register 15 to indicate the results of program execution. The return codes and their meanings are:

Return Code	Meaning
00 (x'00')	Operation was successful.
04 (x'04')	Operation completed, but a warning (W) message was issued during processing.
08 (x'08')	Operation may be incomplete. An error (E) condition caused premature termination.
12 (x'0C')	Operation is incomplete. A severe error (S) condition occurred.

Message Format

The Enhanced Access Control for SCLM messages begin with a unique message identifier, followed by message text which may contain variable information to identify the particular circumstance which caused the message:

```
HSSnnnnx Message text with variable information
```

The message identifier has the format **HSSnnnnx** where:

HSS
: The **program identifier** identifies the message as an Enhanced Access Control for SCLM message. All Enhanced Access Control for SCLM messages begin with HSS.

nnnn
: The **message identification number** is a four-digit number that uniquely identifies each message.

x
: The **severity level** is a letter that indicates the return code, the purpose of the message, and the type of response required. The severity levels, from least to most severe, are:

  I
  : Information. No action is required.

  W
  : Warning. Enhanced Access Control for SCLM detected a possible error condition that the user should evaluate.

  E
  : Error. User action is required before Enhanced Access Control for SCLM can continue processing.

  S
  : Severe. Enhanced Access Control for SCLM processing is suspended until action has been taken.

The text of the message itself follows the diagnostic information and completes the standard format for Enhanced Access Control for SCLM messages.

HSS Messages

HSS0001I HSS SUBSYSTEM ''<ssid>'' HAS BEEN INITIALIZED

Explanation: The MVS subsystem used by Enhanced Access Control for SCLM has been successfully initialized. <ssid> is the MVS subsystem-id.

System Action: Processing continues.

User Response: None required.

HSS0002E HSS SUBSYSTEM ''<ssid>'' INITIALIZATION FAILED. ABEND <abend> LOADING HSSSSFNR

Explanation: The MVS subsystem used by Enhanced Access Control for SCLM failed initialization, because an ABEND occurred while loading the HSSSSFNR program. <abend> is the ABEND code. <ssid> is the MVS subsystem-id.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors. Enhanced Access Control for SCLM is not operational.

User Response: If the ABEND code is S806-04, then correct the job JCL to add a STEPLIB for the SHSSLINK data set in which the HSSSSFNR resides, and resubmit the job. Refer to the MVS System Codes manual for explanations of other ABEND codes. Correct the problem and resubmit the job. If the cause of the problem is indeterminate, contact your IBM representative for help.

HSS0003E HSSSSINT NOT LOADED FROM APF DATA SET, PROCESSING TERMINATED

Explanation: The HSSSSINT program must be loaded from an APF library.

System Action: The HSSSSINT Rule Load Utility terminates. Processing fails with errors.

User Response: Ensure that the HSSSSINT program resides in an APF library. If the HSSSSINT program library was located by a STEPLIB or JOBLIB statement in the job JCL, ensure that all of the program libraries are APF in the STEPLIB or JOBLIB data set concatenation list. Authorize the program library, or correct the job JCL. Resubmit the job.

HSS0004E <ssid>: INVALID COMMAND RECEIVED

Explanation: An invalid command was passed to the MVS subsystem used by Enhanced Access Control for SCLM. <ssid> is the MVS subsystem-id issuing the message.

System Action: The operator command is ignored. Processing continues.

User Response: Check the command for spelling and syntax correctness. Ensure that the command was issued for the correct MVS subsystem. Enter only valid Enhanced Access Control for SCLM commands.

HSS0005I <ssid>: VALIDATION EXIT PROCESSING HAS BEEN DISABLED

Explanation: The DISABLE operator command was processed successfully. Validation Routine processing has been disabled. <ssid> is the MVS subsystem-id issuing the message.

System Action: Data set access validation processing ceases. Data set access violations will occur in cases where Enhanced Access Control for SCLM previously granted access.

User Response: None required. You may use the ENABLE operator command to resume Enhanced Access Control for SCLM validation processing.

HSS0006I <ssid>: VALIDATION EXIT PROCESSING HAS BEEN ENABLED

Explanation: The ENABLE operator command was processed successfully. Validation Routine processing has been enabled. <ssid> is the MVS subsystem-id issuing the message

System Action: Data set access validation processing resumes. The Validation Routine will perform its normal access control services as RACF data set violations occur.

User Response: None required. You may use the DISABLE operator command to cease Enhanced Access Control for SCLM validation processing.

HSS0007E VSAM <function> ERROR RC=<rc>

Explanation: VSAM OPEN, GET or CLOSE processing of the Rule File has failed. <function> is the VSAM function being processed. <rc> is the VSAM return code for the function.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Investigate the cause of the error by checking the DFSMS/MVS Macro Instructions for Data Sets manual for VSAM function return codes. Correct the problem and resubmit the job. If the problem persists, contact your IBM representative for help.

HSS0008I <ssid>: VALIDATION EXIT DYNAMIC INSTALL SUCCESSFUL

Explanation: The INSTALL operator command has successfully completed the dynamic install of the Validation Routine. <ssid> is the MVS subsystem-id issuing the message.

System Action: Processing continues

User Response: None required.

HSS0009E HSS SUBSYSTEM NOT INITIALIZED

Explanation: The MVS subsystem used by Enhanced Access Control for SCLM failed initialization.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors. Enhanced Access Control for SCLM is not operational.

User Response: Examine the SYSPRINT messages in the job output for the HSSSSINT Rule Load Utility step for other error messages. Correct the cause of failure and resubmit the job. If the cause of failure is indeterminate, contact your IBM representative for help.

HSS0010E RACF REQUEST=EXTRACT,TYPE=EXTRACT ERROR, SAFRC=<saf rc>, RACF RC=<rc>, RACF REAS=<reason code>

Explanation: The Validation Routine has issued a RACF EXTRACT request to obtain Profile information. This request has failed. Information regarding the request and the RACF reason and returns codes is provided in the message. <saf-rc> is the RACF SAF call return code. <rc> is the RACF return code. <reason-code> is the RACF reason code.

System Action: Validation for the data set access request fails. Access will not be granted for this data set access request. Validation Routine processing continues for other data set access requests.

User Response: Refer to the Security Server RACROUTE Macro Reference manual for an explanation of the RACF error codes. Take corrective action as appropriate. If the problem persists, contact your IBM representative for help.

HSS0012E OPEN FAILED FOR DDNAME <ddname> RC=<rc>

Explanation: The OPEN of a data set has failed. <ddname> is the DDname of the data set within the job JCL. <rc> is the return code from the OPEN instruction.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Refer to the DFSMS/MVS Macro Instructions for Data Sets manual for an explanation of the OPEN statement return codes. Correct the cause of failure and resubmit the job. If the cause of failure is indeterminate or if the problem persists, contact your IBM representative for help.

HSS0013E DDNAME <ddname> NOT SPECIFIED IN JCL

Explanation: Required DDname <ddname> is missing from the job JCL.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Correct the job JCL to include a DD statement for DDname <ddname>, and resubmit the job.

HSS0014E RDJFCB ERROR FOR DDNAME <ddname> RC=YY

Explanation: The HSSSSINT Rule Load Utility received a bad return code from the RDJFCB macro for DDname <ddname>.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Contact your IBM representative for help.

HSS0015E MVS SUBSYSTEM PARAMETER IS INVALID

Explanation: The job JCL EXEC statement is incorrect for the HSSSSINT Rule Load Utility. The EXEC PARM parameter is invalid. The parameter should appear as PARM='SSID=<ssid>' where <ssid> is a valid MVS subsystem-id name:

- maximum of 4 characters in length
- consisting of the National characters A to Z, 0 to 9, @, #, $
- the first character cannot be numeric

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Correct the EXEC statement PARM parameter in the job JCL, and resubmit the job.

HSS0016E SUBSYSTEM NAMES CAN BE UP TO 4 CHARACTERS LONG; CONSIST OF THE CHARACTERS A TO Z, 0 TO 9, @, #, OR $; AND THE FIRST CHARACTER MAY NOT BE NUMERIC.

Explanation: The value for the MVS subsystem-id parameter on the EXEC statement in the job JCL for the HSSSSINT Rule Load Utility is invalid. MVS subsystem-id names are a maximum of 4 characters in length; consisting of the National characters A to Z, 0 to

9, @, #, $; and the first character cannot be numeric.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Correct the MVS subsystem-id name parameter value on the EXEC statement in the job JCL, and resubmit the job.

HSS0017E A SUBSYSTEM ALREADY EXISTS CALLED <ssid>, IT IS NOT THE SAME NAME AS SPECIFIED ON THE EXEC STATEMENT

Explanation: The MVS subsystem-id parameter on the EXEC statement in the job JCL for the HSSSSINT Rule Load Utility does not match the name of the currently active MVS subsystem for Enhanced Access Control for SCLM. <ssid> is the MVS subsystem-id currently active for Enhanced Access Control for SCLM.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Correct the MVS subsystem-id name parameter value on the EXEC statement in the job JCL. The parameter value must match currently active MVS subsystem-id, shown as <ssid> in the error message text. Correct the job JCL, and resubmit the job.

HSS0018E SWAREQ ERROR FOR DDNAME HSSRULES RC=<rc>

Explanation: The HSSSSINT Rule Load Utility received a bad return code from the SWAREQ macro for DDname HSSRULES.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Contact your IBM representative for help.

HSS0019E DDNAME HSSRULES NOT FOUND IN TIOT TABLE

Explanation: The HSSSSINT Rule Load Utility was unable to locate the DDname HSSRULES in the Task Input/Output Table (TIOT).

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Contact your IBM representative for help.

HSS0020E <ssid>: VALIDATION EXIT ADDRESS NOT DEFINED TO THE SUBSYSTEM

Explanation: The address of the Validation Routine is missing in the communications vector table (CVT) of MVS subsystem used by Enhanced Access Control for SCLM. An error occurred during the initial execution of the HSSSSINT Rule Load Utility, resulting in the failure to insert the Validation Routine address into the MVS subsystem CVT. <ssid> is the MVS subsystem-id.

System Action: Processing fails. The INSTALL operator command is unsuccessful. Enhanced Access Control for SCLM is not operational.

User Response: Run the HSSRELOD utility, and then retry the INSTALL operator command. If the problem persists, contact your IBM representative for help.

HSS0021E <ssid>: VALIDATION EXIT IS ALREADY DYNAMICALLY INSTALLED

Explanation: The Validation Routine is already active from a previous dynamic install via an INSTALL operator command. <ssid> is the MVS subsystem-id issuing the message.

System Action: Processing fails. The INSTALL operator command is unsuccessful. Enhanced Access Control for SCLM remains operational.

User Response: None required. Do not issue successive INSTALL operator commands.

HSS0022I RULE LOAD COMPLETED SUCCESSFULLY

Explanation: The HSSSSINT Rule Load Utility has completed processing successfully.

System Action: Processing continues. The Validation Routine will perform data set access validation against the refreshed Profile rules.

User Response: None required.

HSS0023W INVALID PROFILE NOT LOADED: NO VALID APPLICATIONS FOUND. PROFILE IS <profile>

Explanation: The Rule File contains a Profile definition that does not reference any valid Applications. Either the Profile does not reference any defined Applications, or the referenced Applications are invalid owing to errors with their High and Low Program pairs. <profile> is the Profile name.

System Action: Processing continues for other Profile definitions. The HSSSSINT Rule Load Utility does not load the offending Profile definition into memory and will complete with warnings. The Validation Routine will not validate the offending Profile after the HSSSSINT successfully completes.

User Response: Correct the Profile definition. Either delete this Profile or update the Profile's access rules to specify valid Applications. Resubmit the job.

HSS0024W INVALID PROFILE NOT LOADED: NO USERS SPECIFIED. PROFILE IS <profile>

Explanation: The Rule File contains a Profile definition without any access rules. <profile> is the Profile name.

System Action: Processing continues for other Profile definitions. The HSSSSINT Rule Load Utility does not load the offending Profile definition into memory and will complete with warnings. The Validation Routine will not validate the offending Profile after the HSSSSINT successfully completes.

User Response: Correct the Profile definition. Either delete this Profile or insert Profile access rules. Resubmit the job.

HSS0025W INVALID APPLICATION NOT LOADED: NO VALID PROGRAM PAIRS FOUND, APPLICATION IS <application> <function>

Explanation: The Rule File contains an Application definition that does not have any valid High and Low Program pairs. <application> is the Application name. <function> is the Function name.

System Action: Processing continues for other Application definitions. The HSSSSINT Rule Load Utility does not load the offending Application definition into memory, and will complete with warnings. HSSSSINT will issue error message HSS0026W for Profiles containing access rules that reference the offending Application.

User Response: Correct the Application definition. Either delete this Application, correct the High and Low Program definitions, or insert valid High and Low Program definitions. Resubmit the job.

HSS0026W UNDEFINED APPLICATION REFERENCED IN PROFILE IS IGNORED. PROFILE IS <profile> APPLICATION IS <application> <function>

Explanation: The Rule File contains a Profile that referenced an undefined or invalid Application definition. Message HSS0025W is issued for invalid Applications. <profile> is the Profile name. <application> is the Application name. <function> is the Function name.

System Action: Processing continues for the Profile, however the HSSSSINT Rule Load Utility does not load the access rules referencing the offending Application. The HSSSSINT utility will complete with warnings.

User Response: Correct the Profile or Application definitions. Either delete the Profile access rules that reference the offending Application, or correct the Application definitions. Resubmit the job.

HSS0027E THE RULE FILE DOES NOT CONTAIN ANY VALID PROFILE RECORDS

Explanation: No valid Profiles exist within the Rule File. Either no Profile definitions were found, or all Profile definitions are invalid. Messages HSS0023W or HSS0024W are issued for invalid Profiles.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors, and the in-memory rules are not refreshed. Enhanced Access Control for SCLM retains its previously operational status.

User Response: Review the output for other Profile and Application definition warning messages, and correct these. If the Rule File contains no Profile definitions, then insert a Profile definition. Resubmit the job.

HSS0028E THE RULE FILE DOES NOT CONTAIN ANY VALID APPLICATION RECORDS

Explanation: No valid Application definitions exist within the Rule File. Either no Application definitions were found, or all Application definitions are invalid. Message HSS0025W is issued for invalid Applications.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors, and the in-memory rules are not refreshed. Enhanced Access Control for SCLM retains its previously operational status.

User Response: Review the output for other Application definition warning messages, and correct these. If the Rule File contains no Application definitions, then insert an Application definition. A valid Profile definition is also required. Resubmit the job.

HSS0029E RECORD <number> IN RULE FILE IS INVALID

Explanation: The Rule File contains an invalid record. <number> is the record number in error.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors.

User Response: Delete the offending record from your Rule File. You can achieve this by defining a new Rule File using the IDCAMS utility, and then using the REPRO and SKIP options to copy the valid records from the damaged Rule File into the new Rule File. If the problem persists or the cause of the damage is indeterminate, contact your IBM representative.

HSS0030I RULE FILE <rule-file>

Explanation: The Rule File processed by the HSSSSINT Rule Load Utility is shown as <rule-file>.

System Action: Processing continues.

User Response: None required.

HSS0031E LPA LOAD FAILED (MACRO CSVDYLPA REQUEST=ADD RC=<rc>, RS=<reason-code>)

Explanation: The HSSSSINT Rule Load Utility or HSSRELOD utility received a bad return code from the CSVDYLPA macro when attempting to load the Validation Routine into LPA. <rc> is the CSVDYLPA return code, <reason-code> is the reason code.

System Action: Processing fails. The utility terminates with errors.

User Response: Refer to the MVS Authorized Assembly Services Reference (Volume 1) manual for an explanation of the CSVDYLPA return codes. Take corrective action where appropriate. If the problem persists or is indeterminate, contact your IBM representative for help.

HSS0032E ERROR LOADING POST VALIDATION EXIT ROUTINE. ABEND <abend>

Explanation: The HSSSSINT Rule Load Utility failed to load the Validation Routine owing to an ABEND. <abend> is the ABEND code.

System Action: Processing fails. The HSSSSINT Rule Load Utility terminates with errors. Enhanced Access Control for SCLM is not operational.

User Response: If the ABEND code is S806-04, then correct the job JCL to add a STEPLIB for the SHSSLINK data set in which the HSSSSFNR resides, and resubmit the job. Refer to the MVS System Codes manual for explanations of other ABEND codes. Correct the problem and resubmit the job. If the cause of the problem is indeterminate, contact your IBM representative for help.

HSS0033W RULE LOAD COMPLETED SUCCESSFULLY WITH WARNINGS

Explanation: The HSSSSINT Rule Load Utility has completed with warning messages written to DDname SYSPRINT.

System Action: Processing continues. The in-memory Profile and Application definitions have been refreshed.

User Response: Examine DDname SYSPRINT in the job output to view HSSSSINT warning messages. Correct the Profile and Application definitions where appropriate. Resubmit the job.

HSS0034E <ssid>: RACF NOT FOUND ON SYSTEM, INSTALL/UNINSTALL FAILED

Explanation: The INSTALL or UNINSTALL operator command has failed because RACF was not active on the system. <ssid> is the MVS subsystem-id issuing the message.

System Action: The INSTALL or UNINSTALL command fails.

User Response: Ensure that RACF is active on your system before issuing the INSTALL or UNINSTALL operator commands. If RACF is active and the problem persists, contact your IBM representative for help.

HSS0035I <ssid>: TRACE DISABLED

Explanation: The TRACE,OFF operator command has been processed successfully. TRACE is now disabled. <ssid> is the MVS subsystem-id issuing the message.

System Action: Processing continues. Diagnostic tracing ceases.

User Response: None required.

HSS0036I <ssid>: TRACE ENABLED

Explanation: The TRACE,USER=<user-id> operator command has been processed successfully. TRACE is now enabled for the user specified in the TRACE command. <ssid> is the MVS subsystem-id issuing the message.

System Action: Processing continues. Diagnostic tracing starts, with information written to the system log.

User Response: None required. You may deactivate tracing by issuing the TRACE,OFF operator command.

HSS0037E <ssid>: UNINSTALL FAILED, VALIDATION EXIT NOT DYNAMICALLY INSTALLED

Explanation: The Validation Routine was not dynamically installed via the INSTALL operator command. Therefore it cannot be removed dynamically via the UNINSTALL command. <ssid> is the MVS subsystem-id issuing the message.

System Action: The UNINSTALL operator command fails.

User Response: If you wish to stop Enhanced Access Control for SCLM processing, issue the DISABLE operator command. Enhanced Access Control for SCLM will remain on your system until the next IPL. To remove Enhanced Access Control for SCLM from your system following the next IPL, reverse the SMP ++USERMOD applied to the RACF RACHECK Post Processing exit (ICHRCX02) as described in “Static installation using an SMP/E ++USERMOD” on page 98, and do not run the HSSSSINT Rule Load Utility.

HSS0038E <ssid>: VALIDATION EXIT IS NOT THE PRIMARY EXIT

Explanation: The UNINSTALL operator command has determined that the active RACF RACHECK Post Processing exit (ICHRCX02) is not the Enhanced Access

Control for SCLM Validation Routine. Another product or installation code is the primary exit. <ssid> is the MVS subsystem-id issuing the message.

System Action: UNINSTALL processing fails.

User Response: UNINSTALL cannot be used to remove the Enhanced Access Control for SCLM Validation Routine from you RACF environment. If you wish to stop Enhanced Access Control for SCLM Validation Routine processing, you may use the DISABLE operator command.

HSS0039I <ssid>: VALIDATION EXIT DYNAMIC UNINSTALL SUCCESSFUL

Explanation: The UNINSTALL operator command has completed successfully. <ssid> is the MVS subsystem-id issuing the message.

System Action: The Validation Routine has been removed from the system, therefore data set access validations will no longer occur. Data set violations occur in cases where Enhanced Access Control for SCLM previously granted access. The Enhanced Access Control for SCLM MVS subsystem remains operative.

User Response: None required. You may use the INSTALL operator command to dynamically install the Validation Routine. Enhanced Access Control for SCLM thereafter resumes its former operational status.

HSS0040E HSSRELOD NOT LOADED FROM APF DATA SET, PROCESSING TERMINATED

Explanation: The HSSRELOD program must be loaded from an APF library.

System Action: The HSSRELOD utility terminates. Processing fails with errors.

User Response: Ensure that the HSSRELOD program resides in an APF library. If the HSSRELOD program library was located by a STEPLIB or JOBLIB statement in the job JCL, ensure that all of the program libraries are APF in the STEPLIB or JOBLIB data set concatenation list. Authorize the program library, or correct the job JCL. Resubmit the job.

HSS0041I LOAD OF NEW POST VALIDATION EXIT SUCCESSFUL

Explanation: The HSSRELOD utility has successfully loaded the Validation Routine.

System Action: Processing continues.

User Response: None required.

HSS0042I HSSRELOD PROCESSING COMPLETED RC=<rc>

Explanation: The HSSRELOD utility has completed processing. <rc> is the HSSRELOD return code.

System Action: The HSSRELOD utility terminates.

User Response: If a non-zero return code is indicated, examine the DDname SYSPRINT in the job output to view HSSRELOD warning or error messages. Take corrective action where appropriate. Resubmit the job.

HSS0043I REMOVAL OF OLD POST VALIDATION EXIT SUCCESSFUL

Explanation: The HSSRELOD utility has successfully removed the previously loaded Validation Routine.

System Action: Processing continues.

User Response: None required.

HSS0044E HSSSSINT RULE LOAD PROCESSING FAILED

Explanation: The HSSSSINT Rule Load Utility has completed with error messages written to DDname SYSPRINT.

System Action: HSSSSINT terminates with errors. The in-memory Profile and Application definitions have not been refreshed.

User Response: Examine DDname SYSPRINT in the job output to view HSSSSINT warning and error messages. Correct the Profile and Application definitions where appropriate. Resubmit the job.

HSS0045W ERROR REMOVING OLD POST VALIDATION EXIT

Explanation: The HSSRELOD utility encountered an error when deleting the replaced (old) Validation Routine from memory. The replacement (new) Validation Routine has been loaded into memory and switched to become the active Validation Routine.

System Action: Processing continues. HSSRELOD will complete with warning messages.

User Response: None required. The replacement (new) Validation Routine is active. The replaced (old) Validation Routine, if still present, will be removed at the next system IPL.

HSS0046E Enhanced Access Control for SCLM SUBSYSTEM CVT NOT FOUND

Explanation: The HSSRELOD utility cannot locate the communication vector table (CVT) for the Enhanced Access Control for SCLM MVS subsystem. Either the HSSSSINT Rule Load Utility that installs and initializes the MVS subsystem has not been run, or it failed on its

first execution on the system.

System Action: HSSRELOD processing terminates with errors. Enhanced Access Control for SCLM is not operative.

User Response: If the HSSSSINT Rule Load Utility was not previously run, submit this now to make Enhanced Access Control for SCLM operational. There is no need to resubmit the job for the HSSRELOD utility, as the HSSSSINT utility loads the Validation Routine into memory on its first successful execution.

If the HSSSSINT Rule Load Utility failed during execution, examine DDname SYSPRINT in the job output to view HSSSSINT error messages. Correct the cause of failure and resubmit the HSSSSINT job. When this completes successfully, resubmit the job to execute the HSSRELOD utility.

HSS0047E Enhanced Access Control for SCLM SUBSYSTEM NOT INITIALIZED

Explanation: The HSSRELOD utility has failed because the Enhanced Access Control for SCLM MVS subsystem has not been initialized. Either the HSSSSINT Rule Load Utility that installs and initializes the MVS subsystem has not been run (or completed execution), or it failed on its first execution on the system.

System Action: HSSRELOD processing terminates with errors.

User Response: If the HSSSSINT Rule Load Utility was not previously run, submit this now to make Enhanced Access Control for SCLM operational. There is no need to resubmit the job for the HSSRELOD utility, as the HSSSSINT utility loads the Validation Routine into memory on its first successful execution.

If the HSSSSINT Rule Load Utility executed concurrently with this HSSRELOD utility execution, wait until the HSSSSINT job completes and then resubmit the HSSRELOD job.

If the HSSSSINT Rule Load Utility failed during execution, examine DDname SYSPRINT in the job output to view HSSSSINT error messages. Correct the cause of failure and resubmit the HSSSSINT job. When this completes successfully, resubmit the job to execute the HSSRELOD utility.

HSS0048E UNINSTALL COMMAND REQUIRED BEFORE HSSRELOD UTILITY CAN RUN

Explanation: The UNINSTALL operator command must be issued before the HSSRELOD may be run.

System Action: HSSRELOD processing terminates with errors.

User Response: This sequence of actions is required in order to reload a new copy of the Validation Routine via the HSSRELOD utility:

1. Issue the DISABLE operator command to stop Validation Routine processing.
2. Issue the UNINSTALL operator command to remove the Validation Routine as a RACF exit.
3. Execute the HSSRELOD utility to replace the in-memory copy of the Validation Routine.
4. Issue the INSTALL operator command to insert the Validation Routine as a RACF exit.
5. Issue the ENABLE operator command to start Validation Routine processing.

HSS0049E DISABLE COMMAND REQUIRED BEFORE HSSRELOD UTILITY CAN RUN

Explanation: The DISABLE operator command must be issued before the HSSRELOD may be run.

System Action: HSSRELOD processing terminates with errors.

User Response: This sequence of actions is required in order to reload a new copy of the Validation Routine via the HSSRELOD utility:

1. Issue the DISABLE operator command to stop Validation Routine processing.
2. Issue the UNINSTALL operator command to remove the Validation Routine as a RACF exit.
3. Execute the HSSRELOD utility to replace the in-memory copy of the Validation Routine.
4. Issue the INSTALL operator command to insert the Validation Routine as a RACF exit.
5. Issue the ENABLE operator command to start Validation Routine processing.

HSS0050E LPA DELETE FAILED (MACRO CSVDYLPA REQUEST=DELETE RC=<rc>, RS=<reason-code>)

Explanation: The HSSRELOD utility received a bad return code from the CSVDYLPA macro when attempting to delete the Validation Routine from LPA. <rc> is the CSVDYLPA return code, <reason-code> is the reason code.

System Action: Processing fails. The HSSRELOD utility terminates with errors.

User Response: Refer to the MVS Authorized Assembly Services Reference (Volume 1) manual for an explanation of the CSVDYLPA return codes. Take corrective action where appropriate. If the problem persists or is indeterminate, contact your IBM representative for help.

Appendix A. Suggestions for Application Definitions

Preparing application definitions requires some knowledge of the SCLM program environment. To simplify the definition process, suggested definitions have been prepared for you. Instructions to load these into your Rule File are contained in "HSSRDEFN - Rule File definition JCL" on page 57.

Listed below are the suggested Application definitions:

Application	Function	High Program	Low Program	Comment
SCLM	Browse	ISRSCLM	FLMEB$	Via online TSO/ISPF
SCLM	Build	FLMCMD	FLMB	Via SCLM command interface
		FLMS$SRV	FLMB	Via SCLM subroutine interface
		ISRSCLM	FLMB	Via online TSO/ISPF
SCLM	Edit	CLZRSDRV	CLZRSDRV	Cloud9 edit save processing
		FLMCMD	FLMCMD	SCLM Lock/Unlock processing
		FLMCMD	FLMCXUDI	SCLM PDS directory updating
		FLMCMD	FLME$CRT	EDIT command: SCREATE command
		FLMCMD	FLME$END	EDIT command: END processing
		FLMCMD	FLME$IM	EDIT command: Initial Macro processing
		FLMCMD	FLME$SAV	EDIT command: SAVE command
		FLMCMD	FLME$SMO	EDIT command: SMOVE command
		FLMCMD	FLME$SRE	EDIT command: SREPLACE command
		FLMS$SRV	FLME$CRT	EDIT command: SCREATE command
		FLMS$SRV	FLME$END	EDIT command: END processing
		FLMS$SRV	FLME$IM	EDIT command: Initial Macro processing
		FLMS$SRV	FLME$SAV	EDIT command: SAVE command
		FLMS$SRV	FLME$SMO	EDIT command: SMOVE command
		FLMS$SRV	FLME$SRE	EDIT command: SREPLACE command
		FLMS$SRV	FLME$SRV	SCLM Lock/Unlock processing
		ISRSCLM	FLME$CRT	online editing: SCREATE command
		ISRSCLM	FLME$END	online editing: END processing
		ISRSCLM	FLME$IM	online editing: Initial Macro processing
		ISRSCLM	FLME$SAV	online editing: SAVE command
		ISRSRSLM	FLME$SMO	online editing: SMOVE command
		ISRSCLM	FLME$SRE	online editing: SREPLACE command
SCLM	Export	FLMCMD	FLMXE	Via SCLM command interface
		FLMS$SRV	FLMXE	Via SCLM subroutine interface
		ISRSCLM	FLMXE	Via online TSO/ISPF
SCLM	Import	FLMCMD	FLMXI	Via SCLM command interface
		FLMS$SRV	FLMXI	Via SCLM subroutine interface
		ISRSCLM	FLMXI	Via online TSO/ISPF
SCLM	Migrate	FLMCMD	FLMUM	Via SCLM command interface
		FLMS$SRV	FLMUM	Via SCLM subroutine interface
		ISRSCLM	FLMUM	Via online TSO/ISPF

Suggestions for Application Definitions

Application	Function	High Program	Low Program	Comment
SCLM	Promote	FLMCMD	FLMP	Via SCLM command interface
		FLMS$SRV	FLMP	Via SCLM subroutine interface
		ISRSCLM	FLMP	Via online TSO/ISPF
SCLM	Rptarch	FLMCMD	FLMRA	Via SCLM command interface
		FLMS$SRV	FLMRA	Via SCLM subroutine interface
		ISRSCLM	FLMRA	Via online TSO/ISPF
SCLM	Sample	FLMCMD	FLMDDL	Via SCLM command interface
		FLMS$SRV	FLMDDL	Via SCLM subroutine interface
		ISRSCLM	FLMDDL	Via online TSO/ISPF
SCLM	Utility	FLMCMD	FLMUDU$	Via SCLM command interface
		FLMS$SRV	FLMUDU$	Via SCLM subroutine interface
		ISRSCLM	FLMUDU$	Via online TSO/ISPF

Appendix B. Summary of SCLM Services and High-Low Programs

SCLM Command Level Interface via FLMCMD

The SCLM Command Interface is invoked by using the FLMCMD command via a CLIST, a REXX procedure, or as a TSO command. The chart below summarizes the FLMCMD services, along with the recommended High and Low Program values.

FLMCMD Service	Required Access Privilege	High Program	Low Program	Suggested Application/Function
ACCTINFO	READ	FLMCMD	FLMCMD	*Note 1
AUTHCODE	READ	FLMCMD	FLMCMD	*Note 1
BUILD	UPDATE	FLMCMD	FLMB	SCLM Build
DBUTIL	READ	FLMCMD	FLMCMD	*Note 1
DELETE	UPDATE	FLMCMD	FLMUDU$	SCLM Utility
DELGROUP	UPDATE	FLMCMD	FLMUDU$	SCLM Utility
DSALLOC	NONE	FLMCMD	FLMCMD	*Note 1
EDIT	UPDATE	FLMCMD	*Note 2	SCLM Edit
EXPORT	READ	FLMCMD	FLMXE	SCLM Export
IMPORT	UPDATE	FLMCMD	FLMXI	SCLM Import
LOCK	UPDATE	FLMCMD	FLMCMD	*Note 1
MIGRATE	UPDATE	FLMCMD	FLMUM	SCLM Migrate
NEXTGRP	READ	FLMCMD	FLMCMD	*Note 1
PROMOTE	UPDATE	FLMCMD	FLMP	SCLM Promote
RPTARCH	READ	FLMCMD	FLMRA	SCLM Rptarch
SAVE	UPDATE	FLMCMD	FLMCMD	*Note 1
UNLOCK	UPDATE	FLMCMD	FLMCMD	*Note 1
VERDEL	UPDATE	FLMCMD	FLMCMD	*Note 1

Notes:

1. The High-Low Program combination of FLMCMD-FLMCMD controls multiple SCLM services. These include: ACCTINFO, AUTHCODE, DBUTIL, DSALLOC, LOCK, NEXTGRP, SAVE, UNLOCK, and VERDEL.

   Use of RACF access rules is recommended in cases where READ access is required, as this will simplify rule writing via Enhanced Access Control for SCLM.

   The High-Low Program combination of FLMCMD-FLMCMD appears in the Application SCLM Edit. Use this Application for UPDATE access to data sets that the user is permitted to edit.

   If UPDATE access is required via the FLMCMD-FLMCMD High-Low Program for a data set that the user is not permitted to edit, then you will need to create a new Application definition and grant data set access using that Application.
2. The EDIT service uses multiple Low Program definitions for varying edit functions. These Low Programs are described in "SCLM Edit Services" on page 115.

SCLM Command Level Interface via FLMLNK

The SCLM Subroutine Interface is invoked by issuing a program call to FLMLNK. The chart below summarizes the available FLMLNK services, along with the recommended High and Low Program values.

FLMLNK Service	Required Access Privilege	High Program	Low Program	Suggested Application/Function
ACCTINFO	READ	FLMS$SRV	FLMS$SRV	*Note 1
AUTHCODE	READ	FLMS$SRV	FLMS$SRV	*Note 1
BUILD	UPDATE	FLMS$SRV	FLMB	SCLM Build
DBACCT	READ	FLMS$SRV	FLMS$SRV	*Note 1
DELETE	UPDATE	FLMS$SRV	FLMUDU$	SCLM Utility
DELGROUP	UPDATE	FLMS$SRV	FLMUDU$	SCLM Utility
DSALLOC	None	FLMS$SRV	FLMS$SRV	*Note 1
EDIT	UPDATE	FLMS$SRV	*Note 2	SCLM Edit
END	NONE	FLMS$SRV	FLMS$SRV	*Note 1
EXPORT	READ	FLMS$SRV	FLMXE	SCLM Export
FREE	NONE	FLMS$SRV	FLMS$SRV	*Note 1
IMPORT	UPDATE	FLMS$SRV	FLMXI	SCLM Import
INIT	NONE	FLMS$SRV	FLMS$SRV	*Note 1
LOCK	UPDATE	FLMS$SRV	FLMS$SRV	*Note 1
MIGRATE	UPDATE	FLMS$SRV	FLMUM	SCLM Migrate
NEXTGRP	READ	FLMS$SRV	FLMS$SRV	*Note 1
PARSE	READ	FLMS$SRV	FLMS$SRV	*Note 1
PROMOTE	UPDATE	FLMS$SRV	FLMP	SCLM Promote
SAVE	UPDATE	FLMS$SRV	FLMS$SRV	*Note 1
START	READ	FLMS$SRV	FLMS$SRV	*Note 1
STORE	UPDATE	FLMS$SRV	FLMS$SRV	*Note 1
UNLOCK	UPDATE	FLMS$SRV	FLMS$SRV	*Note 1
VERDEL	UPDATE	FLMS$SRV	FLMS$SRV	*Note 1

Notes:

1. The High-Low Program combination of FLMS$SRV-FLMS$SRV controls multiple SCLM services. These include: ACCTINFO, AUTHCODE, DBACCT, DSALLOC, END, FREE, INIT, LOCK, NEXTGRP, PARSE, SAVE, START, STORE, UNLOCK, and VERDEL.

   Use of RACF access rules is recommended in cases where READ access is required, as this will simplify rule writing via Enhanced Access Control for SCLM.

   The High-Low Program combination of FLMS$SRV-FLMS$SRV appears in the Application SCLM Edit. Use this Application for UPDATE access to data sets that the user is permitted to edit.

   If UPDATE access is required via the FLMS$SRV-FLMS$SRV High-Low Program for a data set that the user is not permitted to edit, then you will need to create a new Application definition and grant data set access using that Application.
2. The EDIT service uses multiple Low Program definitions for varying edit functions. These Low Programs are described in “SCLM Edit Services” on page 115.

SCLM ISPF Online Interface

The chart below summarizes the services available from the SCLM ISPF dialog using foreground execution:

SCLM Service	Required Access Privilege	High Program	Low Program	Suggested Application/Function
BUILD	UPDATE	ISRSCLM	FLMB	SCLM Build
EDIT	UPDATE	ISRSCLM	*Note 1	SCLM Edit
EXPORT	READ	ISRSCLM	FLMXE	SCLM Export
IMPORT	UPDATE	ISRSCLM	FLMXI	SCLM Import
MIGRATE	UPDATE	ISRSCLM	FLMUM	SCLM Migrate
PROMOTE	UPDATE	ISRSCLM	FLMP	SCLM Promote
RPTARCH	READ	ISRSCLM	FLMRA	SCLM Rptarch
SAMPLE	ALTER	ISRSCLM	FLMDDL	SCLM Sample
UTILITIES	UPDATE	ISRSCLM	FLMUDU$	SCLM Utility *Note 2
VIEW	READ	ISRSCLM	FLMEB$	SCLM Browse

Notes:

1. The EDIT service uses multiple Low Program definitions for varying edit functions. These Low Programs are described in "SCLM Edit Services".
2. The Low Program FLMUDU$ controls multiple SCLM services. However, some of these can be defined using their own unique Low Programs. These include: MIGRATE, EXPORT, IMPORT and RPTARCH.

SCLM Edit Services

SCLM editing uses multiple Low Program values. These are summarized in the chart below:

Low Program	Comment
FLME$END	The END command
FLME$IM	Initial-macro processing
FLME$SAV	The SAVE command
FLME$CRT	The SCREATE command
FLME$SMO	The SMOVE command
*Note 1	The SPROF command
FLME$SRE	The SREPLACE command

Notes:

1. The SPROF command does not require a unique Low Program value.

Breeze Interface to SCLM Services

Breeze invokes SCLM services via the FLMCMD Command Level Interface. Refer to "SCLM Command Level Interface via FLMCMD" on page 113 to view these High-Low Program combinations.

Cloud9 Interface to SCLM Services

Cloud9 invokes SCLM services via the FLMCMD Command Level Interface. Refer to "SCLM Command Level Interface via FLMCMD" on page 113 to view these High-Low Program combinations.

Summary of SCLM Services and High-Low Programs

Cloud9 editing of SCLM-managed source members requires the following recommended High and Low Program values.

Cloud9 Service	Required Access Privilege	High Program	Low Program	Suggested Application/Function
EDIT	UPDATE	CLZRSDRV	CLZRSDRV	SCLM Edit *Note 1
	UPDATE	FLMCMD	FLMCMD	SCLM Edit *Note 1
	UPDATE	FLMCMD	FLMCXUDI	SCLM Edit *Note 1

Notes:

1. The Cloud9 High-Low programs serve the following purposes:

High Program	Low Program	Usage
CLZRSDRV	CLZRSDRV	Cloud9 edit-save function
FLMCMD	FLMCMD	SCLM LOCK/UNLOCK that precedes/follows the save
FLMCMD	FLMCXUDI	Other SCLM control processing

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM or Fundi Software may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA
95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED,

INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following are trademarks of International Business Machines Corporation in the United States, or other countries, or both:

- IBM
- MVS
- OS/390
- RACF
- SecureWay
- z/OS

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Index

L

M

O

P

R

S

T

U

V

Readers' Comments — We'd Like to Hear from You

Enhanced Access Control for SCLM for z/OS
User's Guide
Release 1

Publication No. SC27-1591-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC27-1591-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
H150/090
555 Bailey Avenue
San Jose, CA
U.S.A. 95141-9989

Fold and Tape | **Please do not staple** | Fold and Tape

SC27-1591-00

Cut or Fold
Along Line

Program Number: 5655-J64

Printed in U.S.A.

SC27-1591-00