z/OS
2.5

Security Server RACF Data Areas

GA32-0885-50

Note

Before using this information and the product it supports, read the information in "Notices" on page 503.

This edition applies to Version 2 Release 5 of z/OS® (5650-ZOS) and to all subsequent releases and modifications until otherwise indicated in new editions.

Last updated: 2023-06-28

Contents

Tables

About this document

This document supports z/OS (5650-ZOS) and contains information about Resource Access Control Facility (RACF®), which is part of z/OS Security Server.

Who should use this document

This document contains graphic presentations of data areas used by RACF.

This document is for programmers who need specific information about RACF data areas for their programs or for use in diagnosing, modifying, or tuning RACF. Users of this document should have a working knowledge of RACF functions.

How to use this document

The data areas are in alphanumeric sequence by data area acronym. The acronyms are derived by removing the first three characters of the full data area name and the next character too, if it is P. Each data area has up to three sections:

- Header
- Data area map
- Cross reference, if the data area map is long enough.

The Header

The header includes some or all of the following information:

Table 1. The header

Entry	Description
Common Name	The descriptive name of the data area.
Macro ID	The name of the mapping macro for the data area. Mapping macros can be issued in programs to generate a copy of the data area.
DSECT Name	The name of the DSECT (dummy control section) created by the mapping macro.
Owning Component	The component name and component identifier in parentheses.
Eye-Catcher ID	The character-string identifier of the eye catcher (sometimes called the control block ID) within the mapping macro. The offset and length of the eye catcher are also included.
Storage Attributes	The storage attributes of the data area, including the following. **Main Storage** The central storage attributes of the data area. **Virtual Storage** The virtual storage attributes of the data area. **Auxiliary Storage** The spool storage attributes of the data area. **Subpool and Key** The subpool is the area of virtual storage that contains the data area. The key is the storage protection key for the storage represented by the data area.

Table 1. The header (continued)

Entry	Description
Size	The size of the data area in decimal bytes
Created by	The module, macro, or component whose use creates the data area.
Pointed to by	The registers or data area fields that contain the address of the data area.
Serialization	The method used to ensure that one user does not update a data area that is being updated or used by another user. The most common methods used for serialization are: • Lock or locks • ENG or DEQ macros • Compare and Swap (CS) instruction.
Function	A brief description of the use of the data area.

Data area map

Each data area is described field by field. The field descriptions shown in this document are intended as the primary sources of such information. These field descriptions may differ from those found in the programming code.

Here is an example of the field descriptions for the ASCB data area:

Table 2. Example of ASCB data area

Dec	Hex	Type	Len	Name (Dim)	Description
0	(0)	STRUCTURE	384	ASCB	
0	(0)	CHARACTER		ASCBEGIN	Beginning of ASCB
0	(0)	CHARACTER	4	ASCBASCB	Acronym in EBCDIC -ASCB-
4	(4)	ADDRESS	4	ASCBFWDP	Address of next ASCB on ASCB ready queue

For each field in the data area, the data area map provides the following information:

Table 3. Example of the data area map

Type	Description
Offsets	The address of the field, shown in both decimal (DEC) and hexadecimal (HEX in parentheses), relative to the beginning of the data area.

Table 3. Example of the data area map (continued)	
Type	**Description**
Type	The kind of program data defined for this field, as follows: Type Description A-ADDRESS A-type address constant BAL STMT Instruction BITSTRING Bitstring constant CHARACTER Character value DECIMAL Decimal value FIXED Arithmetic signed or unsigned value FLOATING Floating-point binary value HEX Hexadecimal value OFFSET Q-type address constant PACKED Packed decimal value S-ADDRESS S-type address constant SIGNED Arithmetic signed value STRUCTURE Level 1 control block name UNSIGNED Unsigned value V-ADDRESS V-type address constant Y-ADDRESS Y-type address constant ZONED Zoned decimal value
Len	The size of the field in decimal bytes.
Name (Dim)	The name of the field, bit, or mask. Bit or mask names are preceded by a description of bit position and value, as follows: 1. . . . . . . Refers to bit 0. . . . . . . 11 Refers to bits 6 and 7. . . . 1 . . . . Refers to bit 3. 11. . 1111 Refers to bits 0, 1, 4, 5, 6, and 7.
Description	A description of the purpose or meaning of the field, bit, or mask.

Cross Reference

For each data area with more than 25 fields, cross reference shows the following information:

Table 4. Example of a cross reference	
Type	**Description**
Name	The name of the field, bit, or mask.
Hex Offset	The hexadecimal offset of the field into the data area. For bits, the hexadecimal offset of the field containing the bit.
Hex Value	Hexadecimal values are shown only for bits. The hexadecimal value shown implies the position of the bit in the field containing the bit.
Level	Level of the PL/AS declaration for that field.

Bit TCBACTIV in the TCB data area illustrates how to use the hexadecimal value. In the TCB data area, cross reference for the TCBACTIV bit looks like this:

Table 5. Example of TCBACTIV bit

Name	Hex Of f set	Hex Value	Level
TCBACTI V	F0	80	2

In the data area map of the TCB, the TCBACTIV bit appears like this:

```
240   (F0)   FI XED       4   TCBXSCT   Dispatcher intersect control word
240   (F0)   BI TSTRI NG   1   TCBXSCT1   Flagbyte (MDC323)
```

X' F0' is the offset of field TCBXSCT into the TCB. TCBXSCT is a four-byte field, which contains a one-byte field named TCBXSCT1. Both TCBXSCT and TCBXSCT1 have the same offset. The first bit in both fields is named TCBACTIV. Ignoring the other bits in the field TCBXSCT1, if the TCBACTIV bit is on, the value of field TCBXSCT1 would be 1000 0000, which is equivalent to X' 80' . This value (X' 80') is shown both in the description in the data area map and in the column of the cross reference.

Where to find more information

When possible, this information uses cross-document links that go directly to the topic in reference using shortened versions of the document title. For complete titles and order numbers of the documents for all products that are part of z/OS, see *z/OS Information Roadmap*.

To find the complete z/OS library, including the z/OS Documentation, see the z/OS Internet library (www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosInternetLibrary).

To find educational material, see the IBM Education home page (www.ibm.com/training).

RACF courses

The following RACF classroom courses are available in the United States:

ES191
Basics of z/OS RACF Administration

BE870
Effective RACF Administration

ES885
Exploiting the Advanced Features of RACF

IBM® provides various educational offerings for RACF. For more information about classroom courses and other offerings, do any of the following:

- See your IBM representative
- Call 1-800-IBM-TEACH (1-800-426-8322)

Other sources of information

IBM provides customer-accessible discussion areas where RACF may be discussed by customer and IBM participants. Other information is also available through the Internet.

Internet sources

The following resources are available through the Internet to provide additional information about the RACF library and other security-related topics:

- z/OS Internet library (www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zosInternetLibrary)
- IBM Redbooks (www.ibm.com/redbooks)
- Enterprise security (www.ibm.com/systems/z/solutions/enterprise-security.html)
- RACF home page (www.ibm.com/products/resource-access-control-facility/resources)

- RACF download page (github.com/IBM/IBM-Z-zOS/tree/master/zOS-RACF/Downloads)

How to send your comments to IBM

We invite you to submit comments about the z/OS product documentation. Your valuable feedback helps to ensure accurate and high-quality information.

Important: If your comment regards a technical question or problem, see instead "If you have a technical problem" on page xxvii.

Submit your feedback by using the appropriate method for your type of comment or question:

Feedback on z/OS function

If your comment or question is about z/OS itself, submit a request through the IBM RFE Community (www.ibm.com/developerworks/rfe/).

Feedback on IBM Documentation

If your comment or question is about the IBM Documentation functionality, for example search capabilities or how to arrange the browser view, send a detailed email to IBM Documentation at ibmdoc@us.ibm.com.

Feedback on the z/OS product documentation and content

If your comment is about the information that is provided in the z/OS product documentation library, send a detailed email to mhvrcfs@us.ibm.com. We welcome any feedback that you have, including comments on the clarity, accuracy, or completeness of the information.

To help us better process your submission, include the following information:

- Your name, company/university/institution name, and email address
- The following deliverable title and order number: z/OS Security Server RACF Data Areas, GA32-0885-50
- The section title of the specific information to which your comment relates
- The text of your comment.

When you send comments to IBM, you grant IBM a nonexclusive authority to use or distribute the comments in any way appropriate without incurring any obligation to you.

IBM or any other organizations use the personal information that you supply to contact you only about the issues that you submit.

If you have a technical problem

If you have a technical problem or question, do not use the feedback methods that are provided for sending documentation comments. Instead, take one or more of the following actions:

- Go to the IBM Support Portal (support.ibm.com).
- Contact your IBM service representative.
- Call IBM technical support.

Summary of changes

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line to the left of the change.

Note: IBM z/OS policy for the integration of service information into the z/OS product documentation library is documented on the z/OS Internet Library under IBM z/OS Product Documentation Update Policy (www-01.ibm.com/servers/resourcelink/svc00100.nsf/pages/ibm-zos-doc-update-policy?OpenDocument).

Summary of changes for z/OS Version 2 Release 5 (V2R5)

The following changes are made to z/OS Version 2 Release 5 (V2R5).

New

The following information is new.

June 2023

Content has been updated in support of RACF APAR OA63462 and SAF APAR OA63643. In the PTFs for these new APARs, the RACF support for Identity Tokens is enhanced to add support for several new use cases.

- **Generation of IDTs From an Existing ACEE Security Environment:** The new support introduces the capability for applications to use the initACEE callable service to generate an IDT from an ACEE. The returned IDT can be used to authenticate the user using the existing IDT support in RACROUTE REQEST=VERIFY. The new initACEE function can generate an IDT for a protected user without traditional authentication mechanisms.
- **Authentication of a User with IDT Generated from an ACEE:** The new support introduces the capability for RACROUTE REQEST=VERIFY to authenticate a user with an IDT which was generated from an ACEE. With this new support, a user with the protected attribute can optionally be authenticated with an IDT.

In support of RACF APAR OA63462 and SAF APAR OA63643, the following topics in this document are updated:

- Chapter 12, “COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services,” on page 53.
- Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323.

Updates were made in this book and 5 others:

- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS Security Server RACF Command Language Reference*
- *z/OS Security Server RACROUTE Macro Reference*
- *z/OS Security Server RACF Callable Services*
- *z/OS Security Server RACF Macros and Interfaces*

May 2023

Validated Boot for z/OS (APAR OA63507) is a solution that uses digital signatures to provide an initial program load (IPL)-time check that validates that IPL data is intact, not tampered with, and originated from a trusted source. It also enables detection of unauthorized changes to software executables.

RACF support for Validated Boot for z/OS is provided in the PTFs for APARs OA61878 and OA61901.

In support of Validated Boot for z/OS, the following topics in this document are updated:

- The field **PGSN_FUNC_ATTRS@** is added to the program signing (PGSN) section of Chapter 12, “COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services,” on page 53.
- The field **PGSN64_FUNC_ATTR@** is added to the program signing (PGSN64) section of Chapter 14, “COMY: 64-bit enabled SAF callable services,” on page 171
- The bit flag **RCVTVABT** is added to the function availability field, RCVTFLG4, in the Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323.

February 2023

- Information about flag bit FASTLOGX is added to Chapter 25, “FAST: RACROUTE REQUEST=FASTAUTH Parameter List (Request Section),” on page 243. This bit indicates the format of the log stream (LOGSTR) data, which is found at the LOGSTR pointer. This information can be used by requesters of the RACROUTE REQUEST=FASTAUTH service.
- Information about flag bit RFXLOGX is added to Chapter 46, “RFXP: RACROUTE REQUEST=FASTAUTH Exit Parameter List,” on page 389. This bit indicates the format of the log stream (LOGSTR) data, which is found at the LOGSTR pointer. This information can be used by exit routines that are invoked for RACROUTE REQUEST=FASTAUTH requests.

June 2022

- RACF Password Phrase Interval Support provides a new separate password phrase-specific change interval, which can be different than the existing password interval and supports much longer values. The password phrase interval can be set with the new PHRASEINT keyword at the system level with the SETROPTS command and at the user level with the PASSWORD or PHRASE commands. See Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323 and Chapter 48, “RIXP: RACROUTE REQUEST=VERIFY/VERIFYX Exit Parameter List,” on page 405. (APAR OA61951)

V2R5

- The following new constant is added to Chapter 58, “SAFP: SAF Router Parameter List,” on page 473:
  - SAFPRLD0
- The following new information is added to Chapter 19, “DSDT: Data Set Descriptor Table,” on page 209:
  - DSDEVOL
  - DSDEDBObj
  - DSDEFlags
  - DSDEVSAM

Changed

The following information is changed.

V2R5

- The value of RCVTVRMC is changed from 77C0 to 77D0. See Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323.

Deleted

V2R5

No information is deleted in this release.

Summary of changes for z/OS Version 2 Release 4 (V2R4)

The following changes are made to z/OS Version 2 Release 4 (V2R4).

New

The following information is new.

January 2021 refresh

- RACF has added support for a new enhanced PassTicket algorithm option. The original PassTicket algorithm is now referred to as legacy PassTickets.
- A new field has been added to indicate that the enhanced PassTicket Functions are available. See Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323.

October 2020 refresh

- Description was updated for TOKENCR in the Structure TOKEN table. See Chapter 55, “RUTKN: Resource/User Security Token,” on page 457.

June 2020 refresh

- There is no new content for the June 2020 refresh. This publication is being refreshed to correct formatting errors.

Prior to June 2020 refresh

- The following flag has been added to Chapter 1, “ACEE: Accessor Environment Element,” on page 1:
  - ACEESBVR
- The following new constants have been added to Chapter 12, “COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services,” on page 53:
  - ADMN_XTR_DATASET
  - ADMN_XTR_NEXT_DATASET
- The following new flags have been added to Chapter 23, “EVXP: RACF Command Exit Parameter List,” on page 231:
  - EVXOPARM
  - EVXSPEC
  - EVXAUDT
- The following new flag has been added to Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323:
  - RCVTIDT
- The following new constant has been added to Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323:
  - RCVTVRC0
- The following new flags have been added to Chapter 47, “RIPL: RACROUTE REQUEST=TOKENBLD/VERIFY/VERIFYX Parameter List (Request Section),” on page 395:
  - INITPLV0001
  - INITIDTA
  - INITENDPLV0001
- The following new flag has been added to Chapter 48, “RIXP: RACROUTE REQUEST=VERIFY/VERIFYX Exit Parameter List,” on page 405:
  - RIXIDTA
- The following new flag has been added to Chapter 53, “RRPF: Resident Profile Map,” on page 445:

  - DSPFLAGS
- The following new constant has been added to Chapter 58, “SAFP: SAF Router Parameter List,” on page 473:
  - SAFPRLC0

Changed

Prior to June 2020 refresh

- The value of RCVTVRMC has been changed from 77B0 to 77C0. See Chapter 42, “RCVT: RACF Communication Vector Table,” on page 323.
- The value of SAFPCURR has been changed from 24 to 25. See Chapter 58, “SAFP: SAF Router Parameter List,” on page 473

Deleted

No content has been deleted in this edition.

Summary of changes for z/OS Version 2 Release 3 (V2R3)

The following changes are made to z/OS Version 2 Release 3 (V2R3).

New

- ICHPISP (ISP)
  - New flag, RPEFSCPR is added
- ICHPRCVT (RCVT)
  - New flag, RCVTMAIL is added
  - New flags, RCVTFLG4, RCVTMFA3, and RCVTRPFF are added.
- ICHRIXP (RIXP)
  - New flags, RIXFLAG3 and RACPNMFA, are added.
- IRRPCOMP (COMP)
  - New UMAP function code constants, UMAP_R_TO_E and UMAP_E_TO_R, are added.
  - New fields are added for SMOP.
- IRRPCOMY (COMY)
  - New fields are added for SMOP64.
- IRRPRIPL (RIPL)
  - New flag, INITNMFA, is added.
- IRRPRXTW (RXTW)
  - New WORKATTR is added for E-mail address.

Chapter 1. ACEE: Accessor Environment Element

ACEE NOT programming interface information

The following fields are Not Programming Interface information:

- ACEEAMP
- ACEEMDLS
- ACEECGRP
- ACEECLCP
- ACEEGATA
- ACEEPADS
- ACEEOCOX
- ACEEPTDS
- ACEESBVR

ACEE heading information

Common name: Accessor Environment Element (ACEE)

Macro ID: IHAACEE

DSECT name: ACEE

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: ACEE (Offset: 0, Length: 4)

Storage attributes:

Subpool
255 (or as specified by the issuer of RACROUTE REQUEST=VERIFY)

Key
0

Residency
May reside above 16M

Size: 192 bytes (does not include any data pointed to by ACEE)

Created by: RACF or MVS™'s system authorization facility (SAF), depending on the parameters specified on RACROUTE REQUEST=VERIFY

Pointed to by: A field supplied by the issuer of RACROUTE REQUEST=VERIFY. Or, for MVS only: ASXBSENV or TCBSENV. ACEEs pointed to by ASXBSENV or TCBSENV always reside below 16M.

Serialization: See the notes that follow Function.

Function: Maps the ACEE; represents the authorities of a single accessor in the address space.

Notes:

1. When the ACEECHK class is active, a program that updates an ACEE or user token field affecting the user's authorization may cause IRR421I messages to be issued by the security product when the ACEE is checked to determine authorization. IBM recommends that programs do not directly modify authorization-related fields in the ACEE, but instead use interfaces provided by the security product

to create an ACEE with the required security attributes. In cases where this is not possible, consider documenting that your program should be added to the exception list in the ACEECHK class.

2. If you use ACEEIEP, it must point to an area of storage you obtained using a GETMAIN from within the RACINIT post-processing exit (any data obtained by the pre-processing exit will be ignored, and not freed, when an ACEE is retrieved from the VLF cache). RACF frees this area when it frees the ACEE. For RACF to do this, the first word of the area must contain the subpool and the length of the area. The subpool appears in the high-order byte, and the length appears in the next 3 bytes.

   If you do not conform to this requirement in your use of ACEEIEP, you must supply a RACINIT pre-processing exit to free the area and set the ACEEIEP field to 0 when a caller issues a RACINIT DELETE. In certain situations, however, your exit is not called during RACF error recovery, and unpredictable results may occur. Therefore, it is strongly recommended that you adhere to the specified requirements.

   Examples of nonconforming use of ACEEIEP follow:

   a. ACEEIEP contains data, rather than a pointer.
   b. ACEEIEP contains a pointer, however the first word of the area pointed to by ACEEIEP does not contain the subpool and length information for the area.
   c. ACEEIEP contains a pointer, and the first word of the area pointed to contains the subpool and length information for a data area that points to additional area obtained using GETMAIN.

      This situation might not cause an abend, but it results in a failure to free the acquired data area.

   If your use of ACEEIEP does not conform to the specified requirements, or if your data area contains any pointers to other data areas, you must provide an ACEE compression/expansion exit. See *z/OS Security Server RACF System Programmer's Guide* for more information.

   The area that ACEEIEP points to is retrieved with the ACEE. Before reusing ACEEIEP, installation code must process any existing area that ACEEIEP points to. A pointer to storage may be lost if installation code stores over ACEEIEP.

   When reusing ACEEIEP, the storage for the new data that ACEEIEP points to should be in the same subpool as the ACEE. The ACEESP field of the ACEE contains the subpool of the ACEE. For more information about subpool use, see the *z/OS MVS Programming: Assembler Services Guide*.

3. Within an IMS address space, ACEEAPTR is reserved for use by IMS during IMS initialization and signon.
4. Both ACEETRLV and ACEETRDA are 0 if one of the following conditions is met:

   - The NODES class is active and a NODES profile of the form *submitnode***.RUSER.***userid* exists with a UACC of at least UPDATE.
   - The POE's class is not active.
   - Neither TERMID nor POE was specified.
   - There is no matching profile.

   If the level is not specified in the profile, ACEETRLV is 0 even when none of the conditions are met. Similarly, if the DATA is not specified in the profile, ACEETRDA is 0 even when none of the conditions are met.
5. Both ACEEAPLV and ACEEAPDA are 0 if one of the following conditions is met:

   - The NODES class is active and a NODES profile of the form *submitnode***.RUSER.***userid* exists with a UACC of at least UPDATE.
   - The APPL class is not active.
   - APPL was not specified on the RACROUTE REQUEST=VERIFY.
   - No matching profile exists.

   If the level is not specified in the APPL profile, ACEEAPLV is 0 even when none of the conditions are met. Similarly, if the DATA is not specified in the profile, ACEEAPDA is 0 even when none of the conditions are met.

6. The acronym at offset 0 is changed from "ACEE" to "acee" prior to freeing the ACEE storage.
7. If you use ACEE3PTY, you must:
   - Do not use ACEE3PTY ACEE as an address-space ACEE (ASXBSENV) or task ACEE (TCBSENV).
   - Make sure that the ACEE3PTY ACEE is not deleted while it is being used by RACF.
   - The ACEE3PTY must be set to zero, once its value is extracted and before the ACEE which points to the third-party ACEE is deleted and before another 3rd-party RACHECK using that ACEE is performed.
   - The deletion of the ACEE3PTY ACEE must be complete while the resource manager is in the proper key. The ACEE3PTY ACEE is obtained from the same subpool as the ACEE in which it is anchored.

ACEE mapping

Table 6. Structure ACEE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	192	ACEE	Accessor environment element
0	(0)	CHARACTER	4	ACEEACEE	Acronym in EBCDIC -ACEE-
4	(4)	SIGNED	4	ACEECORE	ACEE subpool and length
4	(4)	ADDRESS	1	ACEESP	ACEE subpool number
5	(5)	ADDRESS	3	ACEELEN	Length of ACEE
8	(8)	ADDRESS	1	ACEEVRSN	Version = 1.
9	(9)	CHARACTER	3	ACEESBVR	Reserved for use by security product
12	(C)	ADDRESS	4	ACEEIEP	Reserved for installation. See "Notes®" above for information on using the ACEEIEP field.
16	(10)	ADDRESS	4	ACEEINST	User data address: Points to a 1-byte length field followed by the installation data specified in the user profile. The length includes the 1-byte length field. The address is zero if (1) no valid user ID was provided or (2) no data was present in the profile.
20	(14)	CHARACTER	9	ACEEUSER(0)	User ID information
20	(14)	ADDRESS	1	ACEEUSRL	User ID length
21	(15)	CHARACTER	8	ACEEUSRI	Contains the valid RACF user ID unless (1) the user ID on the verify call was '*BYPASS*' for auditable work that bypasses authorization checking, or (2) no user ID was given so the field contains an '*'.
29	(1D)	CHARACTER	9	ACEEGRP(0)	Group name information
29	(1D)	ADDRESS	1	ACEEGRPL	Group name length
30	(1E)	CHARACTER	8	ACEEGRPN	Valid connect group unless ACEEUSRI is "*" or "BYPASS". For these two cases, ACEEGRPN is "*'".

Table 6. Structure ACEE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
38	(26)	BITSTRING	1	ACEEFLG1	User flags
		1...		ACEESPEC	1 - Special attribute
		.1..		ACEEADSP	1 - Automatic data security protection
		..1.		ACEEOPER	1 - Operations attribute
		...1		ACEEAUDT	1 - Auditor attribute
		.... 1...		ACEELOGU	1 - User is to have most RACF functions logged
		.... .1..		ACEEROA	1 - Read-only auditor attribute
		.... ..1.		ACEEPRIV	1 - User is a started procedure with the privileged attribute
		1		ACEERACF	1 - RACF-defined user
39	(27)	BITSTRING	1	ACEEFLG2	Default universal access
		1...		ACEEALTR	1 - Alter authority to resource
		.1..		ACEECNTL	1 - Control authority to resource
		..1.		ACEEUPDT	1 - Update authority to resource
		...1		ACEEREAD	1 - Read authority to resource
		.... 1...		*	Reserved for compatibility
		.... .1..		*	Reserved
		.... ..1.		*	Reserved
		1		ACEENONE	1 - No authority to resource
40	(28)	BITSTRING	1	ACEEFLG3	Miscellaneous flags
		1...		ACEEGRPA	Access list of group DS to contain 0 - User ID or 1 - Group name and user ID
		.1..		ACEERASP	1 - RACF address space
		..1.		ACEECLNT	1 - Unauthenticated client
		...1		ACEEACLT	1 - Authenticated client
		.... 1...		ACEETSKP	1 - Task level process
		.... .1..		ACEEIUSP	1 - INITUSP has been done
		.... ..1.		ACEEDUID	1 - Default UID being used
		1		ACEENPWR	1 - This is a protected user ID that cannot enter the system with a password
41	(29)	CHARACTER	3	ACEEDATE	Date of RACINIT
44	(2C)	CHARACTER	8	ACEEPROC	Name of started procedure or blanks if not started procedure

Table 6. Structure ACEE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
52	(34)	ADDRESS	4	ACEETRMP	Address that points to the terminal ID. The field is zero for non-terminal users.
56	(38)	BITSTRING	2	ACEEFLG4	Miscellaneous flags 2
		1...		*	Reserved
		.1..		*	Reserved
		..1.		ACEEUATH	1 - User is authorized to define other users
		...1		*	Reserved
		.... 1...		ACEEDASD	1 - User is authorized to protect DASD volumes
		.... .1..		ACEETAPE	.... .1.. ACEETAPE 1 - User is authorized to protect tape volumes
		.... ..1.		ACEETERM	1 - User is authorized to protect terminals
56	(38)	BITSTRING	1	*	Reserved.
58	(3A)	ADDRESS	1	ACEEAPLV	Application level: Contains the level value from the application profile.
59	(3B)	ADDRESS	1	ACEETRLV	POE level: Contains the level value from the general resource profile that protects the port of entry.
60	(3C)	ADDRESS	4	ACEETRDA	POE data address: Points to a 1-byte length field followed by the installation data from the profile that protects the port of entry. The length includes the 1-byte length field.
64	(40)	CHARACTER	8	ACEETRID	An 8-byte area containing the terminal ID. The name is left-aligned and padded on the right with blanks. This field is blank when (1) termid is not specified and (2) either the POE is not specified or the POE class is not terminal.
72	(48)	ADDRESS	4	ACEEAMP	Address first anchored model.
76	(4C)	BITSTRING	4	ACEECLTH	User class authorizations - these bit positions are mapped by the class descriptor entries anchored off the RACF CVT.
80	(50)	ADDRESS	4	ACEECLCP	Anchor for in-storage profile trees built by the RACLIST function.

Table 6. Structure ACEE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
84	(54)	ADDRESS	4	ACEEAPTR	Address field reserved for application usage
88	(58)	CHARACTER	8	ACEEAPLN	Name of application to which user is connected, or blanks if no application specified
96	(60)	ADDRESS	4	ACEEAPDA	Application data address: Points to a 1-byte length field followed by the data from the application profile. The length includes the 1-byte length field.
100	(64)	ADDRESS	4	ACEEUNAM	Address of user name string. The first byte is a length field followed by the name string. The length includes the 1-byte length field.
104	(68)	ADDRESS	4	ACEEMDLS	Address of the data set model name array. If array not obtained by RACINIT or RACROUTE
108	(6C)	ADDRESS	4	ACEECGRP	Address of connect group table.
112	(70)	ADDRESS	4	ACEEGATA	Address of the generic anchor table
116	(74)	ADDRESS	4	ACEEFCGP	Address of table containing the list of groups this user ID is a member of. Built by RACINIT and used by FRACHECK, it is not automatically refreshed.
120	(78)	ADDRESS	4	ACEEDSLP	Address of the list of categories to which this user is allowed access
124	(7C)	CHARACTER	4	ACEEDAT4	4-byte date field formatted ccyydddF where cc is 00 for years 1971-1999 or 01 for years 2000-2070.
128	(80)	ADDRESS	4	ACEEPADS	Address of the list of data sets accessed by controlled programs executed by this user.
132	(84)	BITSTRING	1	ACEESLVL	Maximum security level accessible by this user
133	(85)	BITSTRING	1	ACEEFLG5	Miscellaneous flags
		1...		ACEEMODE	1 - ACEE mode is in 31-bit mode
		.1..		ACEEVMSK	0 - If ACEEPLCL is not zero, it points to a 128-bit mask1 - ACEEPLCL points to a 1024-bit mask
		..1.		ACEED4OK	1 - ACEEDAT4 contains data 0 - ACEEDAT4 not used

Table 6. Structure ACEE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		...1		ACEEXNVR	ENVR object created by another system
		.... 1...		ACEESTOK	1 - An ACEE was built from a TOKEN with a SERVAUTH port of entry, and the SERVAUTH resource name is no longer available.
		.... .1..		ACEENSTE	On if nested ENVR object (in ACEENSTA field) should be used in auth check
		.... ..1.		ACEEDALY	1 - User logged on to an application which only records daily logon statistics
134	(86)	CHARACTER	1	ACEEFLG6	More miscellaneous flags
		1...		ACEERAUI	Restricted access user ID
		.1..		ACEERUAA	"On" if the RESTRICTED user ID can gain UNIX file access by virtue of the OTHER bits (for example, the user ID has READ access to RESTRICTED.FILESYS.ACCESS in the UNIXPRIV class)
		...1		ACEERUAV	A check was made to RESTRICTED.FILESYS.ACCESS for this process so the value of ACEERUAA can be used
		.... 1...		ACEEMFAU	User must authenticate with MFA. On when the user has an active MFA factor and MFADEF class is active.
		.... .111		ACEEMFAA	User authenticated with MFA.
		..1.		*	Reserved
135	(87)	CHARACTER	1	*	Reserved
136	(88)	ADDRESS	4	ACEE3PTY	Address of ACEE created by third-party RACHECK SVC processing
140	(8C)	ADDRESS	4	ACEEPLCL	Pointer to extended class authorization mask, or 0
144	(90)	CHARACTER	8	ACEESUID	Surrogate user ID (AUDIT)
152	(98)	ADDRESS	4	ACEEOCOX	Pointer to O.C.O. extend
156	(9C)	ADDRESS	4	ACEEPTDS	Pointer to first TDS table

Table 6. Structure ACEE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
160	(A0)	ADDRESS	4	ACEEX5PR	Pointer to X500 name pair structure. Structure contains a 4-byte length of structure, followed by two 2-byte lengths, followed by up to 255 bytes of issuers name and up to 255 bytes of subjects name. Name pair storage is the same subpool and addressing mode as the ACEE.
164	(A4)	ADDRESS	4	ACEETOKP	Pointer to UTOKEN in external format
168	(A8)	ADDRESS	4	ACEESRVA	Address of an area containing a 1-byte length followed by the SERVAUTH resource name.
172	(AC)	ADDRESS	4	ACEESRVP	Address of an area containing a 1-byte length followed by the SERVAUTH profile name that granted access to the SERVAUTH resource.
176	(B0)	ADDRESS	4	ACEENSTA	Address of ENVR object representing the address space which created this ACEE
180	(B4)	ADDRESS	4	ACEEICTX	Address of the identity context extension.
184	(B8)	ADDRESS	4	ACEEIDID	Address of distributed identity data (IDID).
188	(BC)	CHARACTER	4	ACEETIME	ACEE creation time

ACEE constants

Table 7. Constants for ACEE

Len	Type	Value	Name	Description
1	DECIMAL	1	ACEEVR01	ACEE version number = 1.
1	DECIMAL	2	ACEEVR02	ACEE version number = 2.
1	DECIMAL	3	ACEEVR03	ACEE version number = 3.
1	DECIMAL	3	ACEECURV	ACEE version number = 3.

ACEE cross reference

Table 8. Cross Reference for ACEE

Name	Offset	Hex Value
ACEE	0	
ACEEACEE	0	
ACEEACLT	28	10
ACEEADSP	26	40
ACEEALTR	27	80
ACEEAMP	48	
ACEEAPDA	60	
ACEEAPLN	58	
ACEEAPLV	3A	
ACEEAPTR	54	
ACEEAUDT	26	10
ACEECGRP	6C	
ACEECLCP	50	
ACEECLNT	28	20
ACEECLTH	4C	
ACEECNTL	27	40
ACEECORE	4	
ACEEDALY	85	
ACEEDASD	38	08
ACEEDATE	29	
ACEEDAT4	7C	
ACEEDSLP	78	
ACEEDUID	28	02
ACEED4OK	85	20
ACEEDAT4	7C	
ACEEFCGP	74	
ACEEFLG1	26	
ACEEFLG2	27	
ACEEFLG3	28	
ACEEFLG4	38	
ACEEFLG5	85	
ACEEFLG6	86	
ACEEGATA	70	
ACEEGRP	1D	
ACEEGRPA	28	80

Table 8. Cross Reference for ACEE (continued)

Name	Offset	Hex Value
ACEEGRPL	1D	
ACEEGRPN	1E	
ACEEICTX	B4	
ACEEIDID	B8	
ACEEIEP	C	
ACEEINST	10	
ACEEIUSP	28	04
ACEELEN	5	
ACEELOGU	26	08
ACEEMDLS	68	
ACEEMFAA	86	08
ACEEMFAU	86	10
ACEEMODE	85	80
ACEENONE	27	01
ACEENPWR	28	01
ACEENSTA	B0	
ACEENSTE	85	04
ACEEOCOX	98	
ACEEOPER	26	20
ACEEPADS	80	
ACEEPLCL	8C	
ACEEPRIV	26	02
ACEEPROC	2C	
ACEEPTDS	9C	
ACEERACF	26	01
ACEERASP	28	40
ACEERAUI	86	80
ACEEREAD	27	10
ACEEROA	26	06
ACEERUAA	86	40
ACEERUAV	86	20
ACEESBVR	9	
ACEESLVL	84	
ACEESP	4	
ACEESPEC	26	80
ACEESRVA	A8	

Table 8. Cross Reference for ACEE (continued)

Name	Offset	Hex Value
ACEESRVP	AC	
ACEESTOK	85	08
ACEESUID	90	
ACEETAPE	38	04
ACEETERM	38	02
ACEETIME	BC	
ACEETOKP	A4	
ACEETRDA	3C	
ACEETRID	40	
ACEETRLV	3B	
ACEETRMP	34	
ACEETSKP	28	08
ACEEUATH	38	20
ACEEUNAM	64	
ACEEUPDT	27	20
ACEEUSER	14	
ACEEUSRI	15	
ACEEUSRL	14	
ACEEVMSK	85	40
ACEEVRSN	8	
ACEEXNVR	85	
ACEEX5PR	A0	
ACEE3PTY	88	

Chapter 2. ACHKL: RACROUTE REQUEST=AUTH Parameter List (Request Section)

ACHKL heading information

Common name: Request-specific portion of the RACROUTE REQUEST=AUTH parameter list

Macro ID: ICHACHKL

DSECT name: ACHKLIST

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Varies with the RELEASE= parameter specified

Created by: RACROUTE REQUEST=AUTH macro

Pointed to by: Address of SAFP plus the offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=AUTH routine

ACHKL mapping

Table 9. Structure ACHKLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	32	ACHKLIST	ACHKPTR RACHECK PARAMETER LIST MAP
0	(0)	ADDRESS	4	ACHKINSW(0)	ADDRESS INSTALLATION DATA
0	(0)	ADDRESS	1	ACHKLENG	LENGTH OF PARAMETER LIST
1	(1)	ADDRESS	3	ACHKINST	ADDRESS INSTALLATION DATA
4	(4)	SIGNED	4	ACHKENTW(0)	ENTITY ADDRESS WORD
4	(4)	BITSTRING	1	ACHKFLG1	FIRST FLAGS BYTE
		1...		ACHKRFI	RACFIND PARAMETER GIVEN
		.1..		ACHKRFIY	RACFIND=YES
		..1.		ACHKENX	ENTITYX IS SPECIFIED

Table 9. Structure ACHKLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		...1		ACHKDSTV	DSTYPE=V
		.... 1...		ACHK31IN	31-BIT ADDRESS LIST INDICATOR
		.... .1..		ACHKLOGF	LOG=NOFAIL
		.... ..1.		ACHKLOGN	LOG=NONE
		.... .11.		ACHKLOGS	LOG=NOSTAT (BOTH ON)
		1		ACHKCSA	ENTITY=(ADDR,CSA)
5	(5)	ADDRESS	3	ACHKENT	ENTITY NAME ADDRESS
8	(8)	SIGNED	4	ACHKCLNW(0)	CLASS NAME ADDRESS WORD
8	(8)	BITSTRING	1	ACHKFLG2	SECOND FLAGS BYTE
		1...		ACHKTALT	ATTR=ALTER
		.111		*	Reserved
		.... 1...		ACHKTCTL	ATTR=CONTROL
		.... .1..		ACHKTUPD	ATTR=UPDATE
		.... ..1.		ACHKTRD	ATTR=READ
		1		*	Reserved
9	(9)	ADDRESS	3	ACHKCLN	CLASS NAME ADDRESS
12	(C)	SIGNED	4	ACHKVOLW	VOLSER ADDRESS WORD
12	(C)	BITSTRING	1	ACHKFLG3	THIRD FLAGS BYTE
		1...		ACHKTAPE	DSTYPE=T
		.1..		ACHKMDEL	DSTYPE=M
		..1.		ACHKPRF	PROFILE ADDRESS GIVEN
		...1		*	Reserved
		.... 1...		ACHKVOL	VOLSER PARM SPECIFIED
		.... .1..		ACHKGEN	GENERIC=YES
		.... ..1.		ACHKPRI	PRIVATE=YES
		1		*	Reserved
13	(D)	ADDRESS	3	ACHKVOLS	VOLSER ADDRESS
16	(10)	ADDRESS	4	ACHKOVOL	OLD VOLSER ADDRESS
20	(14)	ADDRESS	4	ACHKAPPL	APPL NAME ADDRESS
24	(18)	ADDRESS	4	ACHKACEE	ACEE ADDRESS
28	(1C)	ADDRESS	4	ACHKOWNR	OWNER ADDRESS
32	(20)	CHARACTER		ACHKEND	END OF V1.4 LIST
32	(20)	CHARACTER	16	ACHK31	31-BIT-ADDRESS SAF EXTENSION

Table 9. Structure ACHKLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	4	ACHKIN31	31-BIT INSTALLATION DATA ADDRESS
36	(24)	ADDRESS	4	ACHKPR31	31-BIT PROFILE ADDRESS
36	(24)	ADDRESS	4	ACHKENTX	ENTITYX NAME ADDRESS
36	(24)	ADDRESS	4	ACHKEN31	ENTITY NAME/RESOURCE PROFILE ADDRESS
40	(28)	ADDRESS	4	ACHKCL31	CLASS NAME ADDRESS
44	(2C)	ADDRESS	4	ACHKVS31	VOLSER ADDRESS
48	(30)	CHARACTER		ACHK31EN	END OF SAF EXTENSION
48	(30)	CHARACTER	8	ACHK15	RACF 1.5 EXTENSION
48	(30)	ADDRESS	4	ACHKACC1	ACCLVL ADDRESS (1ST PART)
52	(34)	ADDRESS	4	ACHKACC2	ACCLVL ADDRESS (2ND PART)
52	(34)	X'20'	0	ACHKLEN	"ACHK31-ACHKLIST" LENGTH OF V1.4 PARAMETER LIST
52	(34)	X'30'	0	ACHK31LN	"ACHK15-ACHKLIST" LENGTH OF SAF PARAMETER LIST
52	(34)	X'38'	0	ACHKLN15	"*-ACHKLIST" - LENGTH OF V1.5 PARAMETER LIST
56	(38)	CHARACTER	4	ACHK17	RACF 1.7 EXTENSION
56	(38)	ADDRESS	2	ACHKFSEQ	FILESEQ NUMBER
58	(3A)	BITSTRING	1	ACHKFLGT	TAPE FLAG BYTE
		1...		ACHKBLP	TAPELBL SPECIFIED B'00'=STD B'10'=BLP B'01'=NL
		.1..		ACHKNL	TAPE LABEL SPECIFIED NL
		....		ACHKSTD	TAPE LABEL SPECIFIED STD
		..11 1111		*	Reserved
59	(3B)	BITSTRING	1	ACHKFLG4	FOURTH FLAG BYTE
		1...		ACHKEOS	STATUS=ERASE SPECIFIED
		.1..		ACHKEVD	STATUS=EVERDOM SPECIFIED
		..1.		ACHKWRON	STATUS=WRITEONLY SPECIFIED
		...1		ACHKACCS	STATUS=ACCESS SPECIFIED
		 1111		•	Reserved
60	(3C)	CHARACTER	8	ACHK18	RACF 1.8 EXTENSION

Table 9. Structure ACHKLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
60	(3C)	ADDRESS	4	ACHKUSID	USERID POINTER
64	(40)	ADDRESS	4	ACHKGPID	GROUP NAME POINTER
68	(44)	CHARACTER		AACHK18EN	END OF 1.8 EXTENSION
68	(44)	CHARACTER	4	ACHK18X	RACF 1.8X EXTENSION
68	(44)	ADDRESS	4	ACHKDDPR	DDNAME POINTER
68	(44)	CHARACTER		ACHK8XEN	END OF 1.8X EXTENSION
72	(48)	STRUCTURE	20	ACHK19	RACF 1.9 EXTENSION
72	(48)	ADDRESS	4	*	Reserved
76	(4C)	ADDRESS	4	ACHKUTOK	UTOKEN POINTER
80	(50)	ADDRESS	4	ACHKRTOK	RTOKEN POINTER
84	(54)	ADDRESS	4	ACHKLSTR	LOGSTR POINTER
88	(58)	ADDRESS	4	ACHKRCVR	RECVR POINTER
92	(5C)	CHARACTER		ACHK19EN	END OF 1.9 EXTENSION

ACHKL cross reference

Table 10. Cross Reference for ACHKL

Name	Offset	Hex Value
ACHKACCS	3B	10
ACHKACC1	30	
ACHKACC2	34	
ACHKACEE	18	
ACHKAPPL	14	
ACHKCLN	9	
ACHKCLNW	8	
ACHKCL31	28	
ACHKCSA	4	01
ACHKDDPR	44	
ACHKDSTV	4	10
ACHKEND	20	
ACHKENT	5	
ACHKENTW	4	
ACHKENTX	24	
ACHKENX	4	20
ACHKEN31	24	
ACHKEOS	3B	80

Table 10. Cross Reference for ACHKL (continued)

Name	Offset	Hex Value
ACHKEVD	3B	40
ACHKFLGT	3A	
ACHKFLG1	4	
ACHKFLG2	8	
ACHKFLG3	C	
ACHKFLG4	3B	
ACHKFSEQ	38	
ACHKGEN	C	4
ACHKGPID	40	
ACHKINST	1	
ACHKINSW	0	
ACHKIN31	20	
ACHKLENG	0	
ACHKLIST	0	
ACHKLOGF	4	04
ACHKLOGN	4	02
ACHKLOGS	4	06
ACHKLSTR	54	
ACHKMDEL	C	40
ACHKOVOL	10	
ACHKOWNR	1C	
ACHKPRF	C	20
ACHKPRI	C	2
ACHKRCVR	58	
ACHKRFI	4	80
ACHKRFIY	4	40
ACHKRTOK	50	
ACHKTALT	8	80
ACHKTAPE	C	80
ACHKTCTL	8	08
ACHKTLBL	3A	C0
ACHKTRD	8	02
ACHKTUPD	8	04
ACHKUSID	3C	
ACHKUTOK	4C	
ACHKVOL	C	08

Table 10. Cross Reference for ACHKL (continued)

Name	Offset	Hex Value
ACHKVOLS	D	
ACHKVOLW	C	
ACHKVS31	2C	
ACHKWRON	3B	20
ACHK15	30	
ACHK15EN	38	
ACHK17	38	
ACHK17EN	3C	
ACHK18	3C	
ACHK18EN	44	
ACHK18X	44	
ACHK19	48	
ACHK19EN	5C	
ACHK31	20	
ACHK31EN	30	
ACHK31IN	4	08
ACHK8XEN	48	

Chapter 3. ACXP: ACEE Expansion/Compression Exit Parameter List

ACXP heading information

Common name:	RACF ACEE expansion/compression exit parameter list mapping
Macro ID:	IRRACXP
DSECT name:	None
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	**Subpool** Determined by caller **Key** Determined by caller **Residency** Determined by caller
Size:	N/A
Created by:	N/A
Pointed to by:	N/A
Serialization:	None
Function:	Maps the parameter list passed to user exits IRRACX01 and IRRACX02

ACXP mapping

Table 11. Structure IRRACX01_PARMLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	IRRACX01_PARMLIST	
0	(0)	SIGNED	4	IRRACX01_OPERATION	Operation: stash or recall
		1		STASH_OPERATION	"X'00000001'" Stash: IRRACX01 may build and return range table
		.... ..1.		RECALL_OPERATION	"X'00000002'" Recall: Range table is provided by way of parameter IRRACX01_TABLE_PTR
4	(4)	ADDRESS	4	IRRACX01_ACEEPTR	Address of ACEE
8	(8)	ADDRESS	4	IRRACX01_TABLE_PTR	Address of range table

RANGE_TABLE

0	(0)	STRUCTURE	0	RANGE_TABLE	
0	(0)	SIGNED	4	RANGE_COUNT	Number of ranges in the table

Table 11. Structure IRRACX01_PARMLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	SIGNED	4	RANGE_SUBPOOL	Subpool in which the table resides
8	(8)	SIGNED	4	FIRST_RANGE(0)	Array of 1 or more ranges of form:
DATA_RANGE					
0	(0)	STRUCTURE	0	DATA_RANGE	
0	(0)	ADDRESS	4	RANGE_START	Starting virtual address of data
4	(4)	ADDRESS	4	RANGE_END	Ending virtual address of data
8	(8)	SIGNED	4	NEXT_RANGE(0)	

ACXP cross reference

Table 12. Cross Reference for ACXP

Name	Offset	Hex Value
FIRST_RANGE	8	
IRRACX01_ACEEPTR	4	
IRRACX01_OPERATION	0	
IRRACX01_TABLE_PTR	8	
NEXT_RANGE	8	
RANGE_COUNT	0	
RANGE_END	4	
RANGE_START	0	
RANGE_SUBPOOL	4	
RECALL_OPERATION	0	2
STASH_OPERATION	0	1

Chapter 4. AFC: z/OS UNIX System Services audit function codes

AFC programming interface information

AFC is a programming interface.

AFC heading information

Common name: z/OS UNIX System Services common audit function codes

Macro ID: IRRPAFC

DSECT name: N/A

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None (constants only)

Storage attributes:

Subpool
N/A

Key
Any

Residency
Invoker's primary address space

Size: Determined by release (see the mapping)

Created by: N/A

Pointed to by: Constant data only

Serialization: N/A

Function: Contains the constants for the z/OS UNIX System Services audit function codes

AFC constants

Table 13. AFC constants

Value	Type	Len	Name	Description
1	DECIMAL	2	AFC_ACCESS	Check file accessability
2	DECIMAL	2	AFC_CHAUDIT_U	Change user audit options
3	DECIMAL	2	AFC_CHDIR	Change current working directory
4	DECIMAL	2	AFC_CHMOD	Change file modes
5	DECIMAL	2	AFC_CHOWN	Change owner and group of file
6	DECIMAL	2	AFC_DUB	Initialize a process
7	DECIMAL	2	AFC_EXEC	Execute a file
8	DECIMAL	2	AFC_FCHAUDIT_U	Change user audit options when file is open
9	DECIMAL	2	AFC_FCHMOD	Change file modes when file is open

Table 13. AFC constants (continued)

Value	Type	Len	Name	Description
10	DECIMAL	2	AFC_FCHOWN	Change owner and group of file when file is open
11	DECIMAL	2	AFC_GETCWD	Get current working directory
12	DECIMAL	2	AFC_GETPSENT	Get process entry
13	DECIMAL	2	AFC_KILL	Signal a process
14	DECIMAL	2	AFC_LINK	Link to a file
15	DECIMAL	2	AFC_LSTAT	Get file status; do not resolve ending symlink
16	DECIMAL	2	AFC_MKDIR	Make a directory
17	DECIMAL	2	AFC_MKNOD	Make a file node
18	DECIMAL	2	AFC_MOUNT	Mount a file system (using the NOSETUID operand)
19	DECIMAL	2	AFC_OPEN	Open a file
20	DECIMAL	2	AFC_OPENDIR	Open a directory
21	DECIMAL	2	AFC_PATHCONF	Get configurable pathname
22	DECIMAL	2	AFC_PTRACE	Debug a process
23	DECIMAL	2	AFC_READLINK	Read a symbolic link
24	DECIMAL	2	AFC_RENAME	Rename a file
25	DECIMAL	2	AFC_RMDIR	Remove a directory
26	DECIMAL	2	AFC_SETEGID	Set effective GID
27	DECIMAL	2	AFC_SETEUID	Set effective UID
28	DECIMAL	2	AFC_SETGID	Set real/saved and/or effective GID
29	DECIMAL	2	AFC_SETUID	Set real/saved and/or effective UID
30	DECIMAL	2	AFC_STAT	Get file status
31	DECIMAL	2	AFC_SYMLINK	Create a symbolic link
32	DECIMAL	2	AFC_UNLINK	Remove directory entries (Delete a file)
33	DECIMAL	2	AFC_UNMOUNT	Unmount a file system (with the NOSETUID operand)
34	DECIMAL	2	AFC_UTIME	Set file access/modification times
35	DECIMAL	2	AFC_UNDUB_EXIT	Terminate a process
36	DECIMAL	2	AFC_WRITE	Write to a file (Clear setid bits)
37	DECIMAL	2	AFC_CHAUDIT_A	Change auditor audit options
38	DECIMAL	2	AFC_FCHAUDIT_A	Change auditor audit options when file is open
39	DECIMAL	2	AFC_LOOKUP	Path name resolution
40	DECIMAL	2	AFC_TTYNAME	Get path name of terminal
41	DECIMAL	2	AFC_IOCTL	Get path name

Table 13. AFC constants (continued)

Value	Type	Len	Name	Description
42	DECIMAL	2	AFC_GETMNT	Get mount entry
43	DECIMAL	2	AFC_QUIESCE	Quiesce a file system (with the NOSETUID operand)
44	DECIMAL	2	AFC_UNQUIESCE	Unquiesce a file system (with the NOSETUID operand)
45	DECIMAL	2	AFC_VREGISTER	Server registration
46	DECIMAL	2	AFC_VRESOLVEPN	Server resolve path name
47	DECIMAL	2	AFC_VLOOKUP	Server lookup
48	DECIMAL	2	AFC_VREADWRITE	Server read write
49	DECIMAL	2	AFC_VREADDIR	Server read directory
50	DECIMAL	2	AFC_SIGACTION	Change Osigset action
51	DECIMAL	2	AFC_VCREATE	Server create
52	DECIMAL	2	AFC_VMAKEDIR	Server make directory
53	DECIMAL	2	AFC_VSYMLINK	Server symbolic link
54	DECIMAL	2	AFC_VSETATTR	Server set file attributes
55	DECIMAL	2	AFC_VLINK	Server link
56	DECIMAL	2	AFC_VREMOVEDIR	Server remove directory
57	DECIMAL	2	AFC_VREMOVE	Server remove
58	DECIMAL	2	AFC_VRENAME	Server rename
59	DECIMAL	2	AFC_CHATTR	Change file attributes
60	DECIMAL	2	AFC_FCHATTR	Change file attributes
61	DECIMAL	2	AFC_THLMT	Set thread limit
62	DECIMAL	2	AFC_MSGCTL	Message control
63	DECIMAL	2	AFC_MSGGET	Message obtain
64	DECIMAL	2	AFC_MSGRCV	Message receive
65	DECIMAL	2	AFC_MSGSND	Message send
66	DECIMAL	2	AFC_SEMCTL	Semaphore control
67	DECIMAL	2	AFC_SEMGET	Get set of semaphores
68	DECIMAL	2	AFC_SEMOP	Semaphore operations
69	DECIMAL	2	AFC_SHMAT	Shared memory attach
70	DECIMAL	2	AFC_SHMCTL	Shared memory control
71	DECIMAL	2	AFC_SETREGID	Set real and/or effective GID
72	DECIMAL	2	AFC_SHMGET	Shared memory get
73	DECIMAL	2	AFC_WGETIPC	Query IPC status
74	DECIMAL	2	AFC_REMOVE	Remove
75	DECIMAL	2	AFC_SET_MODE	Set mode
76	DECIMAL	2	AFC_SET_MSGQB	Set message queue maximum bytes

Table 13. AFC constants (continued)

Value	Type	Len	Name	Description
77	DECIMAL	2	AFC_SET_GID	Set supplementary groups
78	DECIMAL	2	AFC_PASSWORD	Verify password
79	DECIMAL	2	AFC_LCHOWN	Change owner and group of a symbolic link
80	DECIMAL	2	AFC_TRUNCATE	Truncate a file
81	DECIMAL	2	AFC_PFSCTL	Control function for the physical file system
82	DECIMAL	2	AFC_SETRLIMIT	Set maximum resource consumption
83	DECIMAL	2	AFC_SETPRIORITY	Set process scheduling priority
84	DECIMAL	2	AFC_NICE	Change priority of a process
85	DECIMAL	2	AFC_SETREUID	Set real and/or effective UID
86	DECIMAL	2	AFC_WRITEV	Write on a file
87	DECIMAL	2	AFC_FCHDIR	Change working directory
88	DECIMAL	2	AFC_CHROOT	Change root directory
89	DECIMAL	2	AFC_REALPATH	Resolve path name
90	DECIMAL	2	AFC_STATVFS	Get file system information
91	DECIMAL	2	AFC_BIND	Bind a name to a socket
92	DECIMAL	2	AFC_SOCKET	Create an endpoint for communication
93	DECIMAL	2	AFC_THREAD_SEC	Thread level security
94	DECIMAL	2	AFC_AUTHCHECK	Authority check
95	DECIMAL	2	AFC_ACC_SEND	Send Access Rights
96	DECIMAL	2	AFC_ACC_RECV	Receive Access Rights
96	DECIMAL	2	AFC_ACC_DISC	Discard Access Rights
98	DECIMAL	2	AFC_NEWGRP	Newgrp shell utility
99	DECIMAL	2	AFC_CONSOLE	Console communication service
100	DECIMAL	2	AFC_SERV_INIT	WLM service
101	DECIMAL	2	AFC_SPAWN	SpawnUSERID
102	DECIMAL	2	AFC_SWAP_SERV	Swap services
103	DECIMAL	2	AFC_WLMC	WLM C and C++
104	DECIMAL	2	AFC_LOGIN	__LOGIN SYSTEM CALL
105	DECIMAL	2	AFC_MOUNT_SETUID	Mount a file system (using the SETUID operand)
106	DECIMAL	2	AFC_UNMOUNT_SETUID	Unmount a file system (using the SETUID operand)
107	DECIMAL	2	AFC_QUIESCE_SETUID	Quiesce a file system (using the SETUID operand)
108	DECIMAL	2	AFC_UNQUIESCE_SETUID	Unquiesce a file system (using the SETUID operand)

Table 13. AFC constants (continued)

Value	Type	Len	Name	Description
109	DECIMAL	2	AFC_CHMOUNT	Change mount
110	DECIMAL	2	AFC_CHMOUNT_SETUID	Change mount (setuid)
111	DECIMAL	2	AFC_SETFACL	Add, alter, or delete an access control list
112	DECIMAL	2	AFC_SHUTDOWN_REG	Shutdown registration
113	DECIMAL	2	AFC_EACCESS	Check file access for effective IDs
114	DECIMAL	2	AFC_SETFSECL	Set security label of a UNIX file or directory
115	DECIMAL	2	AFC_POE	System port of entry address
116	DECIMAL	2	AFC_LCHATTR	Change file attributes
117	DECIMAL	2	AFC_UNAVAILABLE	Audit function code not available
118	DECIMAL	2	AFC_MOUNT_NA	Mount no audit
119	DECIMAL	2	AFC_MOUNT_U	User mount
120	DECIMAL	2	AFC_MOUNT_UNA	User mount no audit
121	DECIMAL	2	AFC_UNMOUNT_U	User unmount
122	DECIMAL	2	AFC_UNMOUNT_UNA	User unmount no audit
123	DECIMAL	2	AFC_FSACCESS	File system access
124	DECIMAL	2	AFC_SHMMCV	Shared memory segment mutexes and condition variables
125	DECIMAL	2	AFC_FSEXEC	File system execute access
126	DECIMAL	2	AFC_ENDOF_TAB	End of table

Chapter 5. AUL: RACROUTE REQUEST=AUDIT Parameter List (Request Section)

AUL programming interface information

AUL is a programming interface.

AUL heading information

Common name: Request-specific portion of the RACROUTE REQUEST=AUDIT parameter list

Macro ID: ICHPAUL

DSECT name: AUDLIST

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: 36 bytes

Created by: RACROUTE REQUEST=AUDIT macro

Pointed to by: Address of SAFP plus offset at SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=AUDIT routine

AUL mapping

Table 14. Structure AUDLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	AUDLIST	RACAUDIT parameter list
0	(0)	SIGNED	2	AUDVERS	Parameter list version
2	(2)	SIGNED	2	AUDLEN	Parameter list length
4	(4)	SIGNED	4	AUDEVENT	Address of event name
8	(8)	SIGNED	2	AUDEQUAL	Event code qualifier
10	(A)	SIGNED	2		Reserved
12	(C)	SIGNED	4	AUDCLASS	Address of class name
16	(10)	SIGNED	4	AUDENTYX	Address of entity name

Table 14. Structure AUDLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	SIGNED	4	AUDACEE	Address of ACEE
24	(18)	SIGNED	4	AUDLOGST	Address of LOGSTR data
28	(1C)	BITSTRING	1	AUDRESUL	Result byte
29	(1D)	BITSTRING	3	*	Reserved
32	(20)	SIGNED	4	*(4)	Reserved

AUL cross reference

Table 15. Cross Reference for AUL

Name	Offset	Hex Value
AUDACEE	14	
AUDCLASS	C	
AUDENTYX	10	
AUDEQUAL	8	
AUDEVENT	4	
AUDLEN	2	
AUDLIST	0	
AUDLOGST	18	
AUDRESUL	1C	
AUDVERS	0	

Chapter 6. CCXP: Command Preprocessing Exit (ICHCCX00) Parameter List

CCXP programming interface information

CCXP is a programming interface.

CCXP heading information

Common name:	ICHCCX00 Exit Parameter List
Macro ID:	ICHCCXP
DSECT name:	CCXPL
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	N/A
Size:	16 bytes
Created by:	Commands: DELGROUP (DG), DELUSER (DU), REMOVE (RE)
Pointed to by:	R1 on entry to ICHCCX00
Serialization:	None
Function:	Contains the list of addresses passed to the DELGROUP, DELUSER, and REMOVE command preprocessing installation exit routine

CCXP mapping

Table 16. Structure CCXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		CCXPL	
0	(0)	ADDRESS	4	CCXCALLR	Caller address: points to a 1-byte field identifying the calling command:
		.... 1.1.		CCXDELGR	X'0A' DELGROUP
		.... 1.11		CCXDELUS	X'0B' DELUSER
		.... 11..		CCXREMOV	X'0C' REMOVE

Table 16. Structure CCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	CCXENTIT	Entity address: points to an 8-byte field containing the entity name; the name is left-aligned and padded with blanks. For DELUSER and REMOVE, the field is initialized to the user ID; for DELGROUP, the field is initialized to the group name.
8	(8)	ADDRESS	4	CCXARG	Search argument address: points to an area containing a 1-byte length field followed by a 44-byte field containing the search argument for the data set search. For REMOVE and DELGROUP, the value is set to the group name followed by a period; for DELUSER, the value is set to the userid of the user being deleted, followed by a period. Although the exit can change the value, the length should remain within the range of 1 through 44.
12	(C)	ADDRESS	4	CCXCPPL	CPPL address: points to the command processor parameter list.

Chapter 7. CDXP: Notify/Verify/List Exit Routines Parameter List

CDXP programming interface information

The following fields are Programming Interfaces for IRRVAF01, field validation exit:

- CDXCLPT
- CDXCPPL
- CDXDWPT
- CDXENPT
- CDXKYNM
- CDXNTRYX
- CDXOPTP
- CDXPFNM
- CDXSGNM
- CDXUWPT

CDXP heading information

Common name:	Notify/verify/list exit routines parameter list
Macro ID:	ICHCDXP
DSECT name:	CDXPLIST
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	N/A
Size:	108 bytes
Created by:	RACF dynamic parse
Pointed to by:	R1 at entry to exit
Serialization:	None
Function:	Provides an attachment interface for such products as CICS®/ESA that have data segments in the RACF database and provide exits for RACF's dynamic parse functions. This mapping is also used as an interface to modules defined for the dynamic exit IRRVAF01.

CDXP mapping

Table 17. Structure CDXPLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		CDXPLIST	Exit routines main parameter list

Table 17. Structure CDXPLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	SIGNED	4	CDXNXPT	Address of next parameter list
4	(4)	SIGNED	4	CDXCPPL	Address of the TSO/E CPPL
8	(8)	SIGNED	4	CDXDAPT	Address of data area
12	(C)	SIGNED	4	CDXDWPT	Address of 2 byte length followed by keyword value
16	(10)	SIGNED	4	CDXUWPT	Address of 200 byte message area for use by dynamic validation exit
20	(14)	SIGNED	4	CDXENPT	Address of entry length/name. When CDXNTRYX is OFF (ENTRY): Field points to a 1 byte length followed by entry name. When CDXNTRYX is ON (ENTRYX): Field points to a 2 byte buffer length, followed by a 2 byte actual length, followed by entry name. For IRRVAF01, this field points to the first entry in list.
24	(18)	SIGNED	4	CDXCLPT	Address of 8 byte class name
28	(1C)	CHARACTER	8	CDXPFNM	Profile type name (USER, GROUP, DATASET, GENERAL)
36	(24)	BITSTRING	1	CDXPFTP	Profile type X'01' USER X'02' GROUP X'03' DATASET X'04' GENERAL
37	(25)	BITSTRING	1	CDXOPTP	Operation type
		1...		CDXADD	Add
		.1..		CDXALT	Alter
		..1.		CDXDEL	Delete
		...1		CDXLST	List
38	(26)	BITSTRING	1	CDXFLAGS	Miscellaneous flags
		.... 1...		CDXNTRYX	ENTRYX format specified
		1111 .111		*	Reserved
39	(27)	BITSTRING	1	CDXCLTP	Reserved
40	(28)	CHARACTER	8	CDXSGNM	Segment name
48	(30)	CHARACTER	32	CDXKYNM	Keyword name
80	(50)	CHARACTER	8	CDXTMNM	Template name
88	(58)	CHARACTER	2	CDXTMLN	Template length
90	(5A)	BITSTRING	1	CDXFLAG2	Reserved
92	(5C)			CDXADDL	Additional parameter list if any. See mappings for different parameter lists.
Additional parameter list for validation (verify) exit only					

Table 17. Structure CDXPLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
92	(5C)	SIGNED	4	CDXVMSK	Validation masks
Additional parameter list for list exit only					
92	(5C)	SIGNED	4	CDXLHPT	Address of list heading
96	(60)	SIGNED	4	CDXLFPT	Address of data format
100	(64)	SIGNED	4	CDXANMSK	AND mask to be applied
104	(68)	SIGNED	4	CDXORMSK	OR mask to be applied

CDXP cross reference

Table 18. Cross Reference for CDXP

Name	Offset	Hex Value
CDXADD	25	5C
CDXANMSK	64	
CDXCLPT	25	
CDXCPPL	4	
CDXDAPT	8	
CDXDWPT	C	
CDXENPT	14	
CDXFLAGS	26	
CDXFLAG2	5A	
CDXCLPT	27	
CDXKYNM	30	
CDXLFPT	60	
CDXLHPT	5C	
CDXNXPT	0	
CDXOPTP	26	
CDXORMSK	68	
CDXPFNM	1C	
CDXPFTP	24	
CDXSGNM	28	
CDXTMLN	58	
CDXTMNM	50	
CDXUWPT	10	
CDXVMSK	5C	

Chapter 8. CGRP: Connect Group Name Table Definition

CGRP programming interface information

The following field is not Programming Interface information:

- CGRPGPAT
- When addressed using ACEECGRP, the CGRP data area is not intended for customer use as programming interface information.

CGRP heading information

Common name: Connect group name table definition

Macro ID: ICHPCGRP

DSECT name: CGRP, CGRPENTD

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: CGRP (Offset: 0, Length: 4)

Storage attributes:

Subpool
255 (or as specified by the issuer of RACROUTE REQUEST=VERIFY)

Key
0

Residency
May reside above 16M

Size: 32 bytes plus 24 bytes per connect group

Created by: Various RACF functions

Pointed to by: ACEECGRP or ACEEFCGP field of the ACEE data area

Serialization: None (when ACEEFCGP points to it)

Function: Contains the names of the groups where the ACEEUSRI user ID is a member

CGRP mapping

Table 19. Structure CGRP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		CGRP	Connect group name table
0	(0)	CHARACTER	32	CGRPHADR	CGRP header
0	(0)	CHARACTER	4	CGRPID	Table ID
4	(4)	SIGNED	4	CGRPCORE	CGRP subpool and length
4	(4)	ADDRESS	1	CGRPSP	Subpool number
5	(5)	ADDRESS	3	CGRPLEN	Length of CGRP

Table 19. Structure CGRP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	SIGNED	2	CGRPNUM	Maximum entries in table
10	(A)	ADDRESS	1	CGRPVRSN	Version = 1
11	(B)	ADDRESS	1	*	Reserved
12	(C)	SIGNED	4	CGRPSYNC	Synchronize value
16	(10)	ADDRESS	4	CGRPGPAT	Address of group authorities table, or zero if no such table exists
20	(14)	CHARACTER	4	CGRPGCHG	Group connection status change date
20	(14)	BITSTRING	0	CGRPMAXD	Maximum change date
24	(18)	CHARACTER	8	*	Reserved
32	(20)	CHARACTER	24	CGRPENT	Group name entry
CGRPENTD					
0	(0)	STRUCTURE		CGRPENTD	Group name entry
0	(0)	CHARACTER	8	CGRPNAME	Group name
8	(8)	BITSTRING	1	CGRPIND	Indicators for this entry
		1...		CGRPCHK	Always zero, was revoke indicator
		.1..		CGRPREFR	On if group authority table must be refreshed for this connect group
		..1.		CGRPCOMP	On if group entered into group authority table and no later authority changes were made or the group did not need to be entered into the table
		...1		CGRPPROP	On if this group is owned by its superior group. It indicates the group is part of the subgroup tree for propagation of group authorities.
		 1111		*	Reserved.
9	(9)	BITSTRING	1	CGRPAUTH	Group authority indicators
		1...		CGRPSPEC	On if group-special authority
		..1.		CGRPOPER	On if group-operations authority
		...1		CGRPAUDT	On if group-auditor authority
10	(A)	SIGNED	2	CGRPGPNM	Number of entries in group authority table related to this connect group
12	(C)	ADDRESS	4	CGRPGPTE	Address of first group authority table entry related to this connect group

Table 19. Structure CGRP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	SIGNED	2	CGRPSUPG	RACF 1.9.2 or higher: this field is not used. Index in CGRPENT of entry for superior group of this entry, to which the user is connected.
18	(12)	CHARACTER	6		Reserved

CGRP cross reference

Table 20. Cross Reference for CGRP

Name	Offset	Hex Value
CGRP	0	
CGRPAUDT	29	10
CGRPAUTH	29	
CGRPCHK	28	80
CGRPCOMP	28	20
CGRPCORE	4	
CGRPENT	20	
CGRPGCHG	14	
CGRPGPAT	10	
CGRPGPNM	2A	
CGRPGPTE	2C	
CGRPHADR	0	
CGRPID	0	
CGRPIND	28	
CGRPLEN	5	
CGRPMAXD	20	
CGRPNAME	0	
CGRPNUM	8	
CGRPOPER	29	20
CGRPPROP	28	10
CGRPREFR	28	40
CGRPSP	4	
CGRPSPEC	29	80
CGRPSUPG	30	
CGRPSYNC	C	
CGRPVRSN	A	

Chapter 9. CNST/CNSX (RACF): Class Name and Syntax Table

CNST/CNSX (RACF) programming interface information

The following fields are not programming interface information:

- CNSTGNLP
- CNSTRCLP
- CNSTSTKN

CNST/CNSX (RACF) heading information

Common name: RACF class name and syntax table

Macro ID: ICHPCNST

DSECT name: CNST, CNSX

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by issuer of RACROUTE REQUEST=STAT

Key
Determined by issuer of RACROUTE REQUEST=STAT

Residency
Determined by issuer of RACROUTE REQUEST=STAT

Size: Total length of each entry can be found in RCVT field RCVTCDTL. Each entry has 2 parts: 28 bytes in CNST part 152 bytes in CNSX part

Created by: Issuer of RACROUTE REQUEST=STAT.

Pointed to by: The entry for an individual class should be found using RACROUTE REQUEST=STAT with the CLASS=, COPY=, and COPYLEN= parameters.

Serialization: None

Function: Describes a general resource class to RACF. Each class entry contains the name of the general resource class, the resource name syntax, and control information. There is one entry for each general resource class. CNSTLGT=0 indicates the end of the table.

Each entry has two parts: the CNST part points to the CNSX part.

CNSTRACF mapping

Table 21. Structure CNST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	CNST	Class name/syntax table
0	(0)	SIGNED	2	CNSTLGT	Length of CNST portion of entry

Table 21. Structure CNST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
2	(2)	ADDRESS	1	CNSTID	Class ID number
3	(3)	CHARACTER	8	CNSTNAME	Class name
11	(B)	CHARACTER	8	CNSTXREF	Group/member class name
19	(13)	CHARACTER	3	CNSTSNTX(0)	Class member name syntax
19	(13)	ADDRESS	1	CNSTMAXL	Maximum length of member name
20	(14)	BITSTRING	1	CNSTFRST	Syntax of first character
		1...		CNSTFALP	1 => Alphabetic char allowed
		.1..		CNSTFNAT	1 => National char allowed
		..1.		CNSTFNUM	1 => Numeric char allowed
		...1		CNSTFSPE	1 => Special char allowed
21	(15)	BITSTRING	1	CNSTREMN	Syntax of remaining character
		1...		CNSTRALP	1 => Alphabetic char allowed
		.1..		CNSTRNAT	1 => National char allowed
		..1.		CNSTRNUM	1 => Numeric char allowed
		...1		CNSTRSPE	1 => Special char allowed
22	(16)	BITSTRING	1	CNSTUACC	Default UACC
		1...		CNSTALTR	1 => Alter UACC
		.1..		CNSTCNTL	1 => Control UACC
		..1.		CNSTUPDT	1 => Update UACC
		...1		CNSTREAD	1 => Read UACC
		.... 1...		CNSTEXEC	1 => Execute UACC
		1		CNSTNONE	1 => None UACC
23	(17)	BITSTRING	1	CNSTMFLG	Miscellaneous flags
		1...		CNSTRGRP	1 => Class is resource group
		.1..		CNSTACEE	1 => Use UACC from ACEE
		..1.		CNSTOPER	1 => OPERATIONS attribute applies to this class
		...1		CNSTRACL	1 => RACLIST allowed
		.... 1...		CNSTGENL	1 => GENLIST allowed
		.... .1..		CNSTDSPC	1 => RACLISTed to a data space
		.... ..1.		CNSTXFLG	1 => CNST is in the new format using CNSX
		1		CNSTOWNR	1 => User-installed CDT entry or 0 => IBM-supplied CDT entry
24	(18)	ADDRESS	4	CNSTCNSX	Address of the rest of the class entry fields in CNSX
28	(1C)	CHARACTER	0	CNSTCBLN	Force alignment

Table 21. Structure CNST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
CNSX					
0	(0)	STRUCTURE	152	CNSX	Class entry extension
0	(0)	BITSTRING	4	CNSTMASK	Mask for old option flags
0	(0)	BITSTRING	16	CNSTLMSK	Long mask for option flag
0	(0)	BITSTRING	128	CNSTVMSK	Longer mask for option flag
128	(80)	ADDRESS	4	CNSTGNLP	Pointer to GENLISTed CDT profiles
132	(84)	ADDRESS	4	CNSTRCLP	Pointer to RACLISTed CDT profiles
136	(88)	ADDRESS	1	CNSTDFRC	Default return code; if omitted, defaults to 4
137	(89)	BITSTRING	1	CNSTFLG0	Miscellaneous flags
		1...		CNSTRLRQ	1 ==> Class must be RACLISTed
		.1..		CNSTPRDF	0 ==> Allow profiles to be defined for this class
		..1.		CNSTUSLB	1 ==> SECLABEL required for this class
		...1		CNSTRMAC	1 ==> Reverse MAC checking is done for this class
		.... 1...		CNSTDUP	1 ==> characters 1 - 4 are identical to a previous class
		.... .1..		CNSTCASE	1 ==> Preserve case of profile name (0 = upper case)
		.... ..1.		CNSTEMAC	1 => Equal MAC checking is done for this class
		1		CNSTSGNL	1 => Send ENF signal when this class is refreshed
138	(8A)	ADDRESS	1	CNSTKEYQ	KEYQUAL ==> Number of significant qualifiers
139	(8B)	ADDRESS	1	CNSTORML	Original allowable maximum length of member name, used for 'ENTITY' keyword only.
140	(8C)	BITSTRING	1	CNSTFLG1	Miscellaneous flags
		1...		CNSTDYN	1 ==> Class was defined using dynamic CDT
		.1..		CNSTDDUP	1 ==> Class is duplicated in ICHRRCDE and dynamic CDT
		..1.		CNSTCOPY	1 ==> This is a copy of a CDT
		...1		CNSTNGEN	1 ==> SETR GENERIC is not allowed for the class
141	(8D)	CHARACTER	3	*	Reserved (align to doubleword boundary)

Table 21. Structure CNST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
144	(90)	CHARACTER	8	CNSTSTKN	STOKEN for data space used to RACLIST profiles
152	(80)	CHARACTER		*	Force alignment

CNST/CNSX cross reference

Table 22. Cross Reference for CNSTRACF

Name	Offset	Hex Value
CNST	0	
CNSTACEE	17	40
CNSTALTR	16	80
CNSTCASE	89	06
CNSTCNSX	18	
CNSTCNTL	16	40
CNSTCOPY	8C	20
CNSTDDUP	8C	40
CNSTNGEN	8C	10
CNSTDFRC	88	
CNSTDSPC	17	04
CNSTDUP	89	08
CNSTDYN	8C	80
CNSTEMAC	89	04
CNSTEXEC	16	08
CNSTFALP	14	80
CNSTFLG0	89	
CNSTFLG1	8C	
CNSTFNAT	14	40
CNSTFNUM	14	20
CNSTFRST	14	
CNSTFSPE	14	10
CNSTGENL	17	8
CNSTGNLP	80	
CNSTID	2	
CNSTKEYQ	8A	
CNSTLGT	0	
CNSTLMSK	0	
CNSTMASK	0	

Table 22. Cross Reference for CNSTRACF (continued)

Name	Offset	Hex Value
CNSTMAXL	13	
CNSTMFLG	17	
CNSTMSKS	4	
CNSTNAME	3	
CNSTNONE	16	01
CNSTOPER	17	20
CNSTORML	8B	
CNSTOWNR	17	01
CNSTPRDF	89	40
CNSTRACL	17	10
CNSTRALP	15	80
CNSTRCLP	84	
CNSTREAD	16	10
CNSTREMN	15	
CNSTRGRP	17	80
CNSTRLRQ	89	80
CNSTRMAC	89	10
CNSTRNAT	15	40
CNSTRNUM	15	20
CNSTRSPE	15	10
CNSTSGNL	89	02
CNSTSNTX	13	
CNSTSTKN	90	
CNSTUACC	16	
CNSTUPDT	16	20
CNSTUSLB	89	20
CNSTVMSK	0	
CNSTXFLG	17	02
CNSTXREF	B	
CNSX	0	

Chapter 10. CNST/CNSX (SAF): Class Name and Syntax Table

CNST/CNSX (SAF) programming interface information

CNST/CNSX (SAF) is a programming interface.

CNST/CNSX (SAF) heading information

Common name: RACF class name and syntax table

Macro ID: IRRPCNST

DSECT name: CNST, CNSX

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by issuer of RACROUTE REQUEST=STAT

Key
Determined by issuer of RACROUTE REQUEST=STAT

Residency
Determined by issuer of RACROUTE REQUEST=STAT

Size: 180 bytes

Created by: Storage provided by issuer of RACROUTE REQUEST=STAT.

Pointed to by: The entry for an individual class should be found using RACROUTE REQUEST=STAT with the CLASS=, COPY=, and COPYLEN= parameters.

Serialization: None

Function: Describes syntax information for a general resource class supported by the installed security product. Installations using RACF may want to use CNST/CNSX (RACF) which is used by the ICHPCNST macro, rather than CNST/CNSX (SAF) which is used by the IRRPCNST macro.

Each entry has two parts: the CNST part points to the CNSX part.

CNSTSAF mapping

Table 23. Structure CNST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	CNST	Class name/syntax table
0	(0)	SIGNED	2	CNSTLGT	Length of CNST portion of entry
2	(2)	ADDRESS	1	*	Reserved
3	(3)	CHARACTER	8	CNSTNAME	Class name
11	(B)	CHARACTER	8	*	Reserved

Table 23. Structure CNST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
19	(13)	ADDRESS	1	CNSTMAXL	Maximum length of member name
20	(14)	BITSTRING	1	*	Reserved
24	(18)	ADDRESS	4	CNSTCNSX	Address of the rest of the class entry fields in CNSX
28	(1C)	Character		*	Force alignment
CNSX					
0	(0)	STRUCTURE	152	CNSX	Class entry extension
0	(0)	CHARACTER	139	*	Reserved
139	(8B)	ADDRESS	1	CNSTORML	Original allowable maximum length of member name, used for 'ENTITY' keyword only
140	(8C)	CHARACTER	4	*	Reserved (align to doubleword boundary)
144	(90)	CHARACTER	8	*	Reserved
152	(98)	CHARACTER		*	Force alignment

Chapter 11. CNXP: Command Preprocessing Exit (ICHCNX00) Parameter List

CNXP programming interface information

CNXP is a programming interface.

CNXP heading information

Common name: ICHCNX00 command preprocessing exit parameter list

Macro ID: ICHCNXP

DSECT name: CNXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes: N/A

Size: 48 bytes

Created by:

Commands:
ADDSD, ALTDSD, DELDSD, LISTDSD, PERMIT, SEARCH

Utilities:
IRRUT100

Pointed to by: R1 at entry to ICHCNX00

Serialization: None

Function: Contains the list of addresses passed to the ICHCNX00 command preprocessing exit

CNXP mapping

Table 24. Structure CNXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		CNXPL	
Additional parameter list					
0	(0)	ADDRESS	4	CNXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	CNXCALLR	Caller address: points to a 2-byte field containing a function code and subfunction code identifying the caller:
		BITSTRING	0	CNXRACHK	X'0100' RACHECK
		BITSTRING	0	CNXRDDEF	X'0201' RACDEF DEFINE

Table 24. Structure CNXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		BITSTRING	0	CNXRDREN	X'0202' RACDEF RENAME
		BITSTRING	0	CNXRDADV	X'0203' RACDEF ADDVOL
		BITSTRING	0	CNXRDDEL	X'0204' RACDEF DELETE
		BITSTRING	0	CNXRDCHV	X'0205' RACDEF CHGVOL
		BITSTRING	0	CNXADSET	X'0301' ADDSD SET
		BITSTRING	0	CNXADNOS	X'0302' ADDSD NOSET
		BITSTRING	0	CNXADMOD	X'0303' ADDSD MODEL
		BITSTRING	0	CNXALSET	X'0401' ALTDSD SET
		BITSTRING	0	CNXALNOS	X'0402' ALTDSD NOSET
		BITSTRING	0	CNXDLSET	X'0501' DELDSD SET
		BITSTRING	0	CNXDLNOS	X'0502' DELDSD NOSET
		BITSTRING	0	CNXLDPRE	X'0601' LISTDSD prelocate call
		BITSTRING	0	CNXLDDS	X'0602' LISTDSD DATASET
		BITSTRING	0	CNXLDIDP	X'0603' LISTDSD ID or PREFIX
		BITSTRING	0	CNXPERTO	X'0701' PERMIT TO-resource
		BITSTRING	0	CNXPERFR	X'0702' PERMIT FROM-resource
		BITSTRING	0	CNXSRCPR	X'0801' SEARCH prelocate
4	(4)	BITSTRING	0	CNXSRCPO	X'0802' SEARCH postlocate
4	(4)	BITSTRING	0	CNXUT100	X'0900' ICHUT100
4		BITSTRING	0	CNXRXTRT	X'0D00' RACXTRT
8	(8)	ADDRESS	4	CNXFLAG	Authority flag address: points to a 1-byte field containing the user's authorization to the requested function:
		.... 1...		CNXREAD	X'08' READ
		1...		CNXALTCR	X'80' ALTER or CREATE. In order to issue the SEARCH command for a data set, a user requires at least READ authority. In order to issue LISTDSD for a data set specifying the AUTHUSER or ALL operands, the user must have ALTER authority or the equivalent.
12	(C)	ADDRESS	4	CNXRESNM	Resource name address: points to a 1-byte field containing the resource name length followed by a 44-byte area containing the resource name. The name is left-justified.

Table 24. Structure CNXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	4	CNXOLDNM	Old name address: points to a 1-byte field containing the length of the name followed by a 44-byte area containing the name of the data set that was renamed. The name is left-justified.
20	(14)	ADDRESS	4	CNXVOL	Volume serial address: points to an area containing a 1-byte count field followed by a variable number of 6-byte fields containing volume serial identifiers, each left-justified and padded on the right with blanks.
24	(18)	ADDRESS	4	CNXOLVOL	Old volume serial address: points to a 6-byte area containing the volume serial identifier, left-justified and padded on the right with blanks.
28	(1C)	ADDRESS	4	CNXRESCL	Resource class name address: points to an 8-byte field containing the resource class name (DATASET). See the notes for the LISTDSD and SEARCH commands in the SPL: RACF chapter entitled "RACF Installation Exits."
32	(20)	ADDRESS	4	CNXQUAL	Qualifier address: points to an 8-byte field containing the data set qualifier. The qualifier is left-justified and padded on the right with blanks. This value is initialized to the high-level qualifier of the data set with the exceptions noted in the SPL: RACF "RACF Installation Exits - ICHCNX00. If the exit changes the value, processing continues with the changed value. For ADDSD, RACDEF DEFINE, and RACDEF RENAME, RACF determines if the value is a userid or a group defined to RACF. For the other commands and ICHUT100, RACF determines if the value is a userid.
36	(24)	ADDRESS	4	CNXDSTYP	Data set type address: points to a 1-byte flag field indicating the type of data set:
		1		CNXUNKWN	X'01' unknown
		.1..		CNXGRPDS	X'40' group data set

Table 24. Structure CNXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		CNXUSRDS	X'80' user data set The use of this field is explained in more detail in the following topic, "Return Codes - Command Preprocessing Exit ICHCNX00" in the SPL: RACF.
40	(28)	ADDRESS	4	CNXAUTH	Authority address: points to a 1-byte flag field containing the authority granted by the exit:
		1		CNXNONE	X'01' None
		1...		CNXALTER	X'80' ALTER As noted in the *RACF Installation Exits* chapter in SPL: RACF, this field is used only for the LISTDSD command. It is intended for those cases when the exit gives the user the authority to list the data set description, which requires READ authority, but not list the access list, which requires ALTER authority.
44	(2C)	ADDRESS	4	CNXCPPL	CPPL address: points to the command processor list (mapped by the IKJCPPL macro instruction). The CPPL can be used to prompt or send messages to a TSO user. See the chapter, "RACF Installation Exits" in SPL: RACF. The address is zero in non-TSO cases.

CNXP cross reference

Table 25. Cross Reference for CNXP

Name	Offset	Hex Value
CNXADMOD	4	303
CNXADNOS	4	302
CNXADSET	4	301
CNXALNOS	4	402
CNXALSET	4	401
CNXALTCR	8	80
CNXALTER	28	80
CNXAUTH	28	
CNXCALLR	4	
CNXCPPL	2C	
CNXDLNOS	4	502
CNXDLSET	4	501

Table 25. Cross Reference for CNXP (continued)

Name	Offset	Hex Value
CNXDSTYP	24	
CNXFLAG	8	
CNXGRPDS	24	40
CNXLDDS	4	602
CNXLDIDP	4	603
CNXLDPRE	4	601
CNXLEN	0	
CNXNONE	28	1
CNXOLDNM	10	
CNXOLVOL	18	
CNXPERFR	4	702
CNXPERTO	4	701
CNXPL	0	
CNXQUAL	20	
CNXRACHK	4	100
CNXRDADV	4	203
CNXRDCHV	4	205
CNXRDDEF	4	201
CNXRDDEL	4	204
CNXRDREN	4	202
CNXREAD	8	8
CNXRESCL	1C	
CNXRESNM	C	
CNXRXTRT	4	D00
CNXSRCPO	4	802
CNXSRCPR	4	801
CNXUNKWN	24	1
CNXUSRDS	24	80
CNXUT100	4	900
CNXVOL	14	

Chapter 12. COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services

COMP programming interface information

COMP is a programming interface.

COMP heading information

Common name:	OMVS Common Security Parameter List
Macro ID:	IRRPCOMP
DSECT name:	COMP
Owning component:	SAF (SC1BN)
Eye-catcher ID:	None - this is not a control block definition.
Storage attributes:	**Subpool** N/A **Key** Any **Residency** Invoker's primary address space

Size:

Section / Size
COMP / 28 bytes
IUSP / 8 bytes
CSID / 16 bytes
EXID / 40 bytes
GINF / 28 bytes
GETG / 32 bytes
CHKP / 8 bytes
GMAP / 24 bytes
CKPO / 32 bytes
QRYS / 16 bytes
CMOD / 32 bytes
CLID / 24 bytes
CAUD / 32 bytes
COWN / 40 bytes
UMSK / 8 bytes
KACC / 40 bytes
QRYF / 16 bytes
KFOW / 24 bytes
RAUX / 76 bytes
MKRT / 32 bytes
PTRC / 24 bytes
MFSP / 40 bytes
RAUD / 48 bytes
GUGP / 48 bytes
FORK / 40 bytes
MISP / 32 bytes
IACC / 24 bytes
IOWN / 48 bytes
CKO2 / 40 bytes
GETE / 56 bytes
DKEY / 32 bytes
DINF / 40 bytes
DRUR / 32 bytes
DAUT / 36 bytes
INTA / 68 bytes
ADMN / 24 bytes
UMAP / 32 bytes
CDDL / 24 bytes
KERB / 16 bytes
TKTS / 28 bytes
PKIS / 28 bytes
CACH / 88 bytes
PRXY / 44 bytes
RACL / 40 bytes
PGSN / 16 bytes
WPRV / 12 bytes
SECL / 24 bytes

Created by: Invoker of z/OS UNIX System Services security functions

Pointed to by: Address of COMP is passed in register 1 when invoking z/OS UNIX System Services security functions

Serialization: None

Function: Maps the common input parameter list for the RACF and SAF callable services routers.

Note: For more mapping information about IRRPCOMP, see the parameter list descriptions in *z/OS Security Server RACF Callable Services*.

COMP mapping

Table 26. Structure COMP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	COMP	Common SAF/RACF plist.
0	(0)	ADDRESS	4	COMP_WORKA_STOR@	Address of 1024 byte work area.
4	(4)	ADDRESS	4	COMP_SAFRC_ALET@	Address of ALET for SAF Return Code.
8	(8)	ADDRESS	4	COMP_SAFRC_STOR@	Address of SAF Return Code.
12	(C)	ADDRESS	4	COMP_RACRC_ALET@	Address of ALET for RACF Return Code.
16	(10)	ADDRESS	4	COMP_RACRC_STOR@	Address of RACF Return Code.
20	(14)	ADDRESS	4	COMP_RACSC_ALET@	Address of ALET for RACF Reason Code.
24	(18)	ADDRESS	4	COMP_RACSC_STOR@	Address of RACF Reason Code.
28	(1C)	CHARACTER	1	COMP_SERVR_PARMS	Service routine parameter. Lists begin here.
IUSP					
0	(0)	STRUCTURE	8	IUSP	Mapping for IUSP
0	(4)	ADDRESS	4	IUSP_WORKA_ALET@	Address of ALET for the output area.
4	(4)	ADDRESS	4	IUSP_WORKA_STOR@	Address of a 4 byte output area. This area contains an address that points to data about the user.
CSID					
0	(0)	STRUCTURE	16	CSID	Mapping for CSID
0	(0)	ADDRESS	4	CSID_ID_ALET@	Address of ALET for the UID/GID.
4	(4)	ADDRESS	4	CSID_ID@	Address of the UID/GID.
8	(8)	ADDRESS	4	CSID_IDOA_ALET@	Address of ALET for the output area.
12	(C)	ADDRESS	4	CSID_IDOA@	Address of a 3 word output area. This area contains the new real, effective and saved UID/GIDs.
EXID					
0	(0)	STRUCTURE	40	EXID	Mapping for EXID
0	(0)	ADDRESS	4	EXID_FLAG_ALET@	Address of ALET for a byte flag.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	EXID_FLAG@	Address of a one byte flag that indicates set Uid, Gid, or Both. x'01' Set Uid. x'02' Set Gid. x'03' Set Both.
8	(8)	ADDRESS	4	EXID_UID_ALET@	Address of ALET for Uid to be set.
12	(C)	ADDRESS	4	EXID_UID@	Address of the Uid to be set.
16	(10)	ADDRESS	4	EXID_GID_ALET@	Address of ALET for Gid to be set.
20	(14)	ADDRESS	4	EXID_GID@	Address of the Gid to be set.
24	(18)	ADDRESS	4	EXID_UIDO_ALET@	Address of ALET for Uid output area.
28	(1C)	ADDRESS	4	EXID_UIDO@	Address of Uid output area, which is 3 words filled in by IRRREX00 with the real, effective and saved Uids.
32	(20)	ADDRESS	4	EXID_GIDO_ALET@	Address of ALET for Gid output area.
36	(24)	ADDRESS	4	EXID_GIDO@	Address of Gid output area, which is 3 words filled in by IRRREX00 with the real, effective and saved Gids.
GINF					
0	(0)	STRUCTURE	28	GINF	Mapping for GNIF
0	(0)	ADDRESS	4	GINF_NUM_PARMS@	Address of fullword containing the total number of parameters.
4	(4)	ADDRESS	4	GINF_PARM_ALET@	Address of the ALET of the rest of the parms.
8	(8)	ADDRESS	4	GINF_FUNC_CODE@	Address of halfword function code. Constants for the function codes are supplied below.
12	(C)	ADDRESS	4	GINF_OPTION@	Address of halfword Option for function code.
16	(10)	ADDRESS	4	GINF_RACF_ENTITY@	Address of RACF entity.
20	(14)	ADDRESS	4	GINF_RACF_CLASS@	Address of RACF class.
24	(18)	ADDRESS	4	GINF_RESULT_ENTRIES@	Address of result entries area.
GETG					
0	(0)	STRUCTURE	32	GET	Mapping for GETG
0	(0)	ADDRESS	4	GETG_KEY_ALET@	Address of ALET for the user's key.
4	(4)	ADDRESS	4	GETG_KEY@	Address of a byte that contains the user's key. The key is in the high order 4 bits of the byte.
8	(8)	ADDRESS	4	GETG_GCNT_ALET@	Address of ALET for the Group Count.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	GETG_GCNT@	Address of the group count. (A one word area that contains the number of GIDs in the Grouplist area.)
16	(10)	ADDRESS	4	GETG_GLST_ALET@	Address of ALET for the Group List.
20	(14)	ADDRESS	4	GETG_GLST@	Address of the Group List area in which supplemental GIDs are returned.
24	(18)	ADDRESS	4	GETG_GNUM_ALET@	Address of ALET for the number of groups.
28	(1C)	ADDRESS	4	GETG_GNUM@	Address of the Number Of Groups. A one word area in which the number of supplemental group GIDs is returned.
CHKP					
0	(0)	STRUCTURE	8	CHKP	Mapping for CHKP
0	(0)	ADDRESS	4	CHKP_AUFC_ALET@	Address of ALET for audit function code.
4	(4)	ADDRESS	4	CHKP_AUFC@	Address of the audit function code. (A full word containing the function code that identifies the system call being processed.)
GMAP					
0	(0)	STRUCTURE	24	GMAP	Mapping for GMAP
0	(0)	ADDRESS	4	GMAP_FLAG_ALET@	Address of ALET for one byte flag.
4	(4)	ADDRESS	4	GMAP_FLAG@	Address of the one byte flag: x'00' Search by Gid/Uid. x'01' Search by Group ID/Userid.
8	(8)	ADDRESS	4	GMAP_ID_ALET@	Address of ALET for the Gid/Uid.
12	(C)	ADDRESS	4	GMAP_ID@	Address of a word containing the input Uid/Gid or in which the Uid/Gid is returned.
16	(10)	ADDRESS	4	GMAP_NAME_ALET@	Address of ALET for the Group Name or Userid.
20	(14)	ADDRESS	4	GMAP_NAME@	Address of 8 bytes that contain an input Group Name or Userid or in which the Group name or Userid is returned. The name/id left justified and padded with blanks.
CKPO					
0	(0)	STRUCTURE	32	CKPO	Mapping for CKPO
0	(0)	ADDRESS	4	CKPO_REQT_ALET@	Address of ALET for one byte Request type.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	CKPO_REQT@	Address of the one byte request type: 1 - Audit only request from KILL used when SIGCONT signal is being sent. 2 - KILL request. 3 - GETPSENT request. 4 - OPEN TTY request. 5 - SIGQUEUE req w/ seclabel
8	(8)	ADDRESS	4	CKPO_UIDS_ALET@	Address of ALET for Uids area.
12	(C)	ADDRESS	4	CKPO_UIDS@	Address of 3 word area containing the real, effective, and saved Uids (in that order) for the target process, followed by an 8-byte SECLABEL when request type is 5.
16	(10)	ADDRESS	4	CKPO_PID_ALET@	Address of ALET for target PID word.
20	(14)	ADDRESS	4	CKPO_PID@	Address of the words containing the target process's PID.
24	(18)	ADDRESS	4	CKPO_CODE_ALET@	Address of ALET for signal code word.
28	(1C)	ADDRESS	4	CKPO_CODE@	Address of word containing a code identifying the signal being sent. This word is ignored for request type 3.

QRYS

0	(0)	STRUCTURE	16	QRYS	Mapping for QRYS
0	(0)	ADDRESS	4	QRYS_OPTN_CODE_ALET@	Address of ALET for the one word requested option code.
4	(4)	ADDRESS	4	QRYS_OPTN_CODE@	Address of the one word requested option code. The defined codes are: x'00000001' - NGROUPS_MAX options x'00000002' - POSIX_SAVED_IDS opt
8	(8)	ADDRESS	4	QRYS_OUTP_VAL_ALET@	Address of ALET for one word output value
12	(C)	ADDRESS	4	QRYS_OUTP_VAL@	Address of one word output value

CMOD

0	(0)	STRUCTURE	32	CMOD	Mapping for CMOD
0	(0)	ADDRESS	4	CMOD_MODE_ALET@	Address of ALET for the mode parameter
4	(4)	ADDRESS	4	CMOD_MODE@	Address of a one word mode parameter.
8	(8)	ADDRESS	4	CMOD_FSP_ALET@	Address of ALET for the FSP structure

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	CMOD_FSP@	Address of the FSP structure.
16	(10)	ADDRESS	4	CMOD_FID_ALET@	Address of ALET for the File Id.
20	(14)	ADDRESS	4	CMOD_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
24	(18)	ADDRESS	4	CMOD_CRED_ALET@	Address of ALET for CRED structure.
28	(1C)	ADDRESS	4	CMOD_CRED@	Address of the CRED structure.
CLID					
0	(0)	STRUCTURE	24	CLID	Mapping for CLID
0	(0)	ADDRESS	4	CLID_FSP_ALET@	Address of ALET for the FSP structure.
4	(4)	ADDRESS	4	CLID_FSP@	Address of the FSP structure.
8	(8)	ADDRESS	4	CLID_FID_ALET@	Address of ALET for the File Id.
12	(C)	ADDRESS	4	CLID_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
16	(10)	ADDRESS	4	CLID_CRED_ALET@	Address of ALET for CRED structure.
20	(14)	ADDRESS	4	CLID_CRED@	Address of the CRED structure.
CAUD					
0	(0)	STRUCTURE	32	CAUD	Mapping for CAUD
0	(0)	ADDRESS	4	CAUD_AO_ALET@	Address of ALET for the Audit Options.
4	(4)	ADDRESS	4	CAUD_AO@	Address of 4 bytes that contain an Audit Option for each type of access: Byte 1 - read access audit options. Byte 2 - write access audit options. Byte 3 - execute/ search audit options. Each of the first three bytes contain an Audit Option: x'00' - don't audit any access attempts x'01' - audit successful access. x'02' - audit failed access attempts. Byte 4 - audit flag. In the last byte, the last bit indicates: x'00' - set user audit options. x'01' - set auditor audit options.
8	(8)	ADDRESS	4	CAUD_FSP_ALET@	Address of ALET for the FSP structure.
12	(C)	ADDRESS	4	CAUD_FSP@	Address of the FSP structure.
16	(10)	ADDRESS	4	CAUD_FID_ALET@	Address of ALET for the File Id.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	CAUD_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
24	(18)	ADDRESS	4	CAUD_CRED_ALET@	Address of ALET for CRED structure.
28	(1C)	ADDRESS	4	CAUD_CRED@	Address of the CRED structure.

COWN

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	COWN	Mapping for COWN
0	(0)	ADDRESS	4	COWN_UID_ALET@	Address of ALET for Uid.
4	(4)	ADDRESS	4	COWN_UID@	Address of the Uid to be set as the file owner Uid.
8	(8)	ADDRESS	4	COWN_GID_ALET@	Address of ALET for Gid.
12	(C)	ADDRESS	4	COWN_GID@	Address of the Gid to be set as the file owner Gid.
16	(10)	ADDRESS	4	COWN_FSP_ALET@	Address of ALET for the FSP structure.
20	(14)	ADDRESS	4	COWN_FSP@	Address of the FSP structure.
24	(18)	ADDRESS	4	COWN_FID_ALET@	Address of ALET for the File Id.
28	(1C)	ADDRESS	4	COWN_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
32	(20)	ADDRESS	4	COWN_CRED_ALET@	Address of ALET for CRED structure.
36	(24)	ADDRESS	4	COWN_CRED@	Address of the CRED structure.

UMSK

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	8	UMSK	Mapping for UMSK
0	(0)	ADDRESS	4	UMSK_MODE_ALET@	Address of ALET for the mode parameter
4	(4)	ADDRESS	4	UMSK_MODE@	Address of the mode parameter.

KACC

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	KACC	Mapping for KACC
0	(0)	ADDRESS	4	KACC_ACODE_ALET@	Address of ALET for the access code.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	KACC_ACODE@	Address of a 1 byte access code that contains: x'01' - execute access. x'02' - write access. x'03' - write and execute access. x'04' - read access. x'05' - read and execute access. x'06' - read and write access. x'07' - read, write and execute access. x'81' - directory search access. x'87' - any access.
8	(8)	ADDRESS	4	KACC_FSP_ALET@	Address of ALET for the FSP structure.
12	(C)	ADDRESS	4	KACC_FSP@	Address of the FSP structure.
16	(10)	ADDRESS	4	KACC_FID_ALET@	Address of ALET for the File Id.
20	(14)	ADDRESS	4	KACC_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
24	(18)	ADDRESS	4	KACC_CRED_ALET@	Address of ALET for CRED structure.
28	(1C)	ADDRESS	4	KACC_CRED@	Address of the CRED structure.
32	(20)	ADDRESS	4	KACC_FLAG_ALET@	Address of the ALET for the name flag byte.
36	(24)	ADDRESS	4	KACC_FLAG@	Address of a byte flag that indicates which name is being checked: x'00' - Use the CRED_name_flag to determine path name being checked. Used by lookup. x'01' - The old (or only) name is being checked. Used by all calls except lookup and when rename and link are checking for write access to the parent directory of the new path name file. x'02' - The new name is being checked. Used by rename and link when checking for write access to the parent directory of the new path name file.

QRYF

0	(0)	STRUCTURE	16	QRYF	Mapping for QRYF
0	(0)	ADDRESS	4	QRYF_OCODE_ALET@	Address of ALET for the option code.
4	(4)	ADDRESS	4	QRYF_OCODE@	Address of a 1 word option code that contains: x'00000001'- POSIX_CHOWN_RESTRICTED.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	QRYF_OUTV_ALET@	Address of ALET for the output value.
12	(C)	ADDRESS	4	QRYF_OUTV@	Address of the output value that contains: 0 - POSIX_CHOWN_RESTRICTED. -1 - Not POSIX_CHOWN_RESTRICTED.

KFOW

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	24	KFOW(	Mapping for KFOW
0	(0)	ADDRESS	4	KFOW_FSP_ALET@	Address of ALET for the input FSP.
4	(4)	ADDRESS	4	KFOW_FSP@	Address of the input FSP.
8	(8)	ADDRESS	4	KFOW_FID_ALET@	Address of ALET for the File Id.
12	(C)	ADDRESS	4	KFOW_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
16	(10)	ADDRESS	4	KFOW_CRED_ALET@	Address of ALET for CRED structure.
20	(14)	ADDRESS	4	KFOW_CRED@	Address of the CRED structure.

MKRT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	32	MKRT	Mapping for MKRT
0	(0)	ADDRESS	4	MKRT_MODE_ALET@	Address of ALET for the mode parameter
4	(4)	ADDRESS	4	MKRT_MODE@	Address of the mode parameter.
8	(8)	ADDRESS	4	MKRT_OFSP_ALET@	Address of ALET for the output FSP area.
12	(C)	ADDRESS	4	MKRT_OFSP@	Address of the output FSP area.
16	(10)	ADDRESS	4	MKRT_FID_ALET@	Address of ALET for the File Id.
20	(14)	ADDRESS	4	MKRT_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
24	(18)	ADDRESS	4	MKRT_DSN_ALET@	Address of ALET for the name of the dataset.
28	(1C)	ADDRESS	4	MKRT_DSN@	Address of an area that contains the name of the PDSE/x data set being created. (44 character length)

PTRC

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	24	PTRC	Mapping for PTRC
0	(0)	ADDRESS	4	PTRC_TUID_ALET@	Address of ALET for the target UIDs area.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	PTRC_TUID@	Address of a 3 word area that contains the REAL, EFFECTIVE, and SAVED UIDs (in that order) for the target process, followed by an 8-byte SECLABEL when the high-order bit of the PID is on.
8	(8)	ADDRESS	4	PTRC_TGID_ALET@	Address of ALET for the target GIDs area.
12	(C)	ADDRESS	4	PTRC_TGID@	Address of a 3 word area that contains the REAL, EFFECTIVE, and SAVED GIDs (in that order) for the target process.
16	(10)	ADDRESS	4	PTRC_TPID_ALET@	Address of ALET for the target PID.
20	(14)	ADDRESS	4	PTRC_TPID@	Address of a word that contains the PID of the target process.

MFSP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	()	STRUCTURE	40	MFSP	Mapping for MFSP
0	(0)	ADDRESS	4	MFSP_MODE_ALET@	Address of ALET for the mode parameter
4	(4)	ADDRESS	4	MFSP_MODE@	Address of the mode parameter.
8	(8)	ADDRESS	4	MFSP_OFSP_ALET@	Address of ALET for the output FSP area.
12	(C)	ADDRESS	4	MFSP_OFSP@	Address of the output FSP area.
16	(10)	ADDRESS	4	MFSP_ODFSP_ALET@	Address of ALET for the owning directory FSP.
20	(14)	ADDRESS	4	MFSP_ODFSP@	Address of the owning directory FSP.
24	(18)	ADDRESS	4	MFSP_FID_ALET@	Address of ALET for the File Id.
28	(1C)	ADDRESS	4	MFSP_FID@	Address of the File Id which is a 16 byte area that contains a unique identifier of the file.
32	(20)	ADDRESS	4	MFSP_CRED_ALET@	Address of ALET for CRED structure.
36	(24)	ADDRESS	4	MFSP_CRED@	Address of the CRED structure.

RAUD

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	RAUD	Mapping for RAUD
0	(0)	ADDRESS	4	RAUD_CRED_ALET@	Address of ALET for CRED structure.
4	(4)	ADDRESS	4	RAUD_CRED@	Address of the CRED structure.
8	(8)	ADDRESS	4	RAUD_OFID_ALET@	Address of ALET for the File Id of the old (or only) file.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	RAUD_OFID@	Address of the File Id for the old (or only) file. The File Id is a 16 byte area that contains a unique identifier of the file.
16	(10)	ADDRESS	4	RAUD_OFSP_ALET@	Address of ALET for the FSP for the old (or only) file.
20	(14)	ADDRESS	4	RAUD_OFSP@	Address of the FSP for the old (or only) file.
24	(18)	ADDRESS	4	RAUD_FLAG_ALET@	Address of ALET for the flag byte.
28	(1C)	ADDRESS	4	RAUD_FLAG@	Address of a flag byte that indicates: x'00' - last link was not removed. x'01' - last link was removed for a file. The file is deleted. This parameter is used only on rename and unlink. On a rename, the new name is deleted. On unlink, the 'only' name was deleted.
32	(20)	ADDRESS	4	RAUD_NFID_ALET@	Address of ALET for the File Id of the NEW file. This parameter is used only on a rename.
36	(24)	ADDRESS	4	RAUD_NFID@	Address of the File Id for the NEW file. This parameter is used only on a rename. The File Id is a 16 byte area that contains a unique identifier of the file.
40	(28)	ADDRESS	4	RAUD_NFSP_ALET@	Address of ALET for the FSP for the NEW file. This parameter is used only on a rename.
44	(2C)	ADDRESS	4	RAUD_NFSP@	Address of the FSP for the NEW file. This parameter is used only on a rename.

GUGP

0	(0)	STRUCTURE	48	GUGP	Mapping for GUGP
0	(0)	ADDRESS	4	GUGP_KEY_ALET@	Address of ALET for the user's key.
4	(4)	ADDRESS	4	GUGP_KEY@	Address of a byte that contains the user's key. The key is in the high order 4 bits of the byte.
8	(8)	ADDRESS	4	GUGP_NLEN_ALET@	Address of ALET for the USERID length.
12	(C)	ADDRESS	4	GUGP_NLEN@	Address of 1 byte USERID length.
16	(10)	ADDRESS	4	GUGP_NAME_ALET@	Address of ALET for the USERID.
20	(14)	ADDRESS	4	GUGP_NAME@	Address of 8 byte USERID.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	ADDRESS	4	GUGP_GCNT_ALET@	Address of ALET for the Group Count.
28	(1C)	ADDRESS	4	GUGP_GCNT@	Address of the group count. (A one word area that contains the number of GIDs in the Grouplist area.)
32	(20)	ADDRESS	4	GUGP_GLST_ALET@	Address of ALET for the Group List.
36	(24)	ADDRESS	4	GUGP_GLST@	Address of the Group List area in which supplemental GIDs are returned.
40	(28)	ADDRESS	4	GUGP_GNUM_ALET@	Address of ALET for the number of groups.
44	(2C)	ADDRESS	4	GUGP_GNUM@	Address of the Number Of Groups. A one word area in which the number of supplemental group GIDs is returned.

FORK

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	FORK	Mapping for FORK
0	(0)	ADDRESS	4	FORK_FLAG_ALET@	Address of the ALET for the fork flag.
4	(4)	ADDRESS	4	FORK_FLAG@	Address of a word flag that indicates the current type of processing: x'00' - Fork Parent Processing. x'02' - Fork Parent Processing Extended Data x'01' - Forked Child Processing. x'03' - Forked Child Processing Extended Data
8	(8)	ADDRESS	4	FORK_DATAKEY_ALET@	Address of ALET for the data's storage key.
12	(C)	ADDRESS	4	FORK_DATAKEY@	Address of a word that contains the storage key from which the storage for the data is to be obtained.
16	(10)	ADDRESS	4	FORK_DATALEN_ALET@	Address of ALET for the data length.
20	(14)	ADDRESS	4	FORK_DATALEN@	Address of a word that contains the data length.
24	(18)	ADDRESS	4	FORK_DATA_ALET@	Address of the data ALET.
28	(1C)	ADDRESS	4	FORK_DATA@	Address of the data being passed from the parent to the child via FORK.
32	(20)	ADDRESS	4	FORK_DATASP_ALET@	Address of ALET for the data subpool.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
36	(24)	ADDRESS	4	FORK_DATASP@	Address of a word that contains the subpool from which to obtain storage to hold the data.
MISP					
0	(0)	STRUCTURE	32	MISP	Mapping for MISP
0	(0)	ADDRESS	4	MISP_MODE_ALET@	Address of ALET for the mode.
4	(4)	ADDRESS	4	MISP_MODE@	Address of the mode parameter.
8	(8)	ADDRESS	4	MISP_OISP_ALET@	Address of ALET for the ISP.
12	(C)	ADDRESS	4	MISP_OISP@	Address of the output ISP area.
16	(10)	ADDRESS	4	MISP_OIPC_ALET@	Address of ALET for the output IPCP.
20	(14)	ADDRESS	4	MISP_OIPC@	Address of the IPCP.
24	(18)	ADDRESS	4	MISP_CREI_ALET@	Address of ALET for CREDIPC structure.
28	(1C)	ADDRESS	4	MISP_CREI@	Address of the CREDIPC structure
IACC					
0	(0)	STRUCTURE	24	IACC	Mapping for IACC
0	(0)	ADDRESS	4	IACC_ACODE_ALET@	Address of ALET for the access code.
4	(4)	ADDRESS	4	IACC_ACODE@	Address of a 1 byte access code that contains: x'00' - No access. x'02' - write access. x'04' - read access. x'06' - read and write access
8	(8)	ADDRESS	4	IACC_ISP_ALET@	Address of ALET for the ISP struc.
12	(C)	ADDRESS	4	IACC_ISP@	Address of the ISP structure.
16	(10)	ADDRESS	4	IACC_CREI_ALET@	Address of ALET for CREDIPC struc.
20	(14)	ADDRESS	4	IACC_CREI@	Address of the CREDIPC structure.
IOWN					
0	(0)	STRUCTURE	48	IOWN	Mapping for IOWN
0	(0)	ADDRESS	4	IOWN_CMD_ALET@	Address of ALET for cmd code.
4	(4)	ADDRESS	4	IOWN_CMD@	Address of cmd code.
8	(8)	ADDRESS	4	IOWN_UID_ALET@	Address of ALET for Uid.
12	(C)	ADDRESS	4	IOWN_UID@	Address of the Uid to be set as the ISP owner UID.
16	(10)	ADDRESS	4	IOWN_GID_ALET@	Address of ALET for Gid.
20	(14)	ADDRESS	4	IOWN_GID@	Address of the Gid to be set as the ISP owner GID.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	ADDRESS	4	IOWN_MODE_ALET@	Address of ALET for the mode parameter
28	(1C)	ADDRESS	4	IOWN_MODE@	Address of the mode parameter.
32	(20)	ADDRESS	4	IOWN_ISP_ALET@	Address of ALET for the ISP structure.
36	(24)	ADDRESS	4	IOWN_ISP@	Address of the ISP structure.
40	(28)	ADDRESS	4	IOWN_CREI_ALET@	Address of ALET for CREDIPC structure.
44	(2C)	ADDRESS	4	IOWN_CREI@	Address of the CREDIPC structure.
72	(48)	X'30'	0	IOWN_LEN	"*-IOWN" Constant - fixed length of IOWN.

CKO2

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	CKO2	Mapping for CK02
0	(0)	ADDRESS	4	CKO2_FSP1_ALET@	Address of ALET for the first FSP.
4	(4)	ADDRESS	4	CKO2_FSP1@	Address of the first FSP.
8	(8)	ADDRESS	4	CKO2_FSP2_ALET@	Address of ALET for the second FSP.
12	(C)	ADDRESS	4	CKO2_FSP2@	Address of the second FSP.
16	(10)	ADDRESS	4	CKO2_FID1_ALET@	Address of ALET for the first File ID.
20	(14)	ADDRESS	4	CKO2_FID1@	Address of the first File ID.
24	(18)	ADDRESS	4	CKO2_FID2_ALET@	Address of ALET for the Second File ID.
28	(1C)	ADDRESS	4	CKO2_FID2@	Address of the second File ID.
32	(20)	ADDRESS	4	CKO2_CRED_ALET@	Address of ALET for CRED Structure.
36	(24)	ADDRESS	4	CKO2_CRED@	Address of the CRED structure.

GETE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	56	GETE(0)	Mapping for GETE
0	(0)	ADDRESS	4	GETE_WORKB_ALET@	Address of ALET for 1024 byte work area.
4	(4)	ADDRESS	4	GETE_WORKB@	Address of 1024 byte work area.
8	(8)	ADDRESS	4	GETE_KEY_ALET@	Address of ALET for the user's key
12	(C)	ADDRESS	4	GETE_KEY@	Address of a byte that contains the user's key. The key is in the high order 4 bits of the byte.
16	(10)	ADDRESS	4	GETE_GCNT_ALET@	Address of ALET for the Group Cnt

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	GETE_GCNT@	Address of the group count. (A one word area that contains the number of GIDs in the Grouplist area.)
24	(18)	ADDRESS	4	GETE_GLST_ALET@	Address of ALET for the Group List
28	(1C)	ADDRESS	4	GETE_GLST@	Address of the Group List area in which supplemental GIDs are returned.
32	(20)	ADDRESS	4	GETE_GNUM_ALET@	Address of ALET for the number of groups.
36	(24)	ADDRESS	4	GETE_GNUM@	Address of the Number Of Groups. A one word area in which the number of supplemental group GIDs is returned.
40	(28)	ADDRESS	4	GETE_UID_ALET@	Address of ALET for UIDs.
44	(2C)	ADDRESS	4	GETE_UID@	Address of a 3 word output area. This area contains the real, effective and saved UIDs.
48	(30)	ADDRESS	4	GETE_GID_ALET@	Address of ALET for GIDs.
52	(34)	ADDRESS	4	GETE_GID@	Address of a 3 word output area. This area contains the real, effective and saved GIDs.
DKEY					
0	(0)	STRUCTURE	32	DKEY	Mapping for DKEY callable service
0	(0)	ADDRESS	4	DKEY_FUNC_ALET@	Address of ALET for Function code
4	(4)	ADDRESS	4	DKEY_FUNC@	Address of 1 byte function code
8	(8)	CHARACTER	1	DKEY_ENTITY_ALET@	Address of ALET for the RACF entity
8	(8)	ADDRESS	4	DKEY_USERID_ALET@	Address of ALET for the RACF userid
12	(12)	CHARACTER	1	DKEY_ENTITY@(0)	Address of 247 byte area that contains a 1 byte length followed by a userid of up to 246 characters
12	(12)	ADDRESS	4	DKEY_USERID@	Address of 9 byte area that contains a 1 byte length followed by a userid of up to 8 characters
16	(10)	ADDRESS	4	DKEY_KEY_ALET@	Address of ALET for the DCE key
20	(14)	ADDRESS	4	DKEY_KEY@	Address of the output key area, if retrieving the DCE key, or new key if setting the user's DCE key
24	(18)	ADDRESS	4	DKEY_KEYL_ALET@	Address of ALET for the DCE key length

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	DKEY_KEYL@	Address of fullword containing the DCE key area length
DINF					
0	(0)	STRUCTURE	40	DINF	Mapping for DINF
0	(0)	ADDRESS	4	DINF_FUNC_ALET@	Address of ALET for Function code
4	(4)	ADDRESS	4	DINF_FUNC@	Address of 1 byte function code
8	(8)	ADDRESS	4	DINF_USERID_ALET@	Address of ALET for the RACF userid
12	(C)	ADDRESS	4	DINF_USERID@	Address of 9 byte area that contains a 1 byte length followed by a userid of up to 8 characters
16	(10)	ADDRESS	4	DINF_FIELDL_ALET@	Address of ALET for the Field List
20	(14)	ADDRESS	4	DINF_FIELDL@	Address of an area containing input Field_list
24	(18)	ADDRESS	4	DINF_OUTPUTA_ALET@	Address of ALET for the Output area
28	(1C)	ADDRESS	4	DINF_OUTPUTA@	Address of area which contains the contents of the data obtained
32	(20)	ADDRESS	4	DINF_OUTPUTL_ALET@	Address of ALET for the Output area length
36	(24)	ADDRESS	4	DINF_OUTOUTL@	Address of fullword containing the Output area length
DINF_FIELD_LIST					
0	(0)	STRUCTURE	*	DINF_FIELD_LIST	List of fields to set/get
0	(0)	UNSIGNED	2	DINF_FIELD_LIST_LEN	Len in bytes of field list
2	(2)	UNSIGNED	2	DINF_FIELD_LIST_CNT	Number fields in field list
4	(4)	CHARACTER	*	DINF_FIELD_LIST_ST	Fields start here
DINF_FIELDS					
0	(0)	STRUCTURE	*	DINF_FIELDS	Fields passed from caller
0	(0)	CHARACTER	8	DINF_FIELD_NAME	Name of field
8	(8)	UNSIGNED	2	DINF_FIELD_LEN	Length of field
10	(A)	CHARACTER	*	DINF_FIELD_DATA	Field data
DINF_OUTPUT_AREA					
0	(0)	STRUCTURE	*	DINF_OUTPUT_AREA	Output area from caller
0	(0)	UNSIGNED	2	DINF_OUTPUT_AREA_LEN	Length in bytes of output area
2	(2)	UNSIGNED	2	DINF_OUTPUT_AREA_CNT	Number fields in output data
4	(4)	CHARACTER	*	DINF_OUTPUT_AREA_ST	Output starts here
DINF_OUTPUT_FLD					

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	DINF_OUTPUT_FLD	Data returned for one field
0	(0)	CHARACTER	8	DINF_OUTPUT_FLD_NAME	Name of field
8	(8)	UNSIGNED	2	DINF_OUTPUT_FLD_LEN	Length of this output piece
10	(A)	CHARACTER	*	DINF_OUTPUT_FLD_DATA	Data retrieved

DRUR

0	(0)	STRUCTURE	32	DRUR	Mapping for DRUR
0	(0)	ADDRESS	4	DRUR_FUNC_ALET@	Address of ALET for Function code
4	(4)	ADDRESS	4	DRUR_FUNC@	Address of 1 byte function code
8	(8)	ADDRESS	4	DRUR_PRIN_UUID_ALET@	Address of ALET for the DCE principal UUID
12	(C)	ADDRESS	4	DRUR_PRIN_UUID@	Address of a 36 byte input/output area for the DCE principal UUID

The following two fields address the home cell UUID which is optional. Indicate no home cell UUID passed by having the pointer and alet address a null byte.

16	(10)	ADDRESS	4	DRUR_HOME_UUID_ALET@	Address of ALET for the DCE home cell UUID
20	(14)	ADDRESS	4	DRUR_HOME_UUID@	Address of a 36 byte input/output area for the DCE home cell UUID
24	(18)	ADDRESS	4	DRUR_USERID_ALET@	Address of ALET for the RACF userid
28	(1C)	ADDRESS	4	DRUR_USERID@	Address of a 9 byte input/output area for the RACF userid (1 byte for the length followed by up to 8 bytes for the RACF userid itself)

DAUT

0	(0)	STRUCTURE	36	DAUT	Mapping for DAUT
0	(0)	ADDRESS	4	DAUT_ACEEP@	Address of a full word containing the address of a previously created ACEE (or 0)
4	(4)	ADDRESS	4	DAUT_ALET@	Address of ALET for following variables.
8	(8)	ADDRESS	4	DAUT_PRIN_UUID@	Address of a 36 byte area for the DCE principal UUID
12	(C)	ADDRESS	4	DAUT_HOME_UUID@	Address of a 36 byte area for the DCE home cell UUID
16	(10)	ADDRESS	4	DAUT_USERID@	Address of a 9 byte area for the RACF userid (1 byte for the length followed by up to 8 bytes for the RACF userid itself)

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	DAUT_CLASS_NAME@	Address of an 8 byte area for the RACF class name of a resource. The class name must be left justified and padded to the right with blanks.
24	(18)	ADDRESS	4	DAUT_ENT_NAME@	Address of an area for the RACF resource profile name.
28	(1C)	ADDRESS	4	DAUT_ENT_LEN@	Address of a 2 byte area which contains the length of the resource profile name.
32	(20)	ADDRESS	4	DAUT_ACC_REQ@	Address of a 1 byte area containing the requested access to the resource.

INTA

0	(0)	STRUCTURE	68	INTA	Mapping for INTA
0	(0)	ADDRESS	4	INTA_FUNC@	Address of 1 byte function code
4	(4)	ADDRESS	4	INTA_ATTRIBUTES@	Address of a full word area which contains the attribute flags for the service
8	(8)	ADDRESS	4	INTA_USERID@	Address of 9 byte area for the RACF userid (1 byte for the length followed by up to 8 bytes for the RACF userid itself)
12	(C)	ADDRESS	4	INTA_ACEEP@	Address of a full word input/ output area for the ACEE address.
16	(10)	ADDRESS	4	INTA_APPL_ID@	Address of an 9 byte area which specifies the name of the application being accessed by the user. (1 byte for the length followed by up to 8 bytes for the application name itself)
20	(14)	ADDRESS	4	INTA_PASSWORD@	Address of an 9 byte area containing the user's password or pass ticket. (1 byte for the length followed by up to 8 bytes for the password itself)
24	(18)	ADDRESS	4	INTA_LOGSTRING@	Address of a variable length area containing the log string to be passed to RACROUTE (1 byte for the length followed by up to 255 bytes for the log string itself)
		....		INTA_LAST_PARM_VER1	Variable length parameter list. This was the last parameter for plist version 1

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	INTA_CERTIFICATE@	Address of a variable length area containing the X.509 certificate identifying the user (fullword length followed by the certificate itself. Length of zero indicates certificate not present)
		....		INTA_LAST_PARM_VER2	Variable length parameter list. This was the last parameter for plist version 2
32	(20)	ADDRESS	4	INTA_ENVR_IN@	Address of an area containing the data structure to recreate the security environment
		....		INTA_LAST_PARM_ENVR_IN	Variable length parameter list This could be one of the last parameters for plist ver 3
36	(24)	ADDRESS	4	INTA_ENVR_OUT@	Address of an area containing the data structure to retrieve the security environment
		....		INTA_LAST_PARM_ENVR_OUT	Variable length parameter list This could be one of the last parameters for plist ver 3
40	(28)	ADDRESS	4	INTA_OUTA_STOR@	Address of full word containing the address of an OUSP.
		....		INTA_LAST_PARM_OUTA_STOR	Variable length parameter list. This could be one of the last parameters for plist ver 3
44	(2C)	ADDRESS	4	INTA_X500_NAME_PTR	Address of a full word containing a pointer to the X500 name pair structure. The structure is a 4-Byte structure length, followed by 2 2-byte lengths, followed by up to 255 bytes of issuers name and up to 255 bytes of subjects name.
		....		INTA_LAST_X500_NAME	Variable length parameter list. This could be one of the last parameters for plist ver 4
48	(30)	ADDRESS	4	INTA_VARIABLE_LIST@	Address of an area containing a criteria value data structure. The structure is a 4-byte number of entries, followed by 1 or more entries Each entry is an 8-byte name, followed by a 4-byte value length, followed by the value.
		....		INTA_LAST_PARM_VAR_LIST	Variable length parameter list. This could be one of the last parameters for plist ver 4

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
52	(34)	ADDRESS	4	INTA_SECLABEL@	Address of a 9 byte area containing a security label data structure. The structure is a 1-byte length followed by up to 8 bytes for the security label itself.
56	(38)	ADDRESS	4	INTA_SERVAUTH@	Address of a variable length area for the SERVAUTH resource name. The structure is a 1-byte length followed by up to 64 bytes for the SERVAUTH name itself.
		....		INTA_LAST_PARM_SERVAUTH	Variable length parameter list. This could be one of the last parameters for plist ver 5
60	(3C)	ADDRESS	4	INTA_PHRASE@	Address of a variable length area containing the user's pass phrase, structure is a 1-byte length followed by the pass phrase
		....		INTA_LAST_PARM_PHRASE	Variable length parameter This could be the last parameter
64	(40)	ADDRESS	4	INTA_IDID_AREA@	Address of a fullword containing the address of a variable length area containing distributed identity data (IDID)
		....		INTA_LAST_PARM_IDID	Variable length parameter This could be the last parameter
68	(44)		4	INTA_ACEE2_ALET@	Address of ALET for ACEE 2 parameter.
72	(48)		4	INTA_ACEE2P@	Address of a full word input area for the ACEE 2 address.
76	(4C)		4	INTA_IDTA@	Address of a fullword containing the address of a variable length area containing identity token area (IDTA).
		....		INTA_LAST_PARM_IDTA	Variable length parameter list. This could be the last parameter.
		....		INTA_LAST_PARM	Variable length parameter list. This is the last parameter
Attributes Word Bit Mapping					
0	(0)	STRUCTURE	4	INTA_ATTRIBUTES_MAP	
		1...		INTA_MANAGED	ACEE should be managed (CREATE only)
		.1..		INTA_USP	Initialize a USP for the ACEE (CREATE only)

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		..1.		INTA_TASK_LVL	For CREATE, ON indicates chain ACEE to the TCB. OFF indicates return ACEE address via the Acee_Ptr field. For DELETE, ON indicates get ACEE address from the TCB. OFF indicates ACEE address passed through Acee_Ptr
		...1		INTA_UNAUTH_CLNT	Unauthenticated Client
		.... 1...		INTA_AUTH_CLNT	Authenticated Client
		.... .1..		INTA_MSG_SUPP	ON to suppress RACF messages produced as a result of creating an ACEE
		.... ..1.		INTA_ENVR_RET	ON to return an ENVR object for the ACEE (CREATE only)
		1		INTA_NO_TIMEOUT	ON to create a no timeout managed ACEE (CREATE only)
1	(1)	1...		INTA_OUSP_RET	ON to return an OUSP (CREATE only)
		.1..		INTA_X500_RET	ON to return a X500 name pair (QUERY only)
ADMN					
0	(0)	STRUCTURE	24	ADMN	Mapping for ADMN
0	(0)	ADDRESS	4	ADMN_FUNC@	Address of 1 byte function code
4	(4)	ADDRESS	4	ADMN_PARMLIST@	Address of the function-specific parameter list strcuture
8	(8)	ADDRESS	4	ADMN_USERID@	Address of 9 byte area for the RACF userid under whose authority this service will execute (1 byte for the length followed by up to 8 bytes for the userid itself)
12	(C)	ADDRESS	4	ADMN_ACEEP@	Address of a full word containing the ACEE address under the authority of which this service will execute
16	(10)	ADDRESS	4	ADMN_OUTPUT_SP@	Address of a caller-supplied one byte area containing the subpool in which output messages should be obtained
20	(14)	ADDRESS	4	ADMN_OUTPUT_MSG@	Address of a fullword containing a pointer to the RACF command ouput, which is mapped below. The caller is responsible for freeing this storage.
		1...		ADMN_LAST_PARM	Variable length parameter list. This is the last parameter

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
ADMN_RUNCOMD_MAP					
0	(0)	STRUCTURE	*	ADMN_RUNCOMD_MAP	Mapping for running a command
0	(0)	UNSIGNED	2	ADMN_RUNCOMD_LEN	Length of command string
2	(2)	CHARACTER	*	ADMN_RUNCOMD_CMD	Command String
ADMN_USERADM_MAP					
0	(0)	STRUCTURE	15	ADMN_USRADM_MAP	for Add/Alter/Delete/List user and for Connect/Remove
0	(0)	CHARACTER	9	ADMN_USRADM_USER	User: For Delete user, this is the only required field
0	(0)	UNSIGNED	1	ADMN_USRADM_USER_LEN	Length of User ID
1	(1)	CHARACTER	8	ADMN_USRADM_USER_ID	User ID
9	(9)	CHARACTER	1	*	Reserved
10	(A)	UNSIGNED	2	ADMN_USRADM_ERROFF	Offset to segment/field in error
12	(C)	UNSIGNED	2	ADMN_USRADM_SEG_NUM	Number of segments to update or list
14	(E)	CHARACTER	1	ADMN_USRADM_SEGSTRT	First segment entry
ADMN_USRADM_SEGENTRY					
0	(0)	STRUCTURE	12	ADMN_USRADM_SEGENTRY	Mapping for Segment entry@04A
0	(0)	CHARACTER	8	ADMN_USRADM_SEG_NAME	Profile segment name
8	(8)	CHARACTER	1	ADMN_USRADM_SEG_FLAG	Flag byte for segment
9	(9)	UNSIGNED	2	ADMN_USRADM_FLD_NUM	Number of fields within segment. Set to 0 for the List function
11	(B)	CHARACTER	1	ADMN_USRADM_FLDSTRT	First field entry for segment
ADMN_USRADM_FLDENTRY					
0	(0)	STRUCTURE	*	ADMN_USRADM_FLDENTRY	Mapping for Field entry
0	(0)	CHARACTER	8	ADMN_USRADM_FLD_NAME	Segment field name
8	(8)	CHARACTER	1	ADMN_USRADM_FLD_FLAG	Flag byte for field
9	(9)	UNSIGNED	2	ADMN_USRADM_FLD_LEN	Fld data len
11	(B)	CHARACTER	*	ADMN_USRADM_FLD_DATA	Field data
ADMN_GRPADM_MAP					
0	(0)	STRUCTURE	15	ADMN_GRPADM_MAP	for Add/Alter/Delete/List group
0	(0)	CHARACTER	9	ADMN_GRPADM_GROUP	User: For Delete, this is the only required field
0	(0)	UNSIGNED	1	ADMN_GRPADM_GRP_LEN	Length of Group ID
1	(1)	CHARACTER	8	ADMN_GRPADM_GRP_ID	Group ID
9	(9)	CHARACTER	1	*	Reserved
10	(A)	UNSIGNED	2	ADMN_GRPADM_ERROFF	Offset to segment/field in error

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	UNSIGNED	2	ADMN_GRPADM_SEG_NUM	Number of segments to update
14	(E)	CHARACTER	1	ADMN_GRPADM_SEGSTRT	First segment entry
ADMN_RESADM_MAP					
0	(0)	STRUCTURE	15	ADMN_RESADM_MAP	Mapping for add/alt/del/list resource
0	(0)	CHARACTER	9	ADMN_RESADM_CLASS	Class: not required for data set functions
0	(1)	UNSIGNED	1	ADMN_RESADM_CLAS_LEN	Length of Class Name
1	(1)	CHARACTER	8	ADMN_RESADM_CLAS_NAM	Class Name
9	(9)	CHARACTER	1	*	Reserved
10	(A)	UNSIGNED	2	ADMN_RESADM_ERROFF	Offset to segment/field in error
12	(C)	UNSIGNED	2	DMN_RESADM_SEG_NUM	Number of segments to update
14	(E)	CHARACTER	1	ADMN_RESADM_SEGSTRT	First segment entry
ADMN_SETRADM_MAP					
0	(0)	STRUCTURE	14	ADMN_SETRADM_MAP	Mapping for alter SETROPTS information
0	(0)	CHARACTER	10	*	Reserved
10	(A)	UNSIGNED	2	ADMN_SETRADM_ERROFF	Offset to segment/field in error
12	(C)	UNSIGNED	2	ADMN_SETRADM_SEG_NUM	Number of segments to update
14	(E)	CHARACTER	0	ADMN_SETRADM_SEGSTRT	First segment entry
ADMN_OUTMSG_MAP					
0	(0)	STRUCTURE	17	ADMN_OUTMSG_MAP	R_admin output mapping
0	(0)	ADDRESS	4	ADMN_OUT_NEXT	Addr of next block or 0
4	(4)	CHARACTER	4	ADMN_OUT_EYE	Eye catcher: "RMSG"
8	(8)	UNSIGNED	1	ADMN_OUT_SPID	Subpool ID of this block
9	(9)	UNSIGNED	3	ADMN_OUT_LEN	Total block length
12	(C)	ADDRESS	4	ADMN_OUT_OFF	Offset to first byte after the last message. Offset value is relative to ADMN_OUTMSG_MAP@04A
16	(10)	CHARACTER	1	ADMN_OUT_STRT	First message in block
ADMN_OUT_ENTRY					
0	(0)	STRUCTURE	*	ADMN_OUT_ENTRY	Individual message entry
0	(0)	SIGNED	2	ADMN_OUT_MLEN	Length of this message
2	(2)	CHARACTER	*	ADMN_OUT_MSTR	Variable message string
ADMN_XTRUNL_MAP					
0	(0)	STRUCTURE	14	ADMN_XTRUNL_MAP	R_admin SETROPTS output
0	(0)	CHARACTER	4	ADMN_XTRUNL_EYE	Eye catcher: "RXTR" or "RUNL"

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	UNSIGNED	4	ADMN_XTRUNL_OUTLEN	Total length of output buffer
8	(8)	CHARACTER	4	*	Reserved
12	(C)	UNSIGNED	2	ADMN_XTRUNL_NUM	Number of segment entries for ADMN_XTR_SETR or number of record types for ADMN_UNL_SETR
14	(E)	CHARACTER	0	ADMN_XTRUNL_ENTRY	First segment or record entry
ADMN_UNL_ENTRY					
0	(0)	STRUCTURE	24	ADMN_UNL_ENTRY	Data mapping for ADMN_UNL_SETR
0	(0)	CHARACTER	8	ADMN_UNL_TYPE	SMF Data Unload record type
8	(8)	UNSIGNED	4	ADMN_UNL_LEN	Length of a record of this type
12	(C)	UNSIGNED	4	ADMN_UNL_NUM	Number of records of this type
16	(10)	CHARACTER	8	*	Reserved
24	(18)	CHARACTER	0	ADMN_UNL_RECSTRT	Start of first record of this type
UMAP					
0	(0)	STRUCTURE	32	UMAP	Mapping for UMAP
0	(0)	ADDRESS	4	UMAP_PARM_ALET@	Address of ALET for remaining parameters
4	(4)	ADDRESS	4	UMAP_FUNC@	Address of 2-byte function code
8	(8)	ADDRESS	4	UMAP_OPTION_WORD@	Reserved for future use
12	(C)	ADDRESS	4	UMAP_USERID@	Address of a 9-byte input/output for the RACF userid. The first byte is the length followed by 8 bytes for the RACF userid.
16	(10)	ADDRESS	4	UMAP_CERTIFICATE@	Address of an input area for a digital certificate. First four bytes are a length followed by the digital certificate.
20	(14)	ADDRESS	4	UMAP_APPL_USERID@	Address of an input/output area for an Application ID. The first two bytes contains the length followed by 246 bytes for the Application ID. If not specified on input, length must be zero
		....		UMAP_LAST_PARM	Variable length parameter list (for function codes 1-6).
24	(18)	STRUCTURE	8	UMAP_ID_PROPAGATION	Mapping for ID Propagation portion of UMAP

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	ADDRESS	4	UMAP_DISTINGUISHED_NAME@	Address of an input area that consists of a 2-byte length field followed by the distinguished name (distributed user ID), in UTF-8 format, of up to the maximum length allowed by the RCVT field RCVTDNL (currently 246). If not specified, the length must equal zero.
28	(1C)	ADDRESS	4	UMAP_REGISTRY_NAME@	The name of an area that consists of a 2-byte length field followed by the registry or realm name, in UTF-8 format, of up to the maximum length allowed by the RCVT field RCVTRL (currently 255). If not specified, the length must equal zero.
		....		UMAP_IDPROP_LAST_PARM	Variable length parameter list. (for function code 8)
UMAP_CERTIFICATE					
0	(0)	STRUCTURE	*	UMAP_CERTIFICATE	Digital certificate mapping
0	(0)	SIGNED	4	UMAP_DIGCERT_LEN	Length of digital certificate
4	(4)	CHARACTER	*	UMAP_DIGCERT	Digital certificate
UMAP_USERID					
0	(0)	STRUCTURE	*	UMAP_USERID	RACF userid mapping
0	(0)	UNSIGNED	1	UMAP_RACFID_LEN	Length of RACF user ID
1	(1)	CHARACTER	*	UMAP_RACFID	RACF user ID
UMAP					
0	(0)	STRUCTURE	32	UMAP	Mapping for UMAP
0	(0)	ADDRESS	4	UMAP_PARM_ALET@	Address of ALET for remaining parameters
4	(4)	ADDRESS	4	UMAP_FUNC@	Address of 2-byte function code
UMAP_APPL_USERID					
0	(0)	STRUCTURE	*	UMAP_APPL_USERID	Application ID mapping
0	(0)	UNSIGNED	2	UMAP_APPLID_LEN	Length of Application ID
THIS IS THE LAST PARAMETER FOR FUNCTION CODES 1-6					
2	(2)	CHARACTER	*	UMAP_APPLID	Application ID
UMAP_DISTINGUISHED_NAME_DS					
0	(0)	STRUCTURE	248	UMAP_DISTINGUISHED_NAME_DS	Distinguished name mapping

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	UNSIGNED	2	UMAP_DISTINGUISHED_NAME_LEN	Distinguished name length
2	(2)	CHARACTER	246	UMAP_DISTINGUISHED_NAME	Distinguished name string
UMAP_REGISTRY_NAME_DS					
0	(0)	STRUCTURE	257	UMAP_REGISTRY_NAME_DS	Registry or realm name mapping
0	(0)	UNSIGNED	2	UMAP_REGISTRY_NAME_LEN	Registry or realm name length
THIS IS THE LAST PARAMETER FOR FUNCTION CODE 8					
2	(2)	CHARACTER	255	UMAP_REGISTRY_NAME	Registry or realm name string
CDDL					
0	(0)	STRUCTURE	24	CDDL	Mapping for CDDL
0	(0)	ADDRESS	4	CDDL_FUNC@	Address of 1 byte input function code. Constants for the function codes are supplied below
4	(4)	ADDRESS	4	CDDL_ATTRIBUTES@	Address of full word of input attributes
8	(8)	ADDRESS	4	CDDL_USERID@	Address of a 9 byte input area for the RACF userid (1 byte for the length followed by 8 bytes for the RACF userid itself). A length of 0 indicates the current user is the ring owner.
12	(C)	ADDRESS	4	CDDL_RING@	Address of a variable length input area for the ring name (1 byte for the length followed by up to 237 bytes for the ring name). Parameter ignored for functions CheckStatus, AbortQuery, and DataGetNext.
16	(10)	ADDRESS	4	CDDL_VERSION@	Address of a 4 byte area containing the input version number for function specific parameter list pointed to by CDDL_PARMLIST@. Must be set to 0.
20	(14)	ADDRESS	4	CDDL_PARMLIST@	Address of the input function specific parameter list
		1...		CDDL_LAST_PARM	Last parm in variable length parm list.
CDDL_ATTRIBUTES_MAP					
0	(0)	STRUCTURE	4	CDDL_ATTRIBUTES_MAP	Attribute flags to cause different behaviors in different functions in R_Datalib.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		CDDL_ATT_ALL_KEYTYPES	For functions DataGetFirst and DataGetNext to differentiate between PCICC key type and ICSF key type, DSA key type and PKCS #1 key type, when returning the function specific parameter list field Private_key_type. When this flag is off, R_Datalib treats PCICC key type as an ICSF key type and return value x'00000002', treat DSA key type as a PKCS #1 key type and return value x'00000001'.
		1...		CDDL_ATT_REUSE_RING	For function NewRing to reuse the existing key ring and remove all the certificates from it. When this flag is off, it indicates the creation of a new key ring.
		1...		CDDL_ATT_SET_MIN_SERIAL	For function IncSerialNumber to indicate that the last used serial number field (CERTLUSER) is to be incremented to at least the input serial number.
		1...		CDDL_ATT_TRUST	For function DataPut and DataAlter to set certificate with TRUST status. When this flag is off, it indicates RACF determines the status.
		.1..		CDDL_ATT_HIGHTRUST	For function DataPut and DataAlter to set certificate under CERTAUTH with HIGHTRUST status.
		..1.		CDDL_ATT_NOTRUST	For function DataPut and DataAlter to set certificate with NOTRUST status. When this flag is off, it indicates RACF determines the status.
		1...		CDDL_ATT_DEL_CERT_TOO	For function DataRemove to delete the certificate after it is removed from the ring, if it is not connected to any other rings. When this flag is off, it indicates the certificate is removed from the ring only.
		.1..		CDDL_ATT_DEL_CERT_ALLRINGS	For function DataRemove to delete certificate, even it is connected to another ring.
		..1.		CDDL_ATT_DEL_CERT_FORCE	For function DataRemove to delete certificate, even it is used to generate a request.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		..1.		CDDL_ATT_SKIPAUTH	For function DataGetFirst, DataGetNext, CheckStatus, and GetUpdateCode to indicate RACF authorization check is bypassed for supervisor state or system key caller when request is for a RACF key ring (not applicable to an ICSF token). When this flag is off, it indicates RACF authorization check needs to be performed.

CDDL_ABORT

0	(0)	STRUCTURE	4	CDDL_ABORT	Parameter list for DataAbortQuery
0	(0)	ADDRESS	4	CDDL_PREV_HANDLE	Address of an input/output area from previous DataGetFirst or DataGetNext call.

CDDL_GET_CERT

0	(0)	STRUCTURE	72	CDDL_GET_CERT	Parameter list for DataGetFirst and DataGetNext.
0	(0)	ADDRESS	4	CDDL_RES_HANDLE	Address of input/output area mapped by CDDL_HANDLE_MAP.
4	(4)	CHARACTER	4	CDDL_CERT_USAGE	4 byte output area containing certificate usage flags x'00000000' - Usage is SITE x'00000002' - Usage is CERTAUTH x'00000008' - Usage is PERSONAL x'FFFFFFF5' - reserved bits must be set to zero.
8	(8)	UNSIGNED	4	CDDL_CERT_DEFAULT	Output default indicator. Zero value indicates not default certificate for ring, nonzero indicates this is the default certificate.
12	(C)	UNSIGNED	4	CDDL_CERT_LEN	On input, contains the length of the certificate area pointed to by CDDL_CERT_PTR. On output, contains the actual size of the certificate returned or 0 if no certificate returned.
16	(10)	ADDRESS	4	CDDL_CERT_PTR	Input value specifying address of output certificate data area.
20	(14)	UNSIGNED	4	CDDL_PK_LEN	On input, contains size of private key area pointed to by CDDL_PK_PTR. On output contains the length of the private key returned at address CDDL_PK_PTR or 0 if no private key was returned.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	ADDRESS	4	CDDL_PK_PTR	Input value specifying address of private key output data area.
28	(1C)	UNSIGNED	4	CDDL_PK_TYPE	Output value indicating type of private key. See constants below.
32	(20)	UNSIGNED	4	CDDL_PK_BITSIZE	Output value indicating the size of the private key modules in bits.
36	(24)	UNSIGNED	4	CDDL_LABEL_LEN	On input, contains the length of the field pointed to by CDDL_LABEL_PTR, and must be at least 32. On output, contains the length of the label returned at the address in CDDL_LABEL_PTR, and is 32 or less.
40	(28)	ADDRESS	4	CDDL_LABEL_PTR	Input value specifying the address of the output area to be used for the label name.
44	(2C)	CHARACTER	9	CDDL_RACF_USERID	Input value containing a 1 byte length followed by the certificate owning user ID. The combination of the output label and this field uniquely identify a certificate.
53	(35)	CHARACTER	3	*	Reserved.
56	(38)	UNSIGNED	4	CDDL_SDN_LEN	On input, contains the length of the output buffer pointed to by CDDL_SDN_PTR. On output, contains the length of the BER encoded Subject's Distinguished Name returned in CDDL_SDN_LEN.
60	(3C)	ADDRESS	4	CDDL_SDN_PTR	Input value specifying the address of the output area to be used for the Subjects's Distinguished Name.
64	(40)	UNSIGNED	4	CDDL_RECID_LEN	Output value containing the length of the record ID returned in area pointed to by CDDL_RECID_PTR, or 0 if no record returned.
68	(44)	ADDRESS	4	CDDL_RECID_PTR	Input value specifying the address of a 246 byte area to contain output record ID data.
72	(48)	CHARACTER	4	CDDL_STATUS	Certificate status for input and output. X'80000000' - TRUST. X'40000000' - HIGHTRUST. X'20000000' - NOTRUST. X'00000000' - ANY (input only)

CDDL_CRT_STATUS

0	(0)	STRUCTURE	8	CDDL_CRT_STATUS	Parameter list for CheckStatus

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	UNSIGNED	4	CDDL_CRT_LEN	Address of an input/output area On input, contains the length of the certificate area pointed to by CDDL_CRT_PTR.
4	(4)	ADDRESS	4	CDDL_CRT_PTR	Input value specifying address of output certificate data area
CDDL_GET_UPDATE					
0	(0)	STRUCTURE	4	CDDL_GET_UPDATE	Parm list for GetUpdateCode
0	(0)	UNSIGNED	4	CDDL_RING_SEQN	Ring sequence number
CDDL_ISN					
0	(0)	STRUCTURE	16	CDDL_ISN	Parm List for IncSerialNum
0	(0)	UNSIGNED	4	CDDL_ISN_CLEN	Length of DER Encoded input Digital Certificate
4	(4)	ADDRESS	4	CDDL_ISN_CPTR	Address of DER Encoded input Digital Certificate
8	(8)	CHARACTER	8	CDDL_ISN_SERNUM	Output - Incremented Serial Number of Digital Certificate
CDDL_PUT_CERT					
0	(0)	STRUCTURE	44	CDDL_PUT_CERT	Parm list for DataPut
0	(0)	UNSIGNED	4	CDDL_PCERT_USAGE	A 4 byte input area containing certificate usage flags in the ring. x'00000000' - Usage is SITE. x'00000002' - Usage is CERTAUTH. x'00000008' - Usage is PERSONAL. x'80000000' - Usage is default to the usage of the certificate itself.
4	(4)	UNSIGNED	4	CDDL_PCERT_DEFAULT	A 4 byte input value for default certificate indicator. Zero value indicates this is not default certificate for the ring, nonzero indicates it is.
8	(8)	UNSIGNED	4	CDDL_PCERT_LEN	A 4 byte input value contains the length of the certificate area pointed to by CDDL_PCERT_PTR.
12	(C)	ADDRESS	4	CDDL_PCERT_PTR	An input value specifying address of the input certificate.
16	(10)	UNSIGNED	4	CDDL_PKEY_LEN	A 4 byte input value contains the length of the private key area pointed to be CDDL_PKEY_PTR. Zero indicates no input private key value.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	CDDL_PKEY_PTR	An input value specifying address of the input private key. Ignored if CDDL_PKEY_LEN is zero.
24	(18)	UNSIGNED	4	CDDL_PLABEL_LEN	On input, it contains the length of the label to be assigned to the added certificate. On output, it contains the length of the label of the existing certificate if the input certificate is already installed in RACF.
28	(1C)	ADDRESS	4	CDDL_PLABEL_PTR	On input, it contains the address of the label area. On output, it contains label of the existing certificate if the input certificate is already installed in RACF. The label area must be 32 bytes.
32	(20)	CHARACTER	9	CDDL_PCERT_USERID	On input, it indicates the owner of the certificate, in the format of a 1 byte length followed by the user ID. On output it contains the owner of the existing certificate if the input certificate is already installed in RACF. The 1 byte length must be 8 and the user ID must be left-aligned and padded with blanks.
41	(29)	CHARACTER	3	*	Reserved.
CDDL_REMOVE_CERT					
0	(0)	STRUCTURE	20	CDDL_REMOVE_CERT	Parameter list for DataRemove.
CDDL_RLABEL_LEN					
0	(0)	STRUCTURE	4	CDDL_RLABEL_LEN	A 4 byte input value contains the length of the label of the certificate to be removed pointed by CDDL_RLABEL_PTR.
4	(4)	ADDRESS	4	CDDL_RLABEL_PTR	An input value contains the address of the label of the certificate to be removed.
8	(8)	CHARACTER	9	CDDL_RCERT_USERID	A 9 byte input value indicates the owner of the certificate to be removed, in the format of a 1 byte length followed by the user ID. The 1 byte length must be 8 and the user ID must be left-aligned and padded with blanks.
17	(11)	CHARACTER	3	*	Reserved
CDDL_HANDLE_MAP					

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	20	CDDL_HANDLE_MAP	Mapping of area pointed to by CDDL_RES_HANDLE & CDDL_PREV_HANDLE
0	(0)	ADDRESS	4	CDDL_TOKEN	Reserved for use by the security server. This value must be preserved for subsequent calls to DataGetNext and DataAbortQuery
4	(4)	UNSIGNED	4	CDDL_PREDICATES	Input value specifying the selection criteria. See constants below.
8	(8)	UNSIGNED	4	CDDL_ATTR_ID	Input value specifying the attribute to query on. Ignored if CDDL_PREDICATES is 0. See constants below for possible values.
12	(C)	UNSIGNED	4	CDDL_ATTR_LEN	Input value containing the length of the attributes supplied in CDDL_ATTR_PTR
16	(10)	ADDRESS	4	CDDL_ATTR_PTR	Input value containing the address of query attribute data. Type of data supplied determined by CDDL_ATTR_ID

CDDL_ALTER_CERT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	CDDL_ALTER_CERT	Parameter list for DataAlter
4	(4)	UNSIGNED	4	CDDL_ALABEL_LEN	A 4 byte input value contains the length of the certificate label pointed to by CDDL_ALABEL_PTR.
8	(8)	ADDRESS	4	CDDL_ALABEL_PTR	An input value specifying address of the input certificate label.
12	(C)	UNSIGNED	4	CDDL_ANEW_LABEL_LEN	A 4 byte input value contains the length of the new label area pointed to by CDDL_ANEW_PTR. Zero indicates no input new label value.
16	(10)	ADDRESS	4	CDDL_ANEW_LABEL_PTR	An input value specifying address of the input new label. Ignored if CDDL_ANEW_LABEL_LEN is zero.
20	(14)	CHARACTER	9	CDDL_ACERT_USERID	On input, it indicates the owner of the certificate, in the format of a 1 byte length followed by the user ID. On output, it contains the owner of the existing certificate if the input certificate is already installed in RACF. The 1 byte length must be 8, and the user ID must be left-aligned and padded with blanks.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
29	(1D)	CHARACTER	3	*	Reserved
CDDL_GET_RING					
0	(0)	STRUCTURE	12	CDDL_GET_RING	Parameter list for GetRingInfo
0	(0)	CHARACTER	4	CDDL_RING_SRCH_TYPE	A 4-byte integer input value that identifies more rings to be returned when both ring owner and ring name are specified, based on the following values: X'00000000' - Search for the ring based on thevalues specified by the RACF_user_ID and Ring_name. X'00000001' - Search for all the rings found after the ring is specified by the RACF_user_ID and Ring_name. X'00000002' - Search for all the rings owned by the specified RACF_user_ID find after the ring is specified by RACF_user_ID and Ring_name. X'00000003' - Search for all the rings with the specified Ring_name find after the ring is specified by RACF_user_ID and Ring_name.
4	(4)	UNSIGNED	4	CDDL_RING_RES_LEN	A 4 byte value containing the size of the field pointed to by Ring_result_ptr.
8	(8)	ADDRESS	4	CDDL_RING_RES_PTR	Address of input/output area containing the ring result.
KERBEROS					
0	(0)	STRUCTURE	16	KERB	Mapping for KERB
0	(0)	ADDRESS	4	KERB_FUNC@	Address of 1 byte function code.
4	(4)	ADDRESS	4	KERB_RACF_NAME@	Address of a 9 byte area containing a RACF name preceded by a 1-byte length. x'00' in first byte if not specified.
8	(8)	ADDRESS	4	KERB_NAME@	Address of a 240 byte area for the Kerberos Principal. x'00' in first byte if not specified.
12	(C)	ADDRESS	4	KERB_DATA_AREA@	Address of a caller-supplied area for the field mappings.
		1...		KERB_LAST_PARM	Variable length parameter list. This is the last parameter.
KERB_FLDDATA_MAP					
0	(0)	STRUCTURE	4	KERB_FLDDATA_MAP	Mapping for Field entry
0	(0)	UNSIGNED	2	KERB_FLDDATA_LEN	Length of field struct

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
2	(2)	UNSIGNED	2	KERB_FLD_NUM	Number of fields to update or list
4	(4)	CHARACTER	0	KERB_FLDSTRT	First field entry
KERB_FLDENTRY					
0	(0)	STRUCTURE	*	KERB_FLDENTRY	Mapping for Field entry
0	(0)	CHARACTER	8	KERB_FLD_NAME	Field name
8	(8)	UNSIGNED	2	KERB_FLD_LEN	Field data length
10	(A)	CHARACTER	*	KERB_FLD_DATA	First data
TKTS					
0	(0)	STRUCTURE	28	TKTS	Mapping for TKTS
0	(0)	ADDRESS	4	TKTS_PARM_ALET@	Address of ALET for remaining parameters
4	(4)	ADDRESS	4	TKTS_FUNC@	Address of 2 byte function code.
8	(8)	ADDRESS	4	TKTS_OPTION_WORD@	Address of a fullword containing binary zeros. The area pointed to by this parameter is reserved for future use.
12	(C)	ADDRESS	4	TKTS_CRED_AREA@	Address of an area consisting of a 2 byte length field followed by a ticket, or passticket to be evaluated. For a passticket generation operation, this must be a preallocated buffer which will be filled in with a 2 byte length field and 8 byte passticket value.
16	(10)	ADDRESS	4	TKTS_OPTIONS@	Address of a binary bit string which identifies the function specific processing to be performed. This parameter is unused in the Kerberos return principal function. For passticket function, the bit string is 4 bytes long. If it's value is 1, a passticket is generated and stored in TKTS_CRED_AREA. If the value is 2, the passticket specified in TKTS_CRED_AREA@ is evaluated
20	(14)	ADDRESS	4	TKTS_PRIN_USERID@	For the Kerberos return principal function, this is the address of a pre-allocated storage area containing 242 bytes which is the output principal name preceded by a 2 byte length field. For a passticket operation, this is a 2 byte length field, followed by a 1-8 bytes userid.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(1C)	ADDRESS	4	TKTS_APPLICATION@	Address of area containing 2 byte length followed up to 8 bytes containing the PassTicket application id.
		1...		TKTS_LAST_PARM	Variable length parameter list. This is the last parameter

PGSN

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	16	PGSN	Mapping for PGSN
0	(0)	ADDRESS	4	PGSN_NUM_PARMS@	Address of a fullword containing the total number of parameters included in COMP and PGSN.
4	(4)	ADDRESS	4	PGSN_FUNC@	Address of 2-byte function code. Constants for the function codes are supplied below.
8	(8)	ADDRESS	4	PGSN_FUNC_PARML@	Address of the function-specific parameter list corresponding to the function code. See *z/OS Security Server RACF Callable Services* for function specific parameter lists for callable service R_PgmSignVer.
12	(C)	ADDRESS	4	PGSN_FUNC_ATTRS@	Address of a 4-byte variable that contains the attribute flags for the service.

PKIS

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	PKIS	Mapping for PKIS
0	(0)	ADDRESS	4	PKIS_NUM_PARMS@	Address of a 4-byte variable that contains the number of parameters that follow in the non-request specific portion of the R_PKIServ callable service Parameter List
4	(4)	ADDRESS	4	PKIS_FUNC@	Address of a 2-Byte variable that contains the code of the requested function. Constants for the function codes codes are declared below
8	(8)	ADDRESS	4	PKIS_ATTRIBUTES@	Address of a 4-Byte variable that contains attribute flags for the service
12	(C)	ADDRESS	4	PKIS_LOG_STRING@	Address of a Variable-Length area that contains the LOG string to be passed to RACROUTE (1 byte for the length followed by up to 255 bytes for the LOG string itself)

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	4	PKIS_PARM_VER@	Address of a 4-Byte variable that contains the version number of the Function Specific Parameter List (PKIS_FUNC_PARML@)
20	(14)	ADDRESS	4	PKIS_FUNC_PARML@	Address of the FSPL - Function Specific Parameter List (FSPL = the Parameter List that corresponds to the Function Code)
		....		PKIS_LAST_PARM	Variable length parameter list. This is the last parameter
24	(18)	ADDRESS	4	PKIS_CA_DOMAIN@	Address of the name of the PKI Services certificate authority instance to be invoked.
KERB_FLDENTRY					
0	(0)	STRUCTURE	20	PKIS_GENCERT_MAP	Function Specific Parameter List (FSPL) for GENCERT
0	(0)	CHARACTER	8	PKIS_GENC_EYECATCH	Eyecatcher for the GENCERT FSPL left-aligned blank filled string containing the text 'GENCERT'
8	(8)	UNSIGNED	4	PKIS_GENC_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation PList
12	(C)	ADDRESS	4	PKIS_GENC_CERTPL@	Address of the Digital Certificate generation PList
16	(10)	ADDRESS	4	PKIS_GENC_CERTID@	Address of a 57-byte area that contains the Output Certificate Request ID (the first byte of this area contains the length of the Output Certificate Request ID)
PKIS_CPL_ENTRY					
0	(0)	STRUCTURE	*	PKIS_CPL_ENTRY	Certificate Request PList entry (a single GENCERT PList entry)
0	(0)	CHARACTER	12	PKIS_CPL_FIELD_NAME	Certificate Request Field Name
12	(C)	UNSIGNED	4	PKIS_CPL_FIELD_LEN	Length (in bytes) of the Request Field Name value
16	(10)	CHARACTER	*	PKIS_CPL_FIELD_VALUE	Request Field Name value
PKIS_ATTRIBUTES_MAP					
0	(0)	STRUCTURE	4	PKIS_ATTRIBUTES_MAP	Certificate Request Attribute mapping
		1...		PKIS_SYNCH_CREATE	Synchronous generation flag
0	(0)	BITSTRING	3	*	Reserved
PKIS_EXPORT_MAP					
0	(0)	STRUCTURE	24	PKIS_EXPORT_MAP	Function Specific Parameter List (FSPL) for Export

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	CHARACTER	8	PKIS_EXP_EYECATCH	Eyecatcher for the EXPORT FSPL left-aligned blank filled string containing the text 'EXPORT '
8	(8)	UNSIGNED	4	PKIS_EXP_CERTAN_LEN	Length (in bytes) of the preallocated Certificate Anchor area
12	(C)	ADDRESS	4	PKIS_EXP_CERTAN@	Address of the preallocated Certificate Anchor area
16	(10)	ADDRESS	4	PKIS_EXP_CERTID@	Address of a 57-byte area that contains the Input Certificate Request ID that is used to locate the Digital Certificate to be EXPORTed (the first byte of this area contains the length of the Input Certificate Request ID)
20	(14)	ADDRESS	4	PKIS_EXP_KEYID@	Address of a 41-byte area that contains the KeyID-the hash of the public key generated by PKI Services. The first byte of this area contains the length of the KeyId. This field is used to export a recovery certificate, the only case that the first byte is 40. In all other cases, the length byte should be 0.

PKIS_QUERYREQS_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	36	PKIS_QUERYREQS_MAP	Function Specific Parameter List (FSPL) for QUERYREQS
0	(0)	CHARACTER	8	PKIS_QRYR_EYECATCH	Eyecatcher for QUERYREQS FSPL left-aligned blank filled string containing user specified text, for example, 'QUERYRQS'
8	(8)	UNSIGNED	4	PKIS_QRYR_RESULTL_LEN	Length (in bytes) of the preallocated Results list area
12	(C)	ADDRESS	4	PKIS_QRYR_RESULTL@	Address of the preallocated Results List area
16	(10)	ADDRESS	4	PKIS_QRYR_CERTID@	Address of a 57-byte area that contains the Input Certificate Request ID that is used as a starting point for this query. Only Request IDs located after this Cert ID is returned. The first byte of area contains the length of the Input Certificate Request ID.
20	(14)	UNSIGNED	4	PKIS_QRYR_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Results List area. Zero indicates no limit

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	UNSIGNED	8	PKIS_QRYR_CRIT_STATUS	Value indicating the request status to use as search criteria
28	(1C)	SIGNED	4	PKIS_QRYR_CRIT_DAYS	Value indicating the recent activity to use as additional search criteria
32	(20)	ADDRESS	4	PKIS_QRYR_CRIT_NAME@	Address of a 33-byte area that contains the input requester's name to be used as additional search criteria. The first byte of this area contains the length of the input requester's name.
PKIS_RESL_ENTRY					
0	(0)	STRUCTURE	*	PKIS_RESL_ENTRY	Results List Entry for QUERYREQS or QUERYCERTS (a single entry)
0	(0)	UNSIGNED	1	PKIS_RESL_ENTRY_LEN	Length (in bytes) of the Results List entry value
1	(1)	CHARACTER	*	PKIS_RESL_ENTRY_VALUE	Reserved list entry value
PKIS_REQDETAILS_MAP					
0	(0)	STRUCTURE	28	PKIS_REQDETAILS_MAP	Function Specific Parameter List (FSPL) for REQDETAILS
0	(0)	CHARACTER	8	PKIS_REQD_EYECATCH	Eyecatcher for REQDETAILS FSPL left-aligned blank filled string containing user specified text, for example, 'REQDTAIL'
8	(8)	UNSIGNED	4	PKIS_REQD_SUML_LEN	Length (in bytes) of the preallocated Summary list area
12	(C)	ADDRESS	4	PKIS_REQD_SUML@	Address of the preallocated Summary List area
16	(10)	UNSIGNED	4	PKIS_REQD_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PLIST area
20	(14)	ADDRESS	4	PKIS_REQD_CERTPL@	Address of the pre-allocated Digital Certificate generation type PList details area
24	(18)	ADDRESS	4	KIS_REQD_CERTID@	Address of a 57-byte area that contains the Input Certificate Request ID from which to extract the data (the first byte of this area contains the length of the Input Certificate Request ID)
PKIS					
0	(0)	STRUCTURE	*	PKIS_SUML_ENTRY	Summary List Entry for REQDETAILS CERTDETAILS, or VERIFY (a single entry)

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	UNSIGNED	1	PKIS_SUML_ENTRY_LEN	Length (in bytes) of the Summary List entry value
1	(1)	CHARACTER	*	PKIS_SUML_ENTRY_VALUE	Summary list entry value
PKIS_MODIFYREQS_MAP					
0	(0)	STRUCTURE	44	PKIS_MODIFYREQS_MAP	Function Specific Parameter List (FSPL for MODIFYREQS
0	(0)	CHARACTER	8	PKIS_MODR_EYECATCH	Eyecatcher for MODIFYREQS FSPL left-aligned blank filled string containing user specified text, for example, 'MODREQS'
8	(8)	UNSIGNED	4	PKIS_MODR_ACTION	4 byte binary value indicating the action to be taken against the certificates requests
12	(C)	ADDRESS	4	PKIS_MODR_COMMENT@	Address of a 65-byte area that contains the comment data associated with this action. The first byte of this area contains the length of the actual comment. If the length is x'00' then no comment is recorded.
16	(10)	UNSIGNED	4	PKIS_MODR_CERTIDL_LEN	Length (in bytes) of the Certificate Request id list on input
20	(14)	ADDRESS	4	PKIS_MODR_CERTIDL@	Address of the area containing 1 or more certificate request ids that are to be modified by this request.
24	(18)	UNSIGNED	4	PKIS_MODR_CERTPL_LEN	Length (in bytes) of the Certificate modification plist area
28	(1C)	ADDRESS	4	PKIS_MODR_CERTPL@	Address of the area which is the Certificate modification plist
32	(20)	UNSIGNED	4	PKIS_MODR_ERRL_LEN	Length (in bytes) of the preallocated storage for the Error List output area
36	(24)	ADDRESS	4	PKIS_MODR_ERRL@	Address of the area which contains one or more error results when any of the input requests cannot be modified.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
40	(28)	ADDRESS	4	PKIS_MODR_TSTAMP@	Address of a 20-byte area that contains the time stamp that is associated with the last query upon which the modification request is based. The first byte of the area contains the length of the time stamp value; if the length is x'00' the no time stamp is provided. If a time stamp is provided, the length must be x'13' and the time stamp value must be in the form YYYY/MM/DD hh:mm:ss.
PKIS_QUERYCERTS_MAP					
0	(0)	STRUCTURE	36	PKIS_QUERYCERTS_MAP	Function Specific Parameter List (FSPL) for QUERYCERTS
0	(0)	CHARACTER	8	PKIS_QRYC_EYECATCH	Eyecatcher for QUERYREQS FSPL left-aligned blank filled string containing user specified text, for example, 'QUERYCTS'
8	(8)	UNSIGNED	4	PKIS_QRYC_RESULTL_LEN	Length (in bytes) of the preallocated. Results list area.
12	(C)	ADDRESS	4	PKIS_QRYC_RESULTL@	Address of the preallocated Results List area
16	(10)	ADDRESS	4	PKIS_QRYC_SERIALNUM@	Address of a 17-byte area that contains the Input Certificate Serial Number that is used as a starting point for this query. Only Certificates located after this serial number is returned. The first byte of area contains the length of the Input Certificate Serial Number
20	(14)	UNSIGNED	4	PKIS_QRYC_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Results List area. Zero indicates no limit
24	(18)	UNSIGNED	4	PKIS_QRYC_CRIT_STATUS	Value indicating the request status to use as search criteria
28	(1C)	SIGNED	4	PKIS_QRYC_CRIT_DAYS	Value indicating the recent activity to use as additional search criteria
32	(20)	ADDRESS	4	PKIS_QRYC_CRIT_NAME@	Address of a 33-byte area that contains the input requester's name to be used as additional search criteria. The first byte of this area contains the length of the input requester's name.
PKIS_QUERYCERTS_MAP					

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	PKIS_CERTDETAILS_MAP	Function Specific Parameter List (FSPL) for CERTDETAILS
0	(0)	CHARACTER	8	PKIS_CRTD_EYECATCH	Eyecatcher for CERTDETAILS FSPL left-aligned blank filled string containing user specified text, for example, 'CRTDETLS'
8	(8)	UNSIGNED	4	PKIS_CRTD_SUML_LEN	Length (in bytes) of the preallocated. Summary list area.
12	(C)	ADDRESS	4	PKIS_CRTD_SUML@	Address of the preallocated Summary List area
16	(10)	UNSIGNED	4	PKIS_CRTD_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PLIST area
20	(14)	ADDRESS	4	PKIS_CRTD_CERTPL@	Address of the pre-allocated Digital Certificate generation type PList details area
24	(18)	ADDRESS	4	PKIS_CRTD_SERIALNUM@	Address of a 17-byte area that contains the Input Certificate Serial Number from which to extract the data. The first byte of this area contains the length of the Input Certificate Serial Number.

PKIS_MODIFYCERTS_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	PKIS_MODIFYCERTS_MAP	Function Specific Parameter List (FSPL) for MODIFYCERTS
0	(0)	CHARACTER	8	PKIS_MODC_EYECATCH	Eyecatcher for MODIFYCERTS FSPL left-aligned blank filled string containing user specified text, for example, 'MODCERTS'
8	(8)	UNSIGNED	4	PKIS_MODC_ACTION	4 byte binary value indicating the action to be taken against the certificates identified by the serial number list.
12	(C)	ADDRESS	4	PKIS_MODC_COMMENT@	Address of a 65-byte area that contains the comment data associated with this action. The first byte of this area contains the length of the actual comment. If the length is x'00' then no comment is recorded.
16	(10)	UNSIGNED	4	PKIS_MODC_SERIALNUMSL_LEN	Length (in bytes) of the Serial Numbers list on input.
20	(14)	ADDRESS	4	PKIS_MODC_SERIALNUMSL@	Address of the area containing 1 or more Serial Numbers of certificates that are to be modified by this request.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	UNSIGNED	4	PKIS_MODC_REASON	4 byte binary value indicating the reason for certificate revocation
28	(1C)	ADDRESS	4	PKIS_MODC_REQUESTOREMAIL@	Address of a 33-byte area that contains the new email address of the requester. The first byte of this area contains the length of the email address.
32	(20)	UNSIGNED	4	PKIS_MODC_ERRL_LEN	Length (in bytes) of the preallocated storage for the Error List output area
36	(24)	ADDRESS	4	PKIS_MODC_ERRL@	Address of the area which contains one or more error results when any of the input certificates cannot be modified.

PKIS_QRECOVER_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	PKIS_QRECOVER_MAP	Mapping for QRECOVER function specific parameter list
0	(0)	CHARACTER	8	PKIS_QREC_EYECATCH	Eyecatcher, 8 characters left-aligned blank filled. Actual value set by invoker, for example, 'QRECOVER'
8	(8)	UNSIGNED	4	PKIS_QREC_RESULTL_LEN	Length (in bytes) of the preallocated Results List area
12	(C)	ADDRESS	4	PKIS_QREC_RESULTL@	Address of the preallocated Results List area
16	(10)	UNSIGNED	4	PKIS_QREC_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Results List area. Zero indicates no limit
20	(14)	ADDRESS	4	PKIS_QREC_CRIT_EMAIL@	Address of a 33-byte area that contains the input requester's email address as search criteria
24	(18)	ADDRESS	4	PKIS_QREC_CRIT_PASS@	Address of a 33-byte area that contains the input pass phrase as search criteria

PKIS_SNID_ENTRY

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	PKIS_SNID_ENTRY	Certificate ID or Serial Number (a single entry)
0	(0)	UNSIGNED	1	PKIS_SNID_ENTRY_LEN	Length (in bytes) of the Cert ID or Serial Number entry value
1	(1)	CHARACTER	*	PKIS_SNID_ENTRY_VALUE	Cert ID or Serial Number entry value

PKIS_VERIFY_MAP

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	32	PKIS_VERIFY_MAP	Function Specific Parameter List (FSPL) for VERIFY
0	(0)	CHARACTER	8	PKIS_VERF_EYECATCH	Eyecatcher for VERIFY FSPL left-aligned blank filled string containing user specified text, for example, 'VERIFY '
8	(8)	UNSIGNED	4	PKIS_VERF_SUML_LEN	Length (in bytes) of the preallocated Summary list area
12	(C)	ADDRESS	4	PKIS_VERF_SUML@	Address of the preallocated Summary List area
16	(10)	UNSIGNED	4	PKIS_VERF_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PLIST area
20	(14)	ADDRESS	4	PKIS_VERF_CERTPL@	Address of the pre-allocated Digital Certificate generation type PList details area
24	(18)	UNSIGNED	4	PKIS_VERF_CERT_LEN	Length (in bytes) of the Cert area containing the certificate to verify
28	(1C)	ADDRESS	4	PKIS_VERF_CERT@	Address of the area containing the Certificate to verify

PKIS_REVOKE_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	16	PKIS_REVOKE_MAP	Function Specific Parameter List (FSPL) for REVOKE
0	(0)	CHARACTER	8	PKIS_REVK_EYECATCH	Eyecatcher for REVOKE FSPL left-aligned blank filled string containing user specified text, for example, 'REVOKE '
8	(8)	UNSIGNED	4	PKIS_REVK_REASON	4 byte binary value indicating the reason for the certificate revocation
12	(C)	ADDRESS	4	PKIS_REVK_SERIALNUM@	Address of a 17-byte area that contains the Input Certificate Serial Number to be revoked. The first byte of this area contains the length of the Input Certificate Serial Number.

PKIS_RENEW_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	24	PKIS_RENEW_MAP	Function Specific Parameter List (FSPL) for GENRENEW and REQRENEW
0	(0)	CHARACTER	8	PKIS_RENW_EYECATCH	Eyecatcher for the GENRENEW FSPL and the REQRENEW FSPL left-aligned blank filled string containing the text 'RENEW '

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	UNSIGNED	4	PKIS_RENW_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation PList
12	(C)	ADDRESS	4	PKIS_RENW_CERTPL@	Address of the Digital Certificate generation PList
16	(10)	ADDRESS	4	PKIS_RENW_CERTID@	Address of a 57-byte area that contains the Output Certificate Request ID (the first byte of this area contains the length of the Output Certificate Request ID)
20	(14)	ADDRESS	4	PKIS_RENW_SERIALNUM@	Address of a 17-byte area that contains the input Certificate Serial Number to be renewed. The first byte of this area contains the length of the Input Certificate Serial Number
PKIS_RESPOND_MAP					
0	(0)	STRUCTURE	24	PKIS_RESPOND_MAP	Mapping for RESPOND function specific parameter list
0	(0)	CHARACTER	8	PKIS_RESP_EYECATCH	Eyecatcher, 8 characters left-aligned blank filled. Actual value set by invoker, for example, 'RESPOND'
8	(8)	UNSIGNED	4	PKIS_RESP_RESPONSE_LEN	4 byte area which is the length of the preallocated storage of the response area on input to RESPOND
12	(C)	ADDRESS	4	PKIS_RESP_RESPONSE@	The address of the storage area in which the R_PKIServ service stores the results of the RESPOND if the service was able to successfully retrieve the data
16	(10)	UNSIGNED	4	PKIS_RESP_REQUEST_LEN	Length of the Request area containing the request to verify
20	(14)	ADDRESS	4	PKIS_RESP_REQUEST@	Pointer to the area containing the request to verify
PKIS_SCEPREQ_MAP					
0	(0)	STRUCTURE	24	PKIS_SCEPREQ_MAP	Mapping for SCEPREQ function specific parameter list
0	(0)	CHARACTER	8	PKIS_SCEP_EYECATCH	Eyecatcher, 8 characters left-aligned blank filled. Actual value set by invoker, for example, 'SCEPREQ'
8	(8)	UNSIGNED	4	PKIS_SCEP_RESPONSE_LEN	4 byte area which is the length of the preallocated storage of the response area on input to SCEPREQ

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	PKIS_SCEP_RESPONSE@	The address of the storage area in which the R_PKIServ service stores the results of the SCEPREQ if the service was able to successfully retrieve the data
16	(10)	UNSIGNED	4	PKIS_SCEP_REQUEST_LEN	Length of the Request area containing the request to process
20	(14)	ADDRESS	4	PKIS_SCEP_REQUEST@	Pointer to the area containing the request to process

PKIS_PREREG_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	20	PKIS_PREREG_MAP	Mapping for PREREGISTER function specific parameter list
0	(0)	CHARACTER	8	PKIS_PREG_EYECATCH	Eyecatcher, 8 characters left-aligned blank filled. Actual value set by invoker, for example, 'PREREG'
8	(8)	UNSIGNED	4	PKIS_PREG_CERTPL_LEN	Length in bytes of the preregistration plist
12	(C)	ADDRESS	4	PKIS_PREG_CERTPL@	The address of the preregistration plist
16	(10)	ADDRESS	4	PKIS_PREG_CERTID@	Address of a 57 byte area in which the first byte contains the actual length on return of the certificate request ID

CACH

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	88	CACH	Mapping for CACH
0	(0)	ADDRESS	4	CACH_PARM_ALET@	Address of ALET for remaining parameters not including the ACEE_ALET and ACEE parameters.
4	(4)	ADDRESS	4	CACH_NUM_PARMS@	Address of fullword containing the number of remaining parameters including this one
8	(8)	ADDRESS	4	CACH_FUNC@	Address of 2 byte input function code. Constants for the function codes are supplied below
12	(C)	ADDRESS	4	CACH_OPTION@	Address of 2 byte field containing an option value that further defines the function.
16	(10)	ADDRESS	4	CACH_VERSION@	Address of the data field containing the version (level) of the cache
20	(14)	ADDRESS	4	CACH_VERSION_LEN@	Address of a fullword containing the length of the version field

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	ADDRESS	4	CACH_NAME@	Address of a data field containing the cachename identifying the cache
28	(1C)	ADDRESS	4	CACH_RECNAME_PTR@	Address of a variable length field containing the name of the record to be added or fetched
32	(20)	ADDRESS	4	CACH_RECNAME_LEN@	Address of a fullword containing the length of the record name
36	(24)	ADDRESS	4	CACH_DATA_PTR@	Address of the pointer to the data associated with this record
40	(28)	ADDRESS	4	CACH_DATA_LEN@	Address of a fullword containing the length of the data
44	(2C)	ADDRESS	4	CACH_DATA_TIMEOUT@	Address of a fullword containing the number of seconds before the data times out
48	(30)	ADDRESS	4	CACH_SRC_PTR@	Address of the pointer to a source record
52	(34)	ADDRESS	4	CACH_SRC_LEN@	Address of a fullword containing the length of a source record
56	(38)	ADDRESS	4	CACH_REF_TIMEOUT@	Address of a fullword containing the number of seconds before the reference times out
60	(3C)	ADDRESS	4	CACH_REF_USERID@	Address of an 8 byte area containing a reference userID
64	(40)	ADDRESS	4	CACH_REF@	Address of an 8 byte area containing a reference value
68	(44)	ADDRESS	4	CACH_SUBPOOL@	Address of a 1 byte area containing the subpool for output areas
72	(48)	ADDRESS	4	CACH_ACEE_ALET@	Address of ALET for the ACEE parameter
76	(4C)	ADDRESS	4	CACH_ACEE@	Address of an ACEE
80	(50)	ADDRESS	4	CACH_ICRX@	Address of a fullword containing an ICRX address.
84	(54)	ADDRESS	4	CACH_ICRX_LEN@	Address of a fullword containing the length of the ICRX.

PRXY

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	44	PRXY	Mapping for PRXY
0	(0)	ADDRESS	4	PRXY_PARM_ALET@	Address of ALET for remaining parameters
4	(4)	ADDRESS	4	PRXY_FUNC@	Address of 2 byte input function code. Constants for the function codes are supplied below

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	PRXY_LDAP_HOST@	Address of an area containing a 4 byte length followed by an EBCDIC URL for the LDAP BIND
12	(C)	ADDRESS	4	PRXY_BIND_DN@	Address of an area containing a 4 byte length followed by an EBCDIC DN for the LDAP bind
16	(10)	ADDRESS	4	PRXY_BIND_PW@	Address of an area containing a 4 byte length followed by an EBCDIC password for the LDAP bind
20	(14)	ADDRESS	4	PRXY_USERID@	Address of a 9 byte area containing a 1 byte length followed by up to 8 EBCDIC characters for a host user ID
24	(18)	ADDRESS	4	PRXY_USERDN@	Address of an area containing a 4 byte length followed by an EBCDIC string naming the base DN of an LDAP subtree
28	(38)	ADDRESS	4	PRXY_RESULTS@	Address of a pointer to the results
		1...		PRXY_LAST_PARM0	Last parameter for the original version of the variable length parameter list.
32	(20)	ADDRESS	4	PRXY_FUNC_VERSION@	Address of a 4 byte version number for the function specific parm list
36	(24)	ADDRESS	4	PRXY_FUNC_PARMLIST@	Address of the function specific parameter list.
40	(28)	ADDRESS	4	PRXY_LDAP_ERROR@	Address of an area where an LDAP error message can be returned.
		1...		PRXY_LAST_PARM1	Last parameter for first update of the variable length parameter list.

PRXY_F3_PLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	PRXY_F3_PLIST	Function specific parameter list for PRXY function 3.
0	(0)	UNSIGNED	0	PRXY_F3_OPTYPE	Operation type: X'00' - Add X'01' - Delete X'02' - Modify
1	(1)	BITSTRING	1	PRXY_F3_FLAGS	Request flags.
		1...		PRXY_F3_PWUPD	Reserved for use by the security product. Not for application use. This bit should be set to zero by applications using this interface.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		PRXY_F3_PWUPD2	Reserved for use by the security product. Not for application use. This bit should be set to zero by applications using this interface.
		..1.		PRXY_F3_PWUPD3	Reserved for use by the security product. Not for application use. This bit should be set to zero by applications using this interface.
		...1 1111			Reserved for future use. These bits must be set to 0.
2	(2)	CHARACTER	8	PRXY_F3_CLASS	RACF class name, padded to the right with blanks. Currently, the class names supported are USER, GROUP, CONNECT, and any general resource class except DATASET.
10	(A)	UNSIGNED	2	PRXY_F3_PROFLEN	Length of profile being changed. Must adhere to length requirements for the class name in PRXY_F3_CLASS.
12	(C)	ADDRESS	4	PRXY_F3_PROFNAME@	Address of profile name being added, altered, or deleted. When PRXY_F3_CLASS is CONNECT, the profile name takes the format of <user>. <group>
16	(10)	CHARACTER	8	PRXY_F3_INITIATOR	The user ID who initiated the RACF profile change. If this field contains binary zeros, then RACF uses the identity of the caller.
24	(18)	CHARACTER	22	PRXY_F3_DATETIME	The GMT time of the update in the format: yyyymmddhhiiss.uuuuuuZ where yyyy is the year mm is the month dd is the day hh is the hours ii is the minutes ss is the seconds uuuuuu is the microseconds Z is constant If this field contains binary zeros, R_Proxyserv generates the date and time for you.

RACL

0	(0)	STRUCTURE	40	RACL	Mapping for RACL

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	4	RACL_EDIT_ALET@	Address of the ALET for the EDIT structure
4	(4)	ADDRESS	4	RACL_EDIT@	Address of the EDIT structure
8	(8)	ADDRESS	4	RACL_EDITLEN_ALET@	Address of the ALET for the EDIT structure length
12	(C)	ADDRESS	4	RACL_EDITLEN@	Address of the length of the RACL_Edit structure
16	(10)	ADDRESS	4	RACL_FSP_ALET@	Address of the ALET for the FSP
20	(14)	ADDRESS	4	RACL_FSP@	Address of the FSP structure
24	(18)	ADDRESS	4	RACL_FID_ALET@	Address of the ALET for the File ID
28	(1C)	ADDRESS	4	RACL_FID@	Address of the File ID which is a 16-byte area that contains a unique ID of the file
32	(20)	ADDRESS	4	RACL_CRED_ALET@	Address of the ALET for the CRED structure
36	(24)	ADDRESS	4	RACL_CRED@	Address of the CRED structure
		1...		RACL_LAST_PARM	Variable length parameter list This is the last parameter
RACL_EDIT					
0	(0)	STRUCTURE	4	RACL_EDIT	Function Specific Parameter List (FSPL) for EDIT
0	(0)	UNSIGNED	2	RACL_EDIT_OPTYPE	Operation Type
2	(2)	UNSIGNED	2	RACL_EDIT_ACLTYPE	ACL Type
4	(4)	CHARACTER	0	RACL_EDIT_ACL	An ACL structure containing entries to be modified
RAUX					
0	(0)	STRUCTURE	76	RAUX	Mapping for callable service R_AUDITX
0	(0)	ADDRESS	4	RAUX_NUM_PARMS@	Address of 4-byte area containing the number of parameters in the parameter list.
4	(0)	ADDRESS	4	RAUX_ACEE_ALET@	Name of 4-byte area containing the ALET for the ACEE parameter.
8	(8)	ADDRESS	4	RAUX_ACEE@	Name of area containing the ACEE belonging to the RACF user that appears in the log record.
12	(C)	ADDRESS	4	RAUX_PARM_ALET@	Name of 4-byte area containing the ALET for remaining parameters in parameter list.
16	(10)	ADDRESS	4	RAUX_OPTION_WORD@	Name of 4-byte area containing binary zeros.

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	RAUX_LINK_VALUE@	Name of 8-byte area containing value used to mark related SMF records.
24	(18)	ADDRESS	4	RAUX_ATTRIBUTES@	Name of 4-byte area containing flags set by caller.
28	(1C)	ADDRESS	4	RAUX_COMPONENT@	Name of area that consists of a 4-byte length field followed by character data.
32	(20)	ADDRESS	4	RAUX_FMID@	Name of 7-byte area containing FMID.
36	(24)	ADDRESS	4	RAUX_SUBTYPE@	Name of 4-byte integer with SMF type 83 record subtype
40	(28)	ADDRESS	4	RAUX_EVENT@	Name of 4-byte integer caller initializes with event code
44	(2C)	ADDRESS	4	RAUX_QUALIFIER@	Name of 4-byte integer caller initializes with event code qualifier
48	(30)	ADDRESS	4	RAUX_CLASS@	Name of 8-byte area containing a RACF class name
52	(34)	ADDRESS	4	RAUX_RESOURCE@	Name of area that consists of 4-byte length field followed by resource name.
56	(38)	ADDRESS	4	RAUX_LOG_STRING@	Address of an area containing a 4 byte length followed by text to be written to the SMF record
60	(3C)	ADDRESS	4	RAUX_RELOCATE_COUNT@	Name of 4-byte area containing number of relocate sections
64	(40)	ADDRESS	4	RAUX_RELOCATE_PTR@	Name of area containing address of array of relocate sections
68	(44)	ADDRESS	4	RAUX_MESSAGE_COUNT@	Name of 4-byte integer containing number of message segments
72	(48)	ADDRESS	4	RAUX_MESSAGE_PTR@	Name of area containing zero or address of any array

WPRV

0	(0)	STRUCTURE	12	WPRV	Mapping for callable service R_WRITEPRIV
0	(0)	ADDRESS	4	WPRV_FUNC@	Address of the function byte
4	(4)	ADDRESS	4	WPRV_ACEE_ALET@	Address of the ALET for the ACEE structure
8	(8)	ADDRESS	4	WPRV_ACEE@	Address of the ACEE structure

SECL

0	(0)	STRUCTURE	24	SECL	Mapping for callable service R_SETFSECL

Table 26. Structure COMP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	4	SECL_FSP_ALET@	Address of the ALET for the FSP structure
4	(4)	ADDRESS	4	SECL_FSP@	Address of the FSP structure
8	(8)	ADDRESS	4	SECL_FID_ALET@	Address of the ALET for the File ID
12	(C)	ADDRESS	4	SECL_FID@	Address of the File ID which is a 16-byte area that contains a unique ID of the file
16	(10)	ADDRESS	4	SECL_CRED_ALET@	Address of the ALET for the CRED structure
20	(14)	ADDRESS	4	SECL_CRED@	Address of the CRED structure
SMOP					
0	(0)	STRUCTURE	40	SMOP	Address of fullword containing the total number of parameters.
0	(0)	ADDRESS	4	SMOP_NUM_PARMS@	Address of fullword containing function code.
4	(4)	ADDRESS	4	SMOP_FUNC@	Address of the 4 byte options value.
					x'00000001' - Execute x'00000002' - Precheck x'00000004' - Stop on error (unsupported in RACF) x'00000008' - Clear Sensitive information. These options can be combined.
8	(8)	ADDRESS	4	SMOP_OPTIONS@	Address of fullword containing the request length, in bytes.
12	(C)	ADDRESS	4	SMOP_REQUEST_LEN@	Address of the request.
16	(10)	ADDRESS	4	SMOP_REQUEST@	Address of 64 byte handle.
20	(14)	ADDRESS	4	SMOP_HANDLE@	Address of 9 byte area for the RACF user ID (1-byte for the length followed by up to 8 bytes for the RACF user ID itself)
24	(18)	ADDRESS	4	SMOP_USERID@	Address of input ACEE.
28	(1C)	ADDRESS	4	SMOP_ACEE@	Address of fullword containing the length of the RESULT buffer.
32	(20)	ADDRESS	4	SMOP_RESULT_LEN@	Address of the result buffer.
36	(24)	ADDRESS	4	SMOP_RESULT@	

COMP constants

Table 27. Constants for Comp

Len	Type	Value	Name	Description
4	DECIMAL	28	COMP_LEN	Constant COMP length
4	DECIMAL	8	IUSP_LEN	Constant IUSP length
4	DECIMAL	16	CSID_LEN	Constant CSID length
4	DECIMAL	40	EXID_LEN	Constant EXID length
4	DECIMAL	28	GINF_LEN	Constant GINF length
4	DECIMAL	1	GINF_FUNC_EIM	Get EIM information
2	DECIMAL	1	GINF_OPT1	Varies with function
2	DECIMAL	2	GINF_OPT2	Varies with function
2	DECIMAL	3	GINF_OPT3	Varies with function
4	DECIMAL	32	GETG_LEN	Constant GETG length
4	DECIMAL	8	CHKP_LEN	Constant CHKP length
4	DECIMAL	24	GMAP_LEN	Constant GMAP length
4	DECIMAL	32	CKPO_LEN	Constant CKPO length
4	DECIMAL	1	CKPO_KILL_AUDIT	KILL AUDIT Constant
4	DECIMAL	2	CKPO_KIL	KILL Verification
4	DECIMAL	3	CKPO_GETPSENT	Caller is GETPSENT
4	DECIMAL	4	CKPO_OPEN_STTY	Open Subsidiary TTY
4	DECIMAL	5	CKPO_SIGQUEUE	Sigqueue verification
4	DECIMAL	5	CKPO_MAX_REQ_TYPE	Max Request Type
4	DECIMAL	16	QRYS_LEN	Constant QRYS length
4	DECIMAL	32	CMOD_LEN	Constant CMOD length
4	DECIMAL	24	CLID_LEN	Constant CLID length
4	DECIMAL	32	CAUD_LEN	Constant CAUD length
4	DECIMAL	40	COWN_LEN	Constant COWN length
4	DECIMAL	8	UMSK_LEN	Constant UMSK length
4	DECIMAL	40	KACC_LEN	Constant KACC length
4	DECIMAL	16	QRYF_LEN	Constant QRYF length
4	DECIMAL	24	KFOW_LEN	Constant KFOW length
4	DECIMAL	32	MKRT_LEN	Constant MKRT length
4	DECIMAL	24	PTRC_LEN	Constant PTRC length
4	DECIMAL	40	MFSP_LEN	Constant MFSP length
4	DECIMAL	48	RAUD_LEN	Constant RAUD length
4	DECIMAL	48	GUGP_LEN	Constant GUGP length
4	DECIMAL	40	FORK_LEN	Constant FORK length
4	DECIMAL	0	FORK_PARENT	Fork Parent Processing flag

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
4	DECIMAL	2	FORK_PARENTX	Fork Parent Processing Extended Data flag
4	DECIMAL	1	FORK_CHILD	Fork Child Processing flag
4	DECIMAL	3	FORK_CHILDX	Fork Child Processing Extended Data flag
4	DECIMAL	32	MISP_LEN	Constant MISP length
4	DECIMAL	24	IACC_LEN	Constant IACC length
4	DECIMAL	48	IOWN_LEN	Constant IOWN length
4	DECIMAL	40	CKO2_LEN	Constant CKO2 length
4	DECIMAL	56	GETE_LEN	Constant GETE length
4	DECIMAL	32	DKEY_LEN	Constant DKEY length
1	HEX	01	DKEY_GET	Get DCE key
1	HEX	02	DKEY_PUT	Put DCE key
1	HEX	03	DKEY_GET_LDAP_PW	Get LDAP password
4	DECIMAL	40	DINF_LEN	Constant DINF length
1	HEX	01	DINF_GET	Get the DCE info
1	HEX	02	DINF_PUT	Put the DCE info
4	DECIMAL	32	DRUR_LEN	Constant DRUR length
1	HEX	01	DRUR_GET_RACF_USERID	UUID->RACF userid
1	HEX	02	DRUR_GET_DCE_UUID	RACF userid->UUID
4	DECIMAL	36	DAUT_LEN	Constant DAUT length
1	HEX	02	DAUT_READ	READ access req
1	HEX	04	DAUT_UPDATE	UPDATE access req
1	HEX	08	DAUT_CONTROL	CONTORL access req
1	HEX	80	DAUT_ALTER	ALTER access req
INTA function code values				
1	DECIMAL	1	INTA_CREATE	Create an ACEE
1	DECIMAL	2	INTA_DELETE	Delete the ACEE and USP if applicable
4	DECIMAL	68	INTA_LEN	Constant INTA length
1	DECIMAL	3	INTA_PURGE	Purge space related ACEE management resources
1	DECIMAL	4	INTA_REGSTR	Register certificate for the current userid
1	DECIMAL	5	INTA_DEREGS	Deregister certificate for the current userid
1	DECIMAL	6	INTA_QUERY	Query a certificate for an associated user ID

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
1	DECIMAL	7	INTA_GENIDT	Generate an IDT from an ACEE
ADMIN function code values				
1	DECIMAL	1	ADMN_ADD_USER	Add a user
1	DECIMAL	2	ADMN_DEL_USER	Delete a user
1	DECIMAL	3	ADMN_ALT_USER	Alter a user
1	DECIMAL	4	ADMN_LST_USER	List a user
1	DECIMAL	5	ADMN_RUN_COMD	Run a RACF command
1	DECIMAL	6	ADMN_ADD_GROUP	Add a group
1	DECIMAL	7	ADMN_DEL_GROUP	Delete a group
1	DECIMAL	8	ADMN_ALT_GROUP	Alter a group
1	DECIMAL	9	ADMN_LST_GROUP	List a group
1	DECIMAL	10	ADMN_CONNECT	Connect user to group
1	DECIMAL	11	ADMN_REMOVE	Remove user from group
1	DECIMAL	12	ADMN_ADD_GENRES	Add a resource
1	DECIMAL	13	ADMN_DEL_GENRES	Delete a resource
1	DECIMAL	14	ADMN_ALT_GENRES	Alter a resource
1	DECIMAL	15	ADMN_LST_GENRES	List a resource
1	DECIMAL	16	ADMN_ADD_DS	Add a data set
1	DECIMAL	17	ADMN_DEL_DS	Delete a data set
1	DECIMAL	18	ADMN_ALT_DS	Alter a data set
1	DECIMAL	19	ADMN_LST_DS	List a data set
1	DECIMAL	20	ADMN_PERMIT	Permit a user or group
1	DECIMAL	21	ADMN_ALT_SETR	Alter SETROPTS info
1	DECIMAL	22	ADMN_XTR_SETR	SETROPTS R_admin ext
1	DECIMAL	23	ADMN_UNL_SETR	SETROPS SMF unload ext
4	DECIMAL	24	ADMN_XTR_PWENV	Extract PKCS #7 encrypted password envelope
4	DECIMAL	25	ADMN_XTR_USER	Extract a user profile
4	DECIMAL	26	ADMN_XTR_NEXT_USER	Extract the next user profile
4	DECIMAL	27	ADMN_XTR_GROUP	Extract a group profile
4	DECIMAL	28	ADMN_XTR_NEXT_GROUP	Extract the next group profile
4	DECIMAL	29	ADMN_XTR_CONNECT	Extract the connect profile
4	DECIMAL	30	ADMN_XTR_PPENV	Extract PKCS #7 encrypted password phrase envelope
4	DECIMAL	31	ADMN_XTR_RESOURCE	Extract a general resource profile

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
4	DECIMAL	32	ADMN_XTR_NEXT_RESOURCE	Extract the next general resource profile
4	DECIMAL	33	ADMN_XTR_RRSF	Extract RRSF settings and node definition
4	DECIMAL	34	ADMN_XTR_DATASET	Extract a data set profile
4	DECIMAL	35	ADMN_XTR_NEXT_DATASET	Extract the next data set profile
4	CHARACTER	RMSG	ADMN_OUT_RMSG	eyecatcher in ADMN_OUT_EYE
4	CHARACTER	RXTR	ADMN_XTRUNL_RXTR	eyecatcher in ADMN_XTRUNL_EYE for SETROPTS extract
4	CHARACTER	RUNL	ADMN_XTRUNL_RUNL	eyecatcher in ADMN_XTRUNL_EYE for SETROPTS extract
4	CHARACTER	RXPW	ADMN_OUT_RXPW	eyecatcher in ADMN_XTRPW_EYE for password
4	CHARACTER	RXPP	ADMN_OUT_RXPP	eyecatcher in ADMN_XTRPP_EYE for password phrase
4	CHARACTER	PXTR	ADMN_PROF_PXTR	eyecatcher in ADMN_PROF_EYE
UMAP function codes				
4	DECIMAL	24	UMAP_LEN	fixed length of UMAP
2	DECIMAL	1	UMAP_R_TO_L	Return Lotus Notes® ID for this RACF ID
2	DECIMAL	2	UMAP_L_TO_R	Return RACF ID for this Lotus Notes ID
2	DECIMAL	3	UMAP_R_TO_N	Return NDS ID for this RACF ID
2	DECIMAL	4	UMAP_N_TO_R	Return RACF ID for this NDS ID
2	DECIMAL	5	UMAP_R_TO_K	Return Kerb principal for this RACF ID
2	DECIMAL	6	UMAP_K_TO_R	Return RACF ID for this RACF ID
2	DECIMAL	8	UMAP_DID_TO_R	Return the RACF ID that is mapped by this combination of Distinguished Name and Registry/Realm Name
2	DECIMAL	9	UMAP_R_TO_E	Return e-mail address for this RACF ID
2	DECIMAL	10	UMAP_E_TO_R	Return RACF ID for this e-mail address

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
CDDL function code constants				
1	DECIMAL	1	CDDL_DATA_GET_FIRST	DataGetFirst - locate and return the first certificate in the ring matching the specified criteria
1	DECIMAL	2	CDDL_DATA_GET_NEXT	DataGetNext - locate and return the next certificate in the ring matching the same criteria specified in the previous DataGetFirst/ DataGetNext call
1	DECIMAL	3	CDDL_DATA_ABORT	DataAbortQuery - free resources from previous DataGetFirst or DataGetNext call
1	DECIMAL	4	CDDL_CHECK_STATUS	CheckStatus - return the TRUST/NOTRUST status for a given certificate
1	DECIMAL	5	CDDL_GET_UPDATE_CODE	GetUpdateCode - returns the sequence number for for a ring
1	DECIMAL	6	CDDL_INC_SER_NUM	IncSerialNum - Increments the Serial Number for a Digital Certificate
1	DECIMAL	7	CDDL_NEW_RING	Create a new key ring or re-create a key ring after deleting it
1	DECIMAL	8	CDDL_DATA_PUT	Add a certificate to RACF and connect it to a key ring
1	DECIMAL	9	CDDL_DATA_REMOVE	Remove a certificate from a key ring, and optionally delete it from RACF
1	DECIMAL	10	CDDL_DEL_RING	Delete a key ring
1	DECIMAL	11	CDDL_DATA_REFRESH	Refresh in-storage certificates in RACF if DIGTCERT is raclisted
1	DECIMAL	12	CDDL_DATA_ALTER	Alter the label and status of a certificate
1	DECIMAL	13	CDDL_GET_RING_INFO	Return the key ring information and the certificate information connected to the ring, or rings
4	DECIMAL	24	CDDL_LEN	Constant - length of CDDL
Private Key Type Constants returned in CDDL_PK_TYPE				
4	DECIMAL	1	CDDL_PKCS1	DER encoded PKCS Key
4	DECIMAL	2	CDDL_ICSF	ICSF key token label

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
4	DECIMAL	3	CDDL_PCICC	PCICC key token label
4	DECIMAL	4	CDDL_DSA	DER encoded DSA key
4	DECIMAL	6	CDDL_DH	Diffie-Hellman key
4	DECIMAL	7	CDDL_ECC	ECC key
4	DECIMAL	9	CDDL_ECC_ICSF	ECC key token label
4	DECIMAL	11	CDDL_RSA_TKDS	TKDS token for RSA
4	DECIMAL	13	CDDL_ECC_TKDS	TKDS token for ECC
4	DECIMAL	14	CDDL_DSA_TKDS	TKDS token for DSA
Input selection criteria constants supplied in CDDL_PREDICATES				
4	DECIMAL	0	CDDL_NONE	No selection criteria
4	DECIMAL	1	CDDL_ATTR	Query on a particular attribute
Input query attributes supplied in CDDL_ATTR_ID				
4	DECIMAL	1	CDDL_LABEL	The certificate label, up to 32 characters long
4	DECIMAL	2	CDDL_DEFAULT	The default flag, a 4 byte field specified as 0 for NO, or a nonzero value for YES
4	DECIMAL	3	CDDL_SUBJECT	BER encoded subject's name
KERB Function code values				
4	DECIMAL	16	KERB_LEN	Constant KERB length
1	DECIMAL	1	KERB_GET_USER	Get user information
1	DECIMAL	2	KERB_UPDATE_REVOKECT	Update revoke count
1	DECIMAL	3	KERB_RESET_REVOKECT	Update revoke count
1	DECIMAL	4	KERB_GET_REALM	Get realm information
TKTS Function code values				
4	DECIMAL	24	TKTS_LEN	Constant TKTS length
2	DECIMAL	1	TKTS_RETURN_NAME	Parse and return Kerberos principal name
PKIS Function code constants				
4	DECIMAL	24	PKIS_LEN	Constant - length of PKIS
4	DECIMAL	28	PKIS_LEN_Z8	Constant - length of PKIS for z/OS V1R8
4	DECIMAL	28	PKIS_TOT_LEN	Length of PKIS for current release of z/OS
2	DECIMAL	1	PKIS_GENCERT	GENCERT Function of RACDCERT. Generates a basic X.509 V3 digital certificate

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
2	DECIMAL	2	PKIS_EXPORT	EXPORT Function of RACDCERT. Extracts a digital certificate by Certificate Request ID
2	DECIMAL	3	PKIS_QUERYREQS	QUERYREQS Function of PKI SERVICES. Query certificate requests
2	DECIMAL	4	PKIS_REQDETAILS	REQDETAILS Function of PKI SERVICES. Extract the details of a single certificate request
2	DECIMAL	5	PKIS_MODIFYREQS	MODIFYREQS Function of PKI SERVICES. Modify one or more certificate requests
2	DECIMAL	6	PKIS_QUERYCERTS	QUERYCERTS Function of PKI SERVICES. Query issued certificates
2	DECIMAL	7	PKIS_CERTDETAILS	CERTDETAILS Function of PKI SERVICES. Extract the details of a single issued certificate
2	DECIMAL	8	PKIS_MODIFYCERTS	MODIFYCERTS Function of PKI SERVICES. Modify one or more issued certificates
2	DECIMAL	9	PKIS_REQCERT	REQCERT Function of PKI SERVICES. Request a digital certificate pending the approval of an administrator
2	DECIMAL	10	PKIS_VERIFY	VERIFY Function of PKI SERVICES. Request certificate verification
2	DECIMAL	11	PKIS_REVOKE	REVOKE Function of PKI SERVICES. Request certificate revocation
2	DECIMAL	12	PKIS_GENRENEW	GENRENEW Function of PKI SERVICES. Request auto-approved certificate renewal
2	DECIMAL	13	PKIS_REQRENEW	REQRENEW Function of PKI SERVICES. Request certificate renewal
2	DECIMAL	14	PKIS_RESPOND	Get OCSP response from responder
2	DECIMAL	15	PKIS_SCEPREQ	SCEPREQ Function of PKI Services. Submit a request to PKI Services using SCEP
2	DECIMAL	16	PKIS_PREREGISTER	PREREGISTER Function of PKI Services. Preregister a SCEP user

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
2	DECIMAL	17	PKIS_QRECOVER	QRECOVER Function of PKI Services. To find a list of certificates whose key pairs were generated by PKI Services under the specified requester's email address and pass phrase.
CACH function code constants				
2	DECIMAL	1	CACH_START	Start a new cache
2	DECIMAL	2	CACH_ADD	Add a record to the cache
2	DECIMAL	3	CACH_END	Finish a cache. Make it available and delete any previous cache
2	DECIMAL	4	CACH_FETCH	Fetch a record from the cache
2	DECIMAL	5	CACH_DELETE	Delete the cache
2	DECIMAL	6	CACH_ICTX	Manage a read/write cache
2	DECIMAL	7	CACH_EXTENDED	
4	DECIMAL	80	CACH_LEN_V18	Constant CACH length for z/OS V1R8
4	DECIMAL	44	CACH_LEN	Constant CACH length
2	DECIMAL	88	CACH_LEN_V1R11	
CACH option value constants				
2	DECIMAL	1	CACH_OPT1	Varies with function
2	DECIMAL	2	CACH_OPT2	Varies with function
2	DECIMAL	3	CACH_OPT3	Varies with function
CACH option values for function code 6				
4	DECIMAL	1	CACH_STORE	Store data in the read/write cache
4	DECIMAL	2	CACH_LOCATE	Locate an application data record in the read/write cache
4	DECIMAL	3	CACH_RETRIEVE	Retrieve data from the read/write cache
4	DECIMAL	4	CACH_RETAPPL	Retrieve application data from the read/write cache
4	DECIMAL	5	CACH_REMOVE	Remove a record from the read/write cache
4	DECIMAL	6	CACH_REMEXP	Remove expired records from the read/write cache
4	DECIMAL	7	CACH_DESTROY	Destroy the read/write cache
CACH option values for function code 7				

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
4	DECIMAL	1	CACH_EXT_STORE	Store data in the read/write cache
4	DECIMAL	2	CACH_EXT_RETRVAPPL	Retrieve application data from the read/write cache
4	DECIMAL	3	CACH_EXT_REMOVE	Remove a record from the read/write cache
4	DECIMAL	4	CACH_EXT_STORE_MULT	Store data in the read/write cache and return multi-use ICRX
4	DECIMAL	5	CACH_EXT_ICRX_VAL	Validate an ICRX
PGSN Function code constants				
4	DECIMAL	1	PGSN_FC_SIGINIT	Initialize program signing
4	DECIMAL	2	PGSN_FC_SIGUPDATE	Sign intermediate program data
4	DECIMAL	3	PGSN_FC_SIGFINAL	Finalize program signature
4	DECIMAL	4	PGSN_FC_SIGCLEAN	Terminate signature operation
4	DECIMAL	5	PGSN_FC_VERINIT	Initialize signature verification
4	DECIMAL	6	PGSN_FC_VERUPDATE	Digest intermediate program data
4	DECIMAL	7	PGSN_FC_VERFINAL	Perform final verification
4	DECIMAL	8	PGSN_FC_VERCLEAN	Terminate verification operation
4	DECIMAL	9	PGSN_FC_VERINTER	Interrogate directive
				See *z/OS Security Server RACF Callable Services* for function specific parameter lists for callable service R_PgmSignVer.
PRXY function code constants				
4	DECIMAL	32	PRXY_LEN0	Original PRXY length
4	DECIMAL	44	PRXY_LEN1	PRXY length for first update
2	DECIMAL	1	PRXY_GETDN	Get DN from LDAP
2	DECIMAL	2	PRXY_GETPRIV	Get privilege information from LDAP
2	DECIMAL	3	PRXY_CHANGELOG	Create LDAP change log entry
2	DECIMAL	3	PRXY_F3_PLIST_LEN0	Function 3 specific parameter list length
1	HEX	00	PRXY_F3_OP_ADD	
1	HEX	01	PRXY_F3_OP_DEL	
1	HEX	02	PRXY_F3_OP_MOD	
RACL operation type constants				

Table 27. Constants for Comp (continued)

Len	Type	Value	Name	Description
2	DECIMAL	1	RACL_ADD	ADD the file system ACL
2	DECIMAL	2	RACL_MODIFY	MODIFY the file system ACL
2	DECIMAL	3	RACL_DELETE	DELETE the file system ACL
RACL file system ACL Type Constants				
2	DECIMAL	1	RACL_ACCESS	Access file system ACL
2	DECIMAL	2	RACL_FILEMOD	File Model file system ACL
2	DECIMAL	3	RACL_DIRMOD	Directory Model file system ACL
RAUX Type Constants				
2	DECIMAL	76	RAUX_LEN	Constant RAUX length
WPRV Function code constants				
2	HEX	00	WPRV_Query_WriteDown_Setting	
2	HEX	01	WPRV_Activate_WriteDown	
2	HEX	02	WPRV_InActivate_WriteDown	
2	HEX	03	WPRV_Reset_Writedown	

COMP cross reference

Table 28. Cross Reference for COMP

Name	Offset	Hex Value
ADMN	0	
ADMN_ACEEP@	C	
ADMN_FUNC@	0	
ADMN_GRPADM_ERROFF	A	
ADMN_GRPADM_GROUP	0	
ADMN_GRPADM_GRP_ID	1	
ADMN_GRPADM_GRP_LEN	0	
ADMN_GRPADM_MAP	0	
ADMN_GRPADM_SEG_NUM	C	
ADMN_GRPADM_SEGSTRT	E	
ADMN_LAST_PARM	14	80
ADMN_OUT_ENTRY	0	
ADMN_OUT_EYE	4	
ADMN_OUT_LEN	9	
ADMN_OUT_MLEN	0	
ADMN_OUT_MSTR	2	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
ADMN_OUT_NEXT	0	
ADMN_OUT_OFF	C	
ADMN_OUT_SPID	8	
ADMN_OUT_STRT	10	
ADMN_OUTMSG_MAP	0	
ADMN_OUTPUT_MSG@	14	
ADMN_OUTPUT_SP@	10	
ADMN_PARMLIST@	4	
ADMN_RESADM_CLAS_LEN	0	
ADMN_RESADM_CLAS_NAM	1	
ADMN_RESADM_CLASS	0	
ADMN_RESADM_ERROFF	A	
ADMN_RESADM_MAP	0	
ADMN_RESADM_SEG_NUM	C	
ADMN_RESADM_SEGSTRT	E	
ADMN_RUNCOMD_CMD	2	
ADMN_RUNCOMD_LEN	0	
ADMN_RUNCOMD_MAP	0	
ADMN_SETRADM_ERROFF	A	
ADMN_SETRADM_MAP	0	
ADMN_SETRADM_SEG_NUM	C	
ADMN_SETRADM_SEGSTRT	E	
ADMN_UNL_ENTRY	0	
ADMN_UNL_LEN	8	
ADMN_UNL_NUM	C	
ADMN_UNL_RECSTRT	18	
ADMN_UNL_SETR	23	17
ADMN_UNL_TYPE	0	
ADMN_USERID@	8	
ADMN_USRADM_ERROFF	A	
ADMN_USRADM_FLD_DATA	B	
ADMN_USRADM_FLD_FLAG	8	
ADMN_USRADM_FLD_LEN	9	
ADMN_USRADM_FLD_NAME	0	
ADMN_USRADM_FLD_NUM	9	
ADMN_USRADM_FLDENTRY	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
ADMN_USRADM_FLDSTRT	B	
ADMN_USRADM_MAP	0	
ADMN_USRADM_SEG_FLAG	8	
ADMN_USRADM_SEG_NAME	0	
ADMN_USRADM_SEG_NUM	C	
ADMN_USRADM_SEGENTRY	0	
ADMN_USRADM_SEGSTRT	E	
ADMN_USRADM_USER_ID	1	
ADMN_USRADM_USER_LEN	0	
ADMN_XTR_CONNECT	29	
ADMN_XTR_GROUP	27	
ADMN_XTR_NEXT_GROUP	28	
ADMN_XTR_NEXT_RESOURCE	32	
ADMN_XTR_NEXT_USER	26	
ADMN_XTR_PPENV	30	
ADMN_XTR_PWENV	24	
ADMN_XTR_RESOURCE	31	
ADMN_XTR_RRSF	33	
ADMN_XTR_USER	25	
ADMN_XTRPW_ENV	8	
ADMN_XTRUNL_ENTRY	E	
ADMN_XTRUNL_EYE	0	
ADMN_XTRUNL_MAP	0	
ADMN_XTRUNL_NUM	C	
ADMN_XTRUNL_OUTLEN	4	
CACH	0	
CACH_ACEE_ALET@	48	
CACH_ACEE@	4C	
CACH_DATA_LEN@	28	
CACH_DATA_PTR@	24	
CACH_DATA_TIMEOUT@	2C	
CACH_FUNC@	8	
CACH_NAME@	18	
CACH_NUM_PARMS@	4	
CACH_OPTION@	C	
CACH_PARM_ALET@	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
CACH_RECNAME_LEN@	20	
CACH_RECNAME_PTR@	1C	
CACH_REF_TIMEOUT@	38	
CACH_REF_USERID@	3C	
CACH_REF@	40	
CACH_SRC_LEN@	34	
CACH_SRC_PTR@	30	
CACH_SUBPOOL@	44	
CACH_VERSION_LEN@	14	
CACH_VERSION@	10	
CAUD	0	
CAUD_AO_ALET@	0	
CAUD_AO@	4	
CAUD_CRED_ALET@	18	
CAUD_CRED@	1C	
CAUD_FID_ALET@	10	
CAUD_FID@	14	
CAUD_FSP_ALET@	8	
CAUD_FSP@	C	
CAUD_LEN	0	
CDDL	0	
CDDL_ABORT	0	
CDDL_ATT_ALL_KEYTYPES	0	80
CDDL_ATT_DEL_CERT_ALLRINGS	0	40
CDDL_ATT_DEL_CERT_FORCE	0	20
CDDL_ATT_DEL_CERT_TOO	0	80
CDDL_ATT_HIGHTRUST	0	40
CDDL_ATT_NOTRUST	0	20
CDDL_ATT_REUSE_RING	0	80
CDDL_ATT_SET_MIN_SERIAL	0	80
CDDL_ATT_SKIPAUTH	0	80
CDDL_ATT_TRUST	0	80
CDDL_ATTR_ID	8	
CDDL_ATTR_LEN	C	
CDDL_ATTR_PTR	10	
CDDL_ATTRIBUTES@	20	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
CDDL_ATTRIBUTES_MAP	0	
CDDL_CERT_DEFAULT	8	
CDDL_CERT_LEN	C	
CDDL_CERT_PTR	10	
CDDL_CERT_USAGE	4	
CDDL_CRT_LEN	0	
CDDL_CRT_PTR	4	
CDDL_CRT_STATUS	0	
CDDL_FUNC@	0	
CDDL_GET_CERT	0	
CDDL_GET_UPDATE	0	
CDDL_HANDLE_MAP	0	
CDDL_ISN	0	
CDDL_ISN_CLEN	0	
CDDL_ISN_CPTR	4	
CDDL_ISN_SERNUM	8	
CDDL_LABEL_LEN	24	
CDDL_LABEL_PTR	28	
CDDL_LAST_PARM	14	80
CDDL_PARMLIST@	14	
CDDL_PCERT_DEFAULT	4	
CDDL_PCERT_LEN	8	
CDDL_PCERT_PTR	C	
CDDL_PCERT_USAGE	0	
CDDL_PCERT_USERID	20	
CDDL_PK_BITSIZE	20	
CDDL_PK_LEN	14	
CDDL_PK_PTR	18	
CDDL_PK_TYPE	1C	
CDDL_PKEY_LEN	10	
CDDL_PKEY_PTR	14	
CDDL_PLABEL_LEN	18	
CDDL_PLABEL_PTR	1C	
CDDL_PREDICATES	4	
CDDL_PREV_HANDLE	0	
CDDL_PUT_CERT	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
CDDL_RACF_USERID	2C	
CDDL_RCERT_USERID	8	
CDDL_RECID_LEN	40	
CDDL_RECID_PTR	44	
CDDL_REMOVE_CERT	0	
CDDL_RES_HANDLE	0	
CDDL_RING_RES_LEN	4	
CDDL_RING_RES_PTR	8	
CDDL_RING_SEQN	0	
CDDL_RING_SRCH_TYPE	0	
CDDL_RING@	C	
CDDL_RLABEL_LEN	0	
CDDL_RLABEL_PTR	4	
CDDL_SDN_LEN	38	
CDDL_SDN_PTR	3C	
CDDL_STATUS	48	
CDDL_TOKEN	0	
CDDL_USERID@	8	
CDDL_VERSION@	10	
CHKP	0	
CHKP_AUFC_ALET@	0	
CHKP_AUFC@	4	
CKO2	0	
CKO2_CRED_ALET@	20	
CKO2_CRED@	24	
CKO2_FID1_ALET@	10	
CKO2_FID1@	14	
CKO2_FID2_ALET@	18	
CKO2_FID2@	1C	
CKO2_FSP1_ALET@	0	
CKO2_FSP1@	4	
CKO2_FSP2_ALET@	8	
CKO2_FSP2@	C	
CKPO	0	
CKPO_CODE_ALET@	18	
CKPO_CODE@	1C	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
CKPO_PID_ALET@	10	
CKPO_PID@	14	
CKPO_REQT_ALET@	0	
CKPO_REQT@	4	
CKPO_SIGQUEUE	5	5
CKPO_UIDS_ALET@	8	
CKPO_UIDS@	C	
CLID	0	
CLID_CRED_ALET@	10	
CLID_CRED@	14	
CLID_FID_ALET@	8	
CLID_FID@	C	
CLID_FSP_ALET@	0	
CLID_FSP@	4	
CMOD	0	
CMOD_CRED_ALET@	18	
CMOD_CRED@	1C	
CMOD_FID_ALET@	10	
CMOD_FID@	14	
CMOD_FSP_ALET@	0	
CMOD_FSP@	C	
CMOD_MODE_ALET@	0	
CMOD_MODE@	4	
COMP	0	
COMP_RACRC_ALET@	C	
COMP_RACRC_STOR@	10	
COMP_RACSC_ALET@	14	
COMP_RACSC_STOR@	18	
COMP_SAFRC_ALET@	4	
COMP_SAFRC_STOR@	8	
COMP_SERVR_PARMS	1C	
COMP_WORKA_STOR@	0	
COWN	0	
COWN_CRED_ALET@	20	
COWN_CRED@	24	
COWN_FID_ALET@	18	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
COWN_FID@	1C	
COWN_FSP_ALET@	10	
COWN_FSP@	14	
COWN_GID_ALET@	8	
COWN_GID@	C	
COWN_UID_ALET@	4	
COWN_UID@	0	
CSID	0	
CSID_ID_ALET@	0	
CSID_ID@	4	
CSID_IDOA_ALET@	8	
CSID_IDOA@	C	
DAUT	0	
DAUT_ACC_REQ@	20	
DAUT_ACEEP@	0	
DAUT_ALET@	4	
DAUT_CLASS_NAME@	14	
DAUT_ENT_LEN@	1C	
DAUT_ENT_NAME@	18	
DAUT_HOME_UUID@	C	
DAUT_PRIN_UUID@	8	
DAUT_USERID@	10	
DINF	0	
DINF_FIELD_DATA	A	
DINF_FIELD_LEN	8	
DINF_FIELD_LIST	0	
DINF_FIELD_LIST_CNT	2	
DINF_FIELD_LIST_LEN	0	
DINF_FIELD_LIST_ST	4	
DINF_FIELD_NAME	0	
DINF_FIELDL_ALET@	10	
DINF_FIELDL@	14	
DINF_FIELDS	0	
DINF_FUNC_ALET@	0	
DINF_FUNC@	4	
DINF_OUTPUT_AREA	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
DINF_OUTPUT_AREA_CNT	2	
DINF_OUTPUT_AREA_LEN	0	
DINF_OUTPUT_AREA_ST	4	
DINF_OUTPUT_FLD	0	
DINF_OUTPUT_FLD_DATA	A	
DINF_OUTPUT_FLD_LEN	8	
DINF_OUTPUT_FLD_NAME	0	
DINF_OUTPUTA_ALET@	18	
DINF_OUTPUTA@	1C	
DINF_OUTPUTL_ALET@	20	
DINF_OUTPUTL@	24	
DINF_USERID_ALET@	8	
DINF_USERID@	C	
DKEY	0	
DKEY_FUNC_ALET@	0	
DKEY_FUNC@	4	
DKEY_KEY_ALET@	10	
DKEY_KEY@	14	
DKEY_KEYL_ALET@	18	
DKEY_KEYL@	1C	
DKEY_USERID_ALET@	8	
DKEY_ENTITY_ALET@	8	
DKEY_ENTITY@	12	
DKEY_USERID@	C	
DRUR	0	
DRUR_FUNC_ALET@	0	
DRUR_FUNC@	4	
DRUR_HOME_UUID_ALET@	10	
DRUR_HOME_UUID@	14	
DRUR_PRIN_UUID_ALET@	8	
DRUR_PRIN_UUID@	C	
DRUR_USERID_ALET@	18	
DRUR_USERID@	1C	
EXID	0	
EXID_FLAG_ALET@	0	
EXID_FLAG@	4	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
EXID_GID_ALET@	10	
EXID_GID@	14	
EXID_GIDO_ALET@	20	
EXID_GIDO@	24	
EXID_UID_ALET@	8	
EXID_UID@	C	
EXID_UIDO_ALET@	18	
EXID_UIDO@	1C	
FORK	0	
FORK_DATA_ALET@	18	
FORK_DATA@	1C	
FORK_DATAKEY_ALET@	8	
FORK_DATAKEY@	C	
FORK_DATALEN_ALET@	10	
FORK_DATALEN@	14	
FORK_DATASP_ALET@	20	
FORK_DATASP@	24	
FORK_FLAG_ALET@	0	
FORK_FLAG@	4	
GETE	0	
GETE_GCNT_ALET@	10	
GETE_GCNT@	14	
GETE_GID_ALET@	30	
GETE_GID@	34	
GETE_GLST_ALET@	18	
GETE_GLST@	1C	
GETE_GNUM_ALET@	20	
GETE_GNUM@	24	
GET_INFO@	28	
GETE_KEY_ALET@	8	
GETE_KEY@	C	
GETE_UID_ALET@	28	
GETE_UID@	2C	
GETE_WORKB_ALET@	0	
GETE_WORKB@	4	
GETG	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
GETG_GCNT_ALET@	8	
GETG_GCNT@	C	
GETG_GLST_ALET@	10	
GETG_GLST@	14	
GETG_GNUM_ALET@	18	
GETG_GNUM@	1C	
GETG_KEY_ALET@	0	
GETG_KEY@	4	
GINF	0	
GINF_FUNC_CODE@	8	
GINF_NUM_PARMS@	0	
GINF_OPTION@	C	
GINF_PARM_ALET@	0	
GINF_RACF_CLASS@	14	
GINF_RACF_ENTITY@	10	
GINF_RESULT_ENTRIES@	18	
GMAP	0	
GMAP_FLAG_ALET@	0	
GMAP_FLAG@	4	
GMAP_ID_ALET@	8	
GMAP_ID@	C	
GMAP_NAME_ALET@	10	
GMAP_NAME@	14	
GUGP	0	
GUGP_GCNT_ALET@	18	
GUGP_GCNT@	1C	
GUGP_GLST_ALET@	20	
GUGP_GLST@	24	
GUGP_GNUM_ALET@	28	
GUGP_GNUM@	2C	
GUGP_KEY_ALET@	0	
GUGP_KEY@	4	
GUGP_NAME_ALET@	10	
GUGP_NAME@	14	
GUGP_NLEN_ALET@	8	
GUGP_NLEN@	C	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
IACC	0	
IACC_ACODE_ALET@	0	
IACC_ACODE@	4	
IACC_CREI_ALET@	10	
IACC_CREI@	14	
IACC_ISP_ALET@	8	
IACC_ISP@	C	
INTA	0	
INTA_ACEEP@	28	
INTA_APPL_ID@	10	
INTA_ATTRIBUTES_MAP	0	
INTA_ATTRIBUTES@	4	
INTA_AUTH_CLNT	0	08
INTA_CERTIFICATE@	1C	
INTA_CLIENT	0	18
INTA_ENVR_IN@	20	
INTA_ENVR_OUT@	24	
INTA_ENVR_RET	0	02
INTA_FUNC@	0	
INTA_IDID_AREA@	40	
INTA_LAST_PARM	30	80
INTA_LAST_PARM_ENVR_IN	20	80
INTA_LAST_PARM_ENVR_OUT	24	80
INTA_LAST_PARM_IDID	3C	80
INTA_LAST_PARM_OUTA_STOR	28	80
INTA_LAST_PARM_PHRASE	3C	80
INTA_LAST_PARM_VAR_LIST	30	80
INTA_LAST_PARM_VER1	18	80
INTA_LAST_PARM_VER2	1C	80
INTA_LAST_X500_NAME	2C	80
INTA_LOGSTRING@	18	
INTA_MANAGED	0	80
INTA_MSG_SUPP	0	04
INTA_NO_TIMEOUT	0	01
INTA_OUSP_RET	1	80
INTA_OUTA_STOR@	28	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
INTA_PASSWORD@	14	
INTA_PHRASE@	3C	
INTA_SECLABEL@	34	
INTA_SERVAUTH@	38	
INTA_TASK_LVL	0	20
INTA_UNAUTH_CLNT	0	10
INTA_USERID@	8	
INTA_USP	0	40
INTA_VARIABLE_LIST@	30	
INTA_X500_NAME_PTR	2C	
INTA_X500_RET	1	40
IOWN	0	
IOWN_CMD_ALET@	0	
IOWN_CMD@	4	
IOWN_CREI_ALET@	28	
IOWN_CREI@	2C	
IOWN_GID_ALET@	10	
IOWN_GID@	14	
IOWN_ISP_ALET@	20	
IOWN_ISP@	24	
IOWN_MODE_ALET@	18	
IOWN_MODE@	1C	
IOWN_UID_ALET@	8	
IOWN_UID@	C	
IUSP	0	
IUSP_WORKA_ALET@	0	
IUSP_WORKA_STOR@	4	
KACC	0	
KACC_ACODE_ALET@	0	
KACC_ACODE@	4	
KACC_CRED_ALET@	18	
KACC_CRED@	1C	
KACC_FID_ALET@	10	
KACC_FID@	14	
KACC_FLAG_ALET@	20	
KACC_FLAG@	24	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
KACC_FSP_ALET@	8	
KACC_FSP@	C	
KERB	A	
KERB_DATA_AREA@	C	
KERB_FLD_DATA	A	
KERB_FLD_LEN	8	
KERB_FLD_NAME	0	
KERB_FLD_NUM	2	
KERB_FLDDATA_LEN	0	
KERB_FLDDATA_MAP	0	
KERB_FLDENTRY	0	
KERB_FLDSTRT	4	
KERB_FUNC@	0	
KERB_LAST_PARM	C	80
KERB_NAME@	8	
KERB_RACF_NAME@	4	
KFOW	0	
KFOW_CRED_ALET@	10	
KFOW_CRED@	14	
KFOW_FID_ALET@	8	
KFOW_FID@	C	
KFOW_FSP_ALET@	0	
KFOW_FSP@	4	
MFSP	0	
MFSP_CRED_ALET@	20	
MFSP_CRED@	24	
MFSP_FID_ALET@	18	
MFSP_FID@	1C	
MFSP_MODE_ALET@	0	
MFSP_MODE@	4	
MFSP_ODFSP_ALET@	10	
MFSP_ODFSP@	14	
MFSP_OFSP_ALET@	8	
MFSP_OFSP@	C	
MISP	0	
MISP_CREI_ALET@	18	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
MISP_CREI@	1C	
MISP_MODE_ALET@	0	
MISP_MODE@	4	
MISP_OIPC_ALET@	10	
MISP_OIPC@	14	
MISP_OISP_ALET@	8	
MISP_OISP@	C	
MKRT	0	
MKRT_DSN_ALET@	18	
MKRT_DSN@	1C	
MKRT_FID_ALET@	10	
MKRT_FID@	14	
MKRT_MODE_ALET@	0	
MKRT_MODE@	4	
MKRT_OFSP_ALET@	8	
MKRT_OFSP@	C	
PKIS	0	
PKIS_ATTRIBUTES_MAP	0	
PKIS_ATTRIBUTES@	8	
PKIS_CA_DOMAIN@	18	
PKIS_CERTDETAILS_MAP	0	
PKIS_CPL_ENTRY	0	
PKIS_CPL_FIELD_LEN	C	
PKIS_CPL_FIELD_NAME	0	
PKIS_CPL_FIELD_VALUE	10	
PKIS_CRTD_CERTPL_LEN	10	
PKIS_CRTD_CERTPL@	14	
PKIS_CRTD_EYECATCH	0	
PKIS_CRTD_SERIALNUM@	18	
PKIS_CRTD_SUML_LEN	8	
PKIS_CRTD_SUML@	C	
PKIS_EXP_CERTAN_LEN	8	
PKIS_EXP_CERTAN@	C	
PKIS_EXP_CERTID@	10	
PKIS_EXP_EYECATCH	0	
PKIS_EXPORT_MAP	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
PKIS_FUNC_PARML@	14	
PKIS_FUNC@	4	
PKIS_GENC_CERTID@	10	
PKIS_GENC_CERTPL_LEN	8	
PKIS_GENC_CERTPL@	C	
PKIS_GENC_EYECATCH	0	
PKIS_EXP_KEYID@	14	
PKIS_GENCERT_MAP	0	
PKIS_LAST_PARM	14	80
PKIS_LOG_STRING@	C	
PKIS_MODC_ACTION	8	
PKIS_MODC_COMMENT@	C	
PKIS_MODC_ERRL_LEN	20	
PKIS_MODC_ERRL@	24	
PKIS_MODC_EYECATCH	0	
PKIS_MODC_REASON	18	
PKIS_MODC_REQUESTOREMAIL@	1C	
PKIS_MODC_SERIALNUMSL_LEN	10	
PKIS_MODC_SERIALNUMSL@	14	
PKIS_MODIFYCERTS_MAP	0	
PKIS_MODIFYREQS_MAP	0	
PKIS_MODR_ACTION	8	
PKIS_MODR_CERTIDL_LEN	10	
PKIS_MODR_CERTIDL@	14	
PKIS_MODR_CERTPL_LEN	18	
PKIS_MODR_CERTPL@	1C	
PKIS_MODR_COMMENT@	C	
PKIS_MODR_ERRL_LEN	20	
PKIS_MODR_ERRL@	24	
PKIS_MODR_EYECATCH	0	
PKIS_MODR_TSTAMP@	28	
PKIS_NUM_PARMS@	0	
PKIS_PARM_VER@	10	
PKIS_PREG_CERTID@	16	
PKIS_QREC_CRIT_EMAIL@	14	
PKIS_QREC_CRIT_PASS@	18	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
PKIS_QREC_EYECATCH	0	
PKIS_QREC_NUMENTRIES	10	
PKIS_QREC_RESULTL_LEN	8	
PKIS_QREC_RESULTL@	C	
PKIS_QRECOVER_MAP	0	
PKIS_QRYC_CRIT_DAYS	1C	
PKIS_QRYC_CRIT_NAME@	20	
PKIS_QRYC_CRIT_STATUS	18	
PKIS_QRYC_EYECATCH	0	
PKIS_QRYC_NUMENTRIES	14	
PKIS_QRYC_RESULTL_LEN	8	
PKIS_QRYC_RESULTL@	C	
PKIS_QRYC_SERIALNUM@	10	
PKIS_QRYR_CERTID@	10	
PKIS_QRYR_CRIT_DAYS	1C	
PKIS_QRYR_CRIT_NAME@	20	
PKIS_QRYR_CRIT_STATUS	18	
PKIS_QRYR_EYECATCH	0	
PKIS_QRYR_NUMENTRIES	14	
PKIS_QRYR_RESULTL_LEN	8	
PKIS_QRYR_RESULTL@	C	
PKIS_QUERYCERTS_MAP	0	
PKIS_QUERYREQS_MAP	0	
PKIS_RENEW_MAP	0	
PKIS_RENW_CERTID@	10	
PKIS_RENW_CERTPL_LEN	8	
PKIS_RENW_CERTPL@	C	
PKIS_RENW_EYECATCH	0	
PKIS_RENW_SERIALNUM@	14	
PKIS_RESPOND	14	
PKIS_REQD_CERTID@	18	
PKIS_REQD_CERTPL_LEN	10	
PKIS_REQD_CERTPL@	14	
PKIS_REQD_EYECATCH	0	
PKIS_REQD_SUML_LEN	8	
PKIS_REQD_SUML@	C	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
PKIS_REQDETAILS_MAP	0	
PKIS_RESL_ENTRY	0	
PKIS_RESL_ENTRY_LEN	0	
PKIS_RESL_ENTRY_VALUE	1	
PKIS_REVK_EYECATCH	0	
PKIS_REVK_REASON	8	
PKIS_REVK_SERIALNUM@	C	
PKIS_REVOKE_MAP	0	
PKIS_SCEP_EYECATCH	0	
PKIS_SNID_ENTRY	0	
PKIS_SNID_ENTRY_LEN	0	
PKIS_SNID_ENTRY_VALUE	1	
PKIS_SUML_ENTRY	0	
PKIS_SUML_ENTRY_LEN	0	
PKIS_SUML_ENTRY_VALUE	1	
PKIS_SYNCH_CREATE	0	80
PKIS_VERF_CERT_LEN	18	
PKIS_VERF_CERT@	1C	
PKIS_VERF_CERTPL_LEN	10	
PKIS_VERF_CERTPL@	14	
PKIS_VERF_EYECATCH	0	
PKIS_VERF_SUML_LEN	8	
PKIS_VERF_SUML@	C	
PKIS_VERIFY_MAP	0	
PRXY	0	
PRXY_BIND_DN@	C	
PRXY_BIND_PW@	10	
PRXY_F3_CLASS	2	
PRXY_F3_DATETIME	18	
PRXY_F3_FLAGS	1	
PRXY_F3_INITIATOR	10	
PRXY_F3_OPTYPE	0	
PRXY_F3_PLIST	0	
PRXY_F3_PROFLEN	A	
PRXY_F3_PROFNAME@	C	
PRXY_F3_PWUPD	1	80

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
PRXY_F3_PWUPD2	1	40
PRXY_F3_PWUPD3	1	
PRXY_FUNC@	4	
PRXY_FUNC_VERSION@	20	
PRXY_FUNC_PARMLIST@	24	
PRXY_LAST_PARM0	1C	80
PRXY_LAST_PARM1	28	80
PRXY_LDAP_ERROR@	28	
PRXY_LDAP_HOST@	8	
PRXY_PARM_ALET@	0	
PRXY_RESULTS@	1C	
PRXY_USERDN@	18	
PRXY_USERID@	14	
PTRC	0	
PTRC_TGID_ALET@	C	
PTRC_TGID@	10	
PTRC_TPID_ALET@	14	
PTRC_TPID@	30	
PTRC_TUID_ALET@	0	
PTRC_TUID@	4	
QRYF	0	
QRYF_OCODE_ALET@	0	
QRYF_OCODE@	4	
QRYF_OUTV_ALET@	8	
QRYF_OUTV@	C	
QRYS	0	
QRYS_OPTN_CODE_ALET@	0	
QRYS_OPTN_CODE@	4	
QRYS_OUTP_VAL_ALET@	8	
QRYS_OUTP_VAL@	C	
RACL	0	
RACL_CRED_ALET@	20	
RACL_CRED@	24	
RACL_EDIT	0	
RACL_EDIT_ACL	4	
RACL_EDIT_ACLTYPE	2	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
RACL_EDIT_ALET@	0	
RACL_EDIT_OPTYPE	0	
RACL_EDIT@	4	
RACL_EDITLEN_ALET@	8	
RACL_EDITLEN@	1C	
RACL_FID_ALET@	18	
RACL_FID@	1C	
RACL_FSP_ALET@	10	
RACL_FSP@	14	
RACL_LAST_PARM	24	80
RAUD	0	
RAUD_CRED_ALET@	0	
RAUD_CRED@	4	
RAUD_FLAG_ALET@	18	
RAUD_FLAG@	1C	
RAUD_NFID_ALET@	20	
RAUD_NFID@	24	
RAUD_NFSP_ALET@	28	
RAUD_NFSP@	2C	
RAUD_OFID_ALET@	8	
RAUD_OFID@	C	
RAUD_OFSP_ALET@	10	
RAUD_OFSP@	14	
SECL	0	
SECL_CRED_ALET@	10	
SECL_CRED@	14	
SECL_FID_ALET@	8	
SECL_FID@	C	
SECL_FSP_ALET@	0	
SECL_FSP@	4	
SMOP	0	
SMOP_ACEE@	1C	
SMOP_FUNC@	4	
SMOP_HANDLE@	14	
SMOP_NUM_PARMS@	0	
SMOP_OPTIONS@	8	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
SMOP_REQUEST_LEN@	C	
SMOP_REQUEST@	10	
SMOP_RESULT_LEN@	20	
SMOP_RESULT@	24	
SMOP_USERID@	18	
TKTS	0	
TKTS_CRED_AREA@	C	
TKTS_FUNC@	4	
TKTS_LAST_PARM	14	80
TKTS_OPTION_WORD@	8	
TKTS_OPTIONS@	10	
TKTS_PARM_ALET@	0	
UMAP	0	
UMAP_APPL_USERID	0	
UMAP_APPL_USERID@	14	
UMAP_APPLID	2	
UMAP_APPLID_LEN	0	
UMAP_CERTIFICATE	0	
UMAP_CERTIFICATE@	10	
UMAP_DIGCERT	4	
UMAP_DIGCERT_LEN	0	
UMAP_DISTINGUISHED_NAME	2	
UMAP_DISTINGUISHED_NAME_DS	0	
UMAP_DISTINGUISHED_NAME_LEN	0	
UMAP_DISTINGUISHED_NAME@	18	
UMAP_FUNC@	4	
UMAP_ID_PROPAGATION	18	
UMAP_IDPROP_LAST_PARM	1C	80
UMAP_LAST_PARM	14	80
UMAP_OPTION_WORD@	8	
UMAP_PARM_ALET@	0	
UMAP_RACFID	1	
UMAP_RACFID_LEN	0	
UMAP_REGISTRY_NAME	2	
UMAP_REGISTRY_NAME_DS	0	
UMAP_REGISTRY_NAME_LEN	0	

Table 28. Cross Reference for COMP (continued)

Name	Offset	Hex Value
UMAP_REGISTRY_NAME@	1C	
UMAP_USERID	0	
UMAP_USERID@	C	
UMSK	0	
UMSK_MODE_ALET@	0	
UMSK_MODE@	4	
WPRV	0	
WPRV_ACEE_ALET@	4	
WPRV_ACEE@	8	
WPRV_FUNC@	0	

Chapter 13. COMX: 64-bit enabled SAF callable services

COMX heading information

Common name: 64-bit enabled SAF callable services

Macro ID: IRRPCOMX

DSECT name: COMX, CDDLX, GSEC

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
N/A

Key
Any

Residency
Invoker's primary address space

Size:

Section
Size

COMX
64 bytes

CDDLX
48 bytes

GSEC
32 bytes

Created by: Invoker of 64-bit enabled callable services

Pointed to by: Address of COMX is passed in register 1 when invoking 64-bit enabled callable services

Serialization: None

Function: Maps the common input parameter list for the 64-bit RACF and SAF callable services routers.

COMX mapping

Table 29. Structure COMX

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	COMPX	z/OS UNIX SAF/RACF common parameter list
0	(0)	ADDRESS	8	COMPX_PARMCNT@	Address of number of parameters
8	(8)	ADDRESS	8	COMPX_WORKA_STOR@	Address of 1024 byte work area.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	8	COMPX_SAFRC_ALET@	Address of ALET for SAF Return Code.
24	(18)	ADDRESS	8	COMPX_SAFRC_STOR@	Address of SAF Return Code.
32	(20)	ADDRESS	8	COMPX_RACRC_ALET@	Address of ALET for RACF Return Code.
40	(28)	ADDRESS	8	COMPX_RACRC_STOR@	Address of RACF Return Code.
48	(30)	ADDRESS	8	COMPX_RACSC_ALET@	Address of ALET for RACF Reason Code.
56	(38)	ADDRESS	8	COMPX_RACSC_STOR@	Address of RACF Reason Code.
64	(40)	CHARACTER	1	COMPX_SERVR_PARMS(0)	Parameters for the service routines start here.

CDDLX

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	CDDLX	Mapping for CDDLX.
0	(0)	ADDRESS	8	CDDLX_FUNC@	Address of 1 byte input function code. Constants for the function codes are supplied below.
8	(8)	ADDRESS	8	CDDLX_ATTRIBUTES@	Address of full word of input attributes.
0	(0)	BITSTRING	8	CDDLX_ATTRIBUTES_MAP	Attribute flags to cause different behaviors in different functions in R_Datalib.
		1...		CDDLX_ATT_ALL_KEYTYPES	For functions DataGetFirst and DataGetNext to differentiate between PCICC key type and ICSF key type, DSA key type and PKCS #1 key type, when returning the function specific parameter list field Private_key_type. When this flag is off, R_Datalib treats either key type as an ICSF key type and return value x'00000002', treat DSA key type as a PKCS #1 key type and return value x'00000001'.
		1...		CDDLX_ATT_REUSE_RING	For function NewRing to reuse the existing key ring and remove all of the certificates from it. When this flag is off, it indicates the creation of a new key ring.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		CDDLX_ATT_SET_MIN_SERIAL	For function IncSerialNumber to indicate that the last used serial number field (CERTLUSER) is to be incremented to at least the input serial number.
		1...		CDDLX_ATT_TRUST	For function DataPut and DataAlter to set certificate with TRUST status. When this flag is off, it indicates RACF determines the status.
		.1..		CDDLX_ATT_HIGHTRUST	For function DataPut and DataAlter to set certificate under CERTAUTH with HIGHTRUST status.
		..1.		CDDLX_ATT_NOTRUST	For function DataPut and DataAlter to set certificate with NOTRUST status.
		1...		CDDLX_ATT_DEL_CERT_TOO	For function DataRemove to delete the certificate after it is removed from the ring, if it is not connected to any other rings. When this flag is off, it indicates the certificate is removed from the ring only.
		.1..		CDDLX_ATT_DEL_CERT_ALLRINGS	For function DataRemove to delete certificate, even if it is connected to another ring.
		..1.		CDDLX_ATT_DEL_CERT_FORCE	For function DataRemove to delete certificate, even if it is used to generate a request.
		1...		CDDLX_ATT_SKIPAUTH	For function DataGetFirst, DataGetNext, CheckStatus, and GetUpdateCode to indicate RACF authorization check is bypassed for supervisor state or system key caller when request is for a RACF key ring (not applicable to an ICSFtoken). When this flag is off, it indicates RACF authorization check must be performed.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	8	CDDLX_USERID@	Address of a 9 byte input area for the RACF userid (1 byte for the length followed by 8 bytes for the RACF userid itself). A length of 0 indicates the current user is the ring owner.
24	(18)	ADDRESS	8	CDDLX_RING@	Address of a variable length input area for the ring name (1 byte for the length followed by up to 237 bytes for the ring name). Parameter ignored for functions CheckStatus, AbortQuery, and DataGetNext.
32	(20)	ADDRESS	8	CDDLX_VERSION@	Address of a 4 byte area containing the input version number for function specific parameter list pointed to by CDDLX_PARMLIST@.
40	(28)	ADDRESS	8	CDDLX_PARMLIST@	Address of the input function specific parameter list
		1...		CDDLX_LAST_PARM	Last parameter in variable length parameter list.
CDDLS_ABORT					
0	(0)	STRUCTURE	0	CDDLX_ABORT	Parameter list for DataAbortQuery
0	(0)	ADDRESS	8	CDDLX_PREV_HANDLE	Address of an input/ output area from previous DataGetFirst or DataGetNext call
CDDLX_GET_CERT					
0	(0)	STRUCTURE	0	CDDLX_GET_CERT	Parameter list for DataGetFirst and DataGetNext
0	(0)	ADDRESS	8	CDDLX_RES_HANDLE	Address of input/output area mapped by CDDLX_HANDLE_MAP
8	(8)	UNSIGNED	4	CDDLX_CERT_USAGE	4 byte output area containing certificate usage flags
					X'00000000' - Usage is SITE X'00000002' - Usage is CERTAUTH X'00000008' - Usage is PERSONAL X'FFFFFFF5' - reserved bits must be set to 0

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	UNSIGNED	4	CDDLX_CERT_DEFAULT	Output default indicator. Zero value indicates not default certificate for ring, non-zero indicates this is the default certificate.
16	(10)	UNSIGNED	4	CDDLX_CERT_LEN	On input, contains the length of the certificate area pointed to by CDDLX_CERT_PTR. On output, contains the actual size of the certificate returned or 0 if no certificate returned.
20	(14)	UNSIGNED	4	*	Reserved for alignment
24	(18)	ADDRESS	8	CDDLX_CERT_PTR	Input value specifying address of output certificate data area.
32	(20)	UNSIGNED	4	CDDLX_PK_LEN	On input, contains size of private key area pointed to by CDDLX_PK_PTR. On output contains the length of the private returned at address CDDLX_PK_PTR or 0 if no private key was returned.
36	(24)	UNSIGNED	4	*	Reserved for alignment
40	(28)	ADDRESS	8	CDDLX_PK_PTR	Input value specifying address of private key output data area
48	(30)	UNSIGNED	4	CDDLX_PK_TYPE	Output value indicating type of private key. See constants below.
52	(34)	UNSIGNED	4	CDDLX_PK_BITSIZE	Output value indicating the size of the private key modulus in bits
56	(38)	UNSIGNED	4	CDDLX_LABEL_LEN	On input, contains the length of the field pointed to by CDDLX_LABEL_PTR, and must be at least 32. On output, contains the length of the label returned at the address in CDDLX_LABEL_PTR, and will be 32 or less.
60	(3C)	UNSIGNED	4	*	Reserved for alignment
64	(40)	ADDRESS	8	CDDLX_LABEL_PTR	Input value specifying the address of output area to be used for the label name.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
72	(48)	CHARACTER	9	CDDLX_RACF_USERID	Input value containing a 1 byte length followed by the certificate owning userid. The combination of the output label and this field uniquely identify a certificate
81	(51)	CHARACTER	3	*	Reserved
84	(54)	UNSIGNED	4	CDDLX_SDN_LEN	On input, contains the length of the output buffer pointed to by CDDLX_SDN_PTR. On output, contains the length of the BER encoded Subject's Distinguished Name returned in CDDLX_SDN_LEN.
88	(58)	ADDRESS	8	CDDLX_SDN_PTR	Input value specifying the address of the output area to be used for the Subject's Distinguished Name.
96	(60)	UNSIGNED	4	CDDLX_RECID_LEN	Output value containing the length of the record ID returned in area pointed to by CDDLX_RECID_PTR, or 0 if no record returned. This value must be preserved for subsequent calls to DataGetNext and DataAbortQuery
100	(64)	UNSIGNED	4	*	Reserved for alignment
104	(68)	ADDRESS	8	CDDLX_RECID_PTR	Input value specifying address of a 246 byte area to contain output record ID data. This value must be preserved for subsequent calls to DataGetNext and DataAbortQuery
112	(70)	CHARACTER	4	CDDLX_STATUS	Certificate status X'80000000' - TRUST X'40000000' - HIGHTRUST X'20000000' - NOTRUST X'00000000' - ANY (input only)

CDDLX_CRT_STATUS

0	(0)	STRUCTURE	0	CDDLX_CRT_STATUS	Parameter list for CheckStatus
0	(0)	UNSIGNED	4	CDDLX_CRT_LEN	Input area containing the length of the certificate area pointed to by CDDLX_CRT_PTR.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	UNSIGNED	4	*	Reserved for alignment
8	(8)	ADDRESS	8	CDDLX_CRT_PTR	Input value specifying address of output certificate data area

CDDLX_GET_UPDATE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	CDDLX_GET_UPDATE	Parameter list for GetUpdateCode
0	(0)	UNSIGNED	4	CDDLX_RING_SEQN	Ring sequence number

CDDLX_ISN

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	CDDLX_ISN	Parm List for IncSerialNum
0	(0)	UNSIGNED	4	CDDLX_ISN_CLEN	Length of DER Encoded input Digital Certificate
4	(4)	UNSIGNED	4	*	Reserved for alignment
8	(8)	ADDRESS	8	CDDLX_ISN_CPTR	Address of DER Encoded input Digital Certificate
16	(10)	CHARACTER	8	CDDLX_ISN_SERNUM	Output - Incremented Serial Number of Digital Certificate

CDDLX_PUT_CERT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	68	CDDLX_PUT_CERT	Parameter list for DataPut.
0	(0)	CHARACTER	4	CDDLX_PCERT_USAGE	4 byte input area containing certificate usage flags x'00000000' - Usage is SITE x'00000002' - Usage is CERTAUTH x'00000008' - Usage is PERSONAL x'80000000' - Usage is defaulted to the usage of the cert x'7FFFFFF5' - reserved bits must be set to 0
4	(4)	UNSIGNED	4	CDDLX_PCERT_DEFAULT	input default cert indicator. Non-zero value indicates input certificate is to made the default cert for the ring, zero indicates to not set the certificate as the default
8	(8)	UNSIGNED	4	CDDLX_PCERT_LEN	Input value containing the length of the certificate pointed to by CDDLX_PCERT_PTR.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	CDDLX_PCERT_PTR	Input value specifying address of input certificate data area

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	UNSIGNED	4	CDDLX_PKEY_LEN	A 4 byte input value contains the length of the private key area pointed to by CDDL_PKEY_PTR. Zero indicates no input private key value.
28	(1C)	UNSIGNED	4	*	Alignment Word.
32	(20)	ADDRESS	8	CDDLX_PKEY_PTR	An input value specifying address of the input private key. Ignored if CDDL_PKEY_LEN is zero.
40	(28)	UNSIGNED	4	CDDLX_PLABEL_LEN	On input, it contains the length of the label to be assigned to the added certificate. On output, it contains the length of the label of the existing certificate if the input certificate is already installed in RACF.
44	(2C)	UNSIGNED	4	*	Alignment Word.
48	(30)	ADDRESS	8	CDDLX_PLABEL_PTR	On input, it contains the address of the label area. On output, it contains the label of the existing certificate if the input certificate is already installed in RACF. The label area must be 32 bytes.
56	(38)	CHARACTER	9	CDDLX_PCERT_USERID	On input, it indicates the owner of the certificate, in the format of a 1 byte length followed by the user ID. On output it contains the owner of the existing certificate if the input certificate is already installed in RACF. The 1 byte length must be 8 and the user ID must be left-justified and padded with blanks.
65	(41)	CHARACTER	3	*	Reserved.

CDDLX_REMOVE_CERT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	CDDLX_REMOVE_CERT	Parameter list for DataRemove.
0	(0)	UNSIGNED	4	CDDLX_RLABEL_LEN	A 4 byte input value contains the length of the label of the certificate to be removed pointed by CDDLX_RLABEL_PTR

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	UNSIGNED	4	*	Reserved for alignment.
8	(8)	ADDRESS	8	CDDLX_RLABEL_PTR	An input value contains the address of the label of the certificate to be removed
16	(10)	CHARACTER	9	CDDLX_RCERT_USERID	A 9 byte input value indicates the owner of the certificate to be removed, in the format of a 1 byte length followed by the user ID. The 1 byte length must be 8 and the user ID must be left-justified and padded with blanks.
25	(19)	CHARACTER	3	*	Reserved.
CDDLX_HANDLE_MAP					
0	(0)	STRUCTURE	28	CDDLX_HANDLE_MAP	Mapping of area pointed to by CDDLX_RES_HANDLE & CDDLX_PREV_HANDLE
0	(0)	ADDRESS	4	CDDLX_TOKEN	Reserved for use by the security server. This value must be preserved for subsequent calls to DataGetNext and DataAbortQuery
8	(8)	UNSIGNED	4	CDDLX_PREDICATES	Input value specifying the selection criteria.
12	(C)	UNSIGNED	4	CDDLX_ATTR_ID	Input value specifying the attribute to query on. Ignored if CDDLX_PREDICATES is 0.
16	(10)	UNSIGNED	4	CDDLX_ATTR_LEN	Input value containing the length of the attributes supplied in CDDLX_ATTR_PTR
24	(18)	ADDRESS	8	CDDLX_ATTR_PTR	Input value containing the address of query attribute data. Type of data supplied determined by CDDLX_ATTR_ID
CDDLX_ALTER_CERT					
0	(0)	STRUCTURE	0	CDDLX_ALTER_CERT	Parameter list for DataAlter
4	(0)	UNSIGNED	4	CDDLX_ALABEL_LEN	A 4 byte input value contains the length of the certificate label pointed to by CDDLX_ALABEL_PTR.
8	(8)	ADDRESS	8	CDDLX_ALABEL_PTR	An input value specifying address of the input certificate label.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	UNSIGNED	4	CDDLX_ANEW_LABEL_LEN	A 4 byte value containing the length of the new. Ignored if CDDLX_ANEW_LEN is zero.
20	(14)	ADDRESS	8	CDDLX_ANEW_LABEL_PTR	A value containing the address of the new certificate label. This field is ignored if CDDLX_ANEW_LABEL_LEN is zero.
28	(1C)	CHARACTER	9	CDDLX_ACERT_USERID	On input, it indicates the owner of the certificate, in the format of a 1 byte length followed by the user ID. On output, it contains the owner of the existing certificate if the input certificate is already installed in RACF. The 1 byte length must be 8, and the user ID must be left-aligned and padded with blanks.
37	(25)	CHARACTER	3	*	Reserved

CDDLX_GET_RING

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	16	CDDLX_GET_RING	Parameter list for GetRingInfo.
0	(0)	CHARACTER	4	CDDLX_RING_SRCH_TYPE	A 4 byte input value which identifies more rings to be returned when both ring owner and ring name are specified X'00000000' - Return just the ring with the specified ring owner and ring name X'00000001' - Return all rings after the ring specified by ring owner and ring name X'00000002' - Return all rings with the same owner after the ring specified by ring owner and ring name X'00000003' - Return all rings with the same name after the ring specified by ring owner and ring name LEA
4	(4)	SIGNED	4	CDDLX_RING_RES_LEN	A 4 byte value containing the size of the field pointed to by Ring_result_ptr.
8	(8)	ADDRESS	8	CDDLX_RING_RES_PTR	Address of input/output area containing the ring result.

GSEC

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC	Mapping for GSEC.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	8	GSEC_OPT@	Address of option word (reserved for future use).
8	(8)	ADDRESS	8	GSEC_FUNC@	Address of 2 byte input function code.
16	(10)	ADDRESS	8	GSEC_FUNC_COUNT@	Address of a word containing the number of function specific parameters.
248	(18)	ADDRESS	8	GSEC_PARMLIST@	Address of function specific parameter list.

GSEC_RETNAME

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_RETNAME	Parameter list for Return name
0	(0)	ADDRESS	8	GSEC_RET_FUNC@	Address of Extract function code
8	(8)	ADDRESS	8	GSEC_RET_CRED_LEN@	Address of 4 byte credential length
16	(10)	ADDRESS	8	GSEC_RET_INPUT_CRED@	Address of input credential
24	(18)	ADDRESS	8	GSEC_RET_OID@	Address of 24 byte area for server use
32	(20)	ADDRESS	8	GSEC_RET_TICKET_PRIN@	Address of string buffer with data area of 240 bytes
40	(28)	ADDRESS	8	GSEC_RET_RC@	Address of 4 byte area for return code

GSEC_GSSAPI_COMMON

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_GSSAPI_COMMON	Common parameter list for all GSS-API invocations. These are the first three parms for all GSS-API calls.
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE

GSEC_INITSECCONTEXT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_INITSECCONTEXT	Parameter list for initiate security context
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function.

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function.
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_INIT_TGT_PRIN@	Address of target principal
40	(28)	ADDRESS	8	GSEC_INIT_REQ_FLAGS@	Address of request flags
48	(30)	ADDRESS	8	GSEC_INIT_REQ_EXPIRE@	Address of requested context expiration
56	(38)	ADDRESS	8	GSEC_INIT_CRED_HNDL@	Address of credential handle
64	(40)	ADDRESS	8	GSEC_INIT_CNTXT_HNDL@	Address of returned context handle
72	(48)	ADDRESS	8	GSEC_INIT_RET_FLAGS@	Address of area for return flags
80	(50)	ADDRESS	8	GSEC_INIT_CNTXT_EXPIRE@	Address of area for actual context expiration
88	(58)	ADDRESS	8	GSEC_INIT_OUT_TOKEN@	Address of returned token
GSEC_CONTSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC_CONTSECCONTEXT	Parameter list for Continue initiation of security context
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function.
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function.
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_CONT_CNTXT_HNDL@	Address of context handle returned from INIT_SEC_CONTEXT
40	(28)	ADDRESS	8	GSEC_CONT_TOKLN@	Address of token length
48	(30)	ADDRESS	8	GSEC_CONT_TOKEN@	Address of acceptor context token
56	(38)	ADDRESS	8	GSEC_CONT_RET_FLAGS@	Address of area for return flags
64	(40)	ADDRESS	8	GSEC_CONT_EXPIRE@	Address of area for actual context expiration
GSEC_ACCSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC_ACCSECCONTEXT	Parameter list for Accept security context
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_ACC_TOKLN@	Address of token length
40	(28)	ADDRESS	8	GSEC_ACC_TOKEN@	Address of initiator context token
48	(30)	ADDRESS	8	GSEC_ACC_CONTXT_HNDL@	Address of returned context handle
56	(38)	ADDRESS	8	GSEC_ACC_PRIN@	Address of area for source principal
64	(40)	ADDRESS	8	GSEC_ACC_RET_FLAGS@	Address of area for return flags
72	(48)	ADDRESS	8	GSEC_ACC_CNTXT_EXPIRE@	Address of area for context expiration
80	(50)	ADDRESS	8	GSEC_ACC_OUT_TOKEN@	Address of returned token
88	(58)	ADDRESS	8	GSEC_ACC_CRED_HNDL@	Address of returned credential handle
GSEC_DELSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC_DELSECCONTEXT	Parameter list for Delete security context
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_DEL_CNTXT_HNDL@	Address of context handle
GSEC_RELCRED					
0	(0)	STRUCTURE	0	GSEC_RELCRED	Parameter list for Releasxe credential
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_REL_CRED_HNDL@	Address of credential handle
GSEC_GETMIC					

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_GETMIC	Paramater list for Get MIC for message
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_GET_CONTXT_HNDL@	Address of context handle
40	(28)	ADDRESS	8	GSEC_GET_MSGLN@	Address of message length
48	(30)	ADDRESS	8	GSEC_GET_MSG@	Address of message
56	(38)	ADDRESS	8	GSEC_GET_OUT_TOK@	Address of returned token
GSEC_VERMIC					
0	(0)	STRUCTURE	0	GSEC_VERMIC	Parameter list for Verify MIC for message
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_VER_CONTXT_HNDL@	Address of context handle
40	(28)	ADDRESS	8	GSEC_VER_MSGLN@	Address of message length
48	(30)	ADDRESS	8	GSEC_VER_MSG@	Address of message
56	(38)	ADDRESS	8	GSEC_VER_TOKLN@	Address of length of token
64	(40)	ADDRESS	8	GSEC_VER_TOKEN@	Address of token
GSEC_WRAPMSG					
0	(0)	STRUCTURE	0	GSEC_WRAPMSG	Parameter list for Wrap a message
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_WRAP_CONTXT_HNDL@	Address of context handle

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
40	(28)	ADDRESS	8	GSEC_WRAP_REQ_FLAG@	Address of confidentiality request flag
48	(30)	ADDRESS	8	GSEC_WRAP_MSGLN@	Address of message length
56	(38)	ADDRESS	8	GSEC_WRAP_MSG@	Address of message
64	(40)	ADDRESS	8	GSEC_WRAP_STATE@	Address of area for confidentiality state
72	(48)	ADDRESS	8	GSEC_WRAP_OUT_TOK@	Address of returned token

GSEC_UNWRAPMSG

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_UNWRAPMSG	Parameter list for Unwrap a msg
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_UNWRAP_CONTXT_HNDL@	Address of context handle
40	(28)	ADDRESS	8	GSEC_UNWRAP_TOKLN@	Address of token length
48	(30)	ADDRESS	8	GSEC_UNWRAP_TOKEN@	Address of token
56	(38)	ADDRESS	8	GSEC_UNWRAP_MSG@	Address of returned unwrapped message
64	(40)	ADDRESS	8	GSEC_UNWRAP_STATE@	Address of area for confidentiality state

GSEC_EXPORTSECCONTEXT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_EXPORTSECCONTEXT	Parameter list for Export security context
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_EXPSEC_CONTXT_HNDL@	Address of context handle
40	(28)	ADDRESS	8	GSEC_EXPSEC_OUT_TOK@	Address of returned token

GSEC_IMPORTSECCONTEXT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC_IMPORTSECCONTEXT	Parameter list for Import security context

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_IMPSEC_TOKLN@	Address of length of token
40	(28)	ADDRESS	8	GSEC_IMPSEC_TOKEN@	Address of token
48	(30)	ADDRESS	8	GSEC_IMPSEC_CONTXT_HNDL@	Address of area for context handle

GSEC_EXPORTCRED

0	(0)	STRUCTURE	0	GSEC_EXPORTCRED	Parameter list for Export credential
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_EXPCRD_CRED_HNDL@	Address of credential handle
40	(28)	ADDRESS	8	GSEC_EXPCRD_OUT_TOK@	Address of returned token

GSEC_IMPORTCRED

0	(0)	STRUCTURE	0	GSEC_IMPORTCRED	Parameter list for Import credential
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_IMPCRD_TOKLN@	Address of length of token
40	(28)	ADDRESS	8	GSEC_IMPCRD_TOKEN@	Address of token
48	(30)	ADDRESS	8	GSEC_IMPCRD_CRED_HNDL@	Address of area for credential handle

GSEC_ACQUIRECRED

0	(0)	STRUCTURE	0	GSEC_ACQUIRECRED	Parameter list for Acquire initiator credential

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	8	GSEC_GSS_FUNC@	Address of GSS-API function code
8	(8)	ADDRESS	8	GSEC_GSS_MAJOR@	Address of area for major code returned by function
16	(10)	ADDRESS	8	GSEC_GSS_MINOR@	Address of area for minor code returned by function
24	(18)	ADDRESS	8	GSEC_GSS_ACEE@	Address of input ACEE
32	(20)	ADDRESS	8	GSEC_ACQCRD_REQ_EXPIRE@	Address of requested credential expiration time
40	(28)	ADDRESS	8	GSEC_ACQCRD_CRED_HNDL@	Address of area for credential handle
48	(30)	ADDRESS	8	GSEC_ACQCRD_PRIN@	Address of area for principal name
56	(38)	ADDRESS	8	GSEC_ACQCRD_CRED_EXPIRE@	Address of area for actual credential expiration time

GSEC_PASSTICKET_PARMS

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	32	GSEC_PASSTICKET_PARMS	Parameter list for PassTicket
0	(0)	ADDRESS	8	GSEC_TICKET_OPTIONS	Address of Passticket function code
8	(8)	ADDRESS	8	GSEC_TICKET	Address of area for Passticket to evaluate or generate
16	(10)	ADDRESS	8	GSEC_USERID	Address of passticket user ID
24	(18)	ADDRESS	8	GSEC_APPLICATION	Address of passticket application

GSEC31_RETNAME

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC31_RETNAME	Parameter list for Return principal name
0	(0)	ADDRESS	4	GSEC31_RET_FUNC@	Address of Extract function code
4	(4)	ADDRESS	4	GSEC31_RET_CRED_LEN@	Address of 4 byte credential length
8	(8)	ADDRESS	4	GSEC31_RET_INPUT_CRED@	Address of input credential
12	(C)	ADDRESS	4	GSEC31_RET_OID@	Address of 24 byte area for server use
16	(10)	ADDRESS	4	GSEC31_RET_TICKET_PRIN@	Address of string buffer with data area of 240 bytes.
20	(14)	ADDRESS	4	GSEC31_RET_RC@	Address of 4 byte area for return code

GSEC31_GSSAPI_COMMON

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC31_GSSAPI_COMMON	Common parameter list for all GSS-API invocations. These are the first three parms for all GSS-API calls.
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
GSEC31_INITSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_INITSECCONTEXT	Parameter list for Initiate® Securty context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_INIT_TGT_PRIN@	Address of target principal
20	(14)	ADDRESS	4	GSEC31_INIT_REQ_FLAGS@	Address of request flags
24	(18)	ADDRESS	4	GSEC31_INIT_REQ_EXPIRE@	Address of requested context expiration
28	(1C)	ADDRESS	4	GSEC31_INIT_CRED_HNDL@	Address of credential handle
32	(20)	ADDRESS	4	GSEC31_INIT_CNTXT_HNDL@	Address of returned context handle
36	(24)	ADDRESS	4	GSEC31_INIT_RET_FLAGS@	Address of area for return flags
40	(28)	ADDRESS	4	GSEC31_INIT_CNTXT_EXPIRE@	Address of area for actual context expiration
44	(2C)	ADDRESS	4	GSEC31_INIT_OUT_TOKEN@	Address of returned token
GSEC31_CONTSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_CONTSECCONTEXT	Parameter list for Continue initiation of security context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_CONT_CNTXT_HNDL@	Address of context handle returned from NIT_SEC_CONTEXT
20	(14)	ADDRESS	4	GSEC31_CONT_TOKLN@	Address of token length
24	(18)	ADDRESS	4	GSEC31_CONT_TOKEN@	Address of acceptor context token
28	(1C)	ADDRESS	4	GSEC31_CONT_RET_FLAGS@	Address of area for return flags
32	(20)	ADDRESS	4	GSEC31_CONT_EXPIRE@	Address of area for actual context expiration
GSEC31_ACCSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_ACCSECCONTEXT	Parameter list for Accept security context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_ACC_TOKLN@	Address of token length
20	(14)	ADDRESS	4	GSEC31_ACC_TOKEN@	Address of initiator context token
24	(18)	ADDRESS	4	GSEC31_ACC_CONTXT_HNDL@	Address of returned context handle
28	(1C)	ADDRESS	4	GSEC31_ACC_PRIN@	Address of area for source principal
32	(20)	ADDRESS	4	GSEC31_ACC_RET_FLAGS@	Address of area for return flags
36	(24)	ADDRESS	4	GSEC31_ACC_CNTXT_EXPIRE@	Address of area for context expiration
40	(28)	ADDRESS	4	GSEC31_ACC_OUT_TOKEN@	Address of returned token
44	(2C)	ADDRESS	4	GSEC31_ACC_CRED_HNDL@	Address of returned credential handle
GSEC31_DELSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_DELSECCONTEXT	Parameter list for Delete security context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(0)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4		Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_DEL_CNTXT_HNDL@	Address of context handle
GSEC31_RELCRED					
0	(0)	STRUCTURE	0	GSEC31_RELCRED	Parameter list for Releasxe credential
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4		Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_REL_CRED_HNDL@	Address of credential handle
GSEC31_GETMIC					
0	(0)	STRUCTURE	0	GSEC31_GETMIC	Parameter list for Get MIC for message
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_GET_CONTXT_HNDL@	Address of context handle
20	(14)	ADDRESS	4	GSEC31_GET_MSGLN@	Address of message length
24	(18)	ADDRESS	4	GSEC31_GET_MSG@	Address of message
28	(1C)	ADDRESS	4	GSEC31_GET_OUT_TOK@	Address of returned token
GSEC31_VERMIC					
0	(0)	STRUCTURE	0	GSEC31_VERMIC	Parameter list for Verify MIC for message
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_VER_CONTXT_HNDL@	Address of context handle

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	GSEC31_VER_MSGLN@	Address of message length
24	(18)	ADDRESS	4	GSEC31_VER_MSG@	Address of message
28	(1C)	ADDRESS	4	GSEC31_VER_TOKLN@	Address of length of token
32	(20)	ADDRESS	4	GSEC31_VER_TOKEN@	Address of token
GSEC31_WRAPMSG					
0	(0)	STRUCTURE	0	GSEC31_WRAPMSG	Parameter list for Wrap a message
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_WRAP_CONTXT_HNDL@	Address of context handle
20	(14)	ADDRESS	4	GSEC31_WRAP_REQ_FLAG@	Address of confidentiality request flag
24	(18)	ADDRESS	4	GSEC31_WRAP_MSGLN@	Address of message length
28	(1C)	ADDRESS	4	GSEC31_WRAP_MSG@	Address of message
32	(20)	ADDRESS	4	GSEC31_WRAP_STATE@	Address of area for confidentiality state
36	(24)	ADDRESS	4	GSEC31_WRAP_OUT_TOK@	Address of returned token
GSEC31_UNWRAPMSG					
0	(0)	STRUCTURE	0	GSEC31_UNWRAPMSG	Parameter list for Unwrap a msg
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_UNWRAP_CONTXT_HNDL@	Address of context handle
20	(14)	ADDRESS	4	GSEC31_UNWRAP_TOKLN@	Address of token length
24	(18)	ADDRESS	4	GSEC31_UNWRAP_TOKEN@	Address of token
28	(1C)	ADDRESS	4	GSEC31_UNWRAP_MSG@	Address of returned message
32	(20)	ADDRESS	4	GSEC31_UNWRAP_STATE@	Address of area for conf-identiality state

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
GSEC31_EXPORTSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_EXPORTSECCONTEXT	Parameter list for Export security context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_EXPSEC_CONTXT_HNDL@	Address of context handle
20	(14)	ADDRESS	4	GSEC31_EXPSEC_OUT_TOK@	Address of returned token
GSEC31_IMPORTSECCONTEXT					
0	(0)	STRUCTURE	0	GSEC31_IMPORTSECCONTEXT	Parameter list for Import security context
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_IMPSEC_TOKLN@	Address of length of token
20	(14)	ADDRESS	4	GSEC31_IMPSEC_TOKEN@	Address of token
24	(18)	ADDRESS	4	GSEC31_IMPSEC_CONTXT_HNDL@	Address of area for cntxt handle
GSEC31_EXPORTCRED					
0	(0)	STRUCTURE	0	GSEC31_EXPORTCRED	Parameter list for Export credential
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_EXPCRD_CRED_HNDL@	Address of credential handle
20	(14)	ADDRESS	4	GSEC31_EXPCRD_OUT_TOK@	Address of returned token
GSEC31_IMPORTCRED					

Table 29. Structure COMX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC31_IMPORTCRED	Parameter list for Import credential
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_IMPCRD_TOKLN@	Address of length of token
20	(14)	ADDRESS	4	GSEC31_IMPCRD_TOKEN@	Address of token
24	(18)	ADDRESS	4	GSEC31_IMPCRD_CRED_HNDL@	Address of area for credential handle

GSEC31_ACQUIRECRED

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC31_ACQUIRECRED	Parameter list for Acquire initiator credential
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code
4	(4)	ADDRESS	4	GSEC31_GSS_MAJOR@	Address of area for major code returned by function
8	(8)	ADDRESS	4	GSEC31_GSS_MINOR@	Address of area for minor code returned by function
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_ACQCRD_REQ_EXPIRE@	Address of requested credential expiration time
20	(14)	ADDRESS	4	GSEC31_ACQCRD_CRED_HNDL@	Address of area for credential handle
24	(18)	ADDRESS	4	GSEC31_ACQCRD_PRIN@	Address of area for principal name
28	(1C)	ADDRESS	4	GSEC31_ACQCRD_CRED_EXPIRE@	Address of area for actual credential expiration time

GSEC31_PASSTICKET_PARMS

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	GSEC31_PASSTICKET_PARMS	Parameter list for PassTicket
0	(0)	ADDRESS	4	GSEC31_TICKET_OPTIONS	Address of Passticket function code
4	(4)	ADDRESS	4	GSEC31_TICKET	Address of area for passticket to evaluate or generate
8	(8)	ADDRESS	4	GSEC31_USERID	Address of passticket user ID
12	(C)	ADDRESS	4	GSEC31_APPLICATION	Address of passticket application

COMX constants

Table 30. Constants for COMX

Len	Type	Value	Name	Description
GSEC function code constants				
2	DECIMAL	1	GSEC_EXTRACT	Context token extraction
2	DECIMAL	1	GSEC_RETURN_NAME	ReturnName - Extract principal name from token and return to invoker.
2	DECIMAL	2	GSEC_GSSAPI	Invoke GSS-API service
2	DECIMAL	1	GSEC_INIT_SEC_CONTEXT	Initiate security context
2	DECIMAL	2	GSEC_CONT_SEC_CONTEXT	Continue initiation of security context
2	DECIMAL	3	GSEC_ACC_SEC_CONTEXT	Accept security context
2	DECIMAL	4	GSEC_DEL_SEC_CONTEXT	Delete security context
2	DECIMAL	5	GSEC_REL_CRED	Release credential
2	DECIMAL	6	GSEC_GET_MIC	Get MIC for message
2	DECIMAL	7	GSEC_VER_MIC	Verify MIC for message
2	DECIMAL	8	GSEC_WRAP_MSG	Wrap message
2	DECIMAL	9	GSEC_UNWRAP_MSG	Unwrap message
2	DECIMAL	10	GSEC_EXPORT_SEC_CONTEXT	Export security context
2	DECIMAL	11	GSEC_IMPORT_SEC_CONTEXT	Import security context
2	DECIMAL	12	GSEC_EXPORT_CRED	Export credential
2	DECIMAL	13	GSEC_IMPORT_CRED	Import credential
2	DECIMAL	14	GSEC_ACQUIRE_CRED	Acquire credential
CDDLX function code constants				
1	DECIMAL	1	CDDLX_DATA_GET_FIRST	DataGetFirst - locate and return the first certificate in the ring matching the specified criteria
1	DECIMAL	2	CDDLX_DATA_GET_NEXT	DataGetNext - locate and return the next certificate in the ring matching the same criteria specified in the previous DataGetFirst/DataGetNext call
1	DECIMAL	3	CDDLX_DATA_ABORT	DataAbortQuery - free resources from previous DataGetFirst or DataGetNext call
1	DECIMAL	4	CDDLX_CHECK_STATUS	CheckStatus - return the TRUST/NOTRUST status for a given certificate

Table 30. Constants for COMX (continued)

Len	Type	Value	Name	Description
1	DECIMAL	5	CDDLX_GET_UPDATE_CODE	GetUpdateCode - returns the sequence number for for a ring
1	DECIMAL	6	CDDLX_INC_SER_NUM	IncSerialNum - Increments the Serial Number for a Digital Certificate
1	DECIMAL	7	CDDLX_NEW_RING	Create a new key ring or re-create a key ring after deleting it
1	DECIMAL	8	CDDLX_DATA_PUT	Add a certificate to RACF and connect it to a key ring
1	DECIMAL	9	CDDLX_DATA_REMOVE	Remove a certificate from a key ring, and optionally delete it from RACF
1	DECIMAL	10	CDDLX_DEL_RING	Delete a key ring
1	DECIMAL	11	CDDLX_DATA_REFRESH	Refresh in-storage certificates in RACF if DIGTCERT is raclisted
1	DECIMAL	12	CDDLX_DATA_ALTER	Alter the label and status of a certificate.
1	DECIMAL	13	CDDLX_GET_RING_INFO	Return the key ring information and the certificate information connected to the ring, or rings.
4	DECIMAL	24	CDDLX_LEN	Constant - length of CDDLX
Private key type constants returned in CDDLX_PK_TYPE				
4	DECIMAL	1	CDDLX_PKCS1	DER encoded PKCS Key
4	DECIMAL	2	CDDLX_ICSF	ICSF key token label
4	DECIMAL	3	CDDLX_PCICC	PCICC key token label
4	DECIMAL	4	CDDLX_DSA	DER encoded DSA key
4	DECIMAL	6	CDDLX_DH	Diffie-Hellman key
4	DECIMAL	7	CDDLX_ECC	ECC key
4	DECIMAL	9	CDDLX_ECC_ICSF	ECC key token label
4	DECIMAL	11	CDDLX_RSA_TKDS	TKDS token for RSA
4	DECIMAL	13	CDDLX_ECC_TKDS	TKDS token for ECC
4	DECIMAL	14	CDDLX_DSA_TKDS	TKDS token for DSA
Input selection criteria constants supplied in CDDLX_PREDICATES				
4	DECIMAL	0	CDDLX_LABEL	The certificate label, up to 32 characters long
4	DECIMAL	1	CDDLX_DEFAULT	The default flag, a 4 byte field specified as 0 for NO, or a nonzero value for YES
4	DECIMAL	3	CDDLX_SUBJECT	BER encoded subject's name

COMX cross reference

Table 31. Cross Reference for COMX

Name	Offset	Hex Value
CDDLX	48	
CDDLX_ABORT	0	
CDDLX_ATT_ALL_KEYTYPES	0	80
CDDLX_ATT_DEL_CERT_ALLRINGS	0	40
CDDLX_ATT_DEL_CERT_FORCE	0	20
CDDLX_ATT_DEL_CERT_TOO	0	80
CDDLX_ATT_HIGHTRUST	0	40
CDDLX_ATT_NOTRUST	0	20
CDDLX_ATT_REUSE_RING	0	80
CDDLX_ATT_SET_MIN_SERIAL	0	80
CDDLX_ATT_SKIPAUTH	0	80
CDDLX_ATT_TRUST	0	80
CDDLX_ATTR_ID	C	
CDDLX_ATTR_LEN	10	
CDDLX_ATTR_PTR	18	
CDDLX_ATTRIBUTES@	8	
CDDLX_ATTRIBUTES_MAP	0	
CDDLX_CERT_DEFAULT	C	
CDDLX_CERT_LEN	10	
CDDLX_CERT_PTR	18	
CDDLX_CERT_USAGE	8	
CDDLX_CRT_LEN	0	
CDDLX_CRT_PTR	8	
CDDLX_CRT_STATUS	0	
CDDLX_FUNC@	0	
CDDLX_GET_CERT	0	
CDDLX_GET_UPDATE	0	
CDDLX_HANDLE_MAP	0	
CDDLX_ISN	0	
CDDLX_ISN_CLEN	0	
CDDLX_ISN_CPTR	8	
CDDLX_ISN_SERNUM	10	
CDDLX_LABEL_LEN	38	
CDDLX_PARMLIST@	28	
CDDLX_PK_BITSIZE	34	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
CDDLX_PK_LEN	20	
CDDLX_PK_PTR	28	
CDDLX_PK_TYPE	30	
CDDLX_PREDICATES	8	
CDDLX_PREV_HANDLE	0	
CDDLX_PCERT_DEFAULT	4	
CDDLX_PCERT_LEN	8	
CDDLX_PCERT_PTR	10	
CDDLX_PCERT_USAGE	0	
CDDLX_PCERT_PTR	10	
CDDLX_PCERT_USERID	38	
CDDLX_PKEY_LEN	18	
CDDLX_PKEY_PTR	20	
CDDLX_PLABEL_LEN	28	
CDDLX_PLABEL_PTR	30	
CDDLX_RACF_USERID	48	
CDDLX_RCERT_USERID	10	
CDDLX_RECID_LEN	60	
CDDLX_RECID_PTR	68	
CDDLX_STATUS	70	
CDDLX_REMOVE_CERT	0	
CDDLX_RES_HANDLE	0	
CDDLX_RING_SEQN	0	
CDDLX_RING@	18	
CDDLX_RLABEL_LEN	0	
CDDLX_RLABEL_PTR	8	
CDDLX_SDN_LEN	54	
CDDLX_SDN_PTR	58	
CDDLX_TOKEN	0	
CDDLX_USERID@	10	
CDDLX_VERSION@	20	
COMPX	0	
COMPX_PARMCNT@	0	
COMPX_RACRC_ALET@	20	
COMPX_RACRC_STOR@	28	
COMPX_RACSC_ALET@	30	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
COMPX_RACSC_STOR@	38	
COMPX_SAFRC_ALET@	10	
COMPX_SAFRC_STOR@	18	
COMPX_SERVR_PARMS	40	
COMPX_WORKA_STOR@	8	
GSEC	40	
GSEC_ACC_CNTXT_EXPIRE@	48	
GSEC_ACC_CONTXT_HNDL@	30	
GSEC_ACC_CRED_HNDL@	58	
GSEC_ACC_OUT_TOKEN@	50	
GSEC_ACC_PRIN@	38	
GSEC_ACC_RET_FLAGS@	40	
GSEC_ACC_TOKEN@	28	
GSEC_ACC_TOKLN@	20	
GSEC_ACCSECCONTEXT	0	
GSEC_ACQCRD_CRED_EXPIRE@	38	
GSEC_ACQCRD_CRED_HNDL@	28	
GSEC_ACQCRD_PRIN@	30	
GSEC_ACQCRD_REQ_EXPIRE@	20	
GSEC_ACQUIRECRED	0	
GSEC_APPLICATION	18	
GSEC_CONT_EXPIRE@	40	
GSEC_CONT_RET_FLAGS@	38	
GSEC_CONT_TOKEN@	30	
GSEC_CONT_TOKLN@	28	
GSEC_CONTSECCONTEXT	0	
GSEC_CONTXT_HNDL@	20	
GSEC_DEL_CNTXT_HNDL@	20	
GSEC_DELSECCONTEXT	0	
GSEC_EXPCRD_CRED_HNDL@	20	
GSEC_EXPCRD_OUT_TOK@	28	
GSEC_EXPORTCRED	0	
GSEC_EXPORTSECCONTEXT	0	
GSEC_EXPSEC_CONTXT_HNDL@	20	
GSEC_EXPSEC_OUT_TOK@	28	
GSEC_FUNC_COUNT@	10	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
GSEC_FUNC@	8	
GSEC_GET_CONTXT_HNDL@	20	
GSEC_GET_MSG@	30	
GSEC_GET_MSGLN@	28	
GSEC_GET_OUT_TOK@	38	
GSEC_GETMIC	0	
GSEC_GSS_ACEE@	18	
GSEC_GSS_FUNC@	0	
GSEC_GSS_MAJOR@	8	
GSEC_GSS_MINOR@	10	
GSEC_GSSAPI_COMMON	0	
GSEC_IMPCRD_CRED_HNDL@	30	
GSEC_IMPCRD_TOKEN@	28	
GSEC_IMPCRD_TOKLN@	20	
GSEC_IMPORTCRED	0	
GSEC_IMPORTSECCONTEXT	0	
GSEC_IMPSEC_CONTXT_HNDL@	18	
GSEC_IMPSEC_TOKEN@	28	
GSEC_IMPSEC_TOKLN@	20	
GSEC_INIT_CNTXT_EXPIRE@	50	
GSEC_INIT_CNTXT_HNDL@	40	
GSEC_INIT_CRED_HNDL@	38	
GSEC_INIT_OUT_TOKEN@	58	
GSEC_INIT_REQ_EXPIRE@	30	
GSEC_INIT_REQ_FLAGS@	28	
GSEC_INIT_RET_FLAGS@	48	
GSEC_INIT_TGT_PRIN@	20	
GSEC_INITSECCONTEXT	0	
GSEC_OPT@	0	
GSEC_PARMLIST@	18	
GSEC_PASSTICKET_PARMS	0	
GSEC_REL_CRED_HNDL@	20	
GSEC_RELCRED	0	
GSEC_RET_CRED_LEN@	8	
GSEC_RET_FUNC@	0	
GSEC_RET_INPUT_CRED@	10	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
GSEC_RET_OID@	18	
GSEC_RET_RC@	28	
GSEC_RET_TICKET_PRIN@	20	
GSEC_RETNAME	0	
GSEC_TICKET_OPTIONS	0	
GSEC_TICKET	8	
GSEC_UNWRAP_CONTXT_HNDL@	20	
GSEC_UNWRAP_MSG@	38	
GSEC_UNWRAP_STATE@	40	
GSEC_UNWRAP_TOKEN@	30	
GSEC_UNWRAP_TOKLN@	28	
GSEC_UNWRAPMSG	0	
GSEC_USERID	10	
GSEC_VER_CONTXT_HNDL@	20	
GSEC_VER_MSG@	30	
GSEC_VER_MSGLN@	28	
GSEC_VER_TOKEN@	40	
GSEC_VER_TOKLN@	38	
GSEC_VERMIC	0	
GSEC_WRAP_CONTXT_HNDL@	20	
GSEC_WRAP_MSG@	38	
GSEC_WRAP_MSGLN@	30	
GSEC_WRAP_OUT_TOK@	48	
GSEC_WRAP_REQ_FLAG@	28	
GSEC_WRAP_STATE@	40	
GSEC_WRAPMSG	0	
GSEC31_ACC_CNTXT_EXPIRE@	24	
GSEC31_ACC_CONTXT_HNDL@	18	
GSEC31_ACC_CRED_HNDL@	2C	
GSEC31_ACC_OUT_TOKEN@	28	
GSEC31_ACC_PRIN@	1C	
GSEC31_ACC_RET_FLAGS@	20	
GSEC31_ACC_TOKEN@	14	
GSEC31_ACC_TOKLN@	10	
GSEC31_ACCSECCONTEXT	0	
GSEC31_ACQCRD_CRED_EXPIRE@	1C	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
GSEC31_ACQCRD_CRED_HNDL@	14	
GSEC31_ACQCRD_PRIN@	18	
GSEC31_ACQCRD_REQ_EXPIRE@	10	
GSEC31_ACQUIRECRED	0	
GSEC31_APPLICATION	C	
GSEC31_CONT_CNTXT_HNDL@	10	
GSEC31_CONT_EXPIRE@	20	
GSEC31_CONT_RET_FLAGS@	1C	
GSEC31_CONT_TOKEN@	18	
GSEC31_CONT_TOKLN@	14	
GSEC31_CONTSECCONTEXT	0	
GSEC31_DEL_CNTXT_HNDL@	10	
GSEC31_EXPCRD_CRED_HNDL@	10	
GSEC31_DELSECCONTEXT	0	
GSEC31_EXPCRD_OUT_TOK@	14	
GSEC31_EXPORTSECCONTEXT	0	
GSEC31_EXPSEC_CONTXT_HNDL@	10	
GSEC31_EXPSEC_OUT_TOK@	14	
GSEC31_GET_CONTXT_HNDL@	10	
GSEC31_GET_MSGLN@	14	
GSEC31_GET_OUT_TOK@	1C	
GSEC31_GETMIC	0	
GSEC31_GSS_ACEE@	C	
GSEC31_GSS_FUNC@	0	
GSEC31_GSS_MAJOR@	4	
GSEC31_GSS_MINOR@	8	
GSEC31_GSSAPI_COMMON	0	
GSEC31_IMPCRD_CRED_HNDL@	18	
GSEC31_IMPCRD_TOKEN@	14	
GSEC31_IMPCRD_TOKLN@	10	
GSEC31_IMPORTCRED	0	
GSEC31_IMPORTSECCONTEXT	0	
GSEC31_IMPSEC_TOKEN@	14	
GSEC31_IMPSEC_TOKLN@	10	
GSEC31_INIT_CNTXT_EXPIRE@	28	
GSEC31_INIT_CNTXT_HNDL@	20	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
GSEC31_INIT_CRED_HNDL@	1C	
GSEC31_INIT_OUT_TOKEN@	2C	
GSEC31_INIT_REQ_EXPIRE@	18	
GSEC31_INIT_REQ_FLAGS@	14	
GSEC31_INIT_RET_FLAGS@	24	
GSEC31_INIT_TGT_PRIN@	10	
GSEC31_INITSECCONTEXT	0	
GSEC31_PASSTICKET_PARMS	0	
GSEC31_REL_CRED_HNDL@	10	
GSEC31_RELCRED	0	
GSEC31_RET_CRED_LEN@	4	
GSEC31_RET_FUNC@	0	
GSEC31_RET_INPUT_CRED@	8	
GSEC31_RET_OID@	C	
GSEC31_RET_RC@	14	
GSEC31_RET_TICKET_PRIN@	10	
GSEC31_RETNAME	0	
GSEC31_TICKET	4	
GSEC31_TICKET_OPTIONS	0	
GSEC31_UNWRAP_CONTXT_HNDL@	10	
GSEC31_UNWRAP_MSG@	1C	
GSEC31_UNWRAP_STATE@	20	
GSEC31_UNWRAP_TOKEN@	18	
GSEC31_UNWRAP_TOKLN@	14	
GSEC31_UNWRAPMSG	0	
GSEC31_USERID	8	
GSEC31_VER_CONTXT_HNDL@	10	
GSEC31_VER_MSG@	18	
GSEC31_VER_MSGLN@	14	
GSEC31_VER_TOKEN@	20	
GSEC31_VER_TOKLN@	1C	
GSEC31_VERMIC	0	
GSEC31_WRAP_CONTXT_HNDL@	10	
GSEC31_WRAP_MSG@	1C	
GSEC31_WRAP_MSGLN@	18	
GSEC31_WRAP_OUT_TOK@	24	

Table 31. Cross Reference for COMX (continued)

Name	Offset	Hex Value
GSEC31_WRAP_REQ_FLAG@	14	
GSEC31_WRAP_STATE@	20	
GSEC31_WRAPMSG	0	

Chapter 14. COMY: 64-bit enabled SAF callable services

COMY programming interface information

COMY is a programming interface.

COMY heading information

Common name: SAF Common Security Parameter List (64 bit)

Macro ID: IRRPCOMY

DSECT name: COMY, PGSN64, RAUX64, PKIS64

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
N/A

Key
Any

Residency
Invoker's primary address space

Size:

Section
Size

COMY
56 bytes

PGSN64
32 bytes

PKIS64
60 bytes

RAUX64
80 bytes

Created by: Invoker of 64-bit enabled callable services

Pointed to by: Address of COMY is passed in register 1 when invoking 64-bit enabled callable services

Serialization: None

Function: Maps the common input parameter list for the 64-bit RACF and SAF callable services routers

COMY mapping

Table 32. Structure COMY

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	COMY	SAF enabled callable services.
0	(0)	ADDRESS	8	COMY_WORKA_STOR@	Address of 1024 byte work area
8	(8)	ADDRESS	8	COMY_SAFRC_ALET@	Address of ALET for SAF return code
16	(10)	ADDRESS	8	COMY_SAFRC_STOR@	Address of SAF return code
24	(18)	ADDRESS	8	COMY_RACRC_ALET@	Address of ALET for RACF return code
32	(20)	ADDRESS	8	COMY_RACRC_STOR@	Address of RACF return code
40	(28)	ADDRESS	8	COMY_RACSC_ALET@	Address of ALET for RACF reason code
48	(30)	ADDRESS	8	COMY_RACSC_STOR@	Address of RACF reason code
PGSN64					
0	(0)	STRUCTURE	24	PGSN64	Mapping for PGSN64
0	(0)	ADDRESS	8	PGSN64_NUM_PARMS@	Address of a fullword containing the total number of parameters included in COMY and PGSN64.
8	(8)	ADDRESS	8	PGSN64_FUNC@	Address of a 2-byte function code. See data area COMP for the function code constants.
16	(10)	ADDRESS	8	PGSN64_FUNC_PARML@	Address of the function specific parameter list corresponding to the function code. See *z/OS Security Server RACF Callable Services* for function specific parameter lists for callable service R_PgmSignVer.
24	(18)	ADDRESS	8	PGSN64_FUNC_ATTR@	Address of a 4-byte variable that contains the attribute flags for the service.
RAUX64					
0	(0)	STRUCTURE	152	RAUX64	
0	(0)	ADDRESS	8	RAUX64_NUM_PARMS@	Address of a fullword containing the total number of parameters included in COMY and RAUX64.
4	(4)	ADDRESS	8	RAUX64_ACEE_ALET@	Address of the ALET for the ACEE structure.
8	(8)	ADDRESS	8	RAUX64_ACEE@	Address of the ACEE structure.
12	(C)	ADDRESS	8	RAUX64_PARM_ALET@	Address of ALET for the remaining parameters.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	8	RAUX64_OPTION_WORD@	Address of a fullword containing binary zeros. The area pointed to by this parameter is reserved for future use.
20	(14)	ADDRESS	8	RAUX64_LINK_VALUE@	Address of 8-byte value used to mark a set of related SMF records.
24	(18)	ADDRESS	8	RAUX64_ATTRIBUTES@	Address of a fullword containing flag values.
28	(1C)	ADDRESS	8	RAUX64_COMPONENT@	Address of an area containing a 4-byte length followed by a component or product name.
32	(20)	ADDRESS	8	RAUX64_FMID@	Address of a 7-byte area containing the FMID for the component or product.
36	(24)	ADDRESS	8	RAUX64_SUBTYPE@	Address of a fullword containing the SMF 83 record subtype.
40	(28)	ADDRESS	8	RAUX64_EVENT@	Address of a fullword containing the event code.
44	(2C)	ADDRESS	8	RAUX64_QUALIFIER@	Address of a fullword containing the qualifier for the event code.
48	(30)	ADDRESS	8	RAUX64_CLASS@	Address of an 8-byte area containing a class name.
52	(34)	ADDRESS	8	RAUX64_RESOURCE@	Address of an area containing a 4-byte length followed by the resource name that is covered by a profile defined in the class specified above.
56	(38)	ADDRESS	8	RAUX64_LOG_STRING@	Address of an area containing a 4-byte length followed by text to be written to the SMF record.
60	(3C)	ADDRESS	8	RAUX64_RELOCATE_COUNT@	Address of a fullword containing the number of relocate sections.
64	(40)	ADDRESS	8	RAUX64_RELOCATE_PTR@	Address of an 8-byte pointer to the array of relocate sections.
68	(44)	ADDRESS	8	RAUX64_MESSAGE_COUNT@	Address of a fullword containing the number of message segments.
72	(48)	ADDRESS	8	RAUX64_MESSAGE_PTR@	Address of an 8-byte pointer to the array of message segments.

PKIS64

0	(0)	STRUCTURE		PKIS64	Mapping for PKIS64.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	8	PKIS64_NUM_PARMS@	Address of an 8-byte variable that contains the number of parameters that follow in the non-request specific portion of the R_PKIServ callable service parameter list.
8	(8)	ADDRESS	8	PKIS64_FUNC@	Address of a 2-byte variable that contains the code of the requested function. Constants for the function codes are below.
16	(10)	ADDRESS	8	PKIS64_ATTRIBUTES@	Address of a 4-byte variable that contains attribute flags for the service.
24	(18)	STRUCTURE	8	PKIS64_LOG_STRING@	Address of a variable-length area that contains the LOG string passed to RACROUTE (1-byte for the length followed by up to 255 bytes for the LOG string.
32	(20)	ADDRESS	8	PKIS64_PARM_VER@	Address of a 4-byte variable that contains the version number of the function specific parameter list (PKIS64_FUNC_PARML@).
40	(28)	ADDRESS	8	PKIS64_FUNC_PARML@	Address of the FSPL, Function Specific Parameter List. (FSPL is the parameter list that corresponds to the function code.)
48	(30)	UNSIGNED	4	PKIS64_LEN	Length of the PKIS64 structure with no CA domain.
52	(34)	ADDRESS	8	PKIS64_CA_DOMAIN@	Address of the name of the PKI Services certificate authority instance invoked.
60	(3C)	UNSIGNED	4	PKIS64_TOT_LEN	Length of the PKIS64 structure with CA domain.

PKIS64_GENCERT_MAP

0	(0)	STRUCTURE	32	PKIS64_GENCERT_MAP	Function Specific Parameter List (FSPL) for GENCERT.
0	(0)	CHARACTER	8	PKIS64_GENC_EYECATCH	Eyecatcher for the GENCERT FSPL left-aligned blank filled string containing the text 'GENCERT'.
8	(8)	UNSIGNED	4	PKIS64_GENC_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation PList .

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_GENC_CERTPL@	Address of the Digital Certificate generation PList.
24	(18)	ADDRESS	8	PKIS64_GENC_CERTID@	Address of a 57-Byte area that contains the output Certificate Request ID (the first byte of this area contains the length of the Output Certificate Request ID).
PKIS64_CPL_ENTRY					
0	(0)	STRUCTURE	16	PKIS64_CPL_ENTRY	Certificate Request PList entry (a single GENCERT PList entry).
0	(0)	CHARACTER	12	PKIS64_CPL_FIELD_NAME	Certificate Request Field Name.
12	(C)	UNSIGNED	4	PKIS64_CPL_FIELD_LEN	Length (in bytes) of the Request Field Name value.
16	(10)	CHARACTER	*	PKIS64_CPL_FIELD_VALUE	Request Field Name value.
PKIS64_ATTRIBUTES_MAP					
0	(0)	BITSTRING	4	PKIS64_ATTRIBUTES_MAP	Certificate Request Attribute mapping.
		1...		PKIS64_SYNCH_CREATE	Synchronous generation flag.
0	(0)	.111 1111 1111 1111	*	*	Reserved
PKIS64_EXPORT_MAP					
0	(0)	STRUCTURE	40	PKIS64_EXPORT_MAP	Function Specific Parameter List (FSPL) for EXPORT.
0	(0)	CHARACTER	8	PKIS64_EXP_EYECATCH	Eyecatcher for the EXPORT FSPL left-aligned blank filled string containing the text 'EXPORT '
8	(8)	UNSIGNED	4	PKIS64_EXP_CERTAN_LEN	Length (in bytes) of the preallocated Certificate Anchor area
12	(C)	UNSIGNED	4	*	Reserved for alignment
16	(10)	ADDRESS	8	PKIS64_EXP_CERTAN@	Address of the preallocated Certificate Anchor area
24	(18)	ADDRESS	8	PKIS64_EXP_CERTID@	Address of a 57-Byte area that contains the Input Certificate Request ID that is used to locate the Digital Certificate to be EXPORTed (the first byte of this area contains the length of the Input Certificate Request ID)

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	8	PKIS64_EXP_KEYID@	Address of a 41-byte area that contains the KeyId - the hash of the public key generated by PKI Services. The first byte of this area contains the length of the KeyId. This field is used to export a recovery certificate, the only case that the first byte is 40. In all other cases, the length byte should be 0.

PKIS64_QUERYREQS_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	56	PKIS64_QUERYREQS_MAP	Function Specific Parameter List (FSPL) for QUERYREQS.
0	(0)	CHARACTER	8	PKIS64_QRYR_EYECATCH	Eyecatcher for the QUERYREQS FSPL left-aligned blank filled string containing user specified text, e.g. 'QUERYRQS'.
8	(8)	SIGNED	4	PKIS64_QRYR_RESULTL_LEN	Length (in bytes) of the pre-allocated Results List area.
12	(C)	SIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_QRYR_RESULTL@	Address of the pre-allocated Results List area.
24	(18)	ADDRESS	8	PKIS64_QRYR_CERTID@	Address of a 57-Byte area that contains the Input Certificate Request ID that is to be used as a starting point for this query. Only Request IDs located after this Cert ID is returned. The first byte of area contains the length of the Input Certificate Request ID.
32	(20)	SIGNED	4	PKIS64_QRYR_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Results List area. Zero indicates no limit.
36	(24)	SIGNED	4	PKIS64_QRYR_CRIT_STATUS	Value indicating the request status to use as search criteria.
40	(28)	SIGNED	4	PKIS64_QRYR_CRIT_DAYS	Value indicating the recent activity time to use as additional search criteria.
44	(2C)	SIGNED	4	*	Reserved for alignment
48	(30)	ADDRESS	8	PKIS64_QRYR_CRIT_NAME@	Address of a 33-Byte area that contains the input requestor's name to be used as additional search criteria. The first byte of this area contains the length of the input requestor's name

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
PKIS64_RESL_ENTRY					
0	(0)	STRUCTURE		PKIS64_RESL_ENTRY	Results List Entry for QUERYREQS or QUERYCERTS (a single entry).
0	(0)	UNSIGNED	1	PKIS64_RESL_ENTRY_LEN	Length (in bytes) of the Results List entry value.
1	(1)	CHARACTER	1	PKIS64_RESL_ENTRY_VALUE	Results list entry value.
PKIS64_REQDETAILS_MAP					
0	(0)	STRUCTURE	48	PKIS64_REQDETAILS_MAP	Function Specific Parameter List (FSPL) for REQDETAILS.
0	(0)	CHARACTER	8	PKIS64_REQD_EYECATCH	Eyecatcher for the REQDETAILS FSPL left-aligned blank filled string containing user specified text, e.g. 'REQDTAIL'.
8	(8)	UNSIGNED	4	PKIS64_REQD_SUML_LEN	Length (in bytes) of the pre-allocated Summary List area.
12	(C)	UNSIGNED	4	*	Reserved for alignment
16	(10)	ADDRESS	8	PKIS64_REQD_SUML@	Address of the pre-allocated Summary List area.
24	(18)	UNSIGNED	4	PKIS64_REQD_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PList details area.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_REQD_CERTPL@	Address of the pre-allocated Digital Certificate generation type Plist details area.
40	(28)	ADDRESS	8	PKIS64_REQD_CERTID@	Address of a 57-Byte area that contains the Input Certificate Request ID from which to extract the data(the first byte of this area contains the length of the Input Certificate Request ID).
PKIS64_SUML_ENTRY					
0	(0)	STRUCTURE	0	PKIS64_SUML_ENTRY	Summary List Entry for REQDETAILS, CERTDETAILS, or VERIFY (a single entry).
0	(0)	UNSIGNED	1	PKIS64_SUML_ENTRY_LEN	Length (in bytes) of the Summary List entry value.
1	(1)	CHARACTER	1	PKIS64_SUML_ENTRY_VALUE	Summary list entry value.
PKIS64_MODIFYREQS_MAP					
0	(0)	STRUCTURE	80	PKIS64_MODIFYREQS_MAP	Function Specific Parameter List (FSPL) for MODIFYREQS.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	CHARACTER	8	PKIS64_MODR_EYECATCH	Eyecatcher for the MODIFYREQS FSPL left-aligned blank filled string containing user specified text, e.g. 'MODREQS'.
8	(8)	UNSIGNED	4	PKIS64_MODR_ACTION	4 byte binary value indicating the action to be taken against the certificate requests.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_MODR_COMMENT@	Address of a 65-Byte area that contains the comment data associated with this action. The first byte of this area contains the length of the actual comment. If the length is x'00' then no comment is to be recorded.
24	(18)	UNSIGNED	4	PKIS64_MODR_CERTIDL_LEN	Length (in bytes) of the Certificate Request id list on input.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_MODR_CERTIDL@	Address of the area containing 1 or more certificate request ids that are to be modified by this request.
40	(28)	UNSIGNED	4	PKIS64_MODR_CERTPL_LEN	Length (in bytes) of the Certificate modification plist area.
44	(2C)	UNSIGNED	4	*	Reserved for alignment
48	(30)	ADDRESS	8	PKIS64_MODR_CERTPL@	Address of the area which is the Certificate modification Plist
56	(38)	UNSIGNED	4	PKIS64_MODR_ERRL_LEN	Length (in bytes) of the error list data return area
60	(3C)	UNSIGNED	4	*	Reserved for alignment
64	(40)	ADDRESS	8	PKIS64_MODR_ERRL@	Address of the area where error list information is to be returned.
72	(48)	ADDRESS	8	PKIS64_MODR_TSTAMP@	Address of a 20-byte area that can contain a time stamp in 'YYYY/MM/DD hh:mm:ss' format. The first byte of this area contains the length of the time stamp. If the length is x'00' then no time stamp is provided. Acceptable lengths are '00'x and '13'x only.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
PKIS64_QUERYCERTS_MAP					
0	(0)	STRUCTURE	56	PKIS64_QUERYCERTS_MAP	Function Specific Parameter List (FSPL) for QUERYCERTS.
0	(0)	CHARACTER	8	PKIS64_QRYC_EYECATCH	Eyecatcher for the QUERYCERTS FSPL left-aligned blank filled string containing user specified text, e.g. 'QUERYCTS'.
8	(8)	UNSIGNED	4	PKIS64_QRYC_RESULTL_LEN	Length (in bytes) of the pre-allocated Results list area.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_QRYC_RESULTL@	Address of the pre-allocated Results List area.
24	(18)	ADDRESS	8	PKIS64_QRYC_SERIALNUM@	Address of a 17-Byte area that contains the Input Certificate Serial Number that is used as a starting point for this query. Only Certificates located after this serial number is returned. The first byte of area contains the length of the Input Certificate Serial Number .
32	(20)	UNSIGNED	4	PKIS64_QRYC_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Result List area. Zero indicates no limit.
36	(24)	UNSIGNED	4	PKIS64_QRYC_CRIT_STATUS	Value indicating the request status to use as search criteria.
40	(28)	SIGNED	4	PKIS64_QRYC_CRIT_DAYS	Value indicating the recent activity to use as additional search criteria.
44	(2C)	UNSIGNED	4	*	Reserved for alignment.
48	(30)	ADDRESS	8	PKIS64_QRYC_CRIT_NAME@	Address of a 33-Byte area that contains the input requestor's name to be used as additional search criteria. The first byte of this area contains the length of the input requestor name.
PKIS64_CERTDETAILS_MAP					
0	(0)	STRUCTURE	48	PKIS64_CERTDETAILS_MAP	Function Specific Parameter List (FSPL) for CERTDETAILS.
0	(0)	CHARACTER	8	PKIS64_CRTD_EYECATCH	Eyecatcher for the CERTDETAILS FSPL left-aligned blank filled string containing user specified text, e.g. 'CRTDETLS'.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	UNSIGNED	4	PKIS64_CRTD_SUML_LEN	Length (in bytes) of the pre-allocated Summary List area.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_CRTD_SUML@	Address of the pre-allocated Summary List area.
24	(18)	UNSIGNED	4	PKIS64_CRTD_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PList area.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_CRTD_CERTPL@	Address of the pre-allocated Digital Certificate generation type PList details area.
40	(28)	ADDRESS	8	PKIS64_CRTD_SERIALNUM@	Address of a 17-Byte area that contains the Input Certificate Serial Number from which to extract the data. The first byte of this area contains the length of the Certificate Serial Number.

PKIS64_MODIFYCERTS_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	72	PKIS64_MODIFYCERTS_MAP	Function Specific Parameter List (FSPL) for MODIFYCERTS.
0	(0)	CHARACTER	8	PKIS64_MODC_EYECATCH	Eyecatcher for MODIFYCERTS FSPL left-aligned blank filled string containing user specified text, e.g. 'MODCERTS'.
8	(8)	UNSIGNED	4	PKIS64_MODC_ACTION	4 byte binary value indicating the action to be taken against the certificates identified by the serial number list.
12	(C)	UNSIGNED	4	*	Reserved for alignment
16	(10)	ADDRESS	8	PKIS64_MODC_COMMENT@	Address of a 65-Byte area that contains the comment data associated with this action. The first byte of this area contains the length of the actual comment. If the length is x'00' then no comment will be recorded.
24	(18)	UNSIGNED	4	PKIS64_MODC_SERIALNUMSL_LEN	Length (in bytes) of the Serial Numbers list on input.
28	(1C)	UNSIGNED	4		Reserved for alignment
32	(20)	ADDRESS	8	PKIS64_MODC_SERIALNUMSL@	Address of the area containing 1 or more Serial Numbers of certificates that are to be modified by this request.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
40	(28)	UNSIGNED	4	PKIS64_MODC_REASON	4 byte binary value indicating the reason for certificate revocation
44	(2C)	UNSIGNED	4	*	AReserved for alignment
48	(30)	ADDRESS	8	PKIS64_MODC_REQUESTOREMAIL@	Address of a 33-byte area that contains the new email address of the requester. The first byte of this area contains the length of the email address.
56	(38)	UNSIGNED	4	PKIS64_MODC_ERRL_LEN	Length (in bytes) of the error list data return area
60	(3C)	UNSIGNED	4	*	Reserved for alignment
64	(40)	ADDRESS	8	PKIS64_MODC_ERRL@	Address of the area which is the error list

PKIS64_QRECOVER_MAP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	PKIS64_QRECOVER_MAP	Mapping for QRECOVER function specific parameter list.
0	(0)	CHARACTER	8	PKIS64_QREC_EYECATCH	Eyecatcher, 8 characters left aligned blank filled. Actual value set by invoker, for example 'QRECOVER'.
8	(8)	UNSIGNED	4	PKIS64_QREC_RESULTL_LEN	Length (in bytes) of the pre-allocated Results List area.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_QREC_RESULTL@	Address of the pre-allocated Results List area.
24	(18)	UNSIGNED	4	PKIS64_QREC_NUMENTRIES	Value indicating the maximum number of entries to be returned in the Results List area. Zero indicates no limit.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_QREC_CRIT_EMAIL@	Address of a 33-byte area that contains the input requestor's email address as search criteria.
40	(28)	ADDRESS	8	PKIS64_QREC_CRIT_PASS@	Address of a 33-byte area that contains the input pass phrase as search criteria.

PKIS64_SNID_ENTRY

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		PKIS64_SNID_ENTRY	Certificate ID or Serial Number (a single entry).
0	(0)	UNSIGNED	1	PKIS64_SNID_ENTRY_LEN	Length (in bytes) of the Cert ID or Serial Number entry value.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
1	(1)	CHARACTER	1	PKIS64_SNID_ENTRY_VALUE(0)	Cert ID or Serial Number entry value.
PKIS64_VERIFY_MAP					
0	(0)	STRUCTURE	56	PKIS64_VERIFY_MAP	Function Specific Parameter List (FSPL) for VERIFY.
0	(0)	CHARACTER	8	PKIS64_VERF_EYECATCH	Eyecatcher for the VERIFY FSPL left-aligned blank filled string containing user specified text, for example, 'VERIFY'.
8	(8)	UNSIGNED	4	PKIS64_VERF_SUML_LEN	Length (in bytes) of the pre-allocated Summary List area.
12	(C)	UNSIGNED	4	•	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_VERF_SUML@	Address of the pre-allocated Summary List area.
24	(18)	UNSIGNED	4	PKIS64_VERF_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation type PList details area.
28	(1C)	UNSIGNED	4	•	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_VERF_CERTPL@	Address of the pre-allocated Digital Certificate generation type PList details area.
40	(28)	UNSIGNED	4	PKIS64_VERF_CERT_LEN	Length (in bytes) of the Cert area containing the certificate to verify.
44	(2C)	UNSIGNED	4	*	Reserved for alignment.
48	(30)	ADDRESS	8	PKIS64_VERF_CERT@	Address of the area containing the Certificate to verify.
PKIS64_REVOKE_MAP					
0	(0)	STRUCTURE	24	PKIS64_REVOKE_MAP	Function Specific Parameter List (FSPL) for REVOKE.
0	(0)	CHARACTER	8	PKIS64_REVK_EYECATCH	Eyecatcher for the REVOKE FSPL left-aligned blank filled string containing user specified text, for example, 'REVOKE'.
8	(8)	UNSIGNED	4	PKIS64_REVK_REASON	4 byte binary value indicating the reason for the certificate revocation.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_REVK_SERIALNUM@	Address of a 17-Byte area that contains the Input Certificate Serial Number to be revoked. The first byte of this area contains the length of the Input Certificate Serial Number.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
PKIS64_RENEW_MAP					
0	(0)	STRUCTURE	40	PKIS64_RENEW_MAP	Function Specific Parameter List (FSPL) for GENRENEW and REQRENEW.
0	(0)	CHARACTER	8	PKIS64_RENW_EYECATCH	Eyecatcher for the GENRENEW FSPL and the REQRENEW FSPL left-alligned blank filled string containing the text 'RENEW'.
8	(8)	UNSIGNED	4	PKIS64_RENW_CERTPL_LEN	Length (in bytes) of the Digital Certificate generation PList.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_RENW_CERTPL@	Address of the Digital Certificate generation PList.
24	(18)	ADDRESS	8	PKIS64_RENW_CERTID@	Address of a 57-Byte area that contains the Output Certificate Request ID (the first byte of this area contains the length of the Output Certificate Request ID).
32	(20)	ADDRESS	8	PKIS64_RENW_SERIALNUM@	Address of a 17-Byte area that contains the input Certficate Serial Number to be renewed. The first byte of this area contains the length of the Input Certificate Serial Number.
PKIS64_RESPOND_MAP					
0	(0)	STRUCTURE	40	PKIS64_RESPOND_MAP	Mapping for RESPOND function Specific parameter List.
0	(0)	CHARACTER	8	PKIS64_RESP_EYECATCH	Eyecatcher for the RESPOND FSPL. left-aligned blank filled string containing user specified text, for example, 'RESPOND'.
8	(8)	UNSIGNED	4	PKIS64_RESP_RESPONSE_LEN	Length(in bytes) of the Response area on input to RESPOND.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_RESP_RESPONSE@	Address of the storage area in which the R_PKISERV service stores the results of the RESPOND if the service was able to successfully retrieve the data.
24	(18)	UNSIGNED	4	PKIS64_RESP_REQUEST_LEN	Length (in bytes) of the Request area containing the input request data.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	8	PKIS64_RESP_REQUEST@	Address of the area containing the Request.
PKIS64_SCEPREQ_MAP					
0	(0)	STRUCTURE	40	PKIS64_SCEPREQ_MAP	Function Specific Parm List (FSPL) for SCEPREQ.
0	(0)	CHARACTER	8	PKIS64_SCEP_EYECATCH	Eyecatcher for the SCEPREQ FSPL. left-aligned, blank filled. Actual value set by the invoker, for example, 'SCEPREQ'.
8	(8)	UNSIGNED	4	PKIS64_SCEP_RESPONSE_LEN	4 byte area which is the length of the pre-allocated storage of the response area on input to SCEPREQ.
12	(C)	UNSIGNED	4	*	Reserverd for alignment.
16	(10)	ADDRESS	8	PKIS64_SCEP_RESPONSE@	The address of the storage area in which the R_PKIServ service stores the results of the SCEPREQ if the service was able to successfully retrieve the data.
24	(18)	UNSIGNED	4	PKIS64_SCEP_REQUEST_LEN	Length (in bytes) of the Request area containing the input request data.
28	(1C)	UNSIGNED	4	*	Reserved for alignment.
32	(20)	ADDRESS	8	PKIS64_SCEP_REQUEST@	Address of the area containing the Request to process.
PKIS64_PREREG_MAP					
0	(0)	STRUCTURE	32	PKIS64_PREREG_MAP	Mapping for PREREGISTER function specific parm List.
0	(0)	CHARACTER	8	PKIS64_PREG_EYECATCH	Eyecatcher for the SCEPREQ PRERIGISTER FSPL. left-aligned, blank filled string.Actual value set by invoker, for example, 'PREREG '
8	(8)	UNSIGNED	4	PKIS64_PREG_CERTPL_LEN	Length (in bytes) of the preregistration parameter list.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	PKIS64_PREG_CERTPL@	Address of the preregistration parameter list.
24	(18)	ADDRESS	8	PKIS64_PREG_CERTID@	Address of a 57-Byte area that contains the Output Certificate Request ID (the first byte of this area contains the length of the Output Certificate Request ID).
SMOP64					

Table 32. Structure COMY (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	80	SMOP64	
0	(0)	ADDRESS	8	SMOP64_NUM_PARMS@	Address of fullword containing the total number of parameters.
8	(8)	ADDRESS	8	SMOP64_FUNC@	Address of fullword containing function code.
16	(10)	ADDRESS	8	SMOP64_OPTIONS@	Address of the 4 byte options value. x'00000001' - Execute x'00000002' - Precheck x'00000004' - Stop on error (unsupported in RACF) x'00000008' - Clear Sensitive information. These options can be combined.
24	(18)	ADDRESS	8	SMOP64_REQUEST_LEN@	Address of fullword containing the request length, in bytes.
32	(20)	ADDRESS	8	SMOP64_REQUEST@	Address of the request.
40	(28)	ADDRESS	8	SMOP64_HANDLE@	Address of 64 byte handle.
48	(30)	ADDRESS	8	SMOP64_USERID@	Address of 9 byte area for the RACF user ID (1-byte for the length followed by up to 8 bytes for the RACF user ID itself).
56	(38)	ADDRESS	8	SMOP64_ACEE@	Address of input ACEE.
64	(40)	ADDRESS	8	SMOP64_RESULT_LEN@	Address of fullword containing the length of the RESULT buffer.
72	(48)	ADDRESS	8	SMOP64_RESULT@	Address of the result buffer.

COMY constants

Table 33. Constants for COMY

Len	Type	Value	Name	Description
PKIS64 function codes				
1	HEX	1	PKIS64_GENCER	Generates a basic X.509 V3 digital certificate.
1	HEX	2	PKIS64_EXPORT	Export a certificate by certificate request ID.
1	HEX	3	PKIS64_QUERYREQS	Query on one or more certificate requests.
1	HEX	4	PKIS64_REQDETAILS	Extract details of a single certificate request.

Table 33. Constants for COMY (continued)

Len	Type	Value	Name	Description
1	HEX	5	PKIS64_MODIFYREQS	Modify information on one or more certificate requests.
1	HEX	6	PKIS64_QUERYCERTS	Query on one or more certificates.
1	HEX	7	PKIS64_CERTDETAILS	Extract details of a single certificate.
1	HEX	8	PKIS64_MODIFYCERTS	Modify information on one or more certificates.
1	HEX	9	PKIS64_REQCERT	Request certificate for approval.
1	HEX	0A	PKIS64_VERIFY	Request certificate verification.
1	HEX	0B	PKIS64_REVOKE	Request certificate revocation.
1	HEX	0C	PKIS64_GENRENEW	Request auto-approved certificate renewal.
1	HEX	0D	PKIS64_REQRENEW	Request certificate renewal.
1	HEX	0E	PKIS64_RESPOND	Report certificate status through OCSP.
1	HEX	0F	PKIS64_SCEPREQ	Submit a request to PKI Services using SCEP.
1	HEX	10	PKIS64_PREREGISTER	Preregister a SCEP user.
1	HEX	11	PKIS64_QRECOVER	Query for recovering a certificate and private key.

COMY cross reference

Table 34. Cross Reference for COMY

Name	Offset	Hex Value
COMY	0	
COMY_RACRC_ALET@	18	
COMY_RACRC_STOR@	20	
COMY_RACSC_ALET@	28	
COMY_RACSC_STOR@	30	
COMY_SAFRC_ALET@	8	
COMY_SAFRC_STOR@	10	
COMY_WORKA_STOR@	0	
PGSN64	0	
PGSN64_FUNC@	8	
PGSN64_FUNC_ATTRS@	18	
PGSN64_FUNC_PARML@	10	

Table 34. Cross Reference for COMY (continued)

Name	Offset	Hex Value
PGSN64_NUM_PARMS@	0	
PKIS64	0	
PKIS64_ATTRIBUTES@	10	
PKIS64_ATTRIBUTES_MAP	0	
PKIS64_CA_DOMAIN@	34	
PKIS64_CERTDETAILS_MAP	0	
PKIS64_CPL_ENTRY	0	
PKIS64_CPL_FIELD_LEN	C	
PKIS64_CPL_FIELD_NAME	0	
PKIS64_CPL_FIELD_VALUE	10	
PKIS64_CRTD_CERTPL@	20	
PKIS64_CRTD_CERTPL_LEN	18	
PKIS64_CRTD_EYECATCH	0	
PKIS64_CRTD_SERIALNUM@	28	
PKIS64_CRTD_SUML_LEN	8	
PKIS64_CRTD_SUML@	10	
PKIS64_EXP_CERTAN_LEN	8	
PKIS64_EXP_CERTAN@	10	
PKIS64_EXP_CERTID@	18	
PKIS64_EXP_EYECATCH	0	
PKIS64_EXP_KEYID@	20	
PKIS64_EXPORT_MAP	0	
PKIS64_FUNC@	8	
PKIS64_FUNC_PARML@	28	
PKIS64_GENC_EYECATCH	0	
PKIS64_GENC_CERTID@	18	
PKIS64_GENC_CERTPL_LEN	8	
PKIS64_GENC_CERTPL@	10	
PKIS64_GENCERT_MAP	0	
PKIS64_LOG_STRING@	18	
PKIS64_MODC_ACTION	8	
PKIS64_MODC_COMMENT@	10	
PKIS64_MODC_ERRL@	40	
PKIS64_MODC_ERRL_LEN	38	
PKIS64_MODC_EYECATCH	0	
PKIS64_MODC_REASON	28	

Table 34. Cross Reference for COMY (continued)

Name	Offset	Hex Value
PKIS64_MODC_REQUESTOREMAIL@	30	
PKIS64_MODC_SERIALNUMSL_LEN	18	
PKIS64_MODC_SERIALNUMSL@	20	
PKIS64_MODIFYCERTS_MAP	0	
PKIS64_MODIFYREQS_MAP	0	
PKIS64_MODR_ACTION	8	
PKIS64_MODR_CERTIDL@	20	
PKIS64_MODR_CERTIDL_LEN	18	
PKIS64_MODR_CERTPL@	30	
PKIS64_MODR_CERTPL_LEN	28	
PKIS64_MODR_COMMENT@	10	
PKIS64_MODR_ERRL@	40	
PKIS64_MODR_ERRL_LEN	38	
PKIS64_MODR_EYECATCH	0	
PKIS64_MODR_TSTAMP@	48	
PKIS64_LEN	30	
PKIS64_NUM_PARMS@	0	
PKIS64_PARM_VER@	20	
PKIS64_PREG_CERTID@	18	
PKIS64_PREG_CERTPL@	10	
PKIS64_PREG_CERTPL_LEN	8	
PKIS64_PREG_EYECATCH	0	
PKIS64_PREREG_MAP	0	
PKIS64_QREC_CRIT_EMAIL@	20	
PKIS64_QREC_CRIT_PASS@	28	
PKIS64_QREC_EYECATCH	0	
PKIS64_QREC_NUMENTRIES	18	
PKIS64_QREC_RESULTL_LEN	8	
PKIS64_QREC_RESULTL@	10	
PKIS64_QRECOVER_MAP	0	
PKIS64_QRYC_CRIT_DAYS	28	
PKIS64_QRYC_CRIT_NAME@	30	
PKIS64_QRYC_CRIT_STATUS	24	
PKIS64_QRYC_EYECATCH	0	
PKIS64_QRYC_NUMENTRIES	20	
PKIS64_QRYC_RESULTL_LEN	8	

Table 34. Cross Reference for COMY (continued)

Name	Offset	Hex Value
PKIS64_QRYC_RESULTL@	10	
PKIS64_QRYC_SERIALNUM@	18	
PKIS64_QRYR_CERTID@	18	
PKIS64_QRYR_CRIT_DAYS	28	
PKIS64_QRYR_CRIT_NAME@	30	
PKIS64_QRYR_CRIT_STATUS	24	
PKIS64_QRYR_EYECATCH	0	
PKIS64_QRYR_NUMENTRIES	20	
PKIS64_QRYR_RESULTL_LEN	8	
PKIS64_QRYR_RESULTL@	10	
PKIS64_QUERYCERTS_MAP	0	
PKIS64_QUERYREQS_MAP	0	
PKIS64_RENEW_MAP	0	
PKIS64_RENW_CERTID@	18	
PKIS64_RENW_CERTPL_LEN	8	
PKIS64_RENW_CERTPL@	10	
PKIS64_RENW_EYECATCH	0	
PKIS64_RENW_SERIALNUM@	20	
PKIS64_REQD_CERTID@	28	
PKIS64_REQD_CERTPL_LEN	18	
PKIS64_REQD_CERTPL@	20	
PKIS64_REQD_EYECATCH	0	
PKIS64_REQD_SUML_LEN	8	
PKIS64_REQD_SUML@	10	
PKIS64_REQDETAILS_MAP	0	
PKIS64_RESL_ENTRY	0	
PKIS64_RESL_ENTRY_LEN	0	
PKIS64_RESL_ENTRY_VALUE	1	
PKIS64_RESP_EYECATCH	0	
PKIS64_RESP_REQUEST_LEN	18	
PKIS64_RESP_REQUEST@	20	
PKIS64_RESP_RESPONSE_LEN	8	
PKIS64_RESP_RESPONSE@	10	
PKIS64_RESPOND_MAP	0	
PKIS64_REVK_EYECATCH	0	
PKIS64_REVK_REASON	8	

Table 34. Cross Reference for COMY (continued)

Name	Offset	Hex Value
PKIS64_REVK_SERIALNUM@	10	
PKIS64_REVOKE_MAP	0	
PKIS64_SCEP_EYECATCH	0	
PKIS64_SCEP_REQUEST_LEN	18	
PKIS64_SCEP_REQUEST@	20	
PKIS64_SCEP_RESPONSE_LEN	8	
PKIS64_SCEP_RESPONSE@	10	
PKIS64_SCEPREQ_MAP	0	
PKIS64_SNID_ENTRY	0	
PKIS64_SNID_ENTRY_LEN	0	
PKIS64_SNID_ENTRY_VALUE	1	
PKIS64_SUML_ENTRY	0	
PKIS64_SUML_ENTRY_LEN	0	
PKIS64_SUML_ENTRY_VALUE	1	
PKIS64_TOT_LEN	3C	
PKIS64_VERF_CERT_LEN	28	
PKIS64_VERF_CERT@	30	
PKIS64_VERF_CERTPL_LEN	18	
PKIS64_VERF_CERTPL@	20	
PKIS64_VERF_EYECATCH	0	
PKIS64_VERF_SUML_LEN	8	
PKIS64_VERF_SUML@	10	
PKIS64_VERIFY_MAP	0	
RAUX64	0	
RAUX64_ACEE_ALET@	4	
RAUX64_ACEE@	8	
RAUX64_ATTRIBUTES@	18	
RAUX64_CLASS@	30	
RAUX64_COMPONENT@	1C	
RAUX64_EVENT@	28	
RAUX64_FMID@	20	
RAUX64_LINK_VALUE@	14	
RAUX64_LOG_STRING@	38	
RAUX64_MESSAGE_COUNT@	44	
RAUX64_MESSAGE_PTR@	48	
RAUX64_NUM_PARMS@	0	

Table 34. Cross Reference for COMY (continued)

Name	Offset	Hex Value
RAUX64_OPTION_WORD@	10	
RAUX64_PARM_ALET@	C	
RAUX64_QUALIFIER@	2C	
RAUX64_RELOCATE_COUNT@	3C	
RAUX64_RELOCATE_PTR@	40	
RAUX64_RESOURCE@	34	
RAUX64_SUBTYPE@	24	
SMOP64	0	
SMOP64_ACEE@	38	
SMOP64_FUNC@	8	
SMOP64_HANDLE@	28	
SMOP64_NUM_PARMS@	0	
SMOP64_OPTIONS@	10	
SMOP64_REQUEST_LEN@	18	
SMOP64_REQUEST@	20	
SMOP64_RESULT_LEN@	40	
SMOP64_RESULT@	48	
SMOP64_USERID@	30	

Chapter 15. CRED: z/OS UNIX System Services Credential Structure

CRED programming interface information

CRED is a programming interface.

CRED heading information

Common name: z/OS UNIX System Services credential structure

Macro ID: IRRPCRED

DSECT name: CRED

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: CRED (Offset: 0, Length: 4)

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: 128 bytes

Created by: Invoker of z/OS UNIX security functions

Pointed to by: IRRPCOMP, the common SAF/RACF parameter list for z/OS UNIX security functions

Serialization: N/A

Function: Maps the structure of the audit data passed by the file system to the security function

CRED mapping

Table 35. Structure CRED

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	128	CRED	Open/MVS Credential structure
0	(0)	CHARACTER	4	CREDID	Control Block ID
4	(4)	UNSIGNED	1	CREDSP	Subpool number for this CRED
5	(5)	UNSIGNED	3	CREDLEN	Length of this CRED
8	(8)	UNSIGNED	1	CREDVER	Version number

Table 35. Structure CRED (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
9	(9)	UNSIGNED	1	CREDUTYPE	User type X'01' - RACF defined local user User type X'02' - System function caller Treated like superuser
10	(A)	UNSIGNED	2	CREDFUNCTION	Audit Function Code - identifies the syscall being processed
12	(C)	UNSIGNED	1	CREDNAMEFLG	Name Flag. Indicates which name is being checked. X'01' - the first (or only) name X'02' - the second name
13	(D)	BITSTRING	1	CREDLFSFLAGS	Flag area reserved for LFS (see BPXZCRED for bit definitions)
14	(E)	CHARACTER	2	*	Reserved
16	(10)	CHARACTER	12	CREDPN1(0)	First Path Name: The path name specified by the user on syscall. For rename and link this is the OLD path name.
16	(10)	UNSIGNED	2	*	Reserved
18	(12)	UNSIGNED	2	CREDPN1LEN	Length of first path name
20	(14)	ADDRESS	4	CREDPN1ALET	ALET for first path name
24	(18)	ADDRESS	4	CREDPN1ADDR	Address of first path name
28	(1C)	CHARACTER	12	CREDFN1	First File Name: The name of the part of the first path name that is being checked on the current OPEN/MVS-RACF call.
28	(1C)	UNSIGNED	2	*	Reserved
30	(1E)	UNSIGNED	2	CREDFN1LEN	Length of first file name
32	(20)	ADDRESS	4	CREDFN1ALET	ALET for first file name
36	(24)	ADDRESS	4	CREDFN1ADDR	Address of first file name
40	(28)	CHARACTER	12	CREDPN2	Second Path Name: The new path name specified by the user on a rename or link syscall or the content of the symbolic link on a symlink syscall or the PDSE/x data set name on a mount, unmount, or chmount.
40	(28)	SIGNED	2		Reserved
42	(2A)	UNSIGNED	2	CREDPN2LEN	Length of second path name
44	(2C)	ADDRESS	4	CREDPN2ALET	ALET for second path name
48	(30)	ADDRESS	4	CREDPN2ADDR	Address of second path name

Table 35. Structure CRED (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
52	(34)	CHARACTER	12	CREDFN2(0)	Second File Name: The name of the part of the second path name that is being checked on the current OPEN/MVS RACF call.
52	(34)	UNSIGNED	2	*	Reserved
54	(36)	UNSIGNED	2	CREDFN2LEN	Length of second file name
56	(38)	ADDRESS	4	CREDFN2ALET	ALET for second file name
60	(3C)	ADDRESS	4	CREDFN2ADDR	Address of second file name
64	(40)	CHARACTER	4	*	Reserved
68	(44)	UNSIGNED	4	CREDACLALET	For ck_access, the ALET of access ACL
72	(48)	CHARACTER	4		Reserved
76	(4C)	ADDRESS	4	CREDACLPTR	For ck_access, the address of access for ACL
76	(4C)	ADDRESS	4	CREDACLINFOPTR	Pointer to the CredAclInfo structure
80	(50)	CHARACTER	8	CRED_SECLABEL	Security label to be set
88	(58)	CHARACTER	4	CRED_ACEEPTR	ACEE address for SRB requests
92	(5C)	CHARACTER	8	CRED_ROSECLABEL	Security Label for Read Only files
100	(64)	BITSTRING	1	CREDMISCFLAGS	Miscellaneous flags
101	(65)	CHARACTER	3	*	Reserved
104	(68)	CHARACTER	8	CREDFS(0)	For ck_access, file system name area
104	(68)	ADDRESS	4	CREDFSALET	For ck_access, ALET of the file system
108	(6C)	ADDRESS	4	CREDFSADDR	For ck_access, address of a 44-byte area containing the file system name, padded with blanks
112	(70)	CHARACTER	16	*	Reserved
128	(80)	CHARACTER	0	*	End of Cred

CREDACLINFO

0	(0)	STRUCTURE	80	CREDACLINFO	CREDACLINFO structure
0	(0)	UNSIGNED	4	CREDACCACLLEN	Length of access ACL buffer
4	(4)	UNSIGNED	4	CREDACCACLALET	ALET of access ACL buffer
8	(8)	CHARACTER	4	*	Reserved
12	(C)	ADDRESS	4	CREDACCACL	Address of access ACL
16	(10)	UNSIGNED	4	CREDFILEMODELACLLEN	Length of ACL buffer
20	(14)	UNSIGNED	4	CREDFILEMODELACLALET	ALET of ACL buffer
24	(18)	CHARACTER	4	*	Reserved

Table 35. Structure CRED (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	CREDFILEMODELACL	Address of ACL
32	(20)	UNSIGNED	4	CREDDIRMODELACLLEN	Length of file model ACL buffer
36	(24)	UNSIGNED	4	CREDDIRMODELACLALET	ALET of file model ACL buffer
40	(28)	CHARACTER	4	*	Reserved
44	(2C)	ADDRESS	4	CREDDIRMODELACL	Address of directory model ACL
48	(30)	CHARACTER	4	*	Reserved
52	(34)	UNSIGNED	4	CREDPFILEMODELACLALET	ALET of parent's file model ACL
56	(38)	CHARACTER	4	*	Reserved
60	(3C)	ADDRESS	4	CREDPFILEMODELACL	Address of parent's file model ACL
64	(40)	CHARACTER	4	*	Reserved
68	(44)	UNSIGNED	4	CREDPDIRMODELACLALET	ALET of parent's directory model ACL
72	(48)	CHARACTER	4	*	Reserved
76	(4C)	ADDRESS	4	CREDPDIRMODELACL	Address of parent's directory model ACL
80	(50)	CHARACTER	0	*	

CRED cross reference

Table 36. Cross Reference for CRED

Name	Offset	Hex Value
CRED	0	
CREDFN1	1C	
CREDACCACL	C	
CREDACCACLALET	4	
CREDACCACLLEN	0	
Cred_ACEEptr	58	
CREDACLALET	44	
CREDACLINFO	0	
CREDACLINFOPTR	4C	
CREDACLPTR	48	
CREDDIRMODELACL	2C	
CREDDIRMODELACLALET	24	
CREDDIRMODELACLLEN	20	
CREDFILEMODELACL	1C	
CREDFILEMODELACLALET	14	
CREDFILEMODELACLLEN	10	

Table 36. Cross Reference for CRED (continued)

Name	Offset	Hex Value
CREDFN1ADDR	24	
CREDFN1ALET	20	
CREDFN1LEN	1E	
CREDFN2	34	
CREDFN2ADDR	3C	
CREDFN2ALET	38	
CREDFN2LEN	36	
CREDFS	68	
CREDFSADDR	6C	
CREDFSALET	68	
CREDFUNCTION	A	
CREDID	0	
CREDLEN	5	
CREDLFSFLAGS	D	
CREDMISCFLAGS	64	
CREDNAMEFLG	C	
CREDPDIRMODELACL	4C	
CREDPDIRMODELACLALET	44	
CREDPFILEMODELACL	3C	
CREDPFILEMODELACLALET	34	
CREDPN1	10	
CREDPN1ADDR	18	
CREDPN1ALET	14	
CREDPN1LEN	12	
CREDPN2	28	
CREDPN2ADDR	30	
CREDPN2ALET	2C	
CREDPN2LEN	2A	
Cred_ROSeclabel	5C	
Cred_Seclabel	50	
CREDSP	4	
CREDUTYPE	9	
CREDVER	8	

Chapter 16. CREI: z/OS UNIX System Services Credential Structure for IPC

CREI programming interface information

CREI is a programming interface.

CREI heading information

Common name:	z/OS UNIX System Services credential structure for IPC
Macro ID:	IRRPCREI
DSECT name:	CREI
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	CREI (Offset: 0, Length: 4)
Storage attributes:	**Subpool** Determined by caller **Key** Determined by caller **Residency** Determined by caller
Size:	64 bytes
Created by:	Invoker of z/OS UNIX security functions
Pointed to by:	IRRPCOMP, the common SAF/RACF parameter list for z/OS UNIX security functions
Serialization:	N/A
Function:	Maps the structure of the security credential area used in the IPC system to pass data from the kernel

CREI mapping

Table 37. Structure CREI

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	64	CREI	
0	(0)	CHARACTER	4	CREIID	Control Block ID
4	(4)	UNSIGNED	1	CREISP	Subpool number for this CREI
5	(5)	UNSIGNED	3	CREILEN	Length of this CREI
8	(8)	UNSIGNED	1	CREIVER	Version number
9	(9)	UNSIGNED	1	CREIUTYPE	User type X'01' - RACF defined local user X'02' - System function caller Treated like superuser

Table 37. Structure CREI (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
10	(A)	UNSIGNED	2	CREIFUNCTION	Audit Function Code - identifies the syscall being processed
12	(C)	UNSIGNED	4	CREIIPCKEY	IPC key of the IPC service that is being checked.
16	(10)	UNSIGNED	4	CREIIPCID	IPC identifier of the IPC service that is being checked.
20	(14)	CHARACTER	44	*	Reserved
64	(40)	CHARACTER		*	End of the CREI

CREI constants

Table 38. Constants for CREI

Len	Type	Value	Name	Description
CreiUType value definitions				
1	DECIMAL	1	CREIUTYPERACF	RACF-defined local user
1	DECIMAL	2	CREIUTYPESYS	System function caller
Miscellaneous constants				
4	CHARACTER	CREI	CREI#ID	Acronym
1	DECIMAL	3	CREI#SP	Subpool
3	DECIMAL	64	CREI#LEN	Length
1	DECIMAL	1	CREI#VERSION1	Version 1
1	DECIMAL	1	CREI#VERSION	Current® Version

CREI cross reference

Table 39. Cross Reference for CREI

Name	Offset	Hex Value
CREI	0	
CREIFUNCTION	A	
CREIID	0	
CREIIPCID	10	
CREIIPCKEY	C	
CREILEN	5	
CREISP	4	
CREIUTYPE	9	
CREIVER	8	

Chapter 17. DAUT: RACROUTE REQUEST=DIRAUTH Parameter List (Request Section)

DAUT programming interface information

DAUT is a programming interface.

DAUT heading information

Common name: Request-specific portion of the RACROUTE REQUEST=DIRAUTH parameter list

Macro ID: IRRPDAUT

DSECT name: DAUTPARM

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Varies

Created by: RACROUTE REQUEST=DIRAUTH macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=DIRAUTH routine

DAUT mapping

Table 40. Structure DAUTPARM

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	36	DAUTPARM	IRRPDAUT - Dirauth Parmlist
0	(0)	BITSTRING	1	DAUTLOGP	Auditing option flags
		1...		DAUTASIS	1=ASIS
		.1..		DAUTNFAI	1=NOFAIL
		..1.		DAUTNONE	=NONE
		...1 1111		*	Reserved for LOG keyword
1	(1)	UNSIGNED	1	DAUTPVER	Parmlist version number 00 - original version, length 8 01 - Version 1, variable length

Table 40. Structure DAUTPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
2	(2)	UNSIGNED	2	DAUTPLEN	Parmlist length if other than Version 00
4	(4)	ADDRESS	4	DAUTRTOK	Message RTOKEN address
8	(8)	BITSTRING	1	DAUTFLG1	Option Flags
		1...		DAUTMAC	1=MAC type checking
		.1..		DAUTRVRS	1=Reverse MAC type checking
		..1.		DAUTEQUM	1= Equal MAC type checking
		...1 1111		*	Reserved
9	(9)	BITSTRING	1	DAUTFLG2	Option Flags
		1...		DAUTREAD	"X'80'" 1=READ access
		.1..		DAUTRDWR	"X'40'" 1=READWRITE access
		..1.		DAUTWRIT	"X'20'" 1=WRITE access
		...1 1111		*	Reserved
10	(A)	CHARACTER	2	*	Reserved
12	(C)	ADDRESS	4	DAUTCLAS	Address of an area containing an 8-byte class name
16	(10)	ADDRESS	4	DAUTRSEC	Address of an area containing an 8-byte resource security label
20	(14)	ADDRESS	4	DAUTUSEC	Address of an area containing an 8-byte user security label
24	(18)	ADDRESS	4	DAUTACEE	Address of an area containing an ACEE
28	(1C)	ADDRESS	4	DAUTALET	Address of an area containing a 4-byte ALET for the ACEE
32	(20)	ADDRESS	4	DAUTLSTR	Address of an area containing a 1 byte length followed by a log string

DAUT cross reference

Table 41. Cross Reference for DAUT

Name	Offset	Hex Value
DAUTACEE	18	
DAUTALET	1C	
DAUTASIS	0	80
DAUTCLAS	C	
DAUTEQUM	8	20
DAUTFLG1	8	
DAUTFLG2	9	
DAUTLOGP	0	

Table 41. Cross Reference for DAUT (continued)

Name	Offset	Hex Value
DAUTLSTR	20	
DAUTMAC	8	80
DAUTNFAI	0	40
DAUTNONE	0	20
DAUTPARM	0	
DAUTPLEN	2	
DAUTPVER	1	
DAUTRDWR	9	40
DAUTREAD	9	80
DAUTRSEC	10	
DAUTRVRS	8	40
DAUTRTOK	4	
DAUTUSEC	14	
DAUTWRIT	9	20

Chapter 18. DEXP: Data Encryption Exit Parameter List

DEXP programming interface information

DEXP is a programming interface.

DEXP heading information

Common name: RACF data encryption exit parameter list

Macro ID: ICHDEXP

DSECT name: DEXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
229

Key
0

Size: 32 bytes

Created by: RACXTRT SVC processor

Pointed to by: R1 at entry to ICHDEX01 and ICHDEX11

Serialization: None

Function: Contains the list of addresses passed to the RACF data encryption installation exit

DEXP mapping

Table 42. Structure DEXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		DEXPL	
ICHDEXP					
0	(0)	ADDRESS	4	DEXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	DEXFUNC	Function code address: points to an area containing a 1-word function code whose possible values are as follows:
		…. ….		DEXCRYPT	"X'00000000'" 0 means encrypt the data.

Table 42. Structure DEXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... .1..		DEXCOMP	"X'00000004'" 4 means compare the data.
8	(8)	ADDRESS	4	DEXDLGTH	Data length address: points to a fullword area containing the length of the clear text data and encrypted data fields.
12	(C)	ADDRESS	4	DEXCTEXT	Clear text address: points to an area containing the clear text data (the parameter DEXDLGTH at offset 8 gives the length of the data).
16	(10)	ADDRESS	4	DEXETEXT	Encrypted data address: for the compare function, points to an area containing the encrypted version that is to be compared against the clear text. For the encrypt function, the exit returns the encrypted data to the area pointed to by this address.
20	(14)	ADDRESS	4	DEXTMPLC	Template code address: points to a 1-byte area containing the code describing the template type of the field being worked on. The possible values are:
		1		DEXGROUP	"X'01'" 1 - Group
		.... ..1.		DEXUSER	"X'02'" 2 - User
		11		DEXCNECT	"X'03'" 3 - Connect
		.... .1..		DEXDS	"X'04'" 4 - Data set
		.... .1.1		DEXGENRL	"X'05'" 5 - General
24	(18)	ADDRESS	4	DEXTMPLN	Template name address: points to an 8-byte area containing the template name of the field being worked on.
28	(1C)	ADDRESS	4	DEXPROF	Profile name address: points to an 8-byte area containing (the first part of) the profile name.

DEXP cross reference

Table 43. Cross Reference for DEXP

Name	Offset	Hex Value
DEXCNECT	14	3
DEXCOMP	4	4
DEXCRYPT	4	0

Table 43. Cross Reference for DEXP (continued)

Name	Offset	Hex Value
DEXCTEXT	C	
DEXDLGTH	8	
DEXDS	14	4
DEXETEXT	10	
DEXFUNC	4	
DEXGENRL	14	5
DEXGROUP	14	1
DEXLEN	0	
DEXPL	0	
DEXPROF	1C	
DEXTMPLC	14	
DEXTMPLN	18	
DEXUSER	14	2

Chapter 19. DSDT: Data Set Descriptor Table

DSDT programming interface information

DSDT is **NOT a programming interface.**

DSDT heading information

Common name:	RACF Data Set Descriptor Table
Macro ID:	ICHPDSDT
DSECT name:	DSDT
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	DSDT (Offset: 0, Length: 4)
Storage attributes:	**Subpool** 241 (ECSA) **Key** 0 **Residency** N/A
Size:	168 bytes plus 896 bytes for each RACF primary data set
Created by:	ICHSEC00
Pointed to by:	RCVTDSDT field of the RCVT data area
Serialization:	None
Function:	Describes primary and backup RACF data sets

DSDT mapping

Table 44. Structure DSDT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	DSDT	LOCATED THROUGH RCVT
0	(0)	CHARACTER	4	DSDTID	EBCIDIC ID
4	(4)	SIGNED	4	DSDTNUM	NUMBER OF ENTRIES IN TABLE
8	(8)	ADDRESS	4	DSDTDSDX	POINTER TO EXTENSION
12	(C)	UNSIGNED	1	DSDTVRSN	VERSION NUMBER OF DSDT
13	(D)	CHARACTER	1	DSDTFLAG	DSDT FLAGS OR INDICATORS
		1...		DSDTPFMT	FLAG FOR PRIMARY FORMAT 0 - Indicates OLD format or no PRIMARY data sets specified at IPL. 1 - Indicates RDS Format

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		DSDTBFMT	FLAG FOR BACKUP FORMAT 0 - Indicates OLD format or no BACKUP data sets specified at IPL. 1 - Indicates RDS Format
		..1.		DSDTPXST	EXISTANCE OF PRIMARY DB'S 0 - No PRIMARY Data sets specified at IPL. 1 - At least one PRIMARY data set specified at IPL.
		...1		DSDTBXST	EXISTANCE OF BACKUP DB'S 0 - No BACKUP Data sets specified at IPL. 1 - At least one BACKUP data set specified at IPL.
		 1111		*	Reserved
14	(E)	CHARACTER	2	*	DOUBLE-WORD ALIGNMENT
16	(10)	CHARACTER	8	*	RESERVED FOR EXPANSION

GRS latch sets: Each set contains 1 latch for each data set in the RACF database for a total for DSDTNUM latches.

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	CHARACTER	48	DSDTPLNM	GRS latch set name for Primary
72	(48)	CHARACTER	8	DSDTPLTK	GRS latch set token for Primary
80	(50)	CHARACTER	48	DSDTBLNM	GRS latch set name for Backup
128	(80)	CHARACTER	8	DSDTBLTK	GRS latch set token for Backup

RACF Sysplex Data Sharing fields:

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
136	(88)	BITSTRING	1	DSDTDSFL	Data Sharing Flags
		1...		DSDTDSRQ	
		.111 1111		*	Reserved
137	(89)	UNSIGNED	1	DSDTDSMO	RACF Data Sharing Mode, (See DSDTDSMO constants below for valid values.
138	(8A)	CHARACTER	2	*	Word alignment
140	(8C)	SIGNED	4	*	Reserved

Entry for Primary Data Set:

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
144	(90)	CHARACTER	352	DSDTENTY(*)	ENTRY FOR DATA SET INFORMATION
144	(90)	CHARACTER	176	DSDTPRIM	ENTRY FOR PRIMARY DATA SET
144	(90)	ADDRESS	4	DSDPDCB	PTR DCB PRIMARY RACF DATA SET

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
148	(94)	ADDRESS	4	DSDPDEB	PTR DEB PRIMARY RACF DATA SET
152	(98)	ADDRESS	4	DSDPINDX	PTR TO IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE. ZERO IF NO RESIDENT BLOCKS FOR THE PRIMARY DATA SET.
156	(9C)	ADDRESS	4	DSDPHDR	PTR RACF IN-STORAGE DS HEADER RECORD OR ZERO IF PRIMARY RACF DATA SET IS ON A SHARED DEVICE
160	(A0)	ADDRESS	4	DSDPRUCB	PTR UCB PRIMARY RACF DATA SET
164	(A4)	SIGNED	4	DSDPXLEN	LENGTH OF IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE FOR THE PRIMARY RACF DATA SET
168	(A8)	ADDRESS	4	DSDPBAM	LOCATES IN-STORAGE BAM INFORMATION FOR PRIMARY DATA SET
172	(AC)	ADDRESS	1	DSDPDSNL	LENGTH OF PRIMARY RACF DATA SET NAME
173	(AD)	BITSTRING	1	DSDPSTAT	PRIMARY RACF DATA SET STATUS
		1...		DSDPACTV	THIS DATA SET IS ACTIVE
		.1..		DSDPPRIM	THIS DATA SET IS A PRIMARY
		..1.		DSDPMSTR	THIS DATA SET IS THE MASTER RACF DATA SET. ITS ICB CONTAINS STATUS OPTIONS.
		...1		DSDPRFSH	REFRESH ICB
		.... 1...		DSDPSHR	DATA SET IS (OR WAS) SHARED
		.... .1..		DSDPALTI	ALTERI REQUESTS ARE BACKED-UP
		.... ..1.		DSDPDAT	IN-STORAGE BLOCKS CAN BE DATA BLOCKS
174	(AE)	SIGNED	2	DSDPNREC	# RECORDS PER TRACK PRIMARY DATA SET
176	(B0)	UNSIGNED	1	DSDPRXNO	# IN-STORAGE BUFFERS OR RESIDENT INDEX BLOCKS
177	(B1)	CHARACTER	44	DSDPDSN	DSN OF RACF PRIMARY DATA SET
221	(DD)	UNSIGNED	1	DSDPDSNO	DATA SET SEQUENCE NUMBER
222	(DE)	SIGNED	2	DSDPCBLN	LENGTH OF PRIMARY DSDE

RACF Sysplex Data Sharing fields:

224	(E0)	CHARACTER	76	DSDPSDS	

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
224	(E0)	CHARACTER	16	DSDPSNAM	XES Structure Name for this database
240	(F0)	CHARACTER	16	DSDPCTOK	XES Connect token
256	(100)	CHARACTER	16	DSDPCNAM	XES Connection Name
256	(100)	CHARACTER	4	DSDPCNP	XES Connection Name Prefix
260	(104)	CHARACTER	3	DSDPCNDS	XES Connection Name Dataset
263	(107)	CHARACTER	1	DSDPCNSL	XES Connection Name Slash
264	(108)	CHARACTER	8	DSDPCNSN	XES Connection Name sufffix (System Name)
272	(110)	CHARACTER	12	DSDPVTOK	XES Vector token
284	(11C)	SIGNED	4	DSDPVLEN	XES Vector token length
288	(120)	UNSIGNED	1	DSDPCONS	XES Connect Status, CONSTAT (See CONSTAT Constants below for valid values)
289	(121)	BITSTRING	1	DSDPRBLD	XES Rebuild Status Flags
		1...		DSDPRBIP	Rebuild In Progress
		.111 1111		*	Reserved
290	(122)	BITSTRING	1	DSDPCACF	Caching Flags.
		1...		DSDPDSCM	IXLCACHE message issued
		.1..		DSDPDSVM	IXLVECTR message issued
		..11 1111		*	Reserved
291	(123)	CHARACTER	1	*	Boundary Alignment
		1...		DSDPVDEA	RVARY Deact/Dealloc
		.1..		DSDPCKVL	RVARY Make Volid check
		..1.		DSDPSWIT	RVARY AutoSwitch failed
		...1		DSDPSERR	AutoSwitch non-retryable
		 1111		*	Reserved
292	(124)	CHARACTER	8	*	Reserved 2 words
300	(12C)	CHARACTER	8	DSDPDDNM	Dynamic allocation DDNAME
308	(134)	CHARACTER	612	*	Reserved
164	(A4)	CHARACTER	6	DSDEVOL	Propagated RVARY VOL ID

RACF VSAM Data Set information:

170	(AA)	ADDRESS	4	DSDEDBObj	RACF VSAM data set information
174	(AE)	CHARACTER	1	DSDEFlags	Flags
		1...		DSDEVSAM	1- Data set is VSAM 0- Data set is not VSAM

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
175	(AF)	CHARACTER	1	*	Reserved
Entry for Backup Data Set:					
320	(140)	CHARACTER	176	DSDTBACK	ENTRY FOR BACKUP DATA SET
320	(140)	ADDRESS	4	DSDBDCB	PTR DCB OF BACK-UP DATA SET
324	(144)	ADDRESS	4	DSDBDEB	PTR DEB OF BACK-UP DATA SET
328	(148)	ADDRESS	4	DSDBINDX	PTR TO IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE. ZERO IF NO RESIDENT BLOCKS FOR THE BACK-UP RACF DATA SET
332	(14C)	ADDRESS	4	DSDBHDR	PTR RACF IN-STORAGE DS HEADER RECORD OR ZERO IF BACK-UP RACF DATA SET IS ON A SHARED DEVICE
336	(150)	ADDRESS	4	DSDBRUCB	PTR UCB OF BACK-UP DATA SET
340	(154)	SIGNED	4	DSDBXLEN	LENGTH OF IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE FOR THE BACK-UP RACF DATA SET
344	(158)	ADDRESS	4	DSDBBAM	LOCATES IN-STORAGE BAM INFORMATION FOR BACK-UP DATA SET
348	(15C)	ADDRESS	1	DSDBDSNL	LENGTH OF BACK-UP DATA SET NAME
349	(15D)	BITSTRING	1	DSDBSTAT	STATUS OF BACK-UP DATA SET
		1...		DSDBACTV	THIS DATA SET IS ACTIVE
		.1..		DSDBPRIM	THIS DATA SET IS A PRIMARY
		..1.		DSDBMSTR	THIS DATA SET IS THE MASTER RACF DATA SET. ITS ICB CONTAINS STATUS OPTIONS.
		...1		DSDBRFSH	REFRESH ICB
		.... 1...		DSDBSHR	DATA SET IS (OR WAS) SHARED
		.... .1..		DSDBALTI	ALTERI REQUESTS ARE BACKED-UP
		.... ..1.		DSDBDAT	IN-STORAGE BLOCKS CAN BE DATA BLOCKS
350	(15E)	SIGNED	2	DSDBNREC	# RECORDS PER TRACK BACK-UP DATA SET
352	(160)	UNSIGNED	1	DSDBRXNO	# IN STORAGE BUFFERS OR RESIDENT INDEX BLOCKS
353	(161)	CHARACTER	44	DSDBDSN	DSN OF BACK-UP RACF DATA SET
397	(18D)	UNSIGNED	1	DSDBDSNO	DATA SET SEQUENCE NUMBER

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
398	(18E)	SIGNED	2	DSDBCBLN	LENGTH OF BACKUP DSDE
RACF Sysplex Data Sharing fields:					
400®	(190)	CHARACTER	76	DSDBSDS	
400	(190)	CHARACTER	16	DSDBSNAM	XES Structure Name for this DB
416	(1A0)	CHARACTER	16	DSDBCTOK	XES Connect token
432	(1B0)	CHARACTER	16	DSDBCNAM	XES Connection Name
432	(1B0)	CHARACTER	4	DSDBCNP	XES Connection Name Prefix
436	(1B4)	CHARACTER	3	DSDBCNDS	XES Connection Name Dataset
439	(1B7)	CHARACTER	1	DSDBCNSL	XES Connection Name Slash
440	(1B8)	CHARACTER	8	DSDBCNSN	XES Connection Name suffix (System Name)
448	(1C0)	CHARACTER	12	DSDBVTOK	XES Vector token
460	(1CC)	SIGNED	4	DSDBVLEN	XES Vector token length
464	(1D0)	UNSIGNED	1	DSDBCONS	XES Connect Status, CONSTAT (See CONSTAT Constants below for valid values)
465	(1D1)	BITSTRING	1	DSDBRBLD	XES Rebuild Status Flags
		1...		DSDBRBIP	Rebuild In Progress
		.111 1111		*	Reserved
466	(1D2)	BITSTRING	1	DSDBCACF	Caching Flags.
		1...		DSDBDSCM	IXLCACHE message issued
		.1..		DSDBDSVM	IXLVECTR message issued
		..11 1111		*	Reserved
467	(1D3)	CHARACTER	1	*	Boundary Alignment
468	(1D4)	CHARACTER	8	*	Reserved 2 words
476	(1DC)	CHARACTER	8	DSDBDDNM	Dynamic allocation DDNAME
484	(1E4)	CHARACTER	12	*	Reserved 3 words
Based mapping of a single data set:					
0	(0)	STRUCTURE	176	DSDE	ENTRY FOR DATA SET
0	(0)	ADDRESS	4	DSDEDCB	PTR DCB FOR DATA SET
4	(4)	ADDRESS	4	DSDEDEB	PTR DEB FOR DATA SET
8	(8)	ADDRESS	4	DSDEINDX	PTR TO IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE. ZERO IF NO RESIDENT BLOCKS FOR THE DATA SET

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	DSDEHDR	PTR RACF IN-STORAGE DS HEADER RECORD OR ZERO IF DATA SET IS ON A SHARED DEVICE
16	(10)	ADDRESS	4	DSDERUCB	PTR UCB FOR DATA SET
20	(14)	SIGNED	4	DSDEXLEN	LENGTH OF IN-STORAGE INDEX RELATED CONTROL BLOCKS FOR DATA SET
24	(18)	ADDRESS	4	DSDEBAM	LOCATES IN-STORAGE BAM INFORMATION FOR DATA SET
28	(1C)	ADDRESS	1	DSDEDSNL	LENGTH OF DATA SET NAME
29	(1D)	BITSTRING	1	DSDESTAT	DATA SET STATUS
		1...		DSDEACTV	THIS DATA SET IS ACTIVE
		.1..		DSDEPRIM	THIS DATA SET IS A PRIMARY
		..1.		DSDEMSTR	THIS DATA SET IS THE MASTER RACF RACF DATA SET. ITS ICB CONTAINS STATUS OPTIONS.
		...1		DSDERFSH	REFRESH ICB
		.... 1...		DSDESHR	DATA SET IS (OR WAS) SHARED
		.... .1..		DSDEALTI	ALTERI REQUESTS ARE BACKED-UP
		.... ..1.		DSDEDAT	IN-STORAGE BLOCKS CAN BE DATA BLOCKS
		1		DSDECMS	DATA SET IS VM CMS FILE
30	(1E)	SIGNED	2	DSDENREC	# RECORDS/TRACK ON DATA SET
32	(20)	UNSIGNED	1	DSDERXNO	# IN-STORAGE BUFFERS OR RESIDENT INDEX BLOCKS
33	(21)	CHARACTER	44	DSDEDSN	NAME OF DATA SET
77	(4D)	UNSIGNED	1	DSDEDSNO	DATA SET SEQUENCE NUMBER
78	(4E)	SIGNED	2	DSDECBLN	LENGTH OF DSDE
RACF Sysplex Data Sharing fields:					
80	(50)	CHARACTER	96	DSDESDS	
80	(50)	CHARACTER	16	DSDESNAM	XES Structure Name for this DB
96	(60)	CHARACTER	16	DSDECTOK	XES Connect token
112	(70)	CHARACTER	16	DSDECNAM	XES Connection Name
112	(70)	CHARACTER	4	DSDECNP	XES Connection Name Prefix
116	(74)	CHARACTER	3	DSDECNDS	XES Connection Name Dataset
119	(77)	CHARACTER	1	DSDECNSL	XES Connection Name Slash

Table 44. Structure DSDT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
120	(78)	CHARACTER	8	DSDECNSN	XES Connection Name suffix (System Name)
128	(80)	CHARACTER	12	DSDEVTOK	XES Vector token
140	(8C)	SIGNED	4	DSDEVLEN	XES Vector token length
144	(90)	UNSIGNED	1	DSDECONS	XES Connect Status, CONSTAT (See CONSTAT Constants below for valid values)
145	(91)	BITSTRING	1	DSDERBLD	XES Rebuild Status Flags
		1...		DSDERBIP	Rebuild In Progress
		.111 1111		*	Reserved
146	(92)	BITSTRING	1	DSDECACF	Caching Flags.
		1...		DSDEDSCM	IXLCACHE message issued
		.1..		DSDEDSVM	IXLVECTR message issued
		..11 1111		*	Reserved
147	(93)	CHARACTER	1	*	Boundary Alignment
148	(94)	CHARACTER	8	*	Reserved 2 words
156	(9C)	CHARACTER	8	DSDEDDNM	Dynamic allocation DDNAME
164	(A4)	CHARACTER	12	*	Reserved 3 words

DSDT constants

Table 45. Constants for DSDT

Len	Type	Value	Name	Description
1	DECIMAL	2	DSDTCVSN	CURRENT VERSION NUMBER
4	CHARACTER	DSDT	DSDTIDC	EBCIDIC ID
Constants for RACF Sysplex Data Sharing:				
48	CHARACTER	See	DSDTPRLS	IRRXCF00.LATCHSET.PRIMARY
48	CHARACTER	Desc.	DSDTBKLS	IRRXCF00.LATCHSET.BACKUP
Constants for DSDTDSMO:				
1	DECIMAL	0	DSDTDSNO	DSDTDSMO=DSDTDSNO - NO DATA SHARING ACROSS IRRXCF00
1	DECIMAL	1	DSDTDSHR	DSDTDSMO=DSDTDSHR - IN DATA SHARING MODE, ALL OTHER MEMBERS OF IRRXCF00 IN DATASHARING OR READONLY

Table 45. Constants for DSDT (continued)

Len	Type	Value	Name	Description
1	DECIMAL	2	DSDTDSRO	DSDTDSMO=DSDTDSRO - IN READONLY MODE, ALL OTHER MEMBERS OF IRRXCF00 IN DATASHARING OR READONLY
1	DECIMAL	3	DSDTTRAN	DSDTDSMO=DSDTTRAN - IN TRANSITION MODE, WILL EITHER BE IN READONLY OR DATASHARING AFTER CONNECT SERVICES CALLED.

Constants for CONSTAT fields:

Len	Type	Value	Name	Description
1	DECIMAL	0	DSDTGCON	Good Connection, ICB primed
1	DECIMAL	1	DSDTUCON	Unconnected
1	DECIMAL	2	DSDTNCON	New Connection, ICB not primed from DASD yet.
1	DECIMAL	3	DSDTCCON	Conditional Connection, minimum SF storage not obtained.
1	DECIMAL	4	DSDTBCON	Bad Connection, IXLDISC failed.
1	DECIMAL	5	DSDTOCON	Old (good) connection sustained through ReadOnly mode, ICB not re-primed from CF yet.

QNAME and RNAME for obtaining ENQs on the DSDT:
When an ENQ on the DSDT is needed:
- If RACF has been installed for Data Sharing (DSCADSRQ is set),
 a SCOPE=SYSTEMS ENQ is required to serialize on the DSDT.
- Otherwise, a SCOPE=SYSTEM ENQ is required.

Len	Type	Value	Name	Description
8	CHARACTER	SYSZRACF	DSDTQNAM	
48	CHARACTER	See note.	DSDTRNAM	

Note: The value for DSDTRNAM is DSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDT.

DSDT cross reference

Table 46. Cross Reference for DSDT

Name	Offset	Hex Value
DSDBACTV	15D	80
DSDBALTI	15D	04
DSDBBAM	158	
DSDBCACF	1D2	
DSDBCBLN	18E	
DSDBCNAM	1B0	
DSDBCNDS	1B4	

Table 46. Cross Reference for DSDT (continued)

Name	Offset	Hex Value
DSDBCNP	1B0	
DSDBCNSL	1B7	
DSDBCNSN	1B8	
DSDBCONS	1D0	
DSDBCTOK	1A0	
DSDBDAT	15D	02
DSDBDCB	140	
DSDBDDNM	1DC	
DSDBDEB	144	
DSDBDSCM	1D2	80
DSDBDSN	161	
DSDBDSNL	15C	
DSDBDSNO	18D	
DSDBDSVM	1D2	40
DSDBHDR	14C	
DSDBINDX	148	
DSDBMSTR	15D	20
DSDBNREC	15E	
DSDBPRIM	15D	40
DSDBRBIP	1D1	80
DSDBRBLD	1D1	
DSDBRFSH	15D	10
DSDBRUCB	150	
DSDBRXNO	160	
DSDBSDS	190	
DSDBSHR	15D	08
DSDBSNAM	190	
DSDBSTAT	15D	
DSDBVLEN	1CC	
DSDBVTOK	1C0	
DSDBXLEN	154	
DSDE	0	
DSDEACTV	1D	80
DSDEALTI	1D	04
DSDEBAM	18	
DSDECACF	92	

Table 46. Cross Reference for DSDT (continued)

Name	Offset	Hex Value
DSDECBLN	4E	
DSDECMS	1D	01
DSDECNAM	70	
DSDECNDS	74	
DSDECNP	70	
DSDECNSL	77	
DSDECNSN	78	
DSDECONS	90	
DSDECTOK	60	
DSDEDAT	1D	02
DSDEDBObj	AA	
DSDEDCB	0	
DSDEDDNM	9C	
DSDEDEB	4	
DSDEDSCM	92	80
DSDEDSN	21	
DSDEDSNL	1C	
DSDEDSNO	4D	
DSDEDSVM	92	40
DSDEFlags	AE	
DSDEHDR	C	
DSDEINDX	8	
DSDEMSTR	1D	20
DSDENREC	1E	
DSDEPRIM	1D	40
DSDERBIP	91	80
DSDERBLD	91	
DSDERFSH	1D	10
DSDERUCB	10	
DSDERXNO	20	
DSDESDS	50	
DSDESHR	1D	08
DSDESNAM	50	
DSDESTAT	1D	
DSDEVLEN	8C	
DSDEVOL	A4	

Table 46. Cross Reference for DSDT (continued)

Name	Offset	Hex Value
DSDEVSAM	AE	
DSDEVTOK	80	
DSDEXLEN	14	
DSDPACTV	AD	80
DSDPALTI	AD	04
DSDPBAM	A8	
DSDPCACF	122	
DSDPCBLN	DE	
DSDPCNAM	100	
DSDPCNDS	104	
DSDPCNP	100	
DSDPCNSL	107	
DSDPCNSN	108	
DSDPCONS	120	
DSDPCTOK	F0	
DSDPDAT	AD	02
DSDPDCB	90	
DSDPDDNM	12C	
DSDPDEB	94	
DSDPDSCM	122	80
DSDPDSN	B1	
DSDPDSNL	AC	
DSDPDSNO	DD	
DSDPDSVM	122	40
DSDPHDR	9C	
DSDPINDX	98	
DSDPMSTR	AD	20
DSDPNREC	AE	
DSDPPRIM	AD	40
DSDPRBIP	121	80
DSDPRBLD	121	
DSDPRFSH	AD	10
DSDPRUCB	A0	
DSDPRXNO	B0	
DSDPSDS	E0	
DSDPSHR	AD	08

Table 46. Cross Reference for DSDT (continued)

Name	Offset	Hex Value
DSDPSNAM	E0	
DSDPSTAT	AD	
DSDPSWIT	123	20
DSDPVTOK	110	
DSDPXLEN	A4	
DSDT	0	
DSDTBACK	140	
DSDTBFMT	D	40
DSDTBLNM	50	
DSDTBLTK	80	
DSDTBXST	D	10
DSDTDSDX	8	
DSDTDSFL	88	
DSDTDSMO	89	
DSDTDSRQ	88	80
DSDTENTY	90	
DSDTFLAG	D	
DSDTID	0	
DSDTNUM	4	
DSDTPFMT	D	80
DSDTPLNM	18	
DSDTPLTK	48	
DSDTPRIM	90	
DSDTPXST	D	20
DSDTVRSN	C	

Chapter 20. ENFP: RACF-Parameter List for ENF event code 62 listen exits

ENFP programming interface information

ENFP is a programming interface.

ENFP heading information

Common name:	Mapping macro for RACF ENF event code 62
Macro ID:	IRRPENFP
DSECT name:	ENFP
Owning component:	SAF
Eye-catcher ID:	IRRENF
Storage attributes:	**Subpool** 239 **Key** 0 **Residency** Above
Size:	**24 bytes ('18' in hex)** FREQUENCY = 1 per ENF signal
Created by:	RACF
Pointed to by:	Register 1 on input to ENF listen routine
Serialization:	None
Function:	Maps the input parameter list for ENF event code 62 listen exits

ENFP mapping

Table 47. Structure ENFP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	24	ENFP	RACF ENF parameter list
0	(0)	CHARACTER	6	IRR_ENFID	Control block ID = IRRENF
6	(6)	BITSTRING	2	IRR_ENFVER	Parameter list version
8	(8)	SIGNED	2	IRR_ENFLEN	Parameter list length
10	(A)	CHARACTER	2	*	Reserved
12	(C)	CHARACTER	4	IRR_ENFQ(0)	Qualifier code
12	(C)	CHARACTER	1	IRR_ENFQ_LIST	SETR RACLIST signal
13	(D)	CHARACTER	1	IRR_ENFQ_REFR	SETR RACLIST REFRESH signal

Table 47. Structure ENFP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
14	(E)	CHARACTER	1	IRR_ENFQ_NOLIST	SETR NORACLIST signal
15	(F)	CHARACTER	1	*	Reserved
16	(10)	CHARACTER	8	IRR_ENFCLASS	Class name

ENFP cross reference

Table 48. Cross Reference for ENFP

Name	Offset	Hex Value
ENFP	0	
IRR_ENFCLASS	10	
IRR_ENFID	0	
IRR_ENFLEN	8	
IRR_ENFQ	C	
IRR_ENFQ_LIST	C	
IRR_ENFQ_REFR	D	
IRR_ENFQ_NOLIST	E	
IRR_ENFVER	6	

Chapter 21. ENF2: RACF-Parameter List for ENF event code 71 listen exits

ENF2 programming interface information

ENF2 is a programming interface.

ENF2 heading information

Common name: Mapping macro for RACF ENF event code 71

Macro ID: IRRPENF2

DSECT name: ENF2

Owning component: SAF

Eye-catcher ID: IRREN2

Storage attributes:

Subpool
231

Key
0

Residency
Above

Size:

48 bytes ('30' in hex)
FREQUENCY = 1 per ENF signal

Created by: RACF

Pointed to by: Register 1 on input to ENF listen routine

Serialization: None

Function: Maps the input parameter list for ENF event code 71 listen exits

ENF2 mapping

Table 49. Structure ENF2

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	ENF2	RACF ENF 71 parameter list
0	(0)	CHARACTER	6	IRR_ENF2ID	Control block ID = IRREN2
6	(6)	BITSTRING	2	IRR_ENF2VER	Parameter list version = X'02'
8	(8)	SIGNED	2	IRR_ENF2LEN	Parameter list length
10	(A)	BITSTRING	2	IRR_ENF2FLAGS	Control Flags
10		1...		IRR_ENF2_CONNECT_REVOKE	CONNECT...REVOKE

Table 49. Structure ENF2 (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		IRR_ENF2_VER_INVPWD	RACROUTE REQUEST=VERIFY revoke due to invalid passwords
		..11 1111		*	Reserved
12	(C)	CHARACTER	4	IRR_ENF2Q	Qualifier code
12	(C)	BITSTRING	1	IRR_ENF2Q_CON	CONNECT command when X'80'
				RR_ENF2Q_REM	REMOVE command when X'40'
				IRR_ENF2Q_ALU_REVOKE	ALTUSER REVOKE command when X'20'
				IRR_ENF2Q_DU	DELUSER command when X'10'
				IRR_ENF2Q_DGRP	DELGROUP command when X'08'
				IRR_ENF2Q_VERIFY	RACROUTE REQUEST=VERIFY when X'04'
13	(D)	CHARACTER	3	*	Reserved
16	(10)	CHARACTER	8	IRR_ENF2USER	RACF UserID
24	(18)	CHARACTER	8	IRR_ENF2GROUP	RACF GroupID
32	(20)	CHARACTER	16	*	Reserved

ENF2 cross reference

Table 50. Cross Reference for ENF2

Name	Offset	Hex Value
ENF2	0	
IRR_ENF2ID	0	
IRR_ENF2Flags	A	
IRR_ENF2LEN	8	
IRR_ENF2VER	6	
IRR_ENF2Q	C	
IRR_ENF2Q_ALU_REVOKE	C	
IRR_ENF2Q_CON	C	
IRR_ENF2Q_DU	C	
IRR_ENF2Q_DGRP	C	
IRR_ENF2Q_REM	C	
IRR_ENF2GROUP	18	
IRR_ENF2USER	10	

Chapter 22. ENF3: RACF-Parameter List for ENF event code 79 listen exits

ENF3 programming interface information

ENF3 is a programming interface.

ENF3 heading information

Common name: Mapping macro for RACF ENF event code 79

Macro ID: IRRPENF3

DSECT name: ENF3

Owning component: SAF

Eye-catcher ID: IRREN3

Storage attributes:

Subpool
231

Key
0

Residency
Above

Size:

Variable - header portion is 592 bytes ('250' in hex)
FREQUENCY = 1 per ENF signal

Created by: RACF

Pointed to by: Register 1 on input to ENF listen routine

Serialization: None

Function: Maps the input parameter list for ENF event code 79 listen exits

Note: The ENF3 is a variable length data area, with a fixed length header portion of 592 bytes.

ENF3 mapping

Table 51. Structure ENF3

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	ENF3	RACF ENF 79 parameter list
0	(0)	CHARACTER	6	IRR_ENF3_ID	Control block ID = IRREN3
6	(6)	BITSTRING	2	IRR_ENF3_VERSION	Parameter list version = X'01'
8	(8)	SIGNED	2	IRR_ENF3_LENGTH	Parameter list length
10	(A)	CHARACTER	2	*	Reserved
12	(C)	CHARACTER	4	IRR_ENF3_QUALCODE	Qualifier Code

Table 51. Structure ENF3 (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	BITSTRING	1	IRR_ENF3_QualCode_PE	PERMIT command when X'80'
				IRR_ENF3_QualCode_RDEF	RDEFINE command when X'40'
				IRR_ENF3_QualCode_RALT	RALTER command when X'20'
				IRR_ENF3_QualCode_RDEL	RDELETE command when X'10'
13	(D)	CHARACTER	3	*	Reserved
16	(10)	CHARACTER	8	IRR_ENF3_USERID	RACF UserID
24	(18)	CHARACTER	8	IRR_ENF3_CLASSNAME	RACF Class Name
32	(20)	BITSTRING	4	IRR_ENF3_FLAGS	Control Flags
		1...		IRR_ENF3_PERMIT_ACCESS	PERMIT ACCESS(...)
		.1..		IRR_ENF3_PERMIT_DELETE	PERMIT DELETE
		..11		IRR_ENF3_PERMIT_RESET	PERMIT RESET or PERMIT RESET(ALL), both bits are "ON" (X'30')
				IRR_ENF3_PERMIT_RESET_STD(..1. ....)	PERMIT RESET(STANDARD), first bit is "ON" (X'20')
				IRR_ENF3_PERMIT_RESET_WHEN(..1. ....)	PERMIT RESET(WHEN) second bit is "ON" (X'10')
		.... 1...		IRR_ENF3_UACC_Specified	RDEFINE ... UACC(..) or RALTER ... UACC(..)
		.... .111		*	Reserved
36	(24)	BITSTRING	1	IRR_ENF3_ACCESS_LEVEL	Access Level from: • PERMIT ACCESS (Access Level) • RDEFINE UACC (Access Level) • RALTER UACC (Access Level)
		1...		IRR_ENF3_ACCESS_LEVEL_ALTER	Access level = ALTER
		.1..		IRR_ENF3_ACCESS_LEVEL_CONTROL	Access level = CONTROL
		..1.		IRR_ENF3_ACCESS_LEVEL_UPDATE	Access level = UPDATE
		...1		IRR_ENF3_ACCESS_LEVEL_READ	Access level = READ
		.... 1...		IRR_ENF3_ACCESS_LEVEL_EXECUTE	Access level = EXECUTE
		.... .11.		*	Reserved
		1		IRR_ENF3_ACCESS_LEVEL_NONE	"X'01'" ACCESS(NONE) / UACC(NONE)
37	(25)	SIGNED	1	IRR_ENF3_PERMIT_WHEN_COND	Numerical value of the PERMIT WHEN(Condition) ***

Table 51. Structure ENF3 (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
38	(26)	CHARACTER	2	*	Reserved
40	(28)	SIGNED	2	IRR_ENF3_ADDMEML_MEMBER#	Number of Members in ADDMEM List
42	(2A)	SIGNED	2	IRR_ENF3_DELMEML_MEMBER#	Number of Members in DELMEM List
44	(2C)	SIGNED	2	IRR_ENF3_ADDMEML_LENGTH	Length of ADDMEM Member List
46	(2E)	SIGNED	2	IRR_ENF3_DELMEML_LENGTH	Length of DELMEM Member List
48	(30)	SIGNED	2	IRR_ENF3_ADDMEML_OFFSET	Offset to ADDMEM List Data
50	(32)	SIGNED	2	IRR_ENF3_DELMEML_OFFSET	Offset to DELMEM List Data
52	(34)	CHARACTER	28	*	Reserved
80	(50)	STRUCTURE	256	IRR_ENF3_PROFNAME_DS(0)	RACF Profile Name Data Structure
80	(50)	SIGNED	1	IRR_ENF3_PROFNAME_LENGTH	Length of RACF Profile Name
81	(51)	CHARACTER	255	IRR_ENF3_PROFNAME	RACF Profile Name
336	(150)	STRUCTURE	256	IRR_ENF3_CACLNAME_DS(0)	Conditional Access Name Data Structure
336	(150)	SIGNED	1	IRR_ENF3_CACLNAME_LENGTH	Length of Conditional Access Name
337	(151)	CHARACTER	255	IRR_ENF3_CACLNAME	Conditional Access Name

Note: ***

Numerical values of PERMIT WHEN: IRR_ENF3_PERMIT_WHEN_PROGRAM_Cond = 1
IRR_ENF3_PERMIT_WHEN_CONSOLE_Cond = 2
IRR_ENF3_PERMIT_WHEN_TERMINAL_Cond = 3
IRR_ENF3_PERMIT_WHEN_JESINPUT_Cond = 4
IRR_ENF3_PERMIT_WHEN_APPCPORT_Cond = 5
IRR_ENF3_PERMIT_WHEN_SYSID_Cond = N/A
IRR_ENF3_PERMIT_WHEN_SERVAUTH_Cond = 7
IRR_ENF3_PERMIT_WHEN_CRITERIA_Cond = 8

ENF3 cross reference

Table 52. Cross Reference for ENF3

Name	Offset	Hex Value
ENF3	0	
IRR_ENF3_ACCESS_LEVEL	24	
IRR_ENF3_ADDMEML_LENGTH	2C	

Table 52. Cross Reference for ENF3 (continued)

Name	Offset	Hex Value
IRR_ENF3_ADDMEML_MEMBER#	28	
IRR_ENF3_ADDMEML_OFFSET	30	
IRR_ENF3_CLASSNAME	18	
IRR_ENF3_DELMEM_LIST	2E	
IRR_ENF3_DELMEML_MEMBER#	2A	
IRR_ENF3_DELMEML_OFFSET	32	
IRR_ENF3_FLAGS	20	
IRR_ENF3_ID	0	
IRR_ENF3_LENGTH	8	
IRR_ENF3_PERMIT_WHEN_COND	25	
IRR_ENF3_PROFNAME	51	
IRR_ENF3_PROFNAME_DS	50	
IRR_ENF3_PROFNAME_LENGTH	50	
IRR_ENF3_QUALCODE	C	
IRR_ENF3_USERID	10	
IRR_ENF3_VERSION	6	

Chapter 23. EVXP: RACF Command Exit Parameter List

EVXP programming interface information

EVXP is a programming interface.

EVXP heading information

Common name: RACF Command Exit Parameter List Mapping

Macro ID: IRREVXP

DSECT name: EVXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
3

Key
Caller's

Size: 48 bytes

Created by: RACF Command Processor Envelope

Pointed to by: R1 at entry to IRREVX01

Serialization: None

Function: Contains the parameter list passed to the IRREVX01 exit point for commands

EVXP mapping

Table 53. Structure EVXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	EVXPL	
0	(0)	ADDRESS	4	EVXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	EVXCALLR	Caller address: points to a 1-byte field containing a function code identifying the command issued:
		1		EVXADDGR	"X'01'" ADDGROUP
		.... ..1.		EVXADDSD	"X'02'" ADDSD
		11		EVXADDUS	"X'03'" ADDUSER
		.... .1..		EVXALTDS	"X'04'" ALTDSD

Table 53. Structure EVXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... .1.1		EVXALTGR	"X'05'" ALTGROUP
		.... .11.		EVXALTUS	"X'06'" ALTUSER
		.... .111		EVXCONNE	"X'07'" CONNECT
		.... 1...		EVXDELDS	"X'08'" DELDSD
		.... 1..1		EVXDELGR	"X'09'" DELGROUP
		.... 1.1.		EVXDELUS	"X'0A'" DELUSER
		.... 1.11		EVXLISTD	"X'0B'" LISTDSD
		.... 11..		EVXLISTG	"X'0C'" LISTGRP
		.... 11.1		EVXLISTU	"X'0D'" LISTUSER
		.... 111.		EVXPASSW	"X'0E'" PASSWORD
		 1111		EVXPERMI	"X'0F'" PERMIT
		...1		EVXRALTE	"X'10'" RALTER
		...1 ...1		EVXRDEFI	"X'11'" RDEFINE
		...1 ..1.		EVXRDELE	"X'12'" RDELETE
		...1 ..11		EVXREMOV	"X'13'" REMOVE
		...1 .1..		EVXRLIST	"X'14'" RLIST
		...1 .1.1		EVXSEARC	"X'15'" SEARCH
		...1 .11.		EVXSETRO	"X'16'" SETROPTS
8	(8)	ADDRESS	4	EVXFLAGS	Flag byte address: points to 2 bytes of flags that cannot be changed by the exit (2nd byte all reserved flagspace) Constants for 1st byte...
		1...		EVXPRE	"X'80'" Pre-processing call
		.1..		EVXPOST	"X'40'" Post-processing call
		..1.		EVXOPER	"X'20'" Command issued as operator command
		...1		EVXPARM	"X'10'" Command issued from RACF parmlib second byte flag constants
		.... 1...		EVXAT	"X'08'" Command directed with AT or ONLYAT
		.... .1..		EVXACD	"X'04'" Command directed with automatic direction
		.... ..1.		EVXRASP	"X'02'" Command execution in RACF subsystem
		1		EVXABND	"X'01'" Command abended during execution (only set for post-processing call)

Table 53. Structure EVXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	EVXCMBUF	Command buffer address: points to a 2 byte length of command buffer, 2 byte offset to the first keyword, followed by the command image. Mapped by CMDBUF below
16	(10)	ADDRESS	4	EVXACEE	ACEE address: points to the ACEE of the execution user ID. EVXACEE is 0 if the command was issued from the RACF parameter library.
20	(14)	ADDRESS	4	EVXWORK	Communication word address: points to a word that can be used by the exit to communicate between the pre and post-processing calls to the exit.
24	(18)	ADDRESS	4	EVXCMDRC	Command return code address: points to a word containing the return code from command execution. Always 0 for the pre-processing call.
28	(1C)	ADDRESS	4	EVXABCD	Abend code address: points to a word containing the abend code when the flags indicate that the command abended.
32	(20)	ADDRESS	4	EVXSRCND	Command source node address: points to an 8 byte node name field. If this is the execution of a directed command, this is the originating node. The node name is left justified and padded with blanks. The field is all blanks if this is not the execution of a directed command.
36	(24)	ADDRESS	4	EVXSRCUS	Command source user ID address: points to an 8 byte user ID field. If this is the execution of a directed command, this is the originating user ID. The user ID is left justified and padded with blanks. The field is all blanks if this is not the execution of a directed command.
40	(28)	ADDRESS	4	EVXMSSG	Message text address: points to a 200 byte area initalized to blanks. Can be used to supply message insert for IRRV022I when the pre-processing call sets register 15 to a value other than 0 or 4.

Table 53. Structure EVXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
44	(2C)	ADDRESS	4	EVXOPARM	Output parameters address: points to a flag word initialized to zeroes that can be used by the exit to request various functions.
		1.......		EVXSPEC	"X'80'" Pre-exit requests to run the command with system SPECIAL authority. This is ignored if the command is running with a managed ACEE. A managed ACEE is one created using the initACEE callable service (IRRSIA00) with the INTA_MANAGED option.
		.1......		EVXAUDT	"X'40' Pre-exit" requests to run the command with system AUDITOR authority. This is ignored if the command is running with a managed ACEE. A managed ACEE is one created using the initACEE callable service (IRRSIA00) with the INTA_MANAGED option.

CMDBUF

0	(0)	STRUCTURE		CMDBUF	
0	(0)	ADDRESS	2	CMDBUFL	Length of command buffer
2	(2)	ADDRESS	2	CMDBUFO	Offset in command buffer to the first blank past the command name
		.... ..1.		CMDBUFD	"*" variable length character data

EVXP cross reference

Table 54. Cross Reference for EVXP

Name	Offset	Hex Value
CMDBUFD	0	0
CMDBUFL	0	0
CMDBUFO	0	
EVXABCD	1C	
EVXABND	8	01
EVXACD	8	04
EVXACEE	10	
EVXADDGR	4	01

Table 54. Cross Reference for EVXP (continued)

Name	Offset	Hex Value
EVXADDSD	4	02
EVXADDUS	4	03
EVXALTDS	4	04
EVXALTGR	4	05
EVXALTUS	4	06
EVXAT	8	08
EVXAUDT	2C	40
EVXCALLR	4	
EVXCMBUF	C	
EVXCMDRC	18	
EVXCONNE	4	07
EVXDELDS	4	08
EVXDELGR	4	09
EVXDELUS	4	0A
EVXFLAGS	8	
EVXLEN	0	
EVXLISTD	4	0B
EVXLISTG	4	0C
EVXLISTU	4	0D
EVXMSSG	28	
EVXOPARM	2C	
EVXOPER	8	20
EVXPARM	8	10
EVXPASSW	4	0E
EVXPERMI	4	0F
EVXPL	0	
EVXPOST	8	40
EVXPRE	8	80
EVXRALTE	4	10
EVXRASP	8	02
EVXRDEFI	4	11
EVXRDELE	4	12
EVXREMOV	4	13
EVXRLIST	4	14
EVXSEARC	4	15
EVXSETRO	4	16

Table 54. Cross Reference for EVXP (continued)

Name	Offset	Hex Value
EVXSPEC	2C	80
EVXSRCND	20	
EVXSRCUS	24	
EVXWORK	14	

Chapter 24. FACL: z/OS UNIX System Services access control list

FACL programming interface information

FACL is a programming interface.

FACL heading information

Common name:	z/OS UNIX System Services access control list
Macro ID:	IRRPFACL
DSECT name:	FACL
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	FACL
Storage attributes:	N/A
Size:	Variable
Created by:	Invoker of z/OS UNIX security functions
Pointed to by:	CREDACLINFO
Serialization:	None
Function:	Contains access control list information for a z/OS UNIX file or directory

FACL mapping

Table 55. Structure FACL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	64	FACL	
0	(0)	CHARACTER	64	FACL_MIN_HEADER	Minimum header required by the LFS/PFS for an ACL read request
0	(0)	CHARACTER	4	FACL_ID	Literal ID 'FACL'
4	(4)	UNSIGNED	4	FACL_LEN	Total length of structure
8	(8)	UNSIGNED	2	FACL_NUM_ENTRY	Number of ACL entries
10	(A)	UNSIGNED	2	FACL_LEN_ENTRY	Length of an ACL entry
12	(C)	CHARACTER	16		Reserved
28	(1C)	UNSIGNED	1	FACL_VERS	Version Number for ACL
29	(1D)	BITSTRING	1	FACL_FLAG	ACL flags
		1...		FACL_ACCESS	Access ACL
		.1..		FACL_FILE_MODEL	File model ACL

Table 55. Structure FACL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		..1.		FACL_DIR_MODEL	Directory Model ACL
		...1 1111		*	Reserved
30	(1E)	UNSIGNED	2	FACL_NUM_USERS	Number of user entries
32	(20)	UNSIGNED	2	FACL_ERROFF	Offset(from start of structure) to ACL entry in error
34	(22)	CHARACTER	2	*	Reserved
36	(24)	UNSIGNED	4	FACL_RACF_RETURN_CODE	RACF Return Code
40	(28)	UNSIGNED	4	FACL_RACF_REASON_CODE	RACF Reason Code
44	(2C)	CHARACTER	20	*	Reserved
64	(40)	CHARACTER	0	FACL_HEADER_EXTENSION	Reserved
64	(40)	CHARACTER	0	FACL_ENTRIES	Start of entries
FACL_EDIT					
0	(0)	STRUCTURE	64	FACL_EDIT	
0	(0)	CHARACTER	64	FACL_EDIT_MIN_HEADER	Minimum header required by the LFS/PFS for an ACL read request
0	(0)	CHARACTER	4	FACL_EDIT_HDR_ID	Literal Id 'FACL'
4	(4)	UNSIGNED	4	FACL_EDIT_LEN	Total length of structure (header plus all entries)
8	(8)	UNSIGNED	2	FACL_EDIT_NUM_ENTRY	Number of ACL entries
10	(A)	UNSIGNED	2	FACL_EDIT_LEN_ENTRY	Length of an ACL entry
12	(C)	CHARACTER	16	*	Reserved
28	(1C)	UNSIGNED	1	FACL_EDIT_VERS	Version Number for ACL
29	(1D)	BITSTRING	1	FACL_EDIT_FLAG	ACL flags
		1...		FACL_EDIT_ACCESS	Access ACL
		.1..		FACL_EDIT_FILE_MODEL	File model ACL
		..1.		FACL_EDIT_DIR_MODEL	Directory Model ACL
		...1 1111		*	Reserved
30	(1E)	UNSIGNED	2	FACL_EDIT_NUM_USERS	Number of user entries
32	(20)	UNSIGNED	2	FACL_EDIT_ERROFF	Offset (from start of structure) to ACL entry in error
34	(22)	CHARACTER	2	*	Reserved
36	(24)	UNSIGNED	4	FACL_EDIT_RACF_RETURN_CODE	RACF Return Code
40	(28)	UNSIGNED	4	FACL_EDIT_RACF_REASON_CODE	RACF Reason Code
44	(2C)	CHARACTER	20	*	Reserved
64	(40)	CHARACTER	0	FACL_EDIT_HEADER_EXTENSION(0)	Reserved

Table 55. Structure FACL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
64	(40)	CHARACTER0	1	FACL_EDIT_ENTRIES(0)	Start of entries
FACL_ENTRY					
0	(0)	STRUCTURE	8	FACL_ENTRY	
0	(0)	CHARACTER	2	*	Reserved
2	(2)	BITSTRING	1	FACL_ENTRY_PERMS	Permissions granted byte
		1111 1...		*	Reserved
		.... .111		FACL_PERM_BITS	Permission bits
		.... .1..		FACL_READ	Read access
		.... ..1.		FACL_WRITE	Write access
		1		FACL_EXECUTE	Execute access
3	(3)	CHARACTER	5	FACL_SORT_KEY(0)	Key for Sorting ACL
3	(3)	UNSIGNED	1	FACL_ENTRY_TYPE	Type of ACL entry (user or group)
4	(4)	SIGNED	4	FACL_ENTRY_ID	UID/GID value
8	(8)	CHARACTER	0	FACL_ENTRY_END	
FACL_EDIT_ENTRY					
0	(0)	STRUCTURE	8	FACL_EDIT_ENTRY	
0	(0)	BITSTRING	1	FACL_EDIT_OPTIONS	ACL entry options
		1...		FACL_DEL_ENTRY	For modify: delete entry
		.111 1111		*	Reserved
1	(1)	BITSTRING	1	FACL_PERMS_MASK	Relevant Permissions
		1111 1...		*	Reserved
		.... .111		FACL_PERM_BITS_MASK	Permission bits mask
		.... .1..		FACL_READ_MASK	Read access mask
		.... ..1.		FACL_WRITE_MASK	Write access mask
		1		FACL_EXECUTE_MASK	Execute access mask
2	(2)	BITSTRING	1	FACL_EDIT_PERMS	Relative Permissions
		1111 1...		*	Reserved
		.... .111		FACL_EDIT_PERM_BITS	Permission bits
		.... .1..		FACL_EDIT_READ	Read access
		.... ..1.		FACL_EDIT_WRITE	Write access
		1		FACL_EDIT_EXECUTE	Execute access
3	(3)	BITSTRING	1	FACL_EDIT_TYPE	Type of ACL entry (user or group)
4	(4)	ADDRESS	4	FACL_EDIT_ID	UID/GID value
8	(8)	CHARACTER	1	FACL_EDIT_END	

FACL constants

Table 56. Constants for FACL

Len	Type	Value	Name	Description
4	DECIMAL	64	FACL_LENGTH	
FACL Constants				
4	CHARACTER	FACL	FACL_IDC	ACL ID Constant
1	DECIMAL	1	FACL_CV01	Version 1 of ACL
1	DECIMAL	1	FACL_CVER	Current version of ACL
4	DECIMAL	1024	FACL_MAX_ENTRIES	
4	DECIMAL	8	FACL_ENTRY_LENGTH	
FACL_Entry_Type Values				
1	NUMB HEX	01	FACL_ENTRY_UID	User entry
1	NUMB HEX	02	FACL_ENTRY_GID	Group entry
FACL Constants				
4	DECIMAL	8256	FACL_MAX_ACL_SIZE	

FACL cross reference

Table 57. Cross Reference for FACL

Name	Offset	Hex Value
FACL	0	
FACL_ACCESS	1D	80
FACL_DEL_ENTRY	0	80
FACL_DIR_MODEL	1D	20
FACL_EDIT	0	
FACL_EDIT_ACCESS	1D	80
FACL_EDIT_DIR_MODEL	1D	20
FACL_EDIT_END	8	
FACL_EDIT_ENTRIES	40	
FACL_EDIT_ENTRY	0	
FACL_EDIT_ERROFF	20	
FACL_EDIT_EXECUTE	2	01
FACL_EDIT_FILE_MODEL	1D	40
FACL_EDIT_FLAG	1D	
FACL_EDIT_HDR_ID	0	
FACL_EDIT_HEADER_EXTENSION	40	
FACL_EDIT_ID	4	
FACL_EDIT_LEN	4	
FACL_EDIT_LEN_ENTRY	A	

Table 57. Cross Reference for FACL (continued)

Name	Offset	Hex Value
FACL_EDIT_MIN_HEADER	0	
FACL_EDIT_NUM_ENTRY	8	
FACL_EDIT_NUM_USERS	1E	
FACL_EDIT_OPTIONS	0	
FACL_EDIT_PERM_BITS	2	07
FACL_EDIT_PERMS	2	
FACL_EDIT_RACF_REASON_CODE	28	
FACL_EDIT_RACF_RETURN_CODE	24	
FACL_EDIT_READ	2	04
FACL_EDIT_TYPE	3	
FACL_EDIT_VERS	1C	
FACL_EDIT_WRITE	2	02
FACL_ENTRIES	40	
FACL_ENTRY	0	
FACL_ENTRY_END	8	
FACL_ENTRY_ID	4	
FACL_ENTRY_PERMS	2	
FACL_ENTRY_TYPE	3	
FACL_ERROFF	20	
FACL_EXECUTE	2	01
FACL_EXECUTE_MASK	1	01
FACL_FILE_MODEL	1D	40
FACL_FLAG	1D	
FACL_HEADER_EXTENSION	40	
FACL_ID	0	
FACL_LEN	4	
FACL_LEN_ENTRY	A	
FACL_MIN_HEADER	0	
FACL_NUM_ENTRY	8	
FACL_NUM_USERS	1E	
FACL_PERM_BITS	2	07
FACL_PERM_BITS_MASK	1	07
FACL_PERMS_MASK	1	
FACL_RACF_REASON_CODE	28	
FACL_RACF_RETURN_CODE	24	
FACL_READ	2	04

Table 57. Cross Reference for FACL (continued)

Name	Offset	Hex Value
FACL_READ_MASK	1	04
FACL_SORT_KEY	3	
FACL_VERS	1C	
FACL_WRITE	2	02
FACL_WRITE_MASK	1	02

Chapter 25. FAST: RACROUTE REQUEST=FASTAUTH Parameter List (Request Section)

FAST programming interface information

FAST is a programming interface.

FAST heading information

Common name: Request-specific portion of the RACROUTE REQUEST=FASTAUTH parameter list

Macro ID: None

DSECT name: None

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Variable

Created by: RACROUTE REQUEST=FASTAUTH macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=FASTAUTH routine

FAST mapping

Table 58. Structure FASTPARD

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	28	FASTPARM	FASTAUTH parameters
0	(0)	BITSTRING	1	FASTATTR	ATTR= Flags
		1...		FASTALTR	1 = ALTER requested
		.111		*	Reserved
		.... 1...		FASTCNTL	1 = CONTROL requested
		.... .1..		FASTUPDT	1 = UPDATE requested
		.... ..1.		FASTREAD	1 = READ requested
1	(1)	BITSTRING	1	FASTLOGO	LOG= Flags

Table 58. Structure FASTPARD (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		FASTASIS	LOG=ASIS
		.1..		FASTNFAL	LOG=NOFAIL
		..1.		FASTENTX	ENTITYX specified. If on, FASTENTP points to a name in ENTITYX format.
		...1		FASTACHK	AUTHCHKS=CRITONLY
		.... 1...		FASTLOGX	LOGSTRX is specified. If on, FASTLOGS points to the LOGSTR data in the format that is described by LOGSTRX keyword.
		.... .111		*	Reserved
2	(2)	FIXED	1	FASTPLEN	Parameter List Length: 28 - OS/390® Security Server R3 or earlier 36 - OS/390 Security Server R4 40 - OS/390 Security Server R6 44 - z/OS V1R8
3	(3)	HEX	1	FASTPVER	Parameter List Length: 0 - OS/390 Security Server R3 or earlier 1 - OS/390 Security Server R4 2 - OS/390 Security Server R6 3 - z/OS V1R8
4	(4)	ADDRESS	4	FASTENTP	For ENTITY (FASTENTX=off), points to a field as the maximum length name of the given class, as determined by the class descriptor table. Names in the field are left-aligned and padded with blanks if necessary. For ENTITYX (FASTENTX=on), points to a halfword buffer length followed by a halfword actual length of the resource name not including trailing blanks. If the actual length is zero, then RACF determines the number of characters in the entity name.
8	(8)	ADDRESS	4	FASTCLAS	Address of class name
12	(C)	ADDRESS	4	FASTACEE	Address of ACEE to use
16	(10)	ADDRESS	4	FASTAPPL	Address of application name
20	(14)	ADDRESS	4	FASTWKA	Address of 16 word workarea
24	(18)	ADDRESS	4	FASTINST	Address of installation exit data field

Table 58. Structure FASTPARD (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	FASTALET	ACEEALET address: If nonzero, it points to a fullword ALET value to be used when referencing an ACEE in another address space. This address is always zero for ICHRFX01 and ICHRFX02. FASTALET exists only when FASTPVER is 1 or higher.
32	(20)	ADDRESS	4	FASTLOGS	LOGSTR address: • If FASTLOGX is off, this field points to a 1-byte length field, followed by character data that can be from 0 to 255 bytes long. • If FASTLOGX is on, this field points to a 2-byte length field, followed by a 2 byte ID field, followed by data that can be up to 1096 bytes long. The specified length includes the length and ID field. The format of the data depends on the ID field.
36	(24)	ADDRESS	4	FASTENVR	Address of the ENVR data structure. FASTENVR exists only when FASTPVER is 2 or higher.
40	(28)	ADDRESS	4	FASTCRIT	Address of the CRITERIA data structure. FASTCRIT exists only when FASTPVER is 3 or higher.

FAST cross reference

Table 59. Cross Reference for FAST

Name	Offset	Hex Value
FASTACEE	C	
FASTACHK	1	10
FASTALET	1C	
FASTALTR	0	80
FASTAPPL	10	
FASTASIS	1	80
FASTATTR	0	
FASTCLAS	8	
FASTCNTL	0	08
FASTCRIT	28	
FASTENTP	4	

Table 59. Cross Reference for FAST (continued)

Name	Offset	Hex Value
FASTENTX	1	20
FASTENVR	24	
FASTINST	18	
FASTLOGO	1	
FASTLOGS	20	
FASTLOGX	1	8
FASTNFAL	1	40
FASTPARM	0	
FASTPLEN	2	
FASTPVER	3	
FASTREAD	0	02
FASTUPDT	0	04
FASTWKA	14	

Chapter 26. FC: z/OS UNIX System Services Security Function Code Table

FC programming interface information

FC is a programming interface.

FC heading information

Common name: z/OS UNIX System Services security function code table

Macro ID: IRRPFC

DSECT name: N/A

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: N/A

Storage attributes: N/A

Size: Determined by release (see the mapping)

Created by: N/A

Pointed to by: N/A

Serialization: N/A

Function: Maps the z/OS UNIX security function codes

FC constants

Table 60. Constants for FC

Len	Type	Value	Name	Description
1	DECIMAL	1	IRRSIU00#	Function code 1 - initUS
1	DECIMAL	2	IRRSDU00#	Function code 2 - deleteUSP
1	DECIMAL	3	IRRSMF00#	Function code 3 - makeFSP
1	DECIMAL	4	*	Reserved
1	DECIMAL	5	IRRSMM00#	Function code 5 - set file mode creation mask
1	DECIMAL	6	IRRSKA00#	Function code 6 - check access
1	DECIMAL	7	IRRSKP00#	Function code 7 - check privilege
1	DECIMAL	8	IRRSUM00#	Function code 8 - getUMAP
1	DECIMAL	9	IRRSGM00#	Function code 9 - getGMAP
1	DECIMAL	10	IRRSGG00#	Function code 10 - get supplemental groups

Table 60. Constants for FC (continued)

Len	Type	Value	Name	Description
1	DECIMAL	11	IRRSSU00#	Function code 11 - set UID
1	DECIMAL	12	IRRSEU00#	Function code 12 - set effective UID
1	DECIMAL	13	IRRSSG00#	Function code 13 - set GID
1	DECIMAL	14	IRRSEG00#	Function code 14 - set effective GID
1	DECIMAL	15	IRRSCO00#	Function code 15 - change owner group
1	DECIMAL	16	IRRSCF00#	Function code 16 - change file mode
1	DECIMAL	17	IRRSCA00#	Function code 17 - change file audit options
1	DECIMAL	18	IRRSEX00#	Function code 18 - exec_setUID/GID
1	DECIMAL	19	IRRSAU00#	Function code 19 - audit
1	DECIMAL	20	IRRSKO00#	Function code 20 - check process owner
1	DECIMAL	21	IRRSQS00#	Function code 21 - query security options
1	DECIMAL	22	IRRSQF00#	Function code 22 - query file options
1	DECIMAL	23	IRRSCS00#	Function code 23 - clear_setid
1	DECIMAL	24	IRRSKF00#	Function code 24 - check file owner
1	DECIMAL	25	IRRSMR00#	Function code 25 - make_root_FSP
1	DECIMAL	26	IRRSPT00#	Function code 26 - PTRACE authority check
1	DECIMAL	27	IRRSUG00#	Function code 27 - get users groups
1	DECIMAL	28	IRRSFK00#	Function code 28 - fork exit
1	DECIMAL	29	IRRSMI00#	Function code 29 - makeISP
1	DECIMAL	30	IRRSKI00#	Function code 30 - check IPC access
1	DECIMAL	31	IRRSCI00#	Function code 31 - R_IPC_ctl
1	DECIMAL	32	IRRSC200#	Function code 32 - ck_owner_two_files
1	DECIMAL	33	IRRSGE00#	Function code 33 - get UIDs, GIDs, and supplemental groups
1	DECIMAL	34	IRRSDI00#	Function code 34 - R_dceinfo
1	DECIMAL	35	IRRSDK00#	Function code 35 - R_dcekey

Table 60. Constants for FC (continued)

Len	Type	Value	Name	Description
1	DECIMAL	36	IRRSUD00#	Function code 36 - R_dceruid
1	DECIMAL	37	IRRSDA00#	Function code 37 - R_dceauth
1	DECIMAL	38	IRRSIA00#	Function code 38 - initACEE
1	DECIMAL	39	IRRSEQ00#	Function code 39 - R_admin
1	DECIMAL	40	IRRSIM00#	Function code 40 - R_usermap
1	DECIMAL	41	IRRSDL00#	Function code 41 - R_datalib
1	DECIMAL	44	IRRSPX00#	Function code 44 - R_PKIServ
1	DECIMAL	45	IRRSCH00#	Function code 45 - R_cacheserv
1	DECIMAL	46	IRRSPY00#	Function code 46 - R_proxyserv
1	DECIMAL	47	IRRSCL00#	Function code 47 - R_setfacl
1	DECIMAL	48	IRRSSB00#	Function code 48 - R_setfsecl
1	DECIMAL	49	IRRSWP00#	Function code 49 - R_writepriv
1	DECIMAL	50	IRRSGS00#	Function code 50 - R_GenSec
1	DECIMAL	51	IRRSAX00#	Function code 51 - R_Auditx
1	DECIMAL	52	IRRSGI00#	Function code 52 - R_GetInfo
1	DECIMAL	53	IRRSPS00#	Function code 53 - R_PgmSignVer

FC cross reference

Table 61. Cross Reference for FC

Name	Offset	Hex Value
IRRSAU00#	0	13
IRRSAX00#	0	33
IRRSCA00#	0	11
IRRSCF00#	0	10
IRRSCH00#	0	2D
IRRSCI00#	0	1F
IRRSCL00#	0	2F
IRRSCO00#	0	F
IRRSCS00#	0	17
IRRSC200#	0	20
IRRSDA00#	0	25
IRRSDI00#	0	22
IRRSDK00#	0	23
IRRSDL00#	0	41

Table 61. Cross Reference for FC (continued)

Name	Offset	Hex Value
IRRSDU00#	0	2
IRRSEG00#	0	E
IRRSEQ00#	0	27
IRRSEU00#	0	C
IRRSEX00#	0	12
IRRSFK00#	0	1C
IRRSGE00#	0	21
IRRSGG00#	0	A
IRRSGI00#	0	34
IRRSGM00#	0	9
IRRSGS00#	0	32
IRRSIA00#	0	26
IRRSIM00#	0	28
IRRSIU00#	0	1
IRRSKA00#	0	6
IRRSKF00#	0	18
IRRSKI00#	0	1E
IRRSKO00#	0	14
IRRSKP00#	0	7
IRRSMF00#	0	3
IRRSMI00#	0	1D
IRRSMK00#	0	2A
IRRSMM00#	0	5
IRRSMR00#	0	19
IRRSPK00#	0	2B
IRRSPS00#	0	35
IRRSPT00#	0	1A
IRRSPW00#	0	36
IRRSPX00#	0	2C
IRRSPY00#	0	2E
IRRSQF00#	0	16
IRRSQS00#	0	15
IRRSSB00#	0	30
IRRSSG00#	0	D
IRRSSU00#	0	B
IRRSUD00#	0	24

Table 61. Cross Reference for FC (continued)

Name	Offset	Hex Value
IRRSUG00#	0	1B
IRRSUM00#	0	8
IRRSWP00#	0	31

Chapter 27. FXAP: RACROUTE REQUEST=FASTAUTH Extended Function Exit Parameter List

FXAP programming interface information

FXAP is a programming interface.

FXAP heading information

Common name:	RACROUTE REQUEST=FASTAUTH extended function exit parameter list
Macro ID:	ICHRFXAP
DSECT name:	RFXAPL
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	**Subpool** 229 **Key** 0 **Residency** Can be above 16MB
Size:	Variable
Created by:	RACROUTE REQUEST=FASTAUTH
Pointed to by:	R1 at entry to ICHRFX03 and ICHRFX04
Serialization:	None
Function:	Maps the parameter list passed to the RACROUTE REQUEST=FASTAUTH pre- or post-processing installation exit ICHRFX03 or ICHRFX04

FXAP mapping

Table 62. Structure RFXAPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	ADDRESS	4	ARFXEXPL	Address of current ICHRFX01/ ICHRFX02 parameter list which may be used by ICHRFX03/ ICHRFX04.
4	(4)	ADDRESS	4	ARFXALET	ALET of the Dataspace/Address space containing the profile and profile name. It is set for postprocessing exit only.

Table 62. Structure RFXAPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	ARFXPROF	Address of the profile being used within the dataspace/ Address space. It is set for postprocessing exit only.
12	(C)	UNSIGNED	2	ARFXAPRT	Type of authorization profile name: 1 = no profile used 2 = discrete profile name 3 = generic profile name in internal format 6 = no profile was found It is set for postprocessing exit only
14	(E)	UNSIGNED	2	ARFXAPRL	Length of authorization profile name. It is set for postprocessing exit only.
16	(10)	ADDRESS	4	ARFXAPRF	Address of authorization profile: points to the profile name used for the authorization check. The profile name is in the same data space or address space as the profile. It is set for post-processing exit only.
20	(14)	UNSIGNED	2	ARFXPVER	Parameter List Version: 1 - Extension exists if RCVTAUTU bit is on
22	(16)	UNSIGNED	2	ARFXPLEN	Parameter List Length
24	(18)	UNSIGNED	2	ARFXUSED	Authorization used to grant access. It is set for post processing exit only. The 2-byte area has the following format:
		1...		ARFXNORM	Normal authority was used .
		.1..		ARFXOPER	OPERATIONS authority was used .
		..1.		ARFXTRST	Trusted authority was used.
		...1		ARFXPRIV	Privileged authority was used.
		.... ðððð		*	Reserved
		ðððð ðððð		*	Reserved
26	(1A)	UNSIGNED	2	*	Reserved

FXAP constants

Table 63. Constants for FXAP

Len	Type	Value	Name	Description
1	DECIMAL	1	ARFXV1	Extension exists if RCVTAUTU bit is on.

Chapter 28. GANC: GENERICANCHOR settings mapping

GANC programming interface information

GANC is **NOT a programming interface.**

GANC heading information

Common name: GENERICANCHOR settings mapping

Macro ID: IRRPGANC

DSECT name: GENANC_SETTINGS, GENANC_JOBNAME_DATA

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: N/A

Storage attributes: Subpool 245

Size:

Section
Size

1
8 bytes

2
8 bytes plus an array of variable number of elements, with each element being 16 bytes long, at offset 8

Created by: The SET GENERICANCHOR command

Pointed to by:

Section 1:
Pointer RCVTGANC in the RCVT (ICHPRCVT)

Section 2:
Pointer GENANC_JOBNAME_DATAPTR in DSECT GENANC_SETTINGS

Serialization: ENQ

Function: Contains the generic anchor settings data for system level and jobname level generic anchors

GANC mapping

Table 64. Structure GENANC_SETTINGS

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	8	GENANC_SETTINGS	Generic Anchor Settings
0	(0)	UNSIGNED	4	GENANC_SYSTEM_COUNT	Number of generic anchors allowed for system

Table 64. Structure GENANC_SETTINGS (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	GENANC_JOBNAME_DATAPTR	Pointer to the jobname data which includes the list of jobnames and their associated generic anchor count values
GENANC_JOBNAME_DATA					
0	(0)	STRUCTURE	*	GENANC_JOBNAME_DATA	Jobname data table
0	(0)	CHARACTER	8	GENANC_JOBNAME_HEADER	Header data for jobname table
0	(0)	UNSIGNED	4	GENANC_TABLELEN	Length of table
4	(4)	UNSIGNED	4	GENANC_TOTAL_JOBNAMES	Total entries in table
8	(8)	STRUCTURE	16	GENANC_JOBNAME_ENTRIES(*)	Array of jobname entries
8	(8)	CHARACTER	8	GENANC_JOBNAME	Jobname
16	(10)	UNSIGNED	4	GENANC_JOBCOUNT	Number of generic anchors allowed for jobname
20	(14)	UNSIGNED	1	GENANC_JOBNAME_LENGTH	Length of jobname
21	(15)	BITSTRING	1	GENANC_JOBNAME_FLAGS	Jobname flags
		1...		GENANC_JOBNAME_GENERIC	1 - Jobname is generic
		.111 1111		*	Reserved
22	(16)	CHARACTER	2	*	Reserved

GANC cross reference

Table 65. Cross Reference for GANC

Name	Offset	Hex Value
GENANC_JOBCOUNT	10	
GENANC_JOBNAME	8	
GENANC_JOBNAME_DATA	0	
GENANC_JOBNAME_DATAPTR	4	
GENANC_JOBNAME_ENTRIES	8	
GENANC_JOBNAME_FLAGS	15	
GENANC_JOBNAME_GENERIC	15	80
GENANC_JOBNAME_HEADER	0	
GENANC_JOBNAME_LENGTH	14	
GENANC_TABLELEN	0	
GENANC_TOTAL_JOBNAMES	4	

Table 65. Cross Reference for GANC (continued)

Name	Offset	Hex Value
GENANC_SETTINGS	0	
GENANC_SYSTEM_COUNT	0	

Chapter 29. GAPL: Generic anchor table entry

GAPL programming interface information

GAPL is **NOT a programming interface.**

GAPL heading information

Common name: Generic anchor table entry

Macro ID: ICHGAPL

DSECT name: GENATE, GENPRFL, GENPLEL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: GATE (Offset: 0, Length: 4)

Storage attributes:

Subpool
255 (or as specified by the issuer of RACROUTE REQUEST=VERIFY)

On MVS/ESA: subpool 225 if ACEE is in subpool 255

Key
0

Size:

Section	**Size**
1	56 bytes minimum
2	28 bytes
3	20 bytes plus a variable of unknown length at offset 20
4	16 bytes
5	2 bytes

Created by: ICHGLS00 (LSQA or ELSQA)

Pointed to by:

Section 1:
ACEEGATA in ACEE data area or ATENEXT field in GAPL data area

Section 2:
After section 1 or pointed to by RTEGENL in ISP data area

Serialization: Local lock

Function: Contains descriptor and generic profile names for general resource class or data set high-level qualifier

GAPL mapping

Table 66. Structure GENATE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	56	GENATE	Generic Anchor Table Entry (GATE)
0	(0)	CHARACTER	4	ATEID	Identifier ('GATE')
4	(4)	UNSIGNED	1	ATEVERSN	Version
5	(5)	CHARACTER	3	*	Reserved
8	(8)	UNSIGNED	4	ATESPLN	
8	(8)	UNSIGNED	1	ATESP	SUBPOOL
9	(9)	UNSIGNED	3	ATELN	Length of GATE
12	(C)	CHARACTER	8	ATENM	Class name or HLQ, if DATASET
20	(14)	BITSTRING	1	ATEFLAG	Flags
		1...		ATETYP	Class TYPE: 1 - DATASET, 0 - General Resource
		.111 1111		*	Reserved
21	(15)	UNSIGNED	1	ATEMOBJS	Count of memory objects used for profile name list
22	(16)	UNSIGNED	2	ATEKQLEN	Length of the key qualifier
24	(18)	CHARACTER	4	*	Reserved
28	(1C)	SIGNED	4	ATERCNT	Refresh count
32	(20)	ADDRESS	4	ATEPROFL	Address of profile list - used for GENLIST GATEs
36	(24)	ADDRESS	4	ATENEXT	Address of next entry in anchor table
40	(28)	ADDRESS	8	ATEPRF64	Address of profile name list for those in 64bit storage
48	(30)	BITSTRING	8	ATETMSTP	GATE creation timestamp
56	(38)	CHARACTER	*	ATEKEYQ	Key qualifier of profile list

GENPRFL

0	(0)	STRUCTURE	28	GENPRFL	Profile list block header
0	(0)	CHARACTER	4	PRFLID	Identifier ('GPRF')
4	(4)	UNSIGNED	4	PRFLSPLN	
4	(4)	UNSIGNED	1	PRFLSP	Profile list block SUBPOOL
5	(5)	UNSIGNED	3	PRFLLN	Length of profile list block
8	(8)	ADDRESS	4	PRFLNEXT	Address of next block in profile list
12	(C)	SIGNED	2	PRFLNE	Number of entries in this block
14	(E)	UNSIGNED	2	*	Reserved
16	(10)	SIGNED	2	PRFLLH	Length of header

Table 66. Structure GENATE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
18	(12)	SIGNED	2	PRFLLE	If RACLIST format (fixed-length entries), length of each entry in the profile list. If normal format (variable-length entries), zero (use PLELNL and LENGTH(GENPLEL), instead)
20	(14)	ADDRESS	4	PRFSRTPL	Sorting Factor used by RACLIST
24	(18)	CHARACTER	4	PRFINDX	Address of maximal discrete prefix index structure
28	(1C)	CHARACTER		PRFENT	Start of profile list elements
GENPLEL					
0	(0)	STRUCTURE	*	GENPLEL	Profile list element
0	(0)	CHARACTER	20	GENPLELH	PLE Header
0	(0)	SIGNED	2	PLELNL	Length of resource name
2	(2)	CHARACTER	1	PLELFLGS	Flags
		1...		PLELRTRV	1- profile has been retrieved (or attempted, since it might be missing when the attempt is done)
		.111 1111		*	Reserved
3	(3)	CHARACTER	1	*	Reserved
4	(4)	ADDRESS	4	PLELPRF	Address of profile, or zero if profile has not yet been retrieved (or can not be found). This field is only valid if PLELRTRV is on
8	(8)	UNSIGNED	2	PLELRVRC	Count of RACFVARS variables in resource name
10	(A)	UNSIGNED	2	PLELRVCT	Count of valid RACFVARS variables in resource name
12	(C)	ADDRESS	4	PLELRVRP	Pointer to array of RACFVARS variables in the resource name
16	(10)	UNSIGNED	4	PLELMLSZ	Total amount of space needed for the member lists of all the RACFVARS variables in the resource name
20	(14)	CHARACTER	*	PLELNM	Resource name
PLELRVRS					
0	(0)	STRUCTURE	16	PLELRVRS(*)	Array of RACFVARS variables in the resource name
0	(0)	CHARACTER	8	PLELRVAR	RACFVARS variable
8	(8)	UNSIGNED	2	PLELMEMC	RACFVARS member count
10	(A)	UNSIGNED	2	PLELMEML	RACFVARS member list length

Table 66. Structure GENATE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	PLELMEMP	RACFVARS member list pointer
PLELMEM					
0	(0)	STRUCTURE	*	PLELMEM	RACFVARS member list
0	(0)	UNSIGNED	1	PLELMBRL	RACFVARS member length
1	(1)	CHARACTER	*	PLELMBR	RACFVARS member

GAPL constants

Table 67. Constants for GAPL

Len	Type	Value	Name	Description
1	DECIMAL	2	ATEVCON	GATE Version
4	DECIMAL	28	PRFLHLN	Length of profile list block header

GAPL cross reference

Table 68. Cross Reference for GAPL

Name	Offset	Hex Value
ATEFLAG	14	
ATEID	0	
ATEKEYQ	28	
ATEKQLEN	16	
ATELN	9	
ATEMOBJS	15	
ATENEXT	24	
ATENM	C	
ATEPRF64	28	
ATEPROFL	20	
ATERCNT	1C	
ATESP	8	
ATESPLN	8	
ATETMSTP	30	
ATETYP	14	80
ATEVERSN	4	
GENATE	0	
GENPLEL	0	
GENPLELH	0	
GENPRFL	0	

Table 68. Cross Reference for GAPL (continued)

Name	Offset	Hex Value
PLELFLGS	2	
PLELMBR	1	
PLELMBRL	0	
PLELMEM	0	
PLELMEMC	8	
PLELMEML	A	
PLELMEMP	C	
PLELMLSZ	10	
PLELNL	0	
PLELNM	14	
PLELPRF	4	
PLELRTRV	2	80
PLELRVAR	0	
PLELRVCT	A	
PLELRVRC	8	
PLELRVRP	C	
PLELRVRS	0	
PRFENT	1C	
PRFINDX	18	
PRFLID	0	
PRFLLE	12	
PRFLLH	10	
PRFLLN	5	
PRFLNE	C	
PRFLNEXT	8	
PRFLSP	4	
PRFLSPLN	4	
PRFSRTPL	14	

Chapter 30. GPRFL: 64-bit Storage profile list mapping

GPRFL programming interface information

GPRFL is **NOT a programming interface.**

GPRFL heading information

Common name: 64-bit Storage profile list mapping

Macro ID: IRRGPRFL

DSECT name: GPRFL64, GENPLE2

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: N/A

Storage attributes: 64-bit private storage

Size:

Section	Size
1	48 bytes
2	24 bytes plus a variable of unknown length at offset 24
3	20 bytes
4	16 bytes

Created by: IRRGLS21 in 64-bit private storage

Pointed to by:

Section 1:
Pointer ATEPRF64 in the GATE (ICHGAPL)

Section 2:
The pointer(s) in PR64ELT, the array of pointers to list elements, defined in DSECT GPRFL64

Section 3:
Pointer PLE2RVRP defined in DSECT GENPLE2

Section 4:
Pointer PR64NEXT defined in DSECT GPRFL64

Serialization: Local lock

Function: Contains descriptor and generic profile names for general resource class or data set high-level qualifier

GPRFL mapping

Table 69. Structure GPRFL64

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	GPRFL64	Profile list block header
0	(0)	CHARACTER	4	PR64ID	Identifier ('PR64')
4	(4)	UNSIGNED	4	PR64NE	Number of entries (profile names) in this block (memory object)
8	(8)	CHARACTER	8	*	RESERVED
16	(10)	ADDRESS	8	PR64NEXT	Address of next block in profile list (next memory object)
24	(18)	ADDRESS	8	PR64GP64	Pointer to first memory object containing the profiles themselves - Set only in first memory object
32	(20)	ADDRESS	8	PR64DATA	Pointer to start of data (PLELs, RVARs) in this memory object.
40	(28)	ADDRESS	*	PR64ELT(*)	Array of pointers to list elements
GENPLE2					
0	(0)	STRUCTURE	*	GENPLE2	Profile list element
0	(0)	CHARACTER	24	GENPLE2H	PLE Header
0	(0)	UNSIGNED	2	PLE2NL	Length of profile name
2	(2)	CHARACTER	1	PLE2FLGS	Flags
		1...		PLE2RTRV	1- profile has been retrieved (or attempted, since it might be missing when the attempt is done)
		.111 1111		*	RESERVED
3	(3)	CHARACTER	1	*	RESERVED
4	(4)	UNSIGNED	2	PLE2RVRC	Count of RACFVARS variables in profile name
6	(6)	UNSIGNED	2	PLE2RVCT	Count of valid RACFVARS variables in profile name
8	(8)	ADDRESS	8	PLE2PRF	Address of profile, or zero if profile has not yet been retrieved (or can not be found). This field is only valid if PLE2RTRV is on
16	(10)	ADDRESS	8	PLE2RVRP	Pointer to array of RACFVARS variables in the profile name
24	(18)	CHARACTER	*	PLE2NM	Profile name
PLE2RVRS					
0	(0)	STRUCTURE	20	PLE2RVRS(*)	Array of RACFVARS variables in the profile name
0	(0)	CHARACTER	8	PLE2RVAR	RACFVARS variable
8	(8)	ADDRESS	8	PLE2MEMP	RACFVARS member list ptr

Table 69. Structure GPRFL64 (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	UNSIGNED	2	PLE2MEMC	RACFVARS member count
18	(12)	UNSIGNED	2	PLE2MEML	RACFVARS member list length
GP64HDR					
0	(0)	STRUCTURE	16	GP64HDR	Memory object header for memory objects containing profile data
0	(0)	ADDRESS	8	GP64NXTO	Next memory object in chain
8	(8)	ADDRESS	8	GP64NXTP	Next available chunk in this memory object (for the next profile)

GPRFL cross reference

Table 70. Cross Reference for GPRFL

Name	Offset	Hex Value
GENPLE2	0	
GENPLE2H	0	
GPRFL64	0	
GP64HDR	0	
GP64NXTO	0	
GP64NXTP	8	
PLE2FLGS	2	
PLE2MEMC	10	
PLE2MEML	12	
PLE2MEMP	8	
PLE2NL	0	
PLE2NM	18	
PLE2PRF	8	
PLE2RTRV	2	80
PLE2RVAR	0	
PLE2RVCT	6	
PLE2RVRC	4	
PLE2RVRP	10	
PLE2RVRS	0	
PR64DATA	20	
PR64ELT	28	
PR64GP64	18	
PR64ID	0	

Table 70. Cross Reference for GPRFL (continued)

Name	Offset	Hex Value
PR64NE	4	
PR64NEXT	10	

Chapter 31. GRPF: In-Storage Generic Profile Map

GRPF programming interface information

GRPF is **NOT a programming interface.**

GRPF heading information

Common name: In-storage generic profile map

Macro ID: ICHGRPF

DSECT name: GRPF

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
255 (LSQA or ELSQA)

On MVS/ESA: subpool 225 if ACEE is in subpool 255

Key
0

Size:

Section
Size

1
68 bytes

2
2 bytes plus an unknown number of 9-byte fields at offset 2

3
2 bytes plus a variable of unknown length at offset 2

4
2 bytes plus an unknown number of 2-byte fields at offset 2

5
35 bytes plus a variable of unknown length at offset 35

Created by: ICHGLS00

Pointed to by: PLELPRF field in the GAPL data area

Serialization: Local lock

Function: Describes the structure of an in-storage generic profile

GRPF mapping

Table 71. Structure GRPF

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	74	GRPF	GENERIC PROFILE MAP
0	(0)	UNSIGNED	4	GRPFSPLN	

Table 71. Structure GRPF (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	UNSIGNED	1	GRPFS	AREA SUBPOOL NUMBER
1	(1)	UNSIGNED	3	GRPFL	TOTAL AREA LENGTH
4	(4)	CHARACTER	70	GRPFST	PROFILE DATA
4	(4)	CHARACTER	8	GRPFOWNR	RESOURCE OWNER
12	(C)	BITSTRING	1	GRPFUACC	UNIVERSAL ACCESS
13	(D)	BITSTRING	1	GRPFAUDT	AUDIT FLAGS
14	(E)	BITSTRING	1	GRPFGAUD	GLOBAL AUDIT FLAGS
15	(F)	ADDRESS	1	GRPFLEVL	RESOURCE LEVEL
16	(10)	SIGNED	4	GRPFACOF	OFFSET TO ACCESS
20	(14)	SIGNED	4	GRPFINOF	OFFSET TO INSTALLATION DATA
24	(18)	UNSIGNED	1	GRPFGPIN	GROUP/USER DATASET INDICATOR
25	(19)	BITSTRING	1	GRPFWARN	WARNING VALUE
26	(1A)	UNSIGNED	2	GRPFRTPD	RETENTION PERIOD
28	(1C)	BITSTRING	1	GRPFEOS	ERASE FLAG
29	(1D)	UNSIGNED	1	GRPFSLVL	RESOURCE SECLEVEL
30	(1E)	BITSTRING	1	GRPFLDAY	DAYS OF THE WEEK TERMINAL MAY NOT BE USED
31	(1F)	CHARACTER	8	GRPFNTFY	USERID TO NOTIFY WHEN THIS PROFILE DENIES ACCESS
39	(27)	CHARACTER	3	GRPFLGNT	EARLIEST TIME A TERMINAL MAY BE USED
42	(2A)	CHARACTER	3	GRPFLGFT	LATEST TIME A TERMINAL MAY BE USED
45	(2D)	CHARACTER	3	GRPFTZNE	TIME ZONE OFFSET OF TERMINAL FROM CPU
48	(30)	SIGNED	4	GRPFDPOF	OFFSET TO RESOURCE CATEGORY LIST
52	(34)	SIGNED	4	GRPFA2OF	OFFSET TO CONDITIONAL ACCESS LIST
56	(38)	CHARACTER	8	GRPFSLBL	SECLABEL
64	(40)	SIGNED	4	GRPFIPOF	Offset to IPLOOK
68	(44)	UNSIGNED	4	GRPFA3OF	Offset to PDSE MEMBER Access List
72	(48)	BITSTRING	1	GRPF_MACLOPDS_OPTIONS	MemberACL(PDS(Option)) flags
73	(49)	BITSTRING	1	GRPF_MACLOLIB_OPTIONS	MemberACL(PDSE(Option)) flags

GRPFACL

0	(0)	STRUCTURE	*	GRPFACL	ACCESS LIST

Table 71. Structure GRPF (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	UNSIGNED	2	GRPFACCT	NUMBER OF ENTRIES
2	(2)	CHARACTER	9	GRPFACLE(*)	ACCESS LIST ENTRIES
2	(2)	CHARACTER	8	GRPFACLU	USERID/GRPNAME
10	(A)	BITSTRING	1	GRPFACLA	ACCESS AUTHORITY
GRPFINSD					
0	(0)	STRUCTURE	*	GRPFINSD	INSTALLATION DATA
0	(0)	SIGNED	2	GRPFINSL	LENGTH OF INSTALLATION
2	(2)	CHARACTER	*	GRPFINST	INSTALLATION DATA
GRPFDPT					
0	(0)	STRUCTURE	*	GRPFDPT	CATEGORY LIST
0	(0)	SIGNED	2	GRPFDPTL	NUMBER OF ENTRIES IN CATEGORY LIST
2	(2)	SIGNED	2	GRPFDEPT(*)	CATEGORY LIST
GRPFACL2					
0	(0)	STRUCTURE	*	GRPFACL2	Second Access List
0	(0)	SIGNED	2	GRPFA2CT	NUMBER OF ENTRIES
2	(2)	UNSIGNED	2	GRPFA2LN	ACCESS LIST LENGTH
4	(4)	CHARACTER	20	GRPFAC2L	Entry structure
4	(4)	CHARACTER	8	GRPFAC2E	Progam Name / Flags
4	(4)	CHARACTER	1	GRPPGFLG	Flag byte
5	(5)	CHARACTER	7	GRPA2RST	The rest of name or flags
12	(C)	CHARACTER	8	GRPFAC2U	User / Group Id
20	(14)	BITSTRING	1	GRPFAC2A	Access Authority
21	(15)	UNSIGNED	2	GRPFGACS	Access Count
23	(17)	UNSIGNED	1	GRPFACVL	Variable Entity Length
24	(18)	CHARACTER	*	GRPFACVE	Variable Entity
24	(18)	CHARACTER	8	GRPAC2ID	Class ID
32	(20)	CHARACTER	2	GRPFRSVD	Reserved
34	(22)	UNSIGNED	1	GRPAC2LV	Variable Length
35	(23)	CHARACTER	*	GRPFAC2V	Variable Entity
GRPFIPLK					
0	(0)	CHARACTER	16	GRPFIPLK	IP Lookup value for SERVAUTH class profiles

GRPF constants

Table 72. Constants for GRPF

Len	Type	Value	Name	Description
1	DECIMAL	0	GRPA2DAT	Flag data equate

GRPF cross reference

Table 73. Cross Reference for GRPF

Name	Offset	Hex Value
GRPAC2ID	18	
GRPAC2LV	22	
GRPA2RST	5	
GRPF	0	
GRPFACCT	0	
GRPFACL	0	
GRPFACLA	A	
GRPFACLE	2	
GRPFACLU	2	
GRPFACL2	0	
GRPFACOF	10	
GRPFACVE	18	
GRPFACVL	17	
GRPFAC2A	14	
GRPFAC2E	4	
GRPFAC2L	4	
GRPFAC2U	C	
GRPFAC2V	23	
GRPFAUDT	D	
GRPFA2CT	0	
GRPFA2LN	2	
GRPFA2OF	34	
GRPFDEPT	2	
GRPFDPOF	30	
GRPFDPT	0	
GRPFDPTL	0	
GRPFEOS	1C	
GRPFGACS	15	
GRPFGAUD	E	

Table 73. Cross Reference for GRPF (continued)

Name	Offset	Hex Value
GRPFGPIN	18	
GRPFINOF	14	
GRPFINSD	0	
GRPFINSL	0	
GRPFINST	2	
GRPFIPLK	0	
GRPFIPOF	40	
GRPFL	1	
GRPFLDAY	1E	
GRPFLEVL	F	
GRPFLGFT	2A	
GRPFLGNT	27	
GRPFNTFY	1F	
GRPFOWNR	4	
GRPFRSVD	20	
GRPFRTPD	1A	
GRPFS	0	
GRPFSLBL	38	
GRPFSLVL	1D	
GRPFSPLN	0	
GRPFST	4	
GRPFTZNE	2D	
GRPFUACC	C	
GRPFWARN	19	
GRPPGFLG	4	

Chapter 32. ICRX: Extended identity context reference

ICRX programming interface information

ICRX is a programming interface.

ICRX heading information

Common name: Extended identity context reference (ICRX)

Macro ID: IRRPICRX

DSECT name: ICRX

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: ICRX (Offset 0, length 4)

Storage attributes:

Subpool
Determined by user

Key
Determined by user

Residency
Determined by user

Size: Variable

Created by: RACF or caller of RACF

Pointed to by: Output of R_cacheserv, input for RACROUTE REQUEST=VERIFY

Serialization: None

Function: Holds information needed to retrieve or rebuild an authenticated distributed user's z/OS security environment.

Note: The ICRX is a variable length data area. It consists of the structure shown below, containing data lengths and offsets, followed by the data values. ICRXLEN contains the length of the entire ICRX structure. All offsets are relative to the beginning of the ICRX data area. All data in the ICRX must be contiguous.

ICRX mapping

Table 74. Structure ICRX

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	ICRX	
0	(0)	CHARACTER	4	ICRXID	Literal ID 'ICRX'
4	(4)	UNSIGNED	1	ICRXVERS	Version number
5	(5)	UNSIGNED	1	ICRXOFFN	Number of offsets - 3 for versions 1 and 2

Table 74. Structure ICRX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
6	(6)	CHARACTER	1	ICRXFLGS	Flag bits
		1...		ICRXMULT	Multiple-use ICRX
7	(7)	UNSIGNED	1	ICRXSP	Subpool
8	(8)	UNSIGNED	4	ICRXLEN	Total length of ICRX
12	(C)	UNSIGNED	4	ICRXICRO	Offset to ICR or zero if omitted
16	(10)	UNSIGNED	4	ICRXDIDO	Offset to IDID or zero if omitted
20	(14)	UNSIGNED	4	ICRXUSRO	Offset to RACF userID or zero if omitted

ICRXUSER

0	(0)	STRUCTURE	9	ICRXUSER	Userid buffer
0	(0)	UNSIGNED	1	ICRXUSRL	USERID Length
1	(1)	CHARACTER	8	ICRXUSRI	USERID

ICRXICR

0	(0)	STRUCTURE	16	ICRXICR	Identity context Reference
0	(0)	CHARACTER	8	ICRXREFU	Reference part 1
8	(8)	CHARACTER	8	ICRXREFR	Reference part 2

ICRX constants

Table 75. Constants for ICRX

Len	Type	Value	Name	Description
4	CHARACTER	ICRX	ICRXIDC	ICRX ID Constant
1	DECIMAL	1	ICRXVR01	Version 1 of extended identity context reference
1	DECIMAL	2	ICRXVR02	Version 2 of extended identity context reference
1	DECIMAL	2	ICRXCURV	Current version of extended identity context of ID context reference

ICRX cross reference

Table 76. Cross Reference for ICRX

Name	Offset	Hex Value
ICRX	0	
ICRXDIDO	10	
ICRXFLGS	6	
ICRXICR	0	

Table 76. Cross Reference for ICRX (continued)

Name	Offset	Hex Value
ICRXICRO	C	
ICRXID	0	
ICRXLEN	8	
ICRXMULT	6	80
ICRXOFFN	5	
ICRXREFR	8	
ICRXREFU	0	
ICRXSP	7	
ICRXUSER	0	
ICRXUSRI	1	
ICRXUSRL	0	
ICRXUSRO	14	
ICRXVERS	4	

Chapter 33. ICTX: Distributed Identity Support SAF

ICTX programming interface information

ICTX is a programming interface.

ICTX heading information

Common name: Identity context extension (ICTX)

Macro ID: IRRPICTX

DSECT name: ICTX

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: ICTX (Offset 0, length 4)

Storage attributes:

Subpool
255 (or the ACEE subpool as specified by the issuer of RACROUTE REQUEST=VERIFY)

Key
0

Residency
Resides above 16M

Size: Variable

Created by: RACF

Pointed to by: ACEEICTX

Serialization: None

Function: Holds identity context information about the initial authentication of the user.

Notes:

1. The ICTX is a variable length control block. It consists of the structure shown below, containing data lengths and offsets, followed by the data values. ICTXLEN contains the length of the ICTX structure, 40 bytes for z/OS V1R8, plus the sum of all of the data lengths: ICTXUSRL, ICTXREGL, and so forth.
2. All offsets are relative to the beginning of the ICTX control block. For example, the offset to the authenticated user name (ICTXUSR@) is 40(28) for z/OS V1R8. The offset to the registry name (ICTXREG@) is the sum of the ICTXUSR@ offset value plus the length of the authenticated user name (ICTXUSRL), if ICTXUSRL is 16, ICTXREG@ is 56(38) for z/OS V1R8. The other offset values are calculated similarly.

ICTX mapping

Table 77. Structure ICTX

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	40	ICTX	
0	(0)	CHARACTER	4	ICTXID	Literal id 'ICTX'

Table 77. Structure ICTX (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	UNSIGNED	1	ICTXVERS	Level of id context extension
5	(5)	CHARACTER	2	*	Reserved
7	(7)	UNSIGNED	1	ICTXSP	Subpool number
8	(8)	UNSIGNED	4	ICTXLEN	Length of id context extension
12	(C)	UNSIGNED	2	ICTXUSRL	Length of authenticated user name, or zero if the name is omitted. The user name has a maximum length of 510 bytes.
14	(E)	UNSIGNED	2	ICTXUSR@	Offset to authenticated user name, or zero if omitted
16	(10)	UNSIGNED	2	ICTXREGL	Length of registry name or zero if the name is omitted. The registry name has a maximum length of 255 bytes.
18	(12)	UNSIGNED	2	ICTXREG@	Offset to registry name, or zero if omitted
20	(14)	UNSIGNED	2	ICTXHSTL	Length of host name, or zero if the name is omitted. The host name has a maximum length of 128 bytes.
22	(16)	UNSIGNED	2	ICTXHST@	Offset to host name, or zero if omitted
24	(18)	UNSIGNED	2	ICTXMCHL	Length of authentication mechanism object identifier (OID), or zero if the authentication mechanism is omitted. The authentication mechanism has a maximum length of 16 bytes.
26	(1A)	BITSTRING	2	ICTXMCH@	Offset to authentication mechanism, zero zero if omitted
28	(1C)	CHARACTER	12		Reserved

ICTX constants

Table 78. Constants for ICTX

Len	Type	Value	Name	Description
4	CHARACTER	ICTX	ICTXIDC	ICTX ID constant
1	DECIMAL	1	ICTXVR01	Version 1 of ID context extension
1	DECIMAL	1	ICTXCURV	Current version of ID context extension

ICTX cross reference

Table 79. Cross Reference for ICTX

Name	Offset	Hex Value
ICTX	0	
ICTXID	0	
ICTXVERS	4	
ICTXSP	7	
ICTXLEN	8	
ICTXUSRL	C	
ICTXUSR@	E	
ICTXREGL	10	
ICTXREG@	12	
ICTXHSTL	14	
ICTXHST@	16	
ICTXMCHL	18	
ICTXMCH@	1A	

Chapter 34. IDID: Distributed identity data

IDID programming interface information

IDID is a programming interface.

IDID heading information

Common name: Distributed identity data (IDID)

Macro ID: IRRPIDID

DSECT name: IDID

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: IDID (Offset 0, length 4)

Storage attributes:

Subpool
Same as ACEE when pointed to by ACEE

Key
Same as ACEE when pointed to by ACEE

Residency
Resides above 16M

Size: Variable

Created by: RACF or caller of RACF

Pointed to by: ACEEIDID

Serialization: None

Function: Holds distributed identity information used by RACF's caller to authenticate the user.

Notes:

1. The IDID is a variable length data area. It consists of the structure shown below, containing offsets to data, followed by the data values. IDID_LEN contains the length of the IDID header, plus the sum of the section lengths. Offsets in the IDID header are relative to the beginning of the IDID data area. Offsets within sections are relative to the start of the section. All data in the IDID must be contiguous, and the sections are expected to be in the order shown.
2. IDID is input to the initACEE callable service and RACROUTE REQUEST=VERIFY. This input results in an IDID data area pointed to by the ACEE, residing in the same subpool as the ACEE.

IDID mapping

Table 80. Structure IDID

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	32	IDID	Distributed Identity Data
0	(0)	CHARACTER	4	IDIDID	Literal id 'IDID'

Table 80. Structure IDID (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	UNSIGNED	1	IDIDVERS	Version number
5	(5)	UNSIGNED	1	IDIDOFFN	Number of offsets - 5 for version 1
6	(6)	BINARY	1	IDIDHSHN	Sections to include in the hash for local cache comparison. For version 1: ....1 Include section 1 .... ..1. Include section 2 0000 00.. Do not include other sections If no bits are on, default should be section 1 and 2 only.
7	(7)	UNSIGNED	1	IDIDSP	Subpool - required when passed as IDID= parameter on RACROUTE
8	(8)	UNSIGNED	4	IDIDLEN	Total length of IDID
12	(C)	UNSIGNED	4	IDIDOFF1	Offset to section 1 (IDIDSEC1) or zero if omitted
16	(10)	UNSIGNED	4	IDIDOFF2	Offset to section 2 (IDIDSEC2) or zero if omitted
20	(14)	UNSIGNED	4	IDIDOFF3	Offset to section 3 (IDIDSEC3) or zero if omitted
24	(18)	UNSIGNED	4	IDIDOFF4	Offset to section 4 (IDIDSEC4) or zero if omitted
28	(1C)	UNSIGNED	4	IDIDOFF5	Offset to section 5 (IDIDSEC5) or zero if omitted

IDIDSEC1

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	12	IDIDSEC1	Section 1 of IDID - z/OS section
0	(0)	BINARY	2	IDID1FLG	Flags
2	(2)	CHARACTER	1	*	Reserved
3	(3)	UNSIGNED	1	IDID1NMF	Name format
4	(4)	UNSIGNED	4	IDID1OF1	Offset to user DN (IDID1USR)
8	(8)	UNSIGNED	4	IDID1OF2	Offset to registry name (IDID1REG)

IDID1USR

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDID1USR	User's distinguished name
0	(0)	BINARY	2	IDID1UDL	Length of user's name up to RCVTDNL maximum

Table 80. Structure IDID (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
2	(2)	UTF8	*	IDID1UDN	Distributed client end-user's identity, within registry designated by IDID1REG, in UTF8 form, represented as one of the following: • LDAP string form of the user's X.500 Distinguished Name as defined within the LDAP registry, in canonical form as in RFC2253, and with LDAP special characters escaped with a '\' (UTF-8'92'x). Note that this is identical to what is the result of a WAS wscredential.getUniqueSecurity Name() method invocation run on a WAS server having authenticated the user by way of the LDAP registry. • A simple character string such as a user-ID as defined within a registry.

IDID1REG

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDID1REG	Registry's name
0	(0)	BINARY	2	IDID1RL	Length of registry's name up to RCVTRL maximum
2	(2)	UTF8	*	IDID1RN	Name of original registry in UTF8 format Note that this is identical to what would be the result of a WAS wscredential.getRealmName() method invocation run on a WAS server having authenticated the user by way of the LDAP registry.

IDIDSEC2

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDIDSEC2	Section 2 of IDID-Reserved for additional RACF security information.

IDIDSEC3

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDIDSEC3	Section 3 of IDID-Reserved for alternative security information.

IDIDSEC4

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDIDSEC4	Section 4 of IDID-Reserved for customer use.

IDIDSEC5

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	IDIDSEC5	Section 5 of IDID-Reserved for use by Websphere Application Server.

IDID constants

Table 81. Constants for IDID

Len	Type	Value	Name	Description
4	CHARACTER	IDID	IDIDIDC	IDID ID Constant
1	DECIMAL	1	IDIDVR01	Version 1 of distributed identity data
1	DECIMAL	1	IDIDCURV	Current version of distributed identity data
1	DECIMAL	3	IDIDCURH	Current sections to hash

IDID cross reference

Table 82. Cross Reference for IDID

Name	Offset	Hex Value
IDID	0	
IDID1FLG	0	
IDID1NMF	3	
IDID1OF1	4	
IDID1OF2	8	
IDID1UDL	0	
IDID1USR	0	
IDIDID	0	
IDIDVERS	4	
IDIDOFFN	5	
IDIDHSHN	6	
IDIDSP	7	
IDIDLEN	8	
IDIDOFF1	C	
IDIDOFF2	10	
IDIDOFF3	14	
IDIDOFF4	18	
IDIDOFF5	1C	
IDIDSEC1	0	
IDIDUDN	2	
IDID1REG	0	
IDID1RL	0	
IDID1RN	2	
IDIDSEC2	0	
IDIDSEC3	0	

Table 82. Cross Reference for IDID (continued)

Name	Offset	Hex Value
IDIDSEC4	0	
IDIDSEC5	0	

Chapter 35. IFSP: z/OS UNIX System Services File Security Packet

IFSP programming interface information

IFSP is a programming interface.

IFSP heading information

Common name:	z/OS UNIX System Services file security packet
Macro ID:	IRRPIFSP
DSECT name:	IFSP
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	IFSP
Storage attributes:	N/A
Size:	64 bytes
Created by:	Invoker of z/OS UNIX security functions
Pointed to by:	IRRPCOMP, the common SAF/RACF parameter list for zOS UNIX security functions
Serialization:	None
Function:	Contains z/OS UNIX-related information for a z/OS UNIX file

IFSP mapping

Table 83. Structure IFSP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	64	IFSP	
0	(0)	CHARACTER	4	IFSP_ID	LITERAL ID 'IFSP'
4	(4)	UNSIGNED	1	IFSP_VERS	Version Number for FSP
5	(5)	UNSIGNED	1	*	Reserved
6	(6)	UNSIGNED	2	IFSP_LEN	Length of the FSP
8	(8)	SIGNED	4	IFSP_OWN_UID	OWNING UID
12	(C)	SIGNED	4	IFSP_OWN_GID	OWNING GID
16	(10)	BITSTRING	4	IFSP_PERMISSION	Permission bits
16	(10)	BITSTRING	1	IFSP_OWNER_8	Owner permission bits
		1111 1...		*	Reserved
		.... .111		IFSP_OWNER	Owner perm bits
		.... .1..		IFSP_OWNER_READ	Owner read perm bit

Table 83. Structure IFSP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... ..1.		IFSP_OWNER_WRITE	Owner Write perm bit
		1		IFSP_OWNER_EXECUTE	Owner execute perm bit
17	(11)	BITSTRING	1	IFSP_GROUP_8	Group perm byte size bits
		1111 1...		*	Reserved
		.... .111		IFSP_GROUP	Group perm bits
		.... .1..		IFSP_GROUP_READ	Group read perm bit
		.... ..1.		IFSP_GROUP_WRITE	Group Write perm bit
		1		IFSP_GROUP_EXECUTE	Group execute perm bit
18	(12)	BITSTRING	1	IFSP_OTHER_8	Other perm byte size bits
		1111 1...		*	Reserved
		.... .111		IFSP_OTHER	Other perm bits
		.... .1..		IFSP_OTHER_READ	Other read perm bit
		.... ..1.		IFSP_OTHER_WRITE	Other Write perm bit
		1		IFSP_OTHER_EXECUTE	Other execute perm bit
19	(13)	BITSTRING	8	IFSP_FLAG2	Additional FSP flags
		1...		IFSP_Access_Acl	An access ACL exists
		.1..		IFSP_File_Model_Acl	A file model ACL exists
		..1.		IFSP_Dir_Model_Acl	A directory model ACL exists
		...1 1111		*	Reserved
20	(14)	BITSTRING	4	IFSP_FLAG	File flags
		1...		IFSP_DIRECTORY	File is Directory
20	(14)	BITSTRING	3	*	Reserved
		.... .1..		IFSP_S_ISVTX	S_ISVTX
		.... ..1.		IFSP_S_ISUID	S_ISUID - setuid file
		1		IFSP_S_ISGID	S_ISGID - setgid file
24	(18)	BITSTRING	4	IFSP_USER_AUDIT_OPTS	User Audit Options
24	(18)	BITSTRING	1	IFSP_USER_AUDIT_READ	Read Access Options
25	(19)	BITSTRING	1	IFSP_USER_AUDIT_WRITE	Write Access Options
26	(1A)	BITSTRING	1	IFSP_USER_AUDIT_EXEC_SCH	Execute/Search Access Options
27	(1B)	BITSTRING	1	*	Reserved
28	(1C)	BITSTRING	4	IFSP_ADTR_AUDIT_OPTS(0)	Auditor Audit Options
28	(1C)	BITSTRING	1	IFSP_ADTR_AUDIT_READ	Read Access Options
29	(1D)	BITSTRING	1	IFSP_ADTR_AUDIT_WRITE	Write Access Options
30	(1E)	BITSTRING	1	IFSP_ADTR_AUDIT_EXEC_SCH	Execute/Search Access Options

Table 83. Structure IFSP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
31	(1F)	BITSTRING	1	*	Reserved
32	(20)	CHARACTER	8	IFSP_SECLABEL	Security Label
40	(28)	CHARACTER	24	*	Reserved

IFSP constants

Table 84. Constants for IFSP

Len	Type	Value	Name	Description
4	CHARACTER	IFSP	IFSP_IDC	FSP ID Constant
1	DECIMAL	1	IFSP_CV01	Version 1 of FSP
1	DECIMAL	1	IFSP_CVER	Current version of FSP
1	HEX	00	IFSP_AUD_NONE	Do not Audit any access attempts
1	HEX	01	IFSP_AUD_SUCC	Audit successful access
1	HEX	02	IFSP_AUD_FAIL	Audit failed access attempts

IFSP cross reference

Table 85. Cross Reference for IFSP

Name	Offset	Hex Value
IFSP	0	
IFSP_ADTR_AUDIT_EXEC_SCH	1E	
IFSP_ACCESS_ACL	13	
IFSP_ADTR_AUDIT_OPTS	1C	
IFSP_ADTR_AUDIT_READ	1C	
IFSP_ADTR_AUDIT_WRITE	1D	
IFSP_DIR_MODEL_ACL	13	
IFSP_DIRECTORY	14	80
IFSP_FILE_MODEL_ACL	13	
IFSP_FLAG	14	
IFSP_GROUP	11	04
IFSP_GROUP_EXECUTE	11	01
IFSP_GROUP_READ	11	04
IFSP_GROUP_WRITE	11	02
IFSP_GROUP_8	11	
IFSP_ID	0	
IFSP_LEN	6	

Table 85. Cross Reference for IFSP (continued)

Name	Offset	Hex Value
IFSP_OTHER	12	04
IFSP_OTHER_EXECUTE	12	01
IFSP_OTHER_READ	12	04
IFSP_OTHER_WRITE	12	02
IFSP_OTHER_8	12	
IFSP_OWN_GID	C	
IFSP_OWN_UID	8	
IFSP_OWNER	10	04
IFSP_OWNER_EXECUTE	10	01
IFSP_OWNER_READ	10	04
IFSP_OWNER_WRITE	10	02
IFSP_OWNER_8	10	
IFSP_PERMISSION	10	
IFSP_S_ISGID	17	01
IFSP_S_ISUID	17	02
IFSP_S_ISVTX	17	04
IFSP_SECLABEL	20	
IFSP_USER_AUDIT_EXEC_SCH	1A	
IFSP_USER_AUDIT_OPTS	18	
IFSP_USER_AUDIT_READ	18	
IFSP_USER_AUDIT_WRITE	19	
IFSP_VERS	4	

Chapter 36. IISP: z/OS UNIX System Services IPC Security Packet

IISP heading information

Common name: z/OS UNIX System Services IPC security packet

Macro ID: IRRPIISP

DSECT name: IISP

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: IISP (Offset: 0, Length: 4)

Storage attributes:

Subpool
3

Key
0 (not fetch-protected)

Residency
Private storage of module that creates it

Size: 64 bytes

Created by: Callable service: makeISP (IRRSMI00)

Pointed to by: IRRPCOMP, the common SAF/RACF parameter list for z/OS UNIX security functions

Serialization: None

Function: Maps the structure of the IPC security packet

IISP mapping

Table 86. Structure IISP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	64	IISP	
0	(0)	CHARACTER	4	IISP_ID	LITERAL ID 'IISP'
4	(4)	UNSIGNED	1	IISP_VERS	Version Number for ISP
5	(5)	UNSIGNED	1	*	Reserved
6	(6)	UNSIGNED	2	IISP_LEN	Length of the ISP
8	(8)	UNSIGNED	4	IISP_IPCP_ALET	ALET of IPCP
12	(C)	ADDRESS	4	IISP_IPCP_PTR	Address of IPCP mapped by BPXYIPCP
16	(10)	UNSIGNED	4	IISP_IPCKEY	IPC Key
20	(14)	UNSIGNED	4	IISP_IPCID	IPC ID
24	(18)	CHARACTER	8	IISP_SECLABEL	Security label
32	(20)	CHARACTER	32	*	Reserved

IISP constants

Table 87. Constants for IISP

Len	Type	Value	Name	Description
4	CHARACTER	IISP	IISP_IDC	ISP ID Constant
1	DECIMAL	1	IISP_CV01	Version 1 of ISP
1	DECIMAL	1	IISP_CVER	Current version of ISP
2	DECIMAL	64	IISP#LEN	Length

IISP cross reference

Table 88. Cross Reference for IISP

Name	Offset	Hex Value
IISP	0	
IISP_ID	0	
IISP_IPCID	14	
IISP_IPCKEY	10	
IISP_IPCP_ALET	8	
IISP_IPCP_PTR	C	
IISP_LEN	6	
IISP_SECLABEL	18	
IISP_VERS	4	

Chapter 37. ISP: RACF In-Storage Profile

ISP programming interface information

The following fields are Not Programming Interface information:

- RACRTE
- RACRSE
- RACRNE

ISP heading information

Common name:	RACF in-storage profile
Macro ID:	ICHPISP
DSECT name:	RACRTE, RACRSE, RACRNE, RACRPE, RPEINST, RPEAPPL, RPEACCLE, RPEPTD, RPESESSN, RPEACL2, RPEMEM, RPESESS2
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	None
Storage attributes:	**Subpool** 255 (or as specified by issuer of RACROUTE REQUEST=LIST) **Key** 0 **Residency** May reside above 16M

Size:

Section	**Size**
1	44 bytes
2	8 bytes plus a variable of unknown length at offset 8
3	16 bytes plus a variable of unknown length at offset 16
4	84 bytes
5	1 bytes plus a variable of unknown length at offset 1
6	1 bytes plus a variable of unknown length at offset 1
7	9 bytes per entry in the access list
8	2 bytes per category
9	10 bytes plus 1 to 8 characters at offset 10
10	31 bytes plus a variable of unknown length at offset 31
11	Variable
12	1 byte

Created by: RACROUTE REQUEST=LIST processing

Pointed to by: ACEECLCP field of the ACEE data area. On systems prior to MVS/ESA, CNSTRCLP also points to it. Individual profiles can be located in 2 ways:

1. Using RACROUTE REQUEST=AUTH with ENTITY=(...,CSA or PRIVATE), which returns a copy of the profile mapped by ICHRRPF.
2. For a RACROUTE REQUEST=LIST tree pointed to from the ACEE, using RACROUTE REQUEST=FASTAUTH which returns a pointer to a profile that was used in word 14 of the work area pointed to by WKAREA.

Serialization: None

Function: Contains profiles for general resources in a class plus control information for locating individual profiles

ISP mapping

Table 89. Structure RACRTE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RACRTE	RACLIST CLASS TREE ANCHOR ELEMENT
0	(0)	ADDRESS	4	RTENEXT	ADDRESS OF NEXT ANCHOR OR 0

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	RTECLASS	ADDRESS OF CLASS DESCRIPTOR ENTRY FOR THIS CLASS
8	(8)	ADDRESS	4	RTETREE	ADDRESS OF TOP NODE IN TREE OR 0
12	(C)	ADDRESS	4	RTESTORE	ADDRESS OF STORAGE BLOCK LIST OR 0
16	(10)	CHARACTER	2	RTESPNS(0)	PROFILE AND NODE SUBPOOL NUMBERS
16	(10)	UNSIGNED	1	RTEPSPN	SUBPOOL NUMBER FOR PROFILES
17	(11)	UNSIGNED	1	RTENSPN	SUBPOOL NUMBER FOR TREE NODES
18	(12)	UNSIGNED	1	RTEASPN	SUBPOOL NUMBER OF THIS BLOCK
19	(13)	CHARACTER	1	*	RESERVED FOR RACGLIST USE
20	(14)	ADDRESS	4	RTEGENL	ADDRESS OF GENERIC PROFILE LIST OR 0
24	(18)	SIGNED	4	RTESIZE	TOTAL STORAGE USED FOR RACLISTED PROFILES AND NODES
28	(1C)	SIGNED	4	RTEGNUM	TOTAL NUMBER OF GROUPING PROFILES THAT CONTAIN MEMBERS
32	(20)	CHARACTER	8	RTESTOKN	STOKEN OF GLOBAL=YES DATASPACE
40	(28)	UNSIGNED	4	RTEALET	PASN-AL ALET OF DATASPACE

RACRSE

0	(0)	STRUCTURE	8	RACRSE	RACLIST CLASS TREE STORAGE BLOCK
0	(0)	ADDRESS	4	RSENEXT	ADDRESS OF NEXT STORAGE BLOCK OR 0
4	(4)	SIGNED	2	RSESIZE	LENGTH OF STORAGE BLOCK
6	(6)	UNSIGNED	1	RSEPOOL	SUBPOOL NUMBER OF STORAGE BLOCK
7	(7)	UNSIGNED	1	*	RESERVED
8	(8)	CHARACTER	*	RSESTORE	USEABLE STORAGE (RSESIZE-4 BYTES)

RACRNE

0	(0)	STRUCTURE	20	RACRNE	RACLIST CLASS TREE NODE ELEMENT
0	(0)	ADDRESS	4	RNELEFT	ADDRESS OF LEFT DAUGHTER NODE OR 0

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	RNEPROF	ADDRESS OF PROFILE FOR THIS NODE
8	(8)	ADDRESS	4	RNERIGHT	ADDRESS OF RIGHT DAUGHTER NODE OR 0
12	(C)	SIGNED	4	RNEBAL	TREE BALANCING FACTOR DURING TREE CREATION
12	(C)	ADDRESS	4	RNEUP	POINTER TO MOTHER NODE DURING TREE DELETION
16	(10)	CHARACTER	*	RNEKEY	KEY (LENGTH DETERMINED BY MAXIMUM NAME LENGTH FOR CLASS IN THE CLASS DESCRIPTOR ELEMENT)

RACRPE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	90	RACRPE	RESOURCE PROFILE ELEMENT
0	(0)	UNSIGNED	2	RPEPLEN	PHYSICAL STORAGE LENGTH OF BLOCK
2	(2)	UNSIGNED	2	RPELLEN	LOGICAL LENGTH OF BLOCK
4	(4)	UNSIGNED	2	RPEUCNT	NUMBER OF RESOURCES SHARING THIS PROFILE
6	(6)	CHARCTER	4	RPEATTR	ATTRIBUTE FLAGS
6	(6)	BITSTRING	1	RPEUACC	UNIVERSAL ACCESS
7	(7)	BITSTRING	1	RPEAUDIT	AUDIT FLAGS
8	(8)	BITSTRING	1	RPEGAUD	GLOBAL AUDIT FLAGS
9	(9)	BITSTRING	1	RPELEVEL	RESOURCE LEVEL
10	(A)	UNSIGNED	2	RPEACCNO	NUMBER OF ENTRIES IN ACCESS LIST
12	(C)	UNSIGNED	2	RPEACCOF	OFFSET TO ACCESS LIST
14	(E)	UNSIGNED	2	RPEINSOF	OFFSET TO INSTALLATION DATA
16	(10)	UNSIGNED	2	RPEAPPOF	OFFSET TO APPLICATION DATA
18	(12)	CHARACTER	8	RPEOWNER	OWNER OF RESOURCE PROFILE
26	(1A)	SIGNED	2	RPENUMDP	NUMBER OF CATEGORIES IN LIST
28	(1C)	UNSIGNED	2	RPEDPTOF	OFFSET TO CATEGORY LIST
30	(1E)	BITSTRING	1	RPELDAYS	DAYS TERMINAL MAY NOT BE USED (BIT 0 - SUNDAY, BIT 1 - MONDAY, ...)
31	(1F)	UNSIGNED	1	RPESCLVL	RESOURCE SECURITY LEVEL
32	(20)	CHARACTER	3	RPELOGNT	EARLIEST TIME TERMINAL MAY BE USED (HHMM)
35	(23)	CHARACTER	3	RPELOGFT	LATEST TIME TERMINAL MAY BE USED (HHMM)

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
38	(26)	CHARACTER	8	RPENTFY	USERID TO NOTIFY WHEN THIS PROFILE DENIES ACCESS
46	(2E)	CHARACTER	3	RPETZONE	TIME OFFSET OF TERMINAL FROM CPU. + = EAST - = WEST.
49	(31)	BITSTRING	1	RPEFLAGS	FLAGS FOR IN STORE PROFILE
		1...		RPEFWARN	WARN OPTION SPECIFIED?
		.1..		RPEFSCPW	ICSF CPACF WRAP ON?
		..1.		RPEFSCPR	ICSF CPACF RET ON?
		...1 1111		*	RESERVED
50	(32)	CHARACTER	8	RPESCLBL	SECLABEL
58	(3A)	UNSIGNED	2	RPESESOF	SESSION SEG DATA OFF
60	(3C)	UNSIGNED	2	RPESESLN	SESSION SEG DATA LEN
62	(3E)	UNSIGNED	2	RPEAC2NO	NUMBER OF OCCURRENCES
64	(40)	UNSIGNED	2	RPEAC2LN	CONDITIONAL ACCESS LIST LENGTH
66	(42)	UNSIGNED	2	RPEAC2OF	SECOND ACCESS LIST OFFSET
68	(44)	UNSIGNED	2	RPEMEMCT	NUMBER OF MEMBERS
70	(46)	UNSIGNED	2	RPEMEMLN	LENGTH OF MEMBER LIST
72	(48)	UNSIGNED	2	RPEMEMOF	OFFSET TO MEMBER LIST
74	(4A)	SIGNED	2	RPESE2LN	MORE SESSION DATA LENGTH
76	(4C)	SIGNED	2	RPESE2OF	MORE SESSION DATA OFFSET
78	(4E)	SIGNED	2	RPEIPOFF	OFFSET OF IP LOOKUP VALUE OR 0 IF THERE IS NO IP LOOKUP VALUE
80	(50)	SIGNED	2	RPECSFLN	ICSF SEGMENT INFO LENGTH
82	(52)	UNSIGNED	2	RPECSFOF	ICSF SEGMENT INFO OFFSET
84	(54)	UNSIGNED	2	RPEDIDCT	NUMBER OF DIDLIST1 ENTRIES
86	(56)	UNSIGNED	2	RPEDIDLN	LENGTH OF DISTRIBUTED IDENTITY INFORMATION (DIDLIST1)
88	(58)	UNSIGNED	2	RPEDIDOF	OFFSET TO DISTRIBUTED IDENTITY INFORMATION (DIDLIST1)
90	(5A)	CHARACTER		RPEEND	END OF FIXED PART OF ELEMENT

RPEDID1

0	(0)	STRUCTURE	*	RPEDID1	DIDLIST1 REPEAT GROUP, PART 1
0	(0)	UNSIGNED	1	RPEDIDLL	LABEL LENGTH
1	(1)	CHARACTER	*	RPEDIDLB	LABEL

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
RPEDID2					
0	(0)	STRUCTURE	*	RPEDID2	DIDLIST1 REPEAT GROUP, PART 2
0	(0)	CHARACTER	8	RPEDIDUS	USER ID
8	(8)	UNSIGNED	1	RPEDIDRL	REGISTRY LENGTH
9	(9)	CHARACTER	*	RPEDIDRG	REGISTRY NAME
RPEINST					
0	(0)	STRUCTURE	*	RPEINST	INSTALLATION DATA VARIABLE LENGTH PORTION
0	(0)	UNSIGNED	1	RPEINSTL	INSTALLATION DATA LENGTH
1	(1)	CHARACTER	*	RPEINSTD	INSTALLATION DATA STRING
RPEIPLOK					
0	(0)	CHARACTER	16	RPEIPLOK	IP LOOKUP VALUE FOR SERVAUTH CLASS PROFILES
RPEAPPL					
0	(0)	STRUCTURE	*	RPEAPPL	APPLICATION DATA VARIABLE LENGTH PORTION
0	(0)	UNSIGNED	1	RPEAPPLL	APPLICATION DATA LENGTH
1	(1)	CHARACTER	*	RPEAPPLD	APPLICATION DATA STRING
RPEACCLE(*)					
0	(0)	STRUCTURE	9	RPEACCLE(*)	ACCESS LIST
0	(0)	CHARACTER	8	RPEAUSR	USERID/GROUPNAME
8	(8)	BITSTRING	1	RPEAUSR	ACCESS AUTHORITY
RPEDPTD (*)					
0	(0)	STRUCTURE	2	RPEDPTD (*)	CATEGORY LIST
0	(0)	SIGNED	2	RPEDEPT	CATEGORY
RPESESSN					
0	(0)	STRUCTURE	*	RPESESSN	SESSION DATA
0	(0)	CHARACTER	10	RPESEFIX	FIXED LEN SESSION FIELDS
0	(0)	CHARACTER	4	RPEKYDAT	DATE KEY WAS LAST CHANGED
4	(4)	SIGNED	2	RPEKYINT	# DAYS UNTIL KEY EXPIRES
6	(6)	SIGNED	2	RPEMFAIL	MAX # OF FAILED ATTEMPTS
8	(8)	BITSTRING	1	RPESLSFG	SESSION FLAGS
9	(9)	UNSIGNED	1	RPESKYLN	LENGTH OF SESSION KEY
10	(A)	CHARACTER	*	RPESEVAR	VARIABLE LEN FIELDS
10	(A)	CHARACTER	*	RPESNKEY	SESSION KEY
RPEACL2					

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	RPEACL2	SECOND ACCESS LIST
0	(0)	CHARACTER	20	RPEA2FIX	FIXED LENGTH PORTION OF SECOND ACCESS LIST
0	(0)	CHARACTER	8	RPEA2PGM	PROGRAM NAME OR FLAGS
0	(0)	CHARACTER	1	RPEPGFLG	FLAG BYTE
1	(1)	CHARACTER	7	RPEA2RST	THE REST OF NAME OR FLAGS
8	(8)	CHARACTER	8	RPEA2USR	USERID
16	(10)	BITSTRING	1	RPEA2ACA	ACCESS AUTHORITY
17	(11)	UNSIGNED	2	RPEA2CNT	ACCESS COUNT FIELD
19	(13)	UNSIGNED	1	RPEA2VRL	VARIABLE AREA LENGTH
20	(14)	CHARACTER	*	RPEA2VAR	VARIABLE AREA
20	(14)	CHARACTER	8	RPEA2CLI	CLASS ID.
28	(1C)	CHARACTER	2	RPEA2RSV	RESERVED.
30	(1E)	UNSIGNED	1	RPEA2ELN	ENTITY LENGTH
31	(1F)	CHARACTER	*	RPEA2ENT	ENTITY
RPEICSF					
0	(0)	CHARACTER	9	RPEICSF	ICSF segment data
0	(0)	UNSIGNED	1	RPEICEXP	Symmetric key export option
1	(1)	UNSIGNED	4	RPEICAUS	Asymmetric key usage options
5	(5)	UNSIGNED	2	RPEICOFF	Offset from RPEICSF to start of certificate label information at RPECLABS
7	(7)	UNSIGNED	2	RPEIKLCT	PKDS label count
9	(9)	CHARACTER	0	RPEIKLBS	Start of PKDS length/label pairs, mapped by RPEILABS
RPEILABS					
0	(0)	CHARACTER	*	RPEILABS	Mapping for both PKDS and certificate length/label pairs
0	(0)	UNSIGNED	1	RPEILLN	Label length
1	(1)	CHARACTER	*	RPEILABE	PKDS or certificate label
RPECLABS					
0	(0)	CHARACTER	*	RPECLABS	Certificate label information. This data starts immediately after the final PKDS label.
0	(0)	UNSIGNED	2	RPEICLCT	Certificate label count
1	(1)	CHARACTER	*	RPEICLBS	Start of certificate length/label pairs, mapped by RPEILABS
RPEMEM					

Table 89. Structure RACRTE (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	RPEMEM	MEMBER LIST
0	(0)	UNSIGNED	1	RPEMEML	MEMBER LENGTH
1	(1)	CHARACTER	*	RPEMEMBR	MEMBER
RPESESS2					
0	(0)	STRUCTURE	1	RPESESS2	MORE SESSION
0	(0)	CHARACTER	1	RPESE2FX	MORE SESSION FIXED FIELDS
0	(0)	BITSTRING	1	RPESCONV	CONVERSATION SECURITY

ISP constants

Table 90. Constants for ISP

Len	Type	Value	Name	Description
1	DECIMAL	0	RPEA2DAT	FLAG DATA EQUATE

ISP cross reference

Table 91. Cross Reference for ISP

Name	Offset	Hex Value
RACRNE	0	
RACRPE	0	
RACRSE	0	
RACRTE	0	
RNEBAL	C	
RNEKEY	10	
RNELEFT	0	
RNEPROF	4	
RNERIGHT	8	
RNEUP	C	
RPEACCLE	0	
RPEACCNO	A	
RPEACCOF	C	
RPEACL2	0	
RPEACS	8	
RPEAC2LN	40	
RPEAC2NO	3E	
RPEAC2OF	42	
RPEAPPL	0	

Table 91. Cross Reference for ISP (continued)

Name	Offset	Hex Value
RPEAPPLD	1	
RPEAPPLL	0	
RPEAPPOF	10	
RPEATTR	6	
RPEAUDIT	7	
RPEAUSR	0	
RPEA2ACA	10	
RPEA2CLI	14	
RPEA2CNT	11	
RPEA2ELN	1E	
RPEA2ENT	1F	
RPEA2FIX	0	
RPEA2PGM	0	
RPEA2RST	1	
RPEA2RSV	1C	
RPEA2USR	8	
RPEA2VAR	14	
RPEA2VRL	13	
RPECLABS	0	
RPECSFLN	50	
RPECSFOF	52	
RPEDEPT	0	
RPEDID1	0	
RPEDID2	0	
RPEDIDCT	54	
RPEDIDLB	1	
RPEDIDLL	0	
RPEDIDLN	56	
RPEDIDOF	58	
RPEDIDRL	8	
RPEDIDRG	9	
RPEDIDUS	0	
RPEDPTD	0	
RPEDPTOF	1C	
RPEEND	4E	
RPEFLAGS	31	

Table 91. Cross Reference for ISP (continued)

Name	Offset	Hex Value
RPEFSCPR	31	20
RPEFSCPW	31	40
RPEFWARN	31	80
RPEGAUD	8	
RPEICAUS	1	
RPEICEXP	0	
RPEICLBS	1	
RPEICLCT	0	
RPEICOFF	5	
RPEIKLBS	9	
RPEIKLCT	7	
RPEILLN	0	
RPEILABE	0	
RPEILABS	0	
RPEICSF	0	
RPEINSOF	E	
RPEINST	0	
RPEINSTD	1	
RPEINSTL	0	
RPEKYDAT	0	
RPEKYINT	4	
RPELDAYS	1E	
RPELEVEL	9	
RPELLEN	2	
RPELOGFT	23	
RPELOGNT	20	
RPEMEM	0	
RPEMEMBR	1	
RPEMEMCT	44	
RPEMEML	0	
RPEMEMLN	46	
RPEMEMOF	48	
RPEMFAIL	6	
RPENTFY	26	
RPENUMDP	1A	
RPEOWNER	12	

Table 91. Cross Reference for ISP (continued)

Name	Offset	Hex Value
RPEPGFLG	0	
RPEPLEN	0	
RPESCLBL	32	
RPESCLVL	1F	
RPESCONV	0	
RPESEFIX	0	
RPESESLN	3C	
RPESESOF	3A	
RPESESSN	0	
RPESESS2	0	
RPESEVAR	A	
RPESE2FX	0	
RPESE2LN	4A	
RPESE2OF	4C	
RPESKYLN	9	
RPESLSFG	8	
RPESNKEY	A	
RPETZONE	2E	
RPEUACC	6	
RPEUCNT	4	
RSENEXT	0	
RSEPOOL	6	
RSESIZE	4	
RSESTORE	8	
RTEALET	28	
RTEASPN	12	
RTECLASS	4	
RTEGENL	14	
RTEGNUM	1C	
RTENEXT	0	
RTENSPN	11	
RTEPSPN	10	
RTESIZE	18	
RTESPNS	10	
RTESTOKN	20	
RTESTORE	C	

Table 91. Cross Reference for ISP (continued)

Name	Offset	Hex Value
RTETREE	8	

Chapter 38. OUSP: initUSP Output Parameter List

OUSP programming interface information

OUSP is a programming interface.

OUSP heading information

Common name: Output parameter list for initUSP callable service

Macro ID: IRRPOUSP

DSECT name: OUSP

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Identified in the output parameter list

Key
0

Residency
Invoker's primary address space

Size: 2074 bytes

Created by: initUSP callable service (IRRSIU00)

Pointed to by: Address of OUSP is put into the IUSP by the initUSP callable service

Serialization: N/A

Function: Maps the output information returned by the initUSP service routine

OUSP mapping

Table 92. Structure OUSP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	2074	OUSP	
0	(0)	UNSIGNED	1	OUSP_VRSN	Version Number
1	(1)	UNSIGNED	1	OUSP_WASP	Work area subpool number
2	(2)	SIGNED	2	OUSP_LEN	OUSP_FIXED + HDPNLEN + IPPNLEN
4	(4)	SIGNED	4	OUSP_UID	UID of the user
8	(8)	SIGNED	4	OUSP_GID	GID of the current group
12	(C)	SIGNED	2	OUSP_TSOULEN	Length of OUSP_TSOU
14	(E)	CHARACTER	8	OUSP_TSOU	TSO userid
22	(16)	SIGNED	2	OUSP_HDPNLEN	Length of home directory path name

Table 92. Structure OUSP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
24	(18)	SIGNED	2	OUSP_IPPNLEN	Length of initial program path name
26	(1A)	CHARACTER	*	OUSP_HDPN	Home directory path name
*	*	CHARACTER	*		Initial program path name
OUSP2					
0	(0)	STRUCTURE	*	OUSP2	Extension.
0	(0)	UNSIGNED	2	OUSP_FLAGS	Flags (if OUSP_VRSN is 1).
		1...		OUSP_LIMITSSET	Limits are defined for this user.
		.1..		OUSP_LIMITSDONOTFIT	If on, there is no room for the user limits in the OUSP.
OUSP3					
0	(0)	STRUCTURE	*	OUSP3	Limits area.
0	(0)	SIGNED	2	OUSP_NUMLIMITS	Number of limits in following array
OUSP4					
0	(0)	STRUCTURE	4	OUSP4	32 bit limit value.
0	(0)	SIGNED	4	OUSP_LIMIT32VAL	4 byte limit value.
OUSP5					
0	(0)	STRUCTURE	4	OUSP5	64 bit limit value.
0	(0)	SIGNED	4	OUSP_LIMIT64VAL	3 byte limit value.
0	(0)	CHARACTER	1	OUSP_LIMIT64UNIT	Unit value.
OUSP_LIMIT_ENTRY					
0	(0)	STRUCTURE	5	OUSP_LIMIT_ENTRY	Array entry.
0	(0)	UNSIGNED	1	OUSP_LIMITKEY	Key defining type of limit.
1	(1)	UNSIGNED	3	OUSP_LIMITVALUE	Value of limit.
4	(4)	CHARACTER	1	OUSP_UNITS	Units M, G, T, P (megabytes, gigabytes, terabytes, or petabytes).

OUSP constants

Table 93. Constants for OUSP

Len	Type	Value	Name	Description
Initial program path name.				
1	DECIMAL	1	OUSPVNC	Version Number.
1	DECIMAL	1	OUSPVNCLIMIT	First version with limit support.
1	DECIMAL	1	OUSP_CPUTIMEMAX	Key for CPUTIMEMAX.
1	DECIMAL	7	OUSP_MEMLIMIT	Key for MEMLIMIT

Table 93. Constants for OUSP (continued)

Len	Type	Value	Name	Description
1	DECIMAL	8	OUSP_SHMEMMAX	Key for SHMEMMAX
1	DECIMAL	2	OUSP_ASSIZEMAX	Key for ASSIZEMAX.
1	DECIMAL	3	OUSP_FILEPROCMAX	Key for FILEPROCMAX.
1	DECIMAL	4	OUSP_PROCUSERMAX	Key for PROCUSERMAX.
1	DECIMAL	5	OUSP_THREADSMAX	Key for THREADSMAX.
1	DECIMAL	6	OUSP_MMAPAREAMAX	Key for MMAPAREAMAX.
4	DECIMAL	26	OUSPFIXED	Length of the fixed portion of the OUSP.
4	DECIMAL	2074	OUSP_FREE_LEN	Length of the OUSP to be freed.

OUSP cross reference

Table 94. Cross Reference for OUSP

Name	Offset	Hex Value
OUSP	0	
OUSP2	0	
OUSP3	0	
OUSP4	0	
OUSP5	0	
OUSP_FLAGS	0	
OUSP_GID	8	
OUSP_HDPN	1A	
OUSP_HDPNLEN	16	
OUSP_IPPN	*	
OUSP_IPPNLEN	18	
OUSP_LEN	2	
OUSP_LIMIT64UNIT	0	
OUSP_LIMIT32VAL	0	
OUSP_LIMIT64VAL	0	
OUSP_LIMITKEY	0	
OUSP_LIMIT_ENTRY	0	
OUSP_LIMITSDONOTFIT	0	40
OUSP_LIMITSSET	0	80
OUSP_LIMITVALUE	1	
OUSP_NUMLIMITS	0	
OUSP_TSOU	E	
OUSP_TSOULEN	C	

Table 94. Cross Reference for OUSP (continued)

Name	Offset	Hex Value
OUSP_UID	4	
OUSP_UNITS	4	
OUSP_VRSN	0	
OUSP_WASP	1	

Chapter 39. PWX2: New Password Phrase Exit Parameter List

PWX2 programming interface information

PWX2 is a programming interface.

PWX2 heading information

Common name: New password phrase exit parameter list

Macro ID: ICHPWX2

DSECT name: PWX2L

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes: If called by the ADDUSER, ALTUSER, or PASSWORD command:

Subpool
0

Key
Problem program in the user's address space or 2 in the RACF subsystem address space.

If called by the RACROUTE REQUEST=VERIFY SVC processor:

Subpool
229

Key
0

Size: 44 bytes

Created by: Commands: ADDUSER, ALTUSER, PASSWORD RACROUTE REQUEST=VERIFY SVC processor

Pointed to by: R1 at entry to ICHPWX11

Serialization: None

Function: Contains the list of addresses passed to the new password phrase installation exit

PWX2 mapping

Table 95. Structure PWX2PL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		PWX2PL	
0	(0)	ADDRESS	4	PWX2LEN	Length address: points to a fullword containing the number of fullwords in this parameter list.

Table 95. Structure PWX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	PWX2CLLR	Caller address: points to a 1-byte field containing the calling function identity:
		1		PWX2RINI	"X'01'" RACROUTE REQUEST=VERIFY (RACINIT) Note: If the caller is RACINIT, the ACEE control block might not be present.
		.... ..1.		PWX2PWRD	"X'02'" PASSWORD Command
		11		PWX2ALTU	"X'03'" ALTUSER Command
		.... .1..		PWX2ADDU	"X'04'" ADDUSER Command
8	(8)	ADDRESS	4	PWX2CPPL	CPPL address: points to the TSO command processor parameter list. This applies only to the PASSWORD, ADDUSER, and ALTUSER commands. If the TSO command processor parameter list is absent, the address is zero.
12	(C)	ADDRESS	4	PWX2NEW	New Pass Phrase Address: points to an area of the following format: Offset 0, length 1: Length of new pass phrase. Offset 1, 100 byte field containing the new pass phrase.
16	(10)	ADDRESS	4	PWX2USER	Userid address: points to an area of the following format: Offset 0, length 1: Length of userid. Offset 1, length 8: Userid.
20	(14)	ADDRESS	4	PWX2WA	Exit work area address: points to a fullword whose contents are either: • Zero, for ADDUSER, ALTUSER and PASSWORD commands • The contents of the user work area address that RACINIT processing passes to ICHRIX01 and ICHRIX02.
24	(18)	ADDRESS	4	PWX2CURP	Current pass phrase address: points to an area of the following format: Offset 0, length 1: Length of current pass phrase Offset 1, variable length: current pass phrase.

Table 95. Structure PWX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	PWX2LCDA	Password phrase last change date address: points to a 4-byte area that contains the date of the last pass phrase change .The format of this area is: yyyyddds where: yyyy is the year, ddd is the day, and s is the packed decimal sign
32	(20)	ADDRESS	4	PWX2ACEE	ACEE address: points to the ACEE used. This address may not be available if the caller is RACINIT.
36	(24)	ADDRESS	4	PWX2GRP	Group name address: points to a 9-byte structure containing a 1-byte length field, followed by an 8-byte field containing the connect group name.
40	(28)	ADDRESS	4	PWX2INST	Installation data address: points to an area containing the installation parameters. This address is only available when the caller is RACINIT and the INSTLN parameter was specified.

PWX2 cross reference

Table 96. Cross Reference for PWX2

Name	Offset	Hex Value
PWX2		0
PWX2LEN		0
PWX2CLLR		4
PWX2CPPL		8
PWX2NEW		C
PWX2USER		10
PWX2WA		14
PWX2CURP		18
PWX2LCDA		1C
PWX2ACEE		20
PWX2GRP		24
PWX2INST		28

Chapter 40. PWXP: Password Exit Parameter List

PWXP programming interface information

PWXP is a programming interface.

PWXP heading information

Common name: Password Exit Parameter List

Macro ID: ICHPWXP

DSECT name: PWXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes: If called by the ALTUSER or PASSWORD command:

Subpool
0

Key
Problem program

If called by the RACROUTE REQUEST=VERIFY SVC processor:

Subpool
229

Key
0

Size: 56 bytes

Created by: Commands: ALTUSER, PASSWORD RACROUTE REQUEST=VERIFY SVC processor

Pointed to by: R1 at entry to ICHPWX01

Serialization: None

Function: Contains the list of addresses passed to the password installation exit

PWXP mapping

Table 97. Structure PWXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		PWXPL	
0	(0)	ADDRESS	4	PWXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	PWXCALLR	Caller address: points to a 1-byte field containing the calling function identity:
		1		PWXRINIT	X'01' RACINIT

Table 97. Structure PWXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... ..1.		PWXPWORD	X'02' PASSWORD Command
		11		PWXALTUS	X'03' ALTUSER Command Note: If the caller is RACINIT, the ACEE control block might not be present.
8	(8)	ADDRESS	4	PWXCPPL	CPPL address: points to the TSO command processor parameter list. This applies only to the PASSWORD and ALTUSER commands. If the TSO command processor parameter list is absent, the address is zero.
12	(C)	ADDRESS	4	PWXNEWPW	NEWPASS address: points to an area of the following format: Offset 0, length 1: Length of new password. Offset 1, length 8: New password. If ENCRYPT=NO was specified, the password is treated as if it were already encrypted. If a new password is not specified, the address is zero.
16	(10)	ADDRESS	4	PWXINTVL	INTERVAL address: points to a 4-byte field containing the desired password interval from the PASSWORD command. If this interval is absent, the address is zero.
20	(14)	ADDRESS	4	PWXUSRID	Userid address: points to an area of the following format: Offset 0, length 1: Length of userid. Offset 1, length 8: Userid.
24	(18)	ADDRESS	4	PWXWA	Exit work area address: points to a fullword whose contents are either: - Zero, for ALTUSER and PASSWORD commands - The contents of the user work address that RACINIT processing passes to ICHRIX01 and ICHRIX02.
28	(1C)	ADDRESS	4	PWXCURPW	Current password address points to an area of the following format: Offset 0, length 1: Length of current password. Offset 1, length 8: Current password. If ENCRYPT=NO was specified, the password is treated as if it were already encrypted.

Table 97. Structure PWXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	4	PWXPLCDA	Password Last Change Date Address: points to a 3-byte area that contains the date of the last password change. The format of this area is: yyddds where: 'yy' is the year, 'ddd' is the day, and 's' is the packed decimal sign.
36	(24)	ADDRESS	4	PWXACEE	ACEE address: points to the ACEE used. This address may not be available if the caller is RACINIT.
40	(28)	ADDRESS	4	PWXGROUP	Group name address: points to a 9-byte structure containing a 1-byte length field, followed by an 8-byte field containing the connect group name.
44	(2C)	ADDRESS	4	PWXINSTL	Installation data address: points to an area containing the installation parameters. This address is only available when the caller is RACINIT and the INSTLN parameter was specified.
48	(30)	ADDRESS	4	PWXPWHST	Password history address: points to an area containing the user's password history. The passwords are in masked or encrypted format, with the oldest password first in the list. The format of the area is: a 2-byte count of the entries in the list, and for each entry a 1-byte reserved field followed by an 8-byte field containing the encrypted password. The SETROPTS PASSWORD(HISTORY(n)) option controls the number of past keywords that are kept. **Note:** This address is 0 when RCVTPALG is not 0.
52	(34)	ADDRESS	4	PWXFLAG	Flag byte address: points to a 1-byte field containing the form of the current and new passwords:
		....		PWXCTEXT	X'00' Clear text form
		1		PWXETEXT	X'01' Encrypted form (If ENCRYPT=NO is specified on RACINIT, the password is treated as if it is already encrypted.)

Table 97. Structure PWXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... ..1.		PWXPTKT	X'02' Passticket is passed in the old password field. The new password is in clear text form. This parameter is available only if the caller is RACINIT. In all cases, if a parameter is not present, it's address is zero.
56	(38)	ADDRESS	4	PWXPLCD4	Password Last Change Date Address: points to a 4-byte area that contains the date of the last password change. The format of this area is: yyyyddds where: 'yyyy' is the year, 'ddd' is the day, and 's' is the packed decimal sign.

PWSP cross reference

Table 98. Cross Reference for PWXP

Name	Offset	Hex Value
PWXACEE	24	
PWXALTUS	4	3
PWXCALLR	4	
PWXCPPL	8	
PWXCTEXT	34	0
PWXCURPW	1C	
PWXETEXT	34	1
PWXFLAG	34	
PWXGROUP	28	
PWXINSTL	2C	
PWXINTVL	10	
PWXLEN	0	
PWXNEWPW	C	
PWXPLCDA	20	
PWXPLCD4	38	
PWXPTKT	34	2
PWXPWHST	30	
PWXPWORD	4	2
PWXRINIT	4	1
PWXUSRID	14	
PWXWA	18	

Chapter 41. RCVI: Identity cache communication vector

RCVI programming interface information

RCVI is a programming interface.

RCVI heading information

Common name: RACF Identity cache communication vector

Macro ID: IRRPRCVI

DSECT name: RCVI

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: RACFRCVI (Offset 0, length 8)

Storage attributes:

Subpool
SQA

Key
0

Size: 6880 bytes

Created by: RACF

Pointed to by: RCVTRCVI

Serialization: You should do the following:

1. Ensure RCVTRCVI is not equal to zero.
2. Read the active table pointer and build count with a CDS instruction.
3. Read the specific portion of, or the entire, active table.
4. Check the build count, if it has changed, and go to step 2.

Function: Provides an instorage communication area for identity cache dynamic configuration information.

Note: RCVILREG, RCVILDPH, RCVIAPPL, and RCVIBEDDN are null terminated. The fields are defined 1 byte longer than the valid maximum valid field length, allowing a returned field to be followed by a byte of X'00' in all cases.

RCVI mapping

Table 99. Structure RCVI

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	6880	RCVI	Located through RCVT
0	(0)	CHARACTER	64	RCVIHDR	
0	(0)	CHARACTER	8	RCVIID	EBCDIC ID "RACFRCVI"

Table 99. Structure RCVI (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	CHARACTER	8	RCVIACHK	RCVI active check field
8	(8)	UNSIGNED	4	RCVIBCNT	RCVI build count
12	(C)	ADDRESS	4	RCVIACTV	PTR of active table
16	(10)	ADDRESS	4	RCVIPTR1	PTR to table 1
20	(14)	ADDRESS	4	RCVIPTR2	PTR to table 2
24	(18)	UNSIGNED	4	RCVIVERN	RCVI version number
28	(1C)	CHARACTER	36	*	Reserved
64	(40)	CHARACTER	3408	RCVISID1	Table 1
3472	(D90)	CHARACTER	3408	RCVISID2	Table 2
3472	(D90)	X'1AE0'	0	RCVIL	"*-RCVI" RCVI length

The RCVI consists of 2 blocks: RCVISID1 and RCVISID2

One block is active (pointed to by RCVIACTV), and the other not active.

RCVISAFI

0	(0)	STRUCTURE	3408	RCVISAFI	Located through RCVI
0	(0)	CHARACTER	8	RCVISEYE	SAFID EYECATCHER
8	(8)	CHARACTER	4	RCVISYSI	SYSID if from .SYSID profile
12	(C)	UNSIGNED	1	RCVIUMAP	USEMAP
13	(D)	UNSIGNED	1	RCVIDMAP	DOMAP
14	(E)	UNSIGNED	1	RCVIMRQD	MAPREQUIRED
15	(F)	UNSIGNED	1	*	alignment
16	(10)	UNSIGNED	2	RCVIMTO	MAPPINGTIMEOUT
18	(12)	UNSIGNED	2		alignment
20	(14)	UNSIGNED	4	RCVILRLN	LOCALREGISTRY LENGTH
24	(18)	CHARACTER	256	RCVILREG	LOCALREGISTRY
280	(118)	UNSIGNED	4	RCVILDLN	LDAPHOST LENGTH
284	(11C)	CHARACTER	1025	RCVILDPH	LDAPHOST
1309	(51D)	UNSIGNED	3		alignment
1312	(520)	UNSIGNED	4	RCVIAPLN	APPLDATA LENGTH
1316	(524)	UNSIGNED	1025	RCVIAPPL	APPLDATA
2341	(925)	UNSIGNED	3	*	alignment
2344	(928)	UNSIGNED	4	RCVIBDLN	BINDDN LENGTH
2348	(92C)	CHARACTER	1025	RCVIBDDN	BINDDN
3373	(D2D)	UNSIGNED	3	*	alignment
3376	(D30)	ADDRESS	4	(8)	reserved

RCVI constants

Table 100. Constants for RCVI

Len	Type	Value	Name	Description
4	DECIMAL	1	RCVIV1	Version number
8	CHARACTER	RACFRCVI	RCVIIDC	EBCDIC RCVI ID for the RCVI control block
8	CHARACTER	RCVISID1	RCVIIDC1	EBCDIC RCVI ID for the RCVISID1 control block
8	CHARACTER	RCVISID2	RCVIIDC2	EBCDIC RCVI ID for the RCVISID2 control block

RCVI cross reference

Table 101. Cross Reference for RCVI

Name	Offset	Hex Value
RCVI	0	
RCVIHDR	0	
RCVIID	0	
RCVIACHK	8	
RCVIBCNT	8	
RCVIACTV	C	
RCVIPTR1	10	
RCVIPTR2	14	
RCVIVERN	18	
RCVISID1	40	
RCVISID2	D90	
RCVISAFI	0	
RCVISEYE	0	
RCVISYSI	8	
RCVIUMAP	C	
RCVIDMAP	D	
RCVIMRQD	E	
RCVIMTO	10	
RCVILRLN	14	
RCVILREG	18	
RCVILDLN	118	
RCVILDPH	11C	
RCVIAPLN	520	
RCVIAPPL	524	
RCVIBDLN	928	

Table 101. Cross Reference for RCVI (continued)

Name	Offset	Hex Value
RCVIBDDN	92C	

Chapter 42. RCVT: RACF Communication Vector Table

RCVT programming interface information

RCVT is **NOT a programming interface.** The following fields are the only intended Programming Interfaces in RCVT:

- RCVT
- RCVTAPTR
- RCVTCDTL
- RCVTDATP
- RCVTDNL
- RCVTENVP
- RCVTFLGS
- RCVTFLG1
- RCVTFLG3
- RCVTFLG4
- RCVTFRCP
- RCVTGENT
- RCVTGLBL
- RCVTID
- RCVTIDPV
- RCVTINAC
- RCVTISTL
- RCVTJALL
- RCVTJCHK
- RCVTJSYS
- RCVTJUND
- RCVTJXAL
- RCVTLNOD
- RCVTMFLG
- RCVTMFL1
- RCVTML2F
- RCVTPALG
- RCVTPINV
- RCVTPNLO
- RCVTPTGN
- RCVTRCVI
- RCVTRELS
- RCVTREXP
- RCVTRL
- RCVTRNA
- RCVTROFF

- RCVTRVOK
- RCVTSTAT
- RCVTSTA1
- RCVTTAPE
- RCVTTDSN
- RCVTVERS
- RCVTVRN
- RCVTVRMN
- RCVTWARN
- RCVTWUID

Application Programmers:

The RCVT fields listed above are Programming Interfaces for input only, with the following exceptions:

- RCVTISTL and RCVTAPTR can be both input and output
- RCVTREXP and RCVTFRCP are not part of the application programming interface.

Notes:

1. The 118th bit of the RCVTVCPR field is a programming interface for input only. It can be used to quickly check if the SECLABEL class is active. If the bit is on, the class is active.
2. For external security managers (ESMs) such as RACF or ESMs that are functionally compatible with RACF: The RCVT fields listed above are Programming Interfaces for both input and output. The ESM is responsible for creating the RCVT, attaching it to the communication vector table (CVT), and putting appropriate data into these fields in order to be compatible with RACF and the way that IBM products use the RCVT.

RCVT heading information

Common name:	RACF communication vector table
Macro ID:	ICHPRCVT
DSECT name:	RCVT
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	RCVT (Offset: 0, Length: 4)
Storage attributes:	**Subpool** SQA **Key** 0
Size:	2308 bytes
Created by:	RACF initialization or equivalent
Pointed to by:	CVTRAC
Serialization:	None
Function:	Communication area for information global to RACF functions (or equivalent product functions)

RCVT mapping

Table 102. Structure RCVT

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	2038	RCVT	LOCATED THROUGH CVT
0	(0)	CHARACTER	4	RCVTID	EBCDIC ID
4	(4)	ADDRESS	4	RCVTDCB	PTR DCB OF RACF DATA SET
8	(8)	ADDRESS	4	RCVTDEB	PTR DEB OF RACF DATA SET
12	(C)	ADDRESS	4	RCVTINDX	PTR RACF RESIDENT INDEX TABLE OR ZERO IF NO INDEX BLOCKS RESIDENT
16	(10)	ADDRESS	4	RCVTTEMP	PTR RACF INCORE TEMPLATE TABLE
20	(14)	ADDRESS	4	RCVTHDR	PTR RACF INCORE DS HEADER RECORD OR ZERO IF RACF DATA SET IS ON A SHARED DEVICE
24	(18)	ADDRESS	4	RCVTRIX	PTR RACROUTE REQUEST=VERIFY(X) EXIT
28	(1C)	ADDRESS	4	RCVTRCX	PTR RACROUTE REQUEST=AUTH EXIT
32	(20)	ADDRESS	4	RCVTRDX	PTR RACROUTE REQUEST=DEFINE EXIT
36	(24)	ADDRESS	4	RCVTRUCB	PTR UCB OF RACF DATA SET
40	(28)	SIGNED	4	RCVTXLEN	LENGTH OF INCORE INDEX RELATED CONTROL BLOCKS
44	(2C)	ADDRESS	4	RCVTBAM	LOCATES INCORE BAM INFORMATION
48	(30)	ADDRESS	4	RCVTISTL	RESERVED FOR INSTALLATION
52	(34)	ADDRESS	1	RCVTDSNL	LENGTH OF RACF DATA SET NAME
53	(35)	BITSTRING	1	RCVTSTAT	STATUS
		1...		RCVTRNA	RACF NOT ACTIVE
		.1..		RCVTNLS	BYPASS RACINIT STATISTICS
		..1.		RCVTNDSS	BYPASS DATA SET STATISTICS
		...1		RCVTNTVS	NO TAPE VOLUME STATISTICS
		.... 1...		RCVTNDVS	NO DIRECT ACCESS VOLUME STATISTICS
		.... .1..		RCVTNTMS	NO TERMINAL STATISTICS
		.... ..1.		RCVTNADS	NO ADSP PROTECTION
		1		RCVTEGN	EGN SUPPORT IN EFFECT
54	(36)	SIGNED	2	RCVTNREC	# RECORDS PER TRACK -RACF DS
56	(38)	CHARACTER	44	RCVTDSN	DSN OF RACF DATA SET
100	(64)	CHARACTER	44	RCVTUADS	DSN OF UADS DATA SET OR ZERO

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
144	(90)	CHARACTER	6	RCVTUVOL	VOLID OF UADS DATA SET OR ZERO
150	(96)	BITSTRING	1	RCVTSTA1	STATUS
		1...		RCVTTAPE	TAPE VOLUME PROTECTION IN EFFECT
		.1..		RCVTDASD	DASD VOLME PROTECTION IN EFFECT
		..1.		RCVTDGEN	GENERIC PROFILE CHECKING FOR DATASET CLASS IN EFFECT
		...1		RCVTDGCM	GENERIC COMMAND PROCESSING FOR DATASET CLASS IN EFFECT
		.... 1...		RCVTRDSN	INPUT DATA SET NAME WILL BE USED FOR LOGGING AND MESSAGES
		.... .1..		RCVTJXAL	JES-XBMALLRACF IS IN EFFECT
		.... ..1.		RCVTJCHK	JES-EARLYVERIFY IS IN EFFECT
		1		RCVTJALL	JES-BATCHALLRACF IS IN EFFECT
151	(97)	BITSTRING	1	RCVTAUOP	AUDIT OPTIONS
		1...		*	RESERVED
		.1..		RCVTAGRO	AUDIT GROUP CLASS
		..1.		RCVTAUSE	AUDIT USER CLASS
		...1		RCVTADAT	AUDIT DATASET CLASS
		.... 1...		RCVTADAS	AUDIT DASDVOL CLASS
		.... .1..		RCVTATAP	AUDIT TAPEVOL CLASS
		.... ..1.		RCVTATER	AUDIT TERMINAL CLASS
		1		RCVTAOPR	AUDIT OPERATIONS ATTRIBUTE
152	(98)	BITSTRING	1	RCVTAXTA	RESERVED
153	(99)	BITSTRING	1	RCVTFLGS	STATUS FLAGS
		1...		RCVTROFF	RACF HAS BEEN DEACTIVATED BY THE RVARY COMMAND
		.1..		RCVTRDHD	RACF HAS BEEN RE-ACTIVATED BY RVARY AND REFRESH OF THE RESIDENT ICB IS NECESSARY
		..1.		RCVTSHR	THE RACF DATA SET AT SOME POINT DURING THIS IPL, WAS ON A SHARED DASD DEVICE
		...1		RCVTNDUP	NO DUPLICATE DATA SET NAMES TO BE DEFINED
		.... 1...		RCVT24MD	AT LEAST ONE INSTALLATION EXIT HAS AMODE=24

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... .1..		RCVTRMSG	RACF MESSAGE ICH412I WAS ISSUED
		.... ..1.		RCVTWUID	RACF WORK UNIT IDENTITY SUPPORT EXISTS
		1		RCVTGLBL	GLOBAL=YES SUPPORT EXISTS
154	(9A)	BITSTRING	1	RCVTEROP	TERMINAL VERIFICATION OPTIONS
		1...		RCVTTERP	TERMINAL AUTHORIZATION CHECKING
		.1..		RCVTTUAC	DEFAULT UACC FOR TERMINALS NOT DEFINED TO RACF IF ON - UACC = NONE IF OFF- UACC = READ
		..1.		RCVTAVIO	DO NOT CREATE LOG RECORD FOR COMMAND VIOLATIONS ONLY
		...1		RCVTSAUD	DO NOT AUDIT SPECIAL USER
					RESERVED
155	(9B)	ADDRESS	1	RCVTPINV	GLOBAL MAX PASSWORD INTERNAL VALUE VALID RANGE 1 - 254. This field is also used as the system password phrase interval unless the RCVTPHIN field has a non-zero value.
156	(9C)	ADDRESS	4	RCVTRAU0	PTR TO AUDITING MODULE
160	(A0)	ADDRESS	4	RCVTRIXP	PTR TO RACINIT POST PROCESSING INSTALLATION EXIT RTN
164	(A4)	ADDRESS	4	RCVTRCXP	PTR TO RACCHK POST PROCESSING INSTALLATION EXIT RTN
168	(A8)	ADDRESS	4	RCVTRID0	PTR TO MSC VERIFY RTN

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
172	(AC)	BITSTRING	1	RCVTVERS	VERSION INDICATOR HIGH NIBBLE IS THE VERSION NUMBER, (0=VERSION 1), AND THE LOW NIBBLE IS THE RELEASE NUMBER 0 - VERSION 1 RELEASE 1 1 - VERSION 1 RELEASE 2 2 - VERSION 1 RELEASE 3 4 - VERSION 1 RELEASE 4 5 - VERSION 1 RELEASE 5 6 - VERSION 1 RELEASE 6 7 - VERSION 1 RELEASE 7 8 - VERSION 1 RELEASE 8
		1111		RCVTVRN	VERSION NUMBER IN HIGH NIBBLE
		 1111		RCVTRELS	RELEASE NUMBER IN LOW NIBBLE
173	(AD)	CHARACTER	3	RCVTEXTA	RESERVED
176	(B0)	ADDRESS	4	RCVTAPTR	ADDRESS FIELD RESERVED FOR APPLICATION USE
180	(B4)	ADDRESS	4	RCVTNCX	PTR NAMING CONVENTION EXIT
184	(B8)	ADDRESS	4	RCVTNCDX	PTR NAMING CONVENTION EXIT FOR DELETE FUNCTION
188	(BC)	ADDRESS	4	RCVTCDTP	PTR TO CLASS DESC TABLE
192	(C0)	ADDRESS	4	RCVTREXP	PTR TO RACSTAT MODULE
196	(C4)	ADDRESS	4	RCVTFRCP	PTR TO FRACHECK MODULE
200	(C8)	ADDRESS	4	RCVTFRXP	PTR RACROUTE REQUEST=FASTAUTH EXIT
204	(CC)	ADDRESS	4	RCVTRLX	PTR RACROUTE REQUEST=LIST
208	(D0)	ADDRESS	4	RCVTRLXP	PTR RACROUTE REQUEST=LIST SELECTION EXIT
212	(D4)	ADDRESS	4	RCVTDATP	PTR TO FOUR-BYTE DATE CONVERSION ROUTINE
216	(D8)	ADDRESS	4	RCVTENVP	PTR TO ENVIRONMENT SERVICE ROUTINE (IRRENS00)
220	(DC)	ADDRESS	4	RCVTEIMR	PTR TO EIM REGISTRY NAME
224	(E0)	ADDRESS	4	RCVTDSDT	PTR TO DS DESCIPTOR TAB
228	(E4)	ADDRESS	4	RCVTRNGP	PTR TO RANGE TABLE
232	(E8)	ADDRESS	4	RCVTAUTP	PTR TO RACF AUTHORIZED CALLER TABLE ICHAUTAB
236	(EC)	ADDRESS	4	RCVTPWDX	PTR TO RACF PASSWORD EXIT.

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
240	(F0)	UNSIGNED	1	RCVTHIST	NUMBER OF PASSWORD GENERATIONS TO MAINTAIN AND CHECK AGAINST.
241	(F1)	UNSIGNED	1	RCVTRVOK	NUMBER OF CONSECUTIVE UNSUCCESSFUL ATTEMPTS BEFORE REVOKING A USERID.
242	(F2)	UNSIGNED	1	RCVTWARN	NUMBER OF DAYS REMAINING IN A PASSWORD INTERVAL AFTER WHICH WARNING MESSAGES WILL BE ISSUED.
243	(F3)	UNSIGNED	1	RCVTINAC	INACTIVATE INTERVAL.
244	(F4)	CHARACTER	10	RCVTSNTX(8)	PASSWORD SYNTAX RULES.
244	(F4)	UNSIGNED	1	RCVTSLEN	STARTING LENGTH VALUE.
245	(F5)	CHARACTER	1	RCVTELEN	ENDING LENGTH VALUE.
246	(F6)	CHARACTER	8	RCVTRUL1	CONTENT RULE.
247	(F7)	CHARACTER	1	RCVTRUL2	CONTENT RULE.
248	(F8)	CHARACTER	1	RCVTRUL3	CONTENT RULE.
249	(F9)	CHARACTER	1	RCVTRUL4	CONTENT RULE.
250	(FA)	CHARACTER	1	RCVTRUL5	CONTENT RULE.
251	(FB)	CHARACTER	1	RCVTRUL6	CONTENT RULE.
252	(FC)	CHARACTER	1	RCVTRUL7	CONTENT RULE.
253	(FD)	CHARACTER	1	RCVTRUL8	CONTENT RULE.
324	(144)	BITSTRING	4	RCVTMDEL	MODEL OPTIONS.
		1...		RCVTMGDG	MODEL-GDG IN EFFECT.
		.1..		RCVTMUSR	MODEL-USER IN EFFECT.
		..1.		RCVTMGRP	MODEL-GROUP IN EFFECT.
		...1 1111		*	RESERVED
325	(145)	BITSTRING	1	*	RESERVED
326	(146)	BITSTRING	1	*	RESERVED
327	(147)	BITSTRING	1	*	RESERVED
328	(148)	BITSTRING	1	RCVTWCNT	NUMBER OF VSL ENTRIES
329	(149)	BITSTRING	1	RCVTOPTX	OPTIONS.
		1...		RCVTLGRP	LIST OF GROUP CHKING ACTIVE.
		.111 1111		*	RESERVED
330	(14A)	BITSTRING	1	RCVTALIS	IDMAP2 enablement byte X Sec Model Ident Interoperability
		....		RCVTSTG0	0=stage0, not active, only alias mapping profiles

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1		RCVTSTG1	1=stage1, active, maintain alias index, but also maintain and USE the mapping profiles
		.... ..1.		RCVTSTG2	2=stage2, active, maintain and USE alias index, but still maintain mapping profiles
		11		RCVTSTG3	3=stage3, active, maintain and USE alias index (mapping profiles have been removed)
331	(14B)	CHARACTER	1	*	RESERVED.
332	(14C)	ADDRESS	4	RCVTFRX3	RACROUTE REQUEST=FASTAUTH EXIT ADDR
336	(150)	CHARACTER	8	RCVTVSL(4)	VSL ENTRIES
368	(170)	SIGNED	4	RCVTCGSN	NUMBER OF CONNECT-REMOVE COMMANDS ISSUED THAT ALTERED A USERS AUTHORITY.
372	(174)	UNSIGNED	2	RCVTDNL	Maximum length of distributed user ID - 246 UTF8 characters for 7760.
374	(176)	UNSIGNED	2	RCVTRL	Maximum length of registry name - 255 UTF8 characters for 7760.
376	(178)	UNSIGNED	1	RCVTIDPV	A value of 1 indicates that Identity Propagation 2 services are available on the system.
377	(179)	CHARACTER	3	*	Reserved.
380	(17C)	ADDRESS	4	RCVTRDXP	PTR RACROUTE REQUEST=DEFINE POST PROCESSING EXIT - ICHRDX02
384	(180)	ADDRESS	4	RCVTFPB	BASE FOR FASTPATH TABLE.
388	(184)	ADDRESS	4	RCVTRCVI	Address of RCVI
392	(188)	BITSTRING	4	RCVTFLG1	MISCELLANEOUS OPTIONS.
		1...		RCVTFPDS	FASTPATH FOR DATASET CLASS.
		.1..		RCVTTDSN	TAPE DATA SET PROTECTION IN EFFECT
		..11 1111		*	RESERVED
393	(189)	1...		RCVTPRO	PROTECT-ALL IS IN EFFECT
		.1..		RCVTPROF	1- PROTECT-ALL WARNING IN EFFECT 0 - PROTECT-ALL FAILURE IN EFFECT (THIS FLAG IS IGNORED IF RCVTPRO HAS A VALUE OF '0'B)
		..1.		RCVTEOS	ERASE-ON-SCRATCH IN EFFECT

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		...1		RCVTEOSL	ERASE-ON-SCRATCH BY SECLEVEL IN EFFECT (THIS FLAG IS IGNORED IF RCVTEOS HAS A VALUE OF '0'B)
		.... 1...		RCVTEOSA	ERASE-ON-SCRATCH FOR ALL DATA SETS IN EFFECT (THIS FLAG IGNORED IF RCVTEOS HAS A VALUE OF '0'B)
		.... .111		*	RESERVED
394	(18A)	1...		RCVTPROG	ACCESS CONTROL BY PROGRAM IN EFFECT
394	(18A)	BITSTRING	1	*	RESERVED
396	(18C)	UNSIGNED	2	RCVTRTPD	SYSTEM SECURITY RETENTION PERIOD
398	(18E)	UNSIGNED	1	RCVTSLVL	SECURITY LEVEL FOR ERASE-ON-SCRATCH
399	(18F)	UNSIGNED	1	RCVTQLLN	LENGTH OF SINGLE LEVEL DATASET NAME PREFIX
400	(190)	CHARACTER	9	RCVTQUAL	INSTALLATION CONTROLLED PREFIX FOR SINGLE LEVEL DATASET NAMES, PLUS PERIOD FOR LEVEL
409	(199)	UNSIGNED	1	RCVTSLAU	SECLEVEL TO AUDIT
410	(19A)	BITSTRING	1	RCVTMFLG	MISCELLANEOUS FLAGS
		1...		RCVTVRMF	RACF VERSION, RELEASE, AND MODIFICATION FLAG FOR THE ICQ (TSO) SUPPORT IN 1.8.1
		.1..		RCVT310U	RUNNING MVS/SP 3.1.0 OR UP
		..1.		RCVTXMFR	Extended FASTAUTH available
		...1		RCVTDYNL	Support for Dynamic LNKLST is provided by this security product.
		.... 1...		RCVTD4OK	DATE CONVERSION ROUTINE IS AVAILABLE
		.... .1..		RCVTXRCO	EXTENDED ENVRIN AVAILABLE
		.... ..1.		RCVT4INF	SUPPORT FOR FOUR BYTE DATES ON PROGRAMMING INTERFACES IS AVAILABLE
		1		RCVTTLPS	Task level ACEE support available for callable services.
411	(19B)	BITSTRING	8	RCVTMFL1	MORE MISCELLANEOUS FLAGS
		1...		RCVTXUSP	Extended INITUSP support available

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		RCVTXFAR	FASTAUTH ENVRIN/SETROPTS RACLIST support available
		..1.		RCVTENVS	Environment service (IRRENS00) available
		...1		RCVTX500	X500NAME Support Available
		.... 1...		RCVTPSEN	PGMSECURITY OPTION 0 - BASIC 1 - ENHANCED
		.... .1..		RCVTPSWR	PGMSECURITY ENHANCED 0 - FAILURE 1 - WARNING
		.... ..1.		RCVTLPAP	RACF REQ=FAE ATTCH1ST required for LPA 0 - no 1 - YES
		1		RCVTFGSG	On if objects created by this process should have the owner GID determined by the set-gid bit of the parent directory.
412	(19C)	ADDRESS	4	RCVTSPT	POINTER TO THE STARTED PROCEDURES TABLE (ICHRIN03)
416	(1A0)	ADDRESS	4	RCVTDESX	POINTER TO THE PASSWORD ENCRYP- TION INSTALLATION EXIT (ICHDEX01)
420	(1A4)	ADDRESS	4	RCVTNTAB	POINTER TO THE NAMING CONVENTION TABLE (ICHNCV00)
424	(1A8)	ADDRESS	4	RCVTNRTN	POINTER TO THE NAMING CONVENTION ROUTINE (ICHNRT00)
428	(1AC)	ADDRESS	4	RCVTFRX2	ADDRESS OF FRACHECK POST-PROCESSING INSTALLATION EXIT (ICHRFX02)
432	(1B0)	CHARACTER	8	RCVTPROB	ADDRESSES OF CONTROLLED PROGRAMS LIST ANCHOR BOX
432	(1B0)	ADDRESS	4	RCVTCISP	ADDRESS OF CURRENT ANCHOR FOR CONTROLLED PROGRAMS LIST
436	(1B4)	ADDRESS	4	RCVTOISP	ADDRESS OF OLD ANCHOR FOR CONTROLLED PROGRAMS LIST
440	(1B8)	CHARACTER	8	RCVTSWPW	PASSWORD FOR RVARY SWITCH COMMAND

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
448	(1C0)	CHARACTER	8	RCVTINPW	PASSWORD FOR RVARY INACTIVE COMMAND
456	(1C8)	ADDRESS	4	RCVTLARP	PTR TO LINKAGE ASSIST ROUTINE FOR INSTAL EXITS (ICHLAR00)
460	(1CC)	ADDRESS	4	RCVTCTV0	ADDRESS OF TVTOC UTILITY (ICHCTV00)
464	(1D0)	ADDRESS	4	RCVTPNL0	POINTER TO PROFILE NAME LIST ROUTINE
468	(1D4)	ADDRESS	4	RCVTSKGN	Pointer to session key routine
472	(1D8)	ADDRESS	4	RCVTPTEV	Pointer to Passticket eval Routine
476	(1DC)	ADDRESS	4	RCVTPHRX	Address of new pass phrase exit ICHPWX11
480	(1E0)	ADDRESS	4	RCVTGENT	Address of generic name translate routine IRRGNT00
484	(1E4)	ADDRESS	4	RCVTGANC	Address of GENERICANCHOR keyword settings
488	(1E8)	CHARACTER	8	RCVTLNOD	RRSF local node name. Binary zeros if never set. Name persists even if node deleted.
496	(1F0)	ADDRESS	4	RCVTMXPW	Address of IRRMXPW0 routine
500	(1F4)	UNSIGNED	2	RCVTPHIN	Password phrase interval – valid range 0-65534. When set to a non-zero value, this field is used as the system password phrase interval. Otherwise, the RCVTPINV value should be used as the system password phrase interval.
502	(1F6)	CHARACTER	94	*	Reserved.
596	(254)	ADDRESS	4	RCVTGLS1	ADDRESS OF GENLIST DELETE ROUTINE (ICHGLS01)
600	(258)	ADDRESS	4	RCVTRCVX	ADDRESS OF RCVT EXTENSION AREA
604	(25C)	ADDRESS	4	RCVTLAR2	ADDRESS OF ICHLAR02
608	(260)	ADDRESS	4	RCVTFLT0	ADDRESS OF IRRFLT00
612	(264)	ADDRESS	4	RCVTFLT1	ADDRESS OF IRRFLT01
616	(268)	CHARACTER	4	RCVTVRMN	Contains 7791 starting in z/OS V2R2. To determine the release level, use fields in the MVS CVT.
620	(26C)	SIGNED	4	RCVTVMSP	ICB SYNC COUNT VM 370
624	(270)	SIGNED	4	RCVTVMXA	ICB SYNCH COUNT VM XA
628	(274)	BITSTRING	1	RCVTFLG2	RACF 1.9.0 SETROPTS OPTIONS

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		RCVTSLCL	SETROPTS SECLABELCONTROL - ON
		.1..		RCVTCATD	SETROPTS CATDSNS - ON
		..1.		RCVTMLQT	SETROPTS MLQUIET - ON
		...1		RCVTMLST	SETROPTS MLSTABLE - ON
		.... 1...		RCVTMLS	SETROPTS MLS - ON
		.... .1..		RCVTMLAC	SETROPTS MLACTIVE - ON
		.... ..1.		RCVTGNOW	SETROPTS GENERICOWNER - ON
		1		RCVTAUSL	SETROPTS SECLABELAUDIT - ON
629	(275)	BITSTRING	1	RCVTLOGD	LOGOPTIONS FOR DATASET
		1...		RCVTDLGA	LOGOPTIONS "ALWAYS" FOR THE DATASET CLASS
		.1..		RCVTDLGN	LOGOPTIONS "NEVER" FOR THE DATASET CLASS
		..1.		RCVTDLGS	LOGOPTIONS "SUCCESSES" FOR THE DATASET CLASS
		...1		RCVTDLGF	LOGOPTIONS "FAILURES" FOR THE DATASET CLASS
		 1111		*	Reserved
630	(276)	SIGNED	2	RCVTSINT	LU Session Interval
632	(278)	BITSTRING	1	RCVTML2F	SETROPTS options for RACF z/OS Multi Level Security support
		1...		RCVTMLFS	Seclabels required for File System Objects
		.1..		RCVTMLIP	Seclabels required for IPC Objects
		..1.		RCVTMLNM	Display of Names Restricted by Seclabel
		...1		RCVTSBYS	Seclabel By System requested
		 1111		*	Reserved
633	(279)	BITSTRING	1	RCVTFLG3	Miscellaneous flags
		1...		RCVTDCDT	Dynamic CDT is active
		.1..		RCVTPLC	Allow lower case passwords
		..1.		RCVTCFLD	Custom Fields are in effect
		...1		RCVTAUTU	Authority used is available to authorization exits
		.... 1...		RCVTPSC	Special characters are allowed in passwords
		.... .1..		RCVTXPWD	Extended password support is available

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... ..1.		RCVTMFA	MFA functions are available
		1		RCVTMAIL	E-mail support available
634	(27A)	SIGNED	1	RCVTPMIN	Minimum days between password changes
635	(27B)	UNSIGNED	1	RCVTPALG	Password algorithm in effect: 0 = Existing algorithm as indicated by ICHDEX01 (masking, DES, or installation-defined) 1 = KDFAES
636	(27C)	UNSIGNED	2	RCVTPMEM	Password algorithm memory factor.
638	(27E)	UNSIGNED	2	RCVTPREP	Password algorithm iteration factor.
640	(280)	BITSTRING	1	RCVTFLG4	Function availability bits
		1...		RCVTRPFF	Indicates that the R_Password fast-fail option is available
		.1..		RCVTMFA3	MFA3 Functions (OA20930) are available.
		..1.		RCVTIDT	IDT Functions (OA55926) are available.
		...1		RCVTEPT	Enhanced PassTicket Functions (OA59196) are available.
		.... 1...		RCVTPHIA	Phrase interval functions (OA61951) are available.
		.... .1..		RCVTIDT2	Identity Token 2 Functions (OA63462) are available.
		.... .1..		*	Reserved.
		.... ..1.		RCVTVABT	Validated Boot for z/OS functions (OA63507) are available.
		1		*	Reserved
696	(2B8)	CHARACTER	8	RCVTJSYS	USER-ID from the SETROPTS command JES(NJEUSERID(user-id))
704	(2C0)	CHARACTER	8	RCVTJUND	USER-ID from the SETROPTS command JES(UNDEFINEDUSER(user-id))
712	(2C8)	ADDRESS	4	RCVTTMP2	ADDRESS OF RDS TEMPLATES
716	(2CC)	ADDRESS	4	RCVTRCK4	ADDRESS OF IRRRCK04
720	(2D0)	ADDRESS	4	RCVTSVC0	ADDRESS OF ICHSVC00
724	(2D4)	ADDRESS	4	RCVTPTGN	ADDRESS OF THE PASSTICKET ROUTINE

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
728	(2D8)	ADDRESS	4	RCVTFRX4	ADDRESS OF FASTAUTH POST-PROCESSING INSTALLATION EXIT FOR DATASPACE (ICHRFX04)
732	(2DC)	ADDRESS	4	RCVTDX11	ADDRESS OF ICHDEX11
736	(2E0)	ADDRESS	4	RCVTXLT0	ADDRESS OF IRRRXT02
740	(2E4)	ADDRESS	4	RCVTGLS6	ADDRESS OF ICHGLS06
744	(2E8)	ADDRESS	4	RCVTDPTB	ADDRESS OF DYNAMIC PARSE TABLE
748	(2EC)	ADDRESS	4	RCVTRCK2	ADDRESS OF IRRRCK02
752	(2F0)	ADDRESS	4	RCVTRX10	Address of IRRRXT10
756	(2F4)	ADDRESS	4	RCVTRX11	Address of IRRRXT11
760	(2F8)	ADDRESS	4	RCVTDSPC	Address of IRRDSP00
764	(2FC)	BITSTRING	1	RCVTFL2X	RACF SETROPTS options
		1...		RCVTCMPM	SETROPTS COMPATMODE IS ACTIVE
		.1..		RCVTMLSF	1 - SETROPTS MLS (FAILURES) IS IN EFFECT 0 - SETROPTS MLS (WARNING) IS IN EFFECT
		..1.		RCVTMLAF	1 - SETROPTS MLACTIVE (FAILURES) IS IN EFFECT 0 - SETROPTS MLACTIVE (WARNING) IS IN EFFECT
		...1		RCVTCATF	1 - SETROPTS CATDSNS (FAILURES) IS IN EFFECT 0 - SETROPTS CATDSNS (WARNING) IS IN EFFECT
		.... 1...		RCVTAAPL	SETROPTS APPLAUDIT IS ACTIVE
		.... .1..		RCVTNADC	SETROPTS NOADDCREATOR IS IN EFFECT
		.... ..1.		RCVTGNOE	SETROPTS ENHANCEDGENERICOWNER -ON if active. RCVTGNOW also ON if RCVTGNOE is ON.
		1		*	Reserved
765	(2FD)	BITSTRING	1	RCVTNJEF	NJE Flags
		1...		RCVTJWTO	Flag indicating WTO has been issued for NJE, if "ON" - (1)
		.111 1111		*	Reserved
766	(2FE)	BITSTRING	128	RCVTVCPR	Class protection option mask

Table 102. Structure RCVT (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
894	(37E)	BITSTRING	128	RCVTVCGE	Generic profile checking mask
1022	(3FE)	BITSTRING	128	RCVTVCGC	Generic command checking mask
1150	(47E)	BITSTRING	128	RCVTVRCL	CDT-anchored RACLISTed profile mask
1278	(4FE)	BITSTRING	128	RCVTVGNL	CDT-anchored GENLISTed profile mask
1406	(57E)	BITSTRING	128	RCVTVFPT	Fastpath option mask
1534	(5FE)	BITSTRING	128	RCVTVCAU	Auditing option mask
1662	(67E)	BITSTRING	128	RCVTVCST	Statistics option mask
1790	(6FE)	BITSTRING	128	RCVTVLGA	SETROPTS LOGOPTIONS ALWAYS mask
1918	(77E)	BITSTRING	128	RCVTVLNV	SETROPTS LOGOPTIONS NEVER mask
2046	(7FE)	BITSTRING	128	RCVTVLGS	SETROPTS LOGOPTIONS SUCCESSES mask
2174	(87E)	BITSTRING	128	RCVTVLGF	SETROPTS LOGOPTIONS FAILURES mask
2302	(8FE)	CHARACTER	2	*	Reserved
2304	(900)	UNSIGNED	4	RCVTCDTL	Length of a CDT entry
2308	(904)	CHARACTER		*	END OF RCVT

RCVT constants

Table 103. Constants for RCVT

Len	Type	Value	Name	Description
1	DECIMAL	8	RCVTVERN	VERSION NUMBER VALUE: HIGH NIBBLE IS THE VERSION NUMBER, (0=VERSION 1), AND THE LOW NIBBLE IS THE RELEASE NUMBER
4	CHARACTER	77D0	RCVTVRMC	z/OS Security Server (RACF) FMID
4	CHARACTER	1081	RCVTVR81	Indicates RACF 1.8.1
4	CHARACTER	1090	RCVTVR19	Indicates RACF 1.9.0
4	CHARACTER	1092	RCVTVR92	Indicates RACF 1.9.2
4	CHARACTER	2010	RCVTVR21	Indicates RACF 2.1.0
4	CHARACTER	2020	RCVTVR22	Indicates RACF 2.2.0
4	CHARACTER	2030	RCVTVR23	Indicates RACF 2.3.0
4	CHARACTER	2040	RCVTVR24	Indicates RACF 2.4.0

Table 103. Constants for RCVT (continued)

Len	Type	Value	Name	Description
4	CHARACTER	2060	RCVTVR26	Indicates RACF 2.6.0
4	CHARACTER	2608	RCVTVR28	Indicates HRF2608
4	CHARACTER	7703	RCVTVR73	Indicates HRF7703
4	CHARACTER	7705	RCVTVR75	Indicates HRF7705
4	CHARACTER	7706	RCVTVR76	Indicates HRF7706
4	CHARACTER	7707	RCVTVR77	Indicates HRF7707
4	CHARACTER	7708	RCVTVR78	Indicates HRF7708
4	CHARACTER	7709	RCVTVR79	Indicates HRF7709
4	CHARACTER	7720	RCVTVR70	Indicates HRF7720
4	CHARACTER	7730	RCVTVR30	Indicates HRF7730
4	CHARACTER	7740	RCVTVR40	Indicates HRF7740
4	CHARACTER	7750	RCVTVR50	Indicates HRF7750
4	CHARACTER	7770	RCVTVR71	Indicates HRF7770
4	CHARACTER	7780	RCVTVR72	Indicates HRF7780
4	CHARACTER	7790	RCVTVR90	Indicates HRF7790
4	CHARACTER	77A0	RCVTVRA0	Indicates HRF77A0
4	CHARACTER	77B0	RCVTVRB0	Indicates HRF77B0
4	CHARACTER	77C0	RCVTVRC0	Indicates HRF77C0
4	CHARACTER	77D0	RCVTVRD0	Indicates HRF77D0
4	CHARACTER	1100	RCVTV110	Indicates VM only
4	CHARACTER	RCVT	RCVTIDC	EBCDIC RCVT ID, FOR THE RCVT CONTROL BLOCK
1	DECIMAL	1	RCVTIDP2	Identity Propagation 2 services are available
1	DECIMAL	0	RCVTSTG0	Application Identity Mapping (AIM) STAGE 0
1	DECIMAL	1	RCVTSTG1	1 DECIMAL 1 RCVTSTG1 Application Identity Mapping (AIM) STAGE 1
1	DECIMAL	2	RCVTSTG2	Application Identity Mapping (AIM) STAGE 2
1	DECIMAL	3	RCVTSTG3	Application Identity Mapping (AIM) STAGE 3

RCVT cross reference

Table 104. Cross Reference for RCVT

Name	Offset	Hex Value
RCVT	0	

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTAAPL	2FC	8
RCVTADAS	97	8
RCVTADAT	97	10
RCVTAGRO	97	40
RCVTALIS	14A	
RCVTAOPR	97	1
RCVTAPTR	B0	
RCVTATAP	97	4
RCVTATER	97	2
RCVTAUOP	97	
RCVTAUSE	97	20
RCVTAUSL	274	1
RCVTAUTP	E8	
RCVTAUTU	279	10
RCVTAVIO	9A	20
RCVTAXTA	98	
RCVTBAM	2C	
RCVTCATD	274	40
RCVTCATF	2FC	10
RCVTCDTL	900	
RCVTCDTP	BC	
RCVTCFLD	279	20
RCVTCGSN	170	
RCVTCISP	1B0	
RCVTCMPM	2FC	80
RCVTCTV0	1CC	
RCVTDASD	96	40
RCVTDATP	D4	
RCVTDCB	4	
RCVTDNL	174	
RCVTDCDT	279	80
RCVTDEB	8	
RCVTDESX	1A0	
RCVTDGCM	96	10
RCVTDGEN	96	20
RCVTDLGA	275	80

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTDLGF	275	10
RCVTDLGN	275	40
RCVTDLGS	275	20
RCVTDPTB	2E8	
RCVTDSDT	E0	
RCVTDSN	38	
RCVTDSNL	34	
RCVTDSPC	2F8	
RCVTDX11	2DC	
RCVTDYNL	410	10
RCVTD4OK	19A	08
RCVTEGN	35	01
RCVTELEN	F5	
RCVTENVS	19B	20
RCVTEIMR	DC	
RCVTENVP	D8	
RCVTEOS	189	20
RCVTEOSA	189	08
RCVTEOSL	189	10
RCVTEROP	9A	
RCVTEXTA	AD	
RCVTFGSG	19B	
RCVTFLGS	99	
RCVTFLG1	188	
RCVTFLG2	274	
RCVTFLG3	279	
RCVTFLG4	280	
RCVTFLT0	260	
RCVTFLT1	264	
RCVTFL2X	2FC	
RCVTFPB	180	
RCVTFPDS	188	80
RCVTFRCP	C4	
RCVTFRXP	C8	
RCVTFRX2	1AC	
RCVTFRX3	14C	

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTFRX4	2D8	
RCVTGANC	1E4	
RCVTGENT	1E0	
RCVTGLBL	99	01
RCVTGLS1	254	
RCVTGLS6	2E4	
RCVTGNOE	2FC	02
RCVTGNOW	274	02
RCVTHDR	14	
RCVTHIST	F0	
RCVTID	0	
RCVTIDPV	178	
RCVTINAC	F3	
RCVTINDX	C	
RCVTINPW	1C0	
RCVTISTL	30	
RCVTJALL	96	01
RCVTJCHK	96	02
RCVTJSYS	2B8	
RCVTJUND	2C0	
RCVTJWTO	2FD	80
RCVTJXAL	96	04
RCVTLARP	1C8	
RCVTLAR2	25C	
RCVTLGRP	149	80
RCVTLNOD	1E8	
RCVTLOGD	275	
RCVTLPAP	19B	
RCVTMAIL	279	01
RCVTMDEL	144	
RCVTMFA	279	02
RCVTMFA3	280	40
RCVTMFLG	19A	
RCVTMFL1	19B	
RCVTMGDG	144	80
RCVTMGRP	144	20

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTMLAC	274	04
RCVTMLAF	2FC	20
RCVTMLFS	278	80
RCVTMLIP	278	40
RCVTMLNM	278	20
RCVTML2F	278	
RCVTMLQT	274	20
RCVTMLS	274	08
RCVTMLSF	2FC	40
RCVTMLST	274	10
RCVTMUSR	144	40
RCVTMXPW	1F0	
RCVTNADC	2FC	04
RCVTNADS	35	02
RCVTNCDX	B8	
RCVTNCX	B4	
RCVTNDSS	35	20
RCVTNDUP	99	10
RCVTNDVS	35	8
RCVTNJEF	2FD	
RCVTNLS	35	40
RCVTNREC	36	
RCVTNRTN	1A8	
RCVTNTAB	1A4	
RCVTNTMS	35	04
RCVTNTVS	35	10
RCVTOISP	1B4	
RCVTOPTX	149	
RCVTPALG	27B	
RCVTPHRX	1DC	
RCVTPLC	279	40
RCVTPMEM	27C	
RCVTPNL0	1D0	
RCVTPREP	27E	
RCVTPRO	189	80
RCVTPROB	1B0	

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTPROF	189	40
RCVTPROG	18A	80
RCVTPTGN	2D4	
RCVTPSC	279	08
RCVTPSEN	19B	
RCVTPSWR	19B	
RCVTPWDX	EC	
RCVTQLLN	18F	
RCVTQUAL	190	
RCVTRAU0	9C	
RCVTRCK2	2EC	
RCVTRCK4	2CC	
RCVTRCVI	184	
RCVTRCVX	258	
RCVTRCX	1C	
RCVTRCXP	A4	
RCVTRDHD	99	40
RCVTRDSN	96	08
RCVTRDX	20	
RCVTRDXP	17C	
RCVTRELS	AC	F
RCVTREXP	C0	
RCVTRID0	A8	
RCVTRIX	18	
RCVTRIXP	A0	
RCVTRL	176	
RCVTRLX	CC	
RCVTRLXP	D0	
RCVTRMSG	99	4
RCVTRNA	35	80
RCVTRNGP	E4	
RCVTROFF	99	80
RCVTRPFF	280	80
RCVTRTPD	18C	
RCVTRUCB	24	
RCVTRULS	F6	

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTRUL1	F6	
RCVTRUL2	F7	
RCVTRUL3	F8	
RCVTRUL4	F9	
RCVTRUL5	FA	
RCVTRUL6	FB	
RCVTRUL7	FC	
RCVTRUL8	FD	
RCVTRVOK	F1	
RCVTRX10	2F0	
RCVTRX11	2F4	
RCVTSAUD	9A	10
RCVTSBYS	278	10
RCVTSHR	99	20
RCVTSINT	276	
RCVTSKGN	1D4	
RCVTSLAU	199	
RCVTSLCL	274	80
RCVTSLEN	F4	
RCVTSLVL	18E	
RCVTSNTX	F4	
RCVTSPT	19C	
RCVTSTAT	35	
RCVTSTA1	96	
RCVTSVC0	2D0	
RCVTSWPW	1B8	
RCVTTAPE	96	80
RCVTTDSN	188	40
RCVTTEMP	10	
RCVTTERP	9A	80
RCVTTLPS	19A	01
RCVTTMP2	2C8	
RCVTTUAC	9A	40
RCVTUADS	64	
RCVTUVOL	90	
RCVTVCAU	5FE	

Table 104. Cross Reference for RCVT (continued)

Name	Offset	Hex Value
RCVTVCGC	3FE	
RCVTVCGE	37E	
RCVTVCPR	2FE	
RCVTVCST	67E	
RCVTVERS	AC	
RCVTVFPT	57E	
RCVTVGNL	4FE	
RCVTVLGA	6FE	
RCVTVLGF	87E	
RCVTVLGS	7FE	
RCVTVLNV	77E	
RCVTVMSP	26C	
RCVTVMXA	270	
RCVTVRCL	47E	
RCVTVRMF	19A	80
RCVTVRMN	268	
RCVTVRN	AC	F0
RCVTVSL	150	
RCVTWARN	F2	
RCVTWCNT	148	
RCVTWUID	99	02
RCVTXFAR	19B	40
RCVTXLEN	28	
RCVTXLT0	2E0	
RCVTXMFR	19A	20
RCVTXPWD	279	06
RCVTXRCO	19A	04
RCVTXUSP	19B	80
RCVTX500	19B	10
RCVT24MD	99	08
RCVT310U	19A	40
RCVT4INF	19A	02

Chapter 43. RCXP: RACROUTE REQUEST=AUTH Exit Parameter List

RCXP programming interface information

RCXP is a programming interface.

RCXP heading information

Common name:	RACROUTE REQUEST=AUTH exit parameter list
Macro ID:	ICHRCXP
DSECT name:	RCXPL
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	**Subpool** 229 **Key** 0
Size:	152 bytes
Created by:	RACROUTE REQUEST=AUTH function
Pointed to by:	R1 at entry to ICHRCX01 and ICHRCX02
Serialization:	None
Function:	Contains the list of addresses passed to RACROUTE REQUEST=AUTH function pre- and postprocessing installation exits

RCXP mapping

Table 105. Structure RCXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RCXPL	
0	(0)	ADDRESS	4	RCXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	RCXFLAG	Flag byte 1 address: points to a 1-byte area of the following format:
		....		RCXFINOT	B'00000000' 00.. .... RACFIND was not specified.
		1...		RCXFINO	B'10000000' 10.. .... RACFIND=NO was specified.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		11..		RCXFIYES	B'11000000' 11.. .... RACFIND=YES was specified.
		..1.		RCXFLGNM	B'00100000' ..1. .... ENTITYX was specified.
		...1		RCXDTYPV	B'00010000' ...1 .... DSTYPE=V was specified.
		.... 1...		RCX31BA	B'00001000' .... 1... 31 bit addressing
		.... .1..		RCXLGNOF	B'00000100' .... .1.. LOG=NOFAIL was specified.
		.... ..1.		RCXLGNON	B'00000010' .... ..1. LOG=NONE was specified.
		.... .11.		RCXLGNOS	B'00000110' .... .11. LOG=NOSTAT was specified.
		....		RCXLGNOA	B'00000000' .... .00. LOG=ASIS was specified.
		1		RCXENTTY	B'00000001' ....1 ENTITY=(entity name addr, CSA) or ENTITYX=(entity name addr, CSA) was specified.
8	(8)	ADDRESS	4	RCXFLAG2	Flag byte 2 address: points to a 1-byte area of the following format:
		1...		RCXATTAL	B'10000000' 1000 0000 ATTR=ALTER was specified.
		.... 1...		RCXATTCO	B'00001000' 0000 1000 ATTR=CONTROL was specified.
		.... .1..		RCXATTUP	B'00000100' 0000 0100 ATTR=UPDATE was specified.
		.... ..1.		RCXATTRE	B'00000010' 0000 0010 ATTR=READ was specified (or assumed). This value is derived from the ATTR parameter on the RACHECK macro instruction. Note that bit mapping for ATTR differs from bit mapping for the access code (pointed to by RCXACC, offset 48 in the parm list), which matches the mapping in the RACF data set.
12	(C)	ADDRESS	4	RCXFLAG3	Flag byte 3 address: points to a 1-byte area of the following format:
		1...		RCXDTYPT	B'10000000' 1... .... DSTYPE=T
		.1..		RCXDTYPM	B'01000000' .1.. .... DSTYPE=M was specified.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		....		RCXENDSN	B'00000000' ..0. .... ENTITY=dsname or ENTITYX=dsname; tape volser or DASD volser addr was specified.
		..1.		RCXPRPRA	B'00100000' ..1. .... PROFILE=profile addr was specified.
		...1		RCXSTWRN	B'00010000' ...1 .... STATUS=WARNING was specified. .... 0... Reserved.
		.... .1..		RCXGENYS	B'00000100' .... .1.. GENERIC=YES was specified.
		.... ..1.		RCXPAPRO	B'00000010' .... ..1. Private area profile requested.
		1		RCXDTYPL	B'00000001' ....1 DSTYPE=L was specified
16	(10)	ADDRESS	4	RCXINSTL	INSTLN address: points to an area containing the installation parameters. This address is zero if INSTLN was not specified. None of the system modules specify the INSTLN parameter. It is intended for use by installation-written routines that invoke RACHECK to communicate with the RACHECK preprocessing and postprocessing exit routines. Do not confuse this value with the DATA address (pointed to by RCXDATA, offset 32 in the parm list) that comes from a field in the RACF profile for the resource being checked.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	RCXENORP	ENTITY, ENTITYX, or PROFILE address: points to an area containing the resource name (for ENTITY or ENTITYX) or an area containing the profile (for PROFILE). If ENTITY or ENTITYX is used, this area is 44 bytes long for the DATASET class. For general resource classes, the length is taken from the class descriptor table. The name or number is left-justified and padded on the right with blanks. If the exit changes this value, the RACF profile affected is changed but RACF does not communicate the change to the invoker of RACHECK. For example, if a user's authority to a data set is being checked and the exit changes the entity value, the RACF profile checked is the one named by the changed value, but the data set itself is unchanged. Similar processing applies to the OLDVOL, VOLSER, OWNER, and CLASS parameters. Note: If you change the entity name, also change the qualifier, whose address is in CNXQUAL in the ICHCNXP parameter list, to reflect this change. If the exit changes the volser or class fields, and the racheck uses the conditional access list, an abend306 may later occur.
24	(18)	ADDRESS	4	RCXCLASS	CLASS address: points to an area containing a 1-byte length field containing the classname length followed by a field containing the entity class name.
28	(1C)	ADDRESS	4	RCXVSER	VOLSER address: points to a 6-byte area containing the volume serial number. This address points to an area containing blanks if the class is not DATASET.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	4	RCXDATA	DATA address: points to a 1-byte length field followed by the installation data for the entity specified on RACHECK. This address is zero for the preprocessing routine. This address is zero for the postprocessing routine if (1) no data is present, (2) the profile could not be retrieved, or (3) the preprocessing routine indicated bypassing of RACHECK.
36	(24)	ADDRESS	4	RCXWA	Work area address: points to a fullword of zeroes on the initial entry to the preprocessing routine. An installation can use this field for any purpose. Because this field is set to zeroes before entry to the preprocessing exit, the preprocessing and postprocessing exits can use this work area to communicate with each other.
40	(28)	ADDRESS	4	RCXCOMP	ABEND code address: points to a 4-byte field containing the ABEND code that RACHECK is going to issue. The ABEND code is contained in the low-order 12 bits of the field. The address points to an area containing zeroes if RACHECK is not going to issue an ABEND. (If ABEND processing is to be bypassed by RACHECK, the exit routine can zero the ABEND code. In this case, the exit routine should also set the return code to zero; otherwise, the ABEND reason code will be passed to the RACHECK caller as a return code.) Do not confuse an ABEND issued by RACHECK with one issued by an invoker of RACHECK. If a user is not authorized to a resource, RACHECK will not issue an ABEND, but the invoker of RACHECK might. For example, OPEN might issue a 913 ABEND in this case, although RACHECK completed without any ABEND.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
44	(2C)	ADDRESS	4	RCXRCODE	Return code address: points to a 4-byte field containing either: 1. the return code to be passed back to the RACHECK caller in response to the access request (for the meanings of these return codes, see SPL: Supervisor or SPL: System Macros and Facilities) 2. the reason code user to cause the ABEND to be issued (for the meanings of these ABEND reason codes, see RACF Messages and Codes). Do not confuse this code with the return code from the RACHECK preprocessing or postprocessing exit routines described in SPL: RACF. Changes to the field pointed to by RCXRCODE will not affect the values pointed to by RCXFLAG2 (attempted access) and RCXACC (allowed access). **Note:** Do not confuse this code with the return code from the RACROUTE REQUEST=AUTH preprocessing and postprocessing exits described in *z/OS RACF System Programmer's Guide*.
48	(30)	ADDRESS	4	RCXACC	Access code address: points to a 1-byte field containing the user's authorization to the resource that is being checked:
		1...		RCXALTER	X'80' - ALTER
		.1..		RCXCONTR	X'40' - CONTROL
		..1.		RCXUPDAT	X'20' - UPDATE
		...1		RCXREAD	X'10' - READ
		.... 1..1		RCXEXEC	X'09' - EXECUTE (both x'08' and x'01' are set)
		1		RCXNONE	X'01' - NONE. The area is zero if (1) the profile could not be retrieved, or (2) the preprocessing routine indicated bypassing RACHECK.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
52	(34)	ADDRESS	4	RCXRLVLV	Resource level number address: points to a 1-byte field containing the LEVEL value from the resource profile. This address is zero for the preprocessing routine. This address is zero for the postprocessing routine if (1) the profile could not be retrieved, or (2) the preprocessing routine indicated bypassing of RACHECK.
56	(38)	ADDRESS	4	RCXOLVOL	OLDVOL address: points to a 6-byte area containing the volume serial number of a previously HECKined volume of a multivolume data set or tape volume set. This is blank if OLDVOL was not specified.
60	(3C)	ADDRESS	4	RCXCNXPL	Naming conventions address: points to the parameter list of the ICHCNX00 exit. The ICHCNX00 exit invoked by RACF commands and the ICHUT100 utility allows an installation to modify or eliminate the RACF DASD data set naming convention. Corresponding processing might be required in the RACHECK preprocessing exit, so a parameter list with similar structure and content is passed to it to allow the use of common routines.
64	(40)	ADDRESS	4	RCXAPPLN	APPL name address: points to an eight-byte field containing the application name (if supplied on the RACHECK macro instruction). The name is left-justified and padded with blanks. If the APPL parameter was not specified, the field contains blanks. RACHECK processing does not reference this field; this field is intended to provide additional information for the exit routines.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
68	(44)	ADDRESS	4	RCXACEE	ACEE address: points to a fullword containing the address of the ACEE that is used for RACHECK processing. If the ACEE parameter was not specified on the RACHECK macro instruction, the fullword pointd to by this value contains zeroes, and the ACEE pointed to by TCBSENV in the current task control block (TCB) or ASXBSENV in the address space extension block (ASXB) is used for authority checking.
72	(48)	ADDRESS	4	RCXOWNER	OWNER address: points to an eight-byte area containing an identifier that is to be compared with the OWNER field in the resource profile whose access is being checked. If the OWNER parameter was not specified on the RACHECK macro instruction, the area pointed to by this address contains blanks. Note that use of the owner field causes RACHECK to bypass checking of the OPERATIONS attribute during authority checking.
76	(4C)	ADDRESS	4	RCXLCNTL	Logging control address: points to a fullword that the postprocessing exit can use to control auditing of the resource profiles. On entry, the fullword is set to zero. The exit may change this value to a 4 to unconditionally request logging or to 8 to unconditionally suppress logging of the resource profiles. (Note that you can never override the GLOBALAUDIT option. Also, RCXLCNTL will not suppress other RACHECK auditing: UAUDIT, SAUDIT, OPERAUDIT, LOGOPTIONS, SECLABELAUDIT, SECLEVELAUDIT)
80	(50)	ADDRESS	4	RCXACCLV	ACCLVL value address: points to a 1-byte length field followed by 0 to 8 bytes of data from the first subparameter in the ACCLVL keyword on the RACHECK macro.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
84	(54)	ADDRESS	4	RCXACCLP	ACCLVL parameter list address: points to the parameter list passed as the second subparameter in the ACCLVL keyword on the RACHECK macro.
88	(58)	ADDRESS	4	RCXSEQNO	Address of file sequence number points to a two-byte field containing the file sequence number for a tape data set.
92	(5C)	ADDRESS	4	RCXTFLAG	Address of tape flag byte: points to a 1-byte area of the following format:
		1...		RCXTFBLP	B'10000000' 10.. .... TAPELBL=BLP was specified.
		.1..		RCXTFNL	B'01000000' 01.. .... TAPELBL=NL was specified.
		....		RCXTFSTD	B'00000000' 00.. .... TAPELBL=STD was specified. ..00 0000 Reserved.
96	(60)	ADDRESS	4	RCXFLAG4	Address of fourth flag byte: points to a 1-byte area of the following format:
		1...		RCXSTERA	B'10000000' 1... .... STATUS=ERASE was specified
		.1..		RCXSTEVD	B'01000000' .1.. .... STATUS=EVERDOM was specified
		..1.		RCXSTWRO	B'00100000' ..1. .... STATUS=WRITEONLY was specified
		...1		RCXSTACS	B'00010000' ...1 .... STATUS=ACCESS was specified .... 0000 Reserved.
100	(64)	ADDRESS	4	RCXREASN	RACHECK reason code address: points to a 4-byte field containing the reason code to be used with the return code pointed to by offset 44. See SPL System Macros and Facilities for the meanings of the RACHECK reason codes. Do not confuse this reason code with the ABEND reason code.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
104	(68)	ADDRESS	4	RCXNOTIF	Address of NOTIFY userid: an 8-byte area containing the userid of the user to be notified when RACF detects an unauthorized attempt to access a resource protected by this profile. This field is valid only for the postprocessing exit and only if the PROFILE specified NOTIFY.
108	(6C)	ADDRESS	4	RCXUSRID	Address of USERID for third party RACHECK: an 8-byte area containing the userid of the user whose access authority is to be checked.
112	(70)	ADDRESS	4	RCXGRPID	Address of GROUPID for third party RACHECK: an 8-byte area containing the groupid of the user whose access authority is to be checked. This field is valid only if USERID was also specified on the RACHECK.
116	(74)	ADDRESS	4	RCXDDNAM	Address of DDNAME: points to an 8-byte area containing the ddname associated with the dataset name specified as the ENTITY/ENTITYX name, or 8 blanks if DDNAME was not specified.
120	(78)	ADDRESS	4	RCXFLAG5	Address of fifth flag byte: points to a 1-byte area of the following format:
		1...		RCXSCLNT	ACEE is a DCE unauthenticated client.
		.1..		RCXSSRVR	ACEE is a DCE server.
		..11 1111		*	Reserved
124	(7C)	ADDRESS	4	RCXUTOKN	Address of UTOKEN - a User security TOKEN - the security relevant data that is mapped by ICHRUTKN macro
128	(80)	ADDRESS	4	RCXRTOKN	Address of RTOKEN - a Resource security TOKEN - the security relevant data that is mapped by ICHRUTKN macro
132	(84)	ADDRESS	4	RCXLOGST	Address of LOGSTR: a variable length field that is one byte length followed by the character data that the issuer of RACHECK wants to be appear in the SMF record.

Table 105. Structure RCXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
136	(88)	ADDRESS	4	RCXRECVR	Address of RECVR for TSO RECEIVE command: an 8 byte area containing the userid that should match the id in ACEE in order to pass the authoruty check.
140	(8C)	ADDRESS	4	RCXCDTP	Address of a copy of the CDT entry used for the request. CDT entry is provided for reference only and must not be updated. It is set for postprocessing exit only.
144	(90)	ADDRESS	4	RCXAPROF	Address of authorization profile: points to a 2-byte profile name type followed by a 2-byte profile name length followed by the profile name used for the authorization check. It is set for postprocessing exit only.
148	(94)	ADDRESS	4	RCXAUSED	Address of authority used to grant access. It is set for post-processing exit only. Points to a 2-byte area of the following format:
		1...		RCXANORM	Normal authority was used.
		.1..		RCXAOPER	OPERATIONS authority was used.
		..1.		RCXAEXIT	Installation exit processing was used.
		...1		RCXABYPS	User ID = '*BYPASS*' was used. **Note:** When BYPASS is used, the final RACF return code is 4. This may allow access to a resource, depending on the application.
		.... .000		*	Reserved
		0000 0000		*	Reserved

RCXP cross reference

Table 106. Cross Reference for RCXP

Name	Offset	Hex Value
RCXACC	30	
RCXACCLP	54	
RCXACCLV	50	
RCXACEE	44	
RCXALTER	30	80
RCXABYPS	94	10

Table 106. Cross Reference for RCXP (continued)

Name	Offset	Hex Value
RCXAEXIT	94	20
RCXANORM	94	80
RCXAOPER	94	40
RCXAPPLN	40	
RCXAPROF	90	
RCXATTAL	8	
RCXATTCO	8	8
RCXATTRE	8	2
RCXATTUP	8	4
RCXAUSED	94	
RCXCDTP	8C	
RCXCLASS	18	
RCXCNXPL	3C	
RCXCOMP	28	
RCXCONTR	30	40
RCXDATA	20	
RCXDDNAM	74	
RCXDTYPM	C	40
RCXDTYPT	C	80
RCXDTYPV	4	10
RCXENDSN	C	0
RCXENORP	14	
RCXENTTY	4	1
RCXEXEC	30	9
RCXFINO	4	80
RCXFINOT	4	0
RCXFIYES	4	C0
RCXFLAG	4	
RCXFLAG2	8	
RCXFLAG3	C	
RCXFLAG4	60	
RCXFLAG5	120	
RCXFLGNM	4	20
RCXGENYS	C	4
RCXGRPID	70	
RCXINSTL	10	

Table 106. Cross Reference for RCXP (continued)

Name	Offset	Hex Value
RCXLCNTL	4C	
RCXLEN	0	
RCXLGNOA	4	0
RCXLGNOF	4	4
RCXLGNON	4	2
RCXLGNOS	4	6
RCXLOGST	84	
RCXNONE	30	1
RCXNOTIF	68	
RCXOLVOL	38	
RCXOWNER	48	
RCXPAPRO	C	2
RCXPRPRA	C	20
RCXRCODE	2C	
RCXREAD	30	10
RCXREASN	64	
RCXRECVR	88	
RCXRLVLV	34	
RCXRTOKN	80	
RCXSCLNT	78	80
RCXSEQNO	58	
RCXSSRVR	78	40
RCXSTACS	60	10
RCXSTERA	60	80
RCXSTEVD	60	40
RCXSTWRN	C	10
RCXSTWRO	60	20
RCXTFBLP	5C	80
RCXTFLAG	5C	
RCXTFNL	5C	40
RCXTFSTD	5C	0
RCXUPDAT	30	20
RCXUSRID	6C	
RCXUTOKN	7C	
RCXVSER	1C	
RCXWA	24	

Table 106. Cross Reference for RCXP (continued)

Name	Offset	Hex Value
RCX31BA	4	8

Chapter 44. RDDFL: RACROUTE REQUEST=DEFINE Parameter List (Request Section)

RDDFL programming interface information

RDDFL is a programming interface.

RDDFL heading information

Common name:	Request-specific portion of the RACROUTE REQUEST=DEFINE parameter list
Macro ID:	ICHRDDFL
DSECT name:	RDDFLIST
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	None
Storage attributes:	**Subpool** Determined by caller **Key** Determined by caller **Residency** Determined by caller
Size:	Varies depending on the release parameter specified
Created by:	RACROUTE REQUEST=DEFINE macro
Pointed to by:	Address of SAFP plus the offset in SAFPRACP
Serialization:	None
Function:	Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=DEFINE routine

RDDFL mapping

Table 107. Structure RDDFLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	RDDFLIST	
0	(0)	ADDRESS	4	RDDFINSW	ADDRESS OF INSTALLATION DATA WORD
0	(0)	UNSIGNED	1	RDDFLENG	LENGTH OF RACDEF PARAMETER LIST
1	(1)	ADDRESS	3	RDDFINST	ADDRESS OF INSTALLATION DATA
4	(4)	ADDRESS	4	RDDFENTW(0)	ENTITY ADDRESS WORD
4	(4)	BITSTRING	1	RDDFFLGS	FUNCTIONAL FLAGS
		11..		RDDFCHGV	TYPE=CHGVOL

Table 107. Structure RDDFLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		RDDFTDEL	TYPE=DELETE
		.1..		RDDFTADV	TYPE=ADDVOL
		..1.		RDDFOLDV	OLDVOL SPECIFIED
		...1		RDDFNEWN	NEWNAME SPECIFIED
		.... 1...		RDDF31IN	31-BIT ADDRESS LIST INDICATOR
		.... .1..		RDDFDSTV	DSTYPE=V
		.... ..1.		RDDFMDEL	DSTYPE=M
		1		RDDFSPEC	SPECIAL=YES
5	(5)	ADDRESS	3	RDDFENT	DSNAME ADDRESS
8	(8)	ADDRESS	4	RDDFOVOL	OLD VOLSER ADDRESS
8	(8)	ADDRESS	4	RDDFNNMX	NEW DSNAME ADDRESS (LONG NAME FORMAT)
8	(8)	ADDRESS	4	RDDFNNAM	NEW DSNAME ADDRESS
12	(C)	ADDRESS	4	RDDFVSER	VOLSER ADDRESS
16	(10)	ADDRESS	4	RDDFCLNW	ENTITY CLASS NAME ADDRESS
20	(14)	ADDRESS	4	RDDFMENX	MODEL ENTITYX ADDRESS
20	(14)	ADDRESS	4	RDDFMENT	MODEL ENTITY ADDRESS
24	(18)	ADDRESS	4	RDDFMVOL	MODEL VOLSER ADDRESS
28	(1C)	ADDRESS	4	RDDFACEE	ACEE ADDRESS
32	(20)	ADDRESS	4	RDDFUNIT	UNIT INFORMATION ADDRESS.
36	(24)	ADDRESS	1	RDDFUACC	UACC VALUE.
		1...		RDDFALTR	ALTER AUTHORITY.
		.1..		RDDFCNTL	CONTROL AUTHORITY.
		..1.		RDDFUPD	UPDATE AUTHORITY.
		...1		RDDFREAD	READ AUTHORITY.
		.... 1...		RDDFEXEC	EXEC AUTHORITY (TOGETHER WITH NONE)
		.... 1..1		*	RESERVED
		1		RDDFNONE	NONE AUTHORITY.
37	(25)	UNSIGNED	1	RDDFLVL	LEVEL VALUE. 00 TO 99.
38	(26)	BITSTRING	1	RDDFAUDT	AUDIT VALUE.
		1...		RDDFALL	AUDIT ALL ACCESSES.
		.1..		RDDFSUCC	AUDIT SUCCESSFUL ACCESSES.
		..1.		RDDFFAIL	AUDIT ACCESS THAT FAIL.
		...1		RDDFANON	NO AUDITING.
		.... 11..		RDDFQS	SUCCESS QUALIFIER BITS

Table 107. Structure RDDFLIST (continued)

Offset	Offset	Type	Len	Name(Dim)	Description
Dec	Hex				
		11		RDDFQF	FAILURE QUALIFIER BITS
39	(27)	BITSTRING	1	RDDFFLG2	2ND FLAG BYTE
		1...		RDDFRFI	RACFIND PARAMETER CODED
		.1..		RDDFRFIY	RACFIND=YES
		..1.		RDDFCHKA	CHKAUTH=YES
		...1		RDDFTAPE	DSTYPE=TAPE GIVEN
		.... 1...		RDDFEOS	ERASE=YES GIVEN
		.... .1..		RDDFMGEN	MGENER VALUE B'0'=ASIS B'1'=YES
		.... ..1.		RDDFWARN	WARNING=YES GIVEN
		1		RDDFGEN	GENERIC=YES GIVEN
40	(28)	ADDRESS	4	RDDFOWNR	OWNER ADDRESS.
44	(2C)	ADDRESS	4	RDDFDATA	INSTALLATION-DEFINED DATA ADDRESS.
48	(30)	CHARACTER		RDDFEND	END OF V1.4 LIST
RDDF31					
48	(30)	STRUCTURE	8	RDDF31	31-BIT-ADDRESS SAF EXTENSION
48	(30)	ADDRESS	4	RDDFIN31	31-BIT INSTALLATION DATA ADDRESS
52	(34)	ADDRESS	4	RDDFENTX	31-BIT ENTITYX NAME ADDRESS
52	(34)	ADDRESS	4	RDDFEN31	31-BIT ENTITY NAME ADDRESS
56	(38)	CHARACTER		RDD31END	END OF 31 BIT LIS
RDDF17					
56	(38)	STRUCTURE	48	RDDF17	RACF 1.7 PARAMETER LIST EXTENSION
56	(38)	ADDRESS	4	RDDFACC1	ADDRESS OF ACCLVL (1ST)
60	(3C)	ADDRESS	4	RDDFACC2	ADDRESS OF ACCLVL (2ND)
64	(40)	ADDRESS	4	RDDFSLVL	ADDRESS OF SECLVL DATA
68	(44)	ADDRESS	4	RDDFCATG	ADDRESS OF CATEGORY DATA
72	(48)	ADDRESS	4	RDDFEXDT	ADDRESS OF EXPIRATION DATE
76	(4C)	SIGNED	2	RDDFFSEQ	FILESEQ VALUE
78	(4E)	BITSTRING	1	RDDFFLGT	TAPES FLAG BYTE
		11..		RDDFTLBL	TAPELBL SPECIFIED NL=B'01' STD=B'00' BLP=B'10'
		..11 11..		*	Reserved

Table 107. Structure RDDFLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... ..1.		RDDFEXPX	EXTENDED EXPDT INDICATOR B'1'=EXTENDED EXPDT FORMAT (CCYYDDDF) B'0'=STANDARD EXPDT FORMAT (YYDDDF)
		1		RDDFEXP	EXPDT/RETPD VALUE B'1'=EXPDT B'0'=RETPD
79	(4F)	BITSTRING	1	RDDFISUR	RACDEF ISSUER FLAG BYTE
		1...		RDDFISCM	RACF COMMAND ISSUED RACDEF
		.111 111.		*	Reserved
		1		RDDFPROP	B'1'=RACDEF ISSUED AS A RESULT OF AUTOMATIC DIRECTION
80	(50)	ADDRESS	4	RDDFMCLS	ADDR OF MCLASS
84	(54)	ADDRESS	4	RDDFNOTF	ADDR OF NOTIFY ID
88	(58)	ADDRESS	4	RDDFSTCL	Address of STORCLAS data
92	(5C)	ADDRESS	4	RDDFMGCL	Address of MGMTCLAS d
96	(60)	ADDRESS	4	RDDFRSOW	Address of RESOWNER data
100	(64)	BITSTRING	1	RDDFENV	ENVIR flags
		1...		RDDFVRFY	VERIFY specified
		.1..		RDDFIENX	ENTITYX SPECIFIED
		..1.		RDDFIMEX	MENTX SPECIFIED
		...1		RDDFINMX	NEWNAMX SPECIFIED
101	(65)	UNSIGNED	1	*(3)	Reserved
104	(68)	CHARACTER		RDD17END	END OF V1.7 LIS

RDDF18X

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
104	(68)	STRUCTURE	20	RDDF18X(0)	RACF 1.8X PARAMETER LIST EXTENSION
104	(68)	ADDRESS	4	RDDFDDPR	DDNAME POINTER
108	(6C)	ADDRESS	4	RDDFSLAB	POINTER TO SECLABEL
112	(70)	CHARACTER	12	*	UNUSED
124	(7C)	CHARACTER		RDD8XEND	END OF V1.8X

RDDFL cross reference

Table 108. Cross Reference for RDDFL

Name	Offset	Hex Value
RDDFACC1	38	
RDDFACC2	3C	

Table 108. Cross Reference for RDDFL (continued)

Name	Offset	Hex Value
RDDFACEE	1C	
RDDFALL	26	80
RDDFALTR	24	80
RDDFANON	26	10
RDDFAUDT	26	
RDDFCATG	44	
RDDFCHGV	4	80
RDDFCHKA	27	20
RDDFCLNW	10	
RDDFCNTL	24	40
RDDFDATA	2C	
RDDFDDPR	68	
RDDFDSTV	4	04
RDDFEND	30	
RDDFENT	5	
RDDFENTW	4	
RDDFENTX	34	
RDDFENV	64	
RDDFEN31	34	
RDDFEOS	27	08
RDDFEXDT	48	
RDDFEXEC	24	08
RDDFEXP	4E	01
RDDFEXPX	4E	02
RDDFFAIL	26	20
RDDFFLGS	4	
RDDFFLGT	4E	
RDDFFLG2	27	
RDDFFSEQ	4C	
RDDFGEN	27	01
RDDFIENX	64	40
RDDFIMEX	64	20
RDDFINMX	64	10
RDDFINST	1	
RDDFINSW	0	
RDDFIN31	30	

Table 108. Cross Reference for RDDFL (continued)

Name	Offset	Hex Value
RDDFISCM	4F	80
RDDFISUR	4F	
RDDFLENG	0	
RDDFLIST	0	
RDDFLVL	25	
RDDFMCLS	50	
RDDFMDEL	4	2
RDDFMENT	14	
RDDFMENX	14	
RDDFMGCL	5C	
RDDFMGEN	27	04
RDDFMVOL	18	
RDDFNEWN	4	10
RDDFNNAM	8	
RDDFNNMX	8	
RDDFNONE	24	01
RDDFNOTF	54	
RDDFOLDV	4	20
RDDFOVOL	8	
RDDFOWNR	28	
RDDFPROP	79	01
RDDFQF	26	02
RDDFQS	26	03
RDDFREAD	24	10
RDDFRFI	27	80
RDDFRFIY	27	40
RDDFRSOW	60	
RDDFSLAB	6C	
RDDFSLVL	40	
RDDFSPEC	4	01
RDDFSTCL	58	
RDDFSUCC	26	40
RDDFTADV	4	40
RDDFTAPE	27	10
RDDFTDEL	4	80
RDDFTLBL	4E	80

Table 108. Cross Reference for RDDFL (continued)

Name	Offset	Hex Value
RDDFUACC	24	
RDDFUNIT	20	
RDDFUPD	24	20
RDDFVRFY	64	80
RDDFVSER	C	
RDDFWARN	27	02
RDDF17	38	
RDDF18X	68	
RDDF31	30	
RDDF31IN	4	08
	68	
	38	
	7C	

Chapter 45. RDXP: RACROUTE REQUEST=DEFINE Exit Parameter List

RDXP programming interface information

RDXP is a programming interface.

RDXP heading information

Common name: RACROUTE REQUEST=DEFINE exit parameter list

Macro ID: ICHRDXP

DSECT name: RDXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
229

Key
0

Size: 184 bytes

Created by: RACROUTE REQUEST=DEFINE caller

Pointed to by: R1 at entry to ICHRDX01 and ICHRDX02

Serialization: None

Function: Contains the list of addresses passed to the RACROUTE REQUEST=DEFINE pre- and postprocessing installation exits.

RDXP mapping

Table 109. Structure RDXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RDXPL	
0	(0)	ADDRESS	4	RDXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	RDXFLAG	Flag byte address: points to a 1-byte area of the following format:
		11..		RDXTYPEV	B'11000000' TYPE field 2 bits - values defined below
		....		RDXDEFIN	B'00000000' TYPE=DEFINE was specified (or assumed).

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		RDXADDVO	B'01000000' TYPE=ADDVOL was specified
		1...		RDXDELET	B'10000000' TYPE= DELETE was specified
		11..		RDXCHGVO	B'11000000' TYPE=CHGVOL was specified
		..1.		RDXOLDVO	B'00100000' OLDVOL=old vol addr was specified
		...1		RDXNEWNA	B'00010000' NEWNAME=new dsn address
		.... 1...		RDXRESV8	B'00001000' Reserved
		.... .1..		RDXDSTYV	B'00000100' DSTYPE=V was specified
		.... ..1.		RDXDSTYM	B'00000010' DSTYPE=M was specified
		1		RDXSPECY	B'00000001' SPECIAL=YES was specified Multiple flags can be set, for example: TYPE=DEFINE and NEWNAME indicates a rename request. TYPE=ADDVOL and OLDVOL indicates a DASD data set is being extended to a new volume.
8	(8)	ADDRESS	4	RDXINSTL	INSTLN address: points to an area containing the installation parameters. This address is zero if INSTLN was not specified. The INSTLN parameter is not specified by any system modules but is intended for use by installation-written routines that invoke RACDEF to communicate with the RACDEF preprocessing exit routine. If automatic direction of application updates is active, see related field RDXINPTR. If RDXIPROP is on, this RACDEF has been propagated and the data pointed to by RDXINSTL may have been reformatted on the originating system, and will always start with a one byte length, followed by data.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	ADDRESS	4	RDXENTIT	ENTITY address: points to an area containing the resource name. For the DATASET class, this is 44 bytes long. For general resource classes, the length comes from the class descriptor table. The name is left-justified and padded on the right with blanks. (See note at bottom.)
16	(10)	ADDRESS	4	RDXVORN	OLDVOL or NEWNAME address: points to a 44-byte area containing the old volume serial number (for OLDVOL) or the new data set name (for NEWNAME). This address points to an area containing blanks if the class is not DATASET. (See note at bottom.)
20	(14)	ADDRESS	4	RDXVSER	VOLSER address: points to a 6-byte area containing the volume serial number when specified for both the DATASET and TAPEVOL classes. Otherwise, this address points to an area containing blanks. (See note at bottom.)
24	(18)	ADDRESS	4	RDXCLASS	CLASS address: points to an area containing a 1-byte field containing the classname length followed by the entity class name. (See note at bottom.)

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	RDXMENTI	MENTITY address: points to a 44-byte area containing the name of the profile to be modeled. This area contains blanks if MENTITY was not specified. (See note at bottom.) If supplied by the exit, the MENTITY value must be the name of a DATASET profile. The class of the profile being created can be DATASET or any class defined by a class descriptor. If a DATASET profile is found, the following fields are copied from it to the new profile: access list, level, universal access, owner, installation-defined data, and logging options (auditing flags). This processing occurs only for a DEFINE request without NEWNAME. RACDEF's search for the MENTITY profile starts with a chain of resident profiles pointed to from the ACEEAMP field. Profiles are added to this chain by RACDEF depending on the options set in the flag byte pointed to from offset 36 in the parameter list.
32	(20)	ADDRESS	4	RDXMVSER	MVOLSER address: points to a 6-byte area containing the volume serial number of the data set profile being modeled. This area contains blanks if MVOLSER was not specified or if the class is not DATASET.
36	(24)	ADDRESS	4	RDXFLAGA	Flag byte address: points to a 1-byte area of the following format:
		1...		RDXNDMNS	B'10000000' Continue processing. Treat TYPE=DEFINE and MENTITY not defined to RACF as if MENTITY were not specified.
		.1..		RDXADMCR	B'01000000' Add the MENTITY profile to the chain of profiles pointed to by the ACEEAMP field if the profile is found in the RACF data set.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		..1.		RDXADMCD	B'00100000' Add the MENTITY profile to the chain of profiles pointed to by the ACEEAMP field whether or not the profile is found in the RACF data set. If not found, build a dummy profile with the MENTITY name, MVOLSER value, and not-found indicator.
		...1		RDXADMCN	B'00010000' Add the MENTITY profile to the chain of profiles pointed to by the ACEEAMP field if the profile is not found in the RACF data set. Build a dummy profile with the MENTITY name, MVOLSER value, and not-found indicator.
		 1111		RDXRESV7	B'00001111' Reserved.
40	(28)	ADDRESS	4	RDXCNXPL	Naming conventions address: points to the parameter list of the ICHCNX00 exit. THe ICHCNX00 exit, invoked by RACF commands and the ICHUT100 utility, allows an installation to modify or eliminate the RACF DASD data set naming convention. Corresponding processing might be required in the RACDEF preprocessing and postprocessing exits, so a parameter list with similar structure and content is passed to them to allow use of common routines.
44	(2C)	ADDRESS	4	RDXPROFO	Profile options flag bytes address: points to a 3-byte area of the following format:
		1...		RDXUACCV	B'10000000' Use the UACC value from the installation-supplied profile.
		.1..		RDXLEVLV	B'01000000' Use the LEVEL value from the installation-supplied profile.
		..1.		RDXOWNRI	B'00100000' Use the OWNER value from the installation-supplied profile.
		...1		RDXAUDTI	B'00010000' Use the AUDIT value from the installation-supplied profile.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... 1...		RDXGLAUD	B'00001000' Use GLOBALAUDIT value from the installation-supplied profile.
		.... .1..		RDXIDATA	B'00000100' Use installation data from the installation-supplied profile.
		.... ..1.		RDXACLST	B'00000010' Use the access list from the installation-supplied profile.
		1		RDXWRNNG	B'00000001' Use the WARNING value from installation-supplied profile.
BYTE 1					
		1...		RDXCATIN	B'10000000' Use the CATEGORY value from the installation-supplied profile.
		.1..		RDXSECLV	B'01000000' Use the SECLEVEL value from the installation-supplied profile.
		..1.		RDXEOSCR	"B'00100000'" Use the ERASE value from the installation-supplied profile.
		...1		RDXNOTFY	"B'00010000'" Use the NOTIFY value from the installation-supplied profile.
		.... 1...		RDXURETP	"B'00001000'" Use retention period value from the installation-supplied profile.
		.... .1..		RDXRESFL	"B'00000100'" Use RESFLAG for TAPEVOL from the installation-supplied profile.
		.... ..1.		RDXLGDAY	"B'00000010'" Use the terminal logon days from the installation-supplied profile.
		1		RDXLGTIM	"B'00000001'" Use the terminal logon times from the installation-supplied profile.
BYTE 2					
		1...		RDXLGZON	B'10000000' Use terminal timezone info from the installation-supplied profile.
		.1..		RDXCNDAC	B'01000000" Use conditional access list from the installation-supplied profile.
		..11 1111		RDXPRESV	B'00111111' Reserved

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
48	(30)	ADDRESS	4	RDXPROFA	Installation-supplied profile address: points to a profile in the format of that returned by RACHECK ENTITY=(addr,CSA). The profile options flag byte determines the values used in this profile. (On entry to the preprocessing exit, this address is zero.) If both MENTITY processing and installation-supplied profile processing are requested, values from the MENTITY profile override RACDEF defaults and values specified for use from installation-supplied profile override MENTITY profile values.
52	(34)	ADDRESS	4	RDXACEE	ACEE address: points to a fullword containing the address of an ACEE that will be used for RACDEF processing. If the ACEE parameter was not specified on the RACDEF macro instruction or changed by the exit, the fullword pointed to by this value contains zeroes and RACDEF processing uses the ACEE pointed to by TCBSENV in the current task control block (TCB) or ASXBSENV in the address space extension block (ASXB).
56	(38)	ADDRESS	4	RDXUNIT	UNIT Information address: points to an area prefixed by a one-byte length field that contains the length of the UNIT information. If the length is 4, it is assumed the UNIT information contains the UCB coded information. If the length is 5 to 8 characters, it is assumed the UNIT information contains the generic unit information (such as 3330-1 or SYSDA). If this address value is zero or the length field is zero, it is assumed that UNIT information is absent. (See note at bottom.)
60	(3C)	ADDRESS	4	RDXUACC	UACC address: points to a one-byte area containing the universal access authority to be placed in the resource profile being defined. (See note at bottom.) The UACC value has the following format:
		1...		RDXALTER	B'10000000' ALTER authority

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		RDXCONTR	B'01000000' CONTROL authority
		..1.		RDXUPDAT	B'00100000' UPDATE authority
		...1		RDXREADA	B'00010000' READ authority
		.... 111.		RDXRESV9	B'00001110' Reserved.
		1		RDXNONEA	B'00000001' NONE authority
64	(40)	ADDRESS	4	RDXLEVEL	LEVEL address: points to a one-byte area containing the level value to be placed in the new resource profile. This value must be in the range of 00 to 99.
68	(44)	ADDRESS	4	RDXAUDIT	AUDIT address: points to a one-byte area containing the audit flags to be placed in the new resource profile. The AUDIT flag area has the following format:
		1...		RDXAUACC	B'10000000' Audit all accesses.
		.1..		RDXAUSAC	B'01000000' Audit all successful accesses.
		..1.		RDXAUAAF	B'00100000' Audit all access attempts that fail.
		...1		RDXNOAUD	B'00010000' No auditing.
		.... 11..		RDXQSUCC	B'00001100' Qualifier for successful access attempts. The qualifier is of the following format: 00-READ, 01-UPDATE, 10-CONTROL, 11-ALTER
		11		RDXQFAIL	B'00000011' Qualifier for unsuccessful access attempts. The qualifier is of the following format: 00-READ, 01-UPDATE, 10-CONTROL, 11-ALTER
		....		RDXSREAD	B'00000000' Successful READ
		.... .1..		RDXSUPDA	B'00000100' Successful UPDATE
		.... 1...		RDXSCONT	B'00001000' Successful CONTROL
		.... 11..		RDXSALTE	B'00001100' Successful ALTER
		....		RDXUREAD	B'00000000' Unsuccessful READ
		1		RDXUUPDA	B'00000001' Unsuccessful UPDATE
		.... ..1.		RDXUCONT	B'00000010' Unsuccessful CONTROL
		11		RDXUALTE	B'00000011' Unsuccessful ALTER

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
72	(48)	ADDRESS	4	RDXOWNER	OWNER address: points to an eight-byte area containing the owner name to be placed in the new resource profile. This owner name must be a RACF-defined userid or group name. If there is no owner, this field contains blanks or zeroes to indicate the information is absent.
76	(4C)	ADDRESS	4	RDXDATA	DATA address: points to a variable length area of the following format: Offset 0, length 1: Length of data information. Offset 1, variable length: data information.
80	(50)	ADDRESS	4	RDXFLAG2	Flag Byte 2 Address: points to a 1-byte area of the following format. (See note.)
		....		RDXRFDNS	B'00000000' 00.. .... RACFIND was not specified
		1...		RDXRFDNO	B'10000000' 10.. .... RACFIND=NO was specified
		11..		RDXRFDYS	B'11000000' 11.. .... RACFIND=YES was specified
		..1.		RDXCKAYS	B'00100000' ..1. .... CHKAUTH=YES was specified
		...1		RDXDSTYT	B'00010000' ...1 .... DSTYPE=TAPE was specified
		.... 1...		RDXERAYS	B'00001000' .... 1... ERASE=YES was specified
		....		RDXMGNAI	B'00000000' .... .0.. MGENER=ASIS was specified
		.... .1..		RDXMGNYS	B'00000100' .... .1.. MGENER=YES was specified
		.... ..1.		RDXWNGYS	B'00000010' .... ..1. WARNING=YES was specified
		1		RDXGNRCY	B'00000001' ....1 GENERIC=YES was specified

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
84	(54)	ADDRESS	4	RDXCOMP	Completion code address: points to a 4-byte field containing the ABEND code that RACDEF is going to issue. The completion code is contained in the low-order 12 bits of the field. The address points to an area containing zeroes if RACDEF is not going to issue an ABEND. (If ABEND processing is to be bypassed by RACDEF, the exit routine can zero the completion code. In this case, the exit routine should also set the return code to zero; otherwise, the ABEND reason code will be passed to the RACDEF caller as a return code.) Do not confuse an ABEND issued by RACDEF with one issued by an invoker of RACDEF. If a user is not authorized to a resource, RACDEF will not issue an ABEND, but the invoker of RACDEF might. For example, OPEN might issue a 913 ABEND in this case, although RACDEF completed without any ABEND.
88	(58)	ADDRESS	4	RDXRCODE	Return code address: points to a 4-byte field containing either: 1. the return code to be passed back to the RACDEF caller in response to the define request (for the meanings of these return codes, see SPL: Supervisor or SPL: System Macros and Facilities) 2. the reason code used to cause the ABEND to be issued (for the meanings of these ABEND reason codes, see RACF Messages and Codes). **Note:** Do not confuse this code with the return code from the RACDEF preprocessing or postprocessing exit routines described in SPL: RACF.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
92	(5C)	ADDRESS	4	RDXREAS	Reason code address: points to a 4-byte field containing the reason code to be passed back to the RACDEF caller in response to the define request (for the meanings of these reason codes, see SPL: System Macros and Facilities).
96	(60)	ADDRESS	4	RDXWA	Exit work area: points to a fullword of zeroes on the initial entry to the preprocessing routine. An installation can use this field for any purpose. Because this field is set to zeroes before entry to the preprocessing exit, the preprocessing and postprocessing exits can use this work area to communicate with each other.
100	(64)	ADDRESS	4	RDXFLAG3	Third flag byte: points to a 1-byte area of the following format:
		....		RDXTLSTD	B'00000000' 00.. .... TAPELBL=STD was specified
		.1..		RDXTLNL	B'01000000' 01.. .... TAPELBL=NL was specified
		1...		RDXTLBLP	B'10000000' 10.. .... TAPELBL=BLP was specified
104	(68)	ADDRESS	4	RDXACCLV	Address of ACCLVL value: points to a 1-byte length field followed by 0 to 8 bytes of data specified by the first subparameter of the ACCLVL parameter on the RACDEF or RACROUTE REQUEST=DEFINE macro
108	(6C)	ADDRESS	4	RDXACCLP	Address of ACCLVL parameter: points to the parameter list passed as the second subparameter of the ACCLVL keyword on the RACDEF or RACROUTE REQUEST=DEFINE macro. If automatic direction of application updates is active, see related field RDXACPTR. If RDXIPROP is on, this RACDEF has been propagated and the data pointed to by RDXACCLP may have been reformatted on the originating system, and will always start with a one byte length, followed by data.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
112	(70)	ADDRESS	4	RDXSECLP	Address of SECLVL parameter: points to a fullword count field followed by the same number of security level (SECLVL) values (currently either 0 or 1). This security level is the same as the numeric part of the installation-defined security level.
116	(74)	ADDRESS	4	RDXCATEG	Address of CATEGORY parameter: points to a fullword count field followed by the same number of binary halfword category values. Each category value identifies an installation-defined value.
120	(78)	ADDRESS	4	RDXSEQNO	Address of file sequence number: points to a 2-byte field containing the file sequence number for a tape data set
124	(7C)	ADDRESS	4	RDXRETPD	Address of security retention period: points to a 2 byte field containing the retention period.
128	(80)	ADDRESS	4	RDXISSUR	Address of a 1-byte flag which indicates whether the RACDEF was issued by a command or was propagated by automatic direction
		1...		RDXICMND	B'10000000' 1... .... RACDEF issued by a command processor.
		.111 111.		RDXIRESV	B'01111110' .111 111. Reserved.
		1		RDXIPROP	B'00000001' ....1 RACDEF issued as a result of automatic direction.
132	(84)	ADDRESS	4	RDXNOTIF	Address of NOTIFY userid: points to an 8-byte area containing the userid of the user to be notified when RACF detects an unauthorized attempt to access a resource protected by this profile.
136	(88)	ADDRESS	4	RDXMCLAS	Address of a 1-byte length field followed by a 1 to 8 byte field containing the class of the model entity pointed to by RDXMENTI.
140	(8C)	ADDRESS	4	RDXSTCLA	Address of storage class: points to a 2 byte length followed by a 1 to 8 byte storage class name.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
144	(90)	ADDRESS	4	RDXMGCLA	Address of management class: points to a 2 byte length followed by a 1 to 8 byte management class name.
148	(94)	ADDRESS	4	RDXRESOW	Address of resource owner: points to a 2 byte length followed by a 1 to 8 byte resource owner name.
152	(98)	ADDRESS	4	RDXENVIR	Address of a 1-byte flag which indicates what ENVIR was coded
		1...		RDXEVRFY	B'10000000' 1... .... ENVIR=VERIFY coded
		.111 1111		RDXERESV	B'01111111' .111 1111 Reserved.
156	(9C)	ADDRESS	4	RDXDDNAM	Address of DDNAME: points to an 8-byte area containing the ddname associated with the dataset name specified as the ENTITY name, or 8 blanks if DDNAME was not specified. Note: If the exit changes this value, the RACF profile being processed is changed, but RACF does not communicate the change to the invoker of RACDEF. For example, if a data set is being defined to RACF and the exit changes the entity value, the RACF profile defined will show the data set itself is unchanged. Similar processing applies to the OLDVOL, NEWNAME, VOLSER, UNIT, UACC, LEVEL, AUDIT, OWNER, DATA, and CLASS parameters.
160	(A0)	ADDRESS	4	RDXSLABL	Address of SECLABL; points to an eight byte field which contains the security label.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
164	(A4)	ADDRESS	4	RDXINPTR	Address of INSTLN data to be propagated: If INSTLN was specified, and RDXIPROP is off indicating that this is the originating system, this address points to a 256 byte field containing binary zeros. If automatic direction of application updates is active, and INSTLN is to be propagated, the exit must set the first byte of this area to the length of the data, followed by the data. If INSTLN contains pointers, the data pointed to must be moved to this field. The area pointed to by this field will be sent to the remote system as INSTLN. If RDXIPROP is on, or RDXINSTL is zero, RDXINPTR will be zero.
168	(A8)	ADDRESS	4	RDXACPTR	Address of ACCLVL parameter data to be propagated: If the second subparameter of the ACCLVL keyword was specified, and RDXIPROP is off indicating that this is the originating system, this address points to a 256 byte field containing binary zeros. If automatic direction of application updates is active, and the ACCLVL parameter data is to be propagated, the exit must set the first byte of this area to the length of the data, followed by the data. If the data contains pointers, the data pointed to must be moved to this field. The area pointed to by this field will be sent to the remote system as the second subparameter of the ACCLVL keyword. If RDXIPROP is on, or RDXACCLP is zero, RDXACPTR will be zero.
172	(AC)	ADDRESS	4	RDXCDTP	Address of a copy of the CDT entry used for the request. CDT entry is provided for reference only and must not be updated. It is set for postprocessing exit only.

Table 109. Structure RDXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
176	(B0)	ADDRESS	4	RDXMCDTP	Address of a copy of the CDT entry for the model class used for the request. CDT entry is provided for reference only and must not be updated. It is set for postprocessing exit only.
180	(B4)	ADDRESS	4	RDXAUSED	Address of authority used to grant access. It is set for post-processing exit only. Points to a 2-byte area of the following format:
		1...		RDXANORM	B'10000000' 1... .... Normal authority was used
		.1..		RDXASPEC	B'01000000' .1.. .... SPECIAL authority was used
		..1.		RDXAOPER	B'00100000' ..1. .... OPERATIONS authority was used
		...1		RDXAEXIT	B'00010000' ...1 .... Installation exit processing was used
		 0000		*	Reserved
		0000 0000		*	Reserved

Note: If the exit changes this value, the RACF profile being processed is changed, but RACF does not communicate the change to the invoker of RACDEF. For example, if a data set is being defined to RACF and the exit changes the entity value, the RACF profile defined shows the data set itself is unchanged. Similar processing applies to the OLDVOL, NEWNAME, VOLSER, UNIT, UACC, LEVEL, AUDIT, OWNER, DATA, and CLASS parameters.

RDXP cross reference

Table 110. Cross Reference for RDXP

Name	Offset	Hex Value
RDXACCLP	6C	
RDXACCLV	68	
RDXACEE	34	
RDXACLST	2C	2
RDXACPTR	A8	4
RDXADDVO	4	40
RDXADMCD	24	20
RDXADMCN	24	10
RDXADMCR	24	40
RDXAEXIT	B4	10
RDXALTER	3C	80

Table 110. Cross Reference for RDXP (continued)

Name	Offset	Hex Value
RDXANORM	B4	80
RDXAOPER	B4	20
RDXASPEC	B4	40
RDXAUAAF	44	20
RDXAUACC	44	80
RDXAUDIT	44	
RDXAUDTI	2C	10
RDXAUSAC	44	40
RDXAUSED	B4	
RDXCATEG	74	
RDXCATIN	2C	80
RDXCDTP	AC	
RDXCHGVO	4	C0
RDXCKAYS	50	20
RDXCLASS	18	
RDXCNDAC	2C	40
RDXCNXPL	28	
RDXCOMP	54	
RDXCONTR	3C	40
RDXDATA	4C	
RDXDDNAM	9C	
RDXDEFIN	4	0
RDXDELET	4	80
RDXDSTYM	4	2
RDXDSTYT	50	10
RDXDSTYV	4	4
RDXENTIT	C	
RDXENVIR	98	
RDXEOSCR	2C	20
RDXERAYS	50	8
RDXERESV	98	7F
RDXEVRFY	98	80
RDXFLAG	4	
RDXFLAGA	24	
RDXFLAG2	50	
RDXFLAG3	64	

Table 110. Cross Reference for RDXP (continued)

Name	Offset	Hex Value
RDXGLAUD	2C	8
RDXGNRCY	50	1
RDXICMND	80	80
RDXIDATA	2C	4
RDXINPTR	A4	4
RDXINSTL	8	
RDXIPROP	80	1
RDXIRESV	80	7E
RDXISSUR	80	
RDXLEN	0	
RDXLEVEL	40	
RDXLEVLV	2C	40
RDXLGDAY	2C	2
RDXLGTIM	2C	1
RDXLGZON	2C	80
RDXMCDTP	B0	
RDXMCLAS	88	
RDXMENTI	1C	
RDXMGCLA	90	
RDXMGNAI	50	0
RDXMGNYS	50	4
RDXMVSER	20	
RDXNDMNS	24	80
RDXNEWNA	4	10
RDXNOAUD	44	10
RDXNONEA	3C	1
RDXNOTFY	2C	10
RDXNOTIF	84	
RDXOLDVO	4	20
RDXOWNER	48	
RDXOWNRI	2C	20
RDXPRESV	2C	3F
RDXPROFA	30	
RDXPROFO	2C	
RDXQFAIL	44	3
RDXQSUCC	44	C

Table 110. Cross Reference for RDXP (continued)

Name	Offset	Hex Value
RDXRCODE	58	
RDXREADA	3C	10
RDXREAS	5C	
RDXRESFL	2C	4
RDXRESOW	94	
RDXRESV7	24	F
RDXRESV8	4	8
RDXRESV9	3C	E
RDXRETPD	7C	
RDXRFDNO	50	80
RDXRFDNS	50	0
RDXRFDYS	50	C0
RDXSALTE	44	C
RDXSCONT	44	8
RDXSECLP	70	
RDXSECLV	2C	40
RDXSEQNO	78	
RDXSLABL	A0	
RDXSPECY	4	1
RDXSREAD	44	0
RDXSTCLA	8C	
RDXSUPDA	44	4
RDXTLBLP	64	80
RDXTLNL	64	40
RDXTLSTD	64	0
RDXTYPEV	4	C0
RDXUACC	3C	
RDXUACCV	2C	80
RDXUALTE	44	3
RDXUCONT	44	2
RDXUNIT	38	
RDXUPDAT	3C	20
RDXUREAD	44	0
RDXURETP	2C	8
RDXUUPDA	44	1
RDXVORN	10	

Table 110. Cross Reference for RDXP (continued)

Name	Offset	Hex Value
RDXVSER	14	
RDXWA	60	
RDXWNGYS	50	2
RDXWRNNG	2C	1

Chapter 46. RFXP: RACROUTE REQUEST=FASTAUTH Exit Parameter List

RFXP programming interface information

RFXP is a programming interface.

RFXP heading information

Common name: RACROUTE REQUEST=FASTAUTH exit parameter list

Macro ID: ICHRFXP

DSECT name: RFXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Size: Variable

Created by: RACROUTE REQUEST=FASTAUTH caller

Pointed to by: R1 at entry to RACROUTE REQUEST=FASTAUTH

Serialization: None

Function: Contains the list of addresses passed to the RACROUTE REQUEST=FASTAUTH pre- and postprocessing installation exits ICHRFX01 and ICHRFX02

RFXP mapping

Table 111. Structure RFXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		RFXPL	
0	(0)	BITSTRING	1	RFXFLAGS	Authority flags: contains the requested access authority. Access authority requested is:
		.... ..1.		RFXREAD	X'02' - READ,
		.... .1..		RFXUPDAT	X'04' - UPDATE,
		.... 1...		RFXCONTR	X'08' - CONTROL, and

Table 111. Structure RFXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		RFXALTER	X'80' - ALTER. Note: These bit mappings are identical to the bit mappings for the RACHECK ATTR parameter pointed to by RCXFLAGS at offset 8 in the RACHECK preprocessing and postprocessing exit parameter list.
1	(1)	BITSTRING	1	RFXFLAG2	LOG FLAGS and MISC flags contains the types of access attempts to be recorded. LOG values are:
		1...		RFXASIS	X'80' - ASIS,
		.1..		RFXNFAIL	X'40' - NOFAIL. MISC FLAGS
		..1.		RFXNENTX	X'20' - ENTITYX specified If on, RFXENTIT points to a name in ENTITYX format.
		...1		RFXNACHK	X'10' - AUTHCHKS=CRITONLY
		.... 1...		RFXLOGX	X’08’ – LOGSTRX is specified. If on, RFXLOGS points to the LOGSTR data in the format that is described by LOGSTRX keyword.
		.... .111		*	Reserved
2	(2)	FIXED	1	RFXPLEN	Parameter List Length: 28 - OS/390 Security Server R3 or earlier 36 - OS/390 Security Server R4 40 - OS/390 Security Server V2R6 44 - z/OS R8
3	(3)	HEX	1	RFXPVERS	Parameter List Version:
4	(4)	ADDRESS	4	RFXENTIT	For ENTITY (RFXNENTX=off), points to a field as the maximum length name of the given class, as determined by the class descriptor table. Names in the field are left-justified and padded with blanks if necessary. For ENTITYX (RFXNENTX=on), points to a halfword buffer length, followed by a halfword actual length, followed by a one or more character name that may or may not be padded at the end with blanks.

Table 111. Structure RFXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	RFXCLASS	CLASS name address: points to an 8-byte field containing the class name; the name is left-justified and padded with blanks if necessary.
12	(C)	ADDRESS	4	RFXACEE	ACEE address: points to the ACEE that RACF uses for authorization checking. If this address is 0, RACF will use the ACEE pointed to by TCBSENV in the current task control block (TCB) or ASXBSENV in the address space extension block (ASXB).
16	(10)	ADDRESS	4	RFXAPPL	APPL name address: points to an 8-byte field containing the name of the application requesting authorization checking. This name is left-justified and padded with blanks if necessary. If no application name is specified, the address is 0.
20	(14)	ADDRESS	4	RFXWA	WKAREA address: points to 16 fullwords of storage; the exit can use the first 15 fullwords. Because the FRACHECK preprocessing exit cannot issue SVCs, this area is intended to be used by the exit as a work area or register save area.
24	(18)	ADDRESS	4	RFXANSTL	INSTLN address: points to the value specified on the INSTLN parameter on the FRACHECK macro instruction. If the INSTLN parameter was not specified, the address is 0. No system modules specify the INSTLN parameter; it is intended for use by installation-written routines that invoke FRACHECK to communicate with the FRACHECK preprocessing exit routine.
28	(1C)	ADDRESS	4	RFXALET	ACEEALET address: If non-zero, it points to a fullword ALET value to be used when referencing an ACEE in another address space. This address will always be zero for ICHRFX01 and ICHRFX02.

Table 111. Structure RFXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
32	(20)	ADDRESS	4	RFXLOGS	LOGSTR address: • If RFXLOGX is off, this field points to a 1-byte length field, followed by character data that can be from 0 to 255 bytes long. • If RFXLOGX is on, this field points to a 2-byte length field, followed by a 2 byte ID field, followed by data that can be up to 1096 bytes long. The specified length includes the length and ID field. The format of the data depends on the ID field.
36	(24)	ADDRESS	4	RFXPENVR	ENVR address: If non-zero, it points to an area containing a data structure to be used for authority checking. This field is only present if RFXPVERS is 2 or higher.
40	(28)	ADDRESS	4	RFXPCRIT	CRITERIA address: If non-zero, it points to an area containing a data structure to be used for criteria checking. This field is only present if RFXPVERS is 3 or higher.

RFXP constants

Table 112. Constants for RFXP

Len	Type	Value	Name	Description
1	DECIMAL	0	RFXV0	OS/390 Security Server R3 or earlier
1	DECIMAL	1	RFXV1	OS/390 Security Server R4
1	DECIMAL	2	RFXV2	OS/390 Security Server R6 extension
1	DECIMAL	3	RFXV3	z/OS R8 extension

RFXP cross reference

Table 113. Cross Reference for RFXP

Name	Offset	Hex Value
RFXACEE	C	
RFXALET	1C	
RFXALTER	0	80
RFXANSTL	18	
RFXAPPL	10	

Table 113. Cross Reference for RFXP (continued)

Name	Offset	Hex Value
RFXASIS	1	80
RFXCLASS	8	
RFXCONTR	0	8
RFXENTIT	4	
RFXFLAGS	0	
RFXFLAG2	1	
RFXLOGS	20	
RFXLOGX	1	8
RFXNACHK	1	10
RFXNENTX	1	20
RFXNFAIL	1	40
RFXPCRIT	28	
RFXPENVR	24	
RFXPLEN	2	
RFXPVERS	3	
RFXREAD	0	2
RFXUPDAT	0	4
RFXWA	14	

Chapter 47. RIPL: RACROUTE REQUEST=TOKENBLD/ VERIFY/VERIFYX Parameter List (Request Section)

RIPL programming interface information

RIPL is a programming interface.

RIPL heading information

Common name: Request-specific portion of the RACROUTE REQUEST=TOKENBLD, VERIFY, or VERIFYX parameter list

Macro ID: IRRPRIPL

DSECT name: INITPARM

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Varies depending on release and function

Created by: RACROUTE REQUEST=TOKENBLD, VERIFY, or VERIFYX macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=TOKENBLD, VERIFY, or VERIFYX routine

RIPL mapping

Table 114. Structure INITPARM

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	INITPARM	RACINIT INPUT SVC PARMLIST MAP
0	(0)	ADDRESS	1	INITLEN	PARM LIST LENGTH (28)
1	(1)	UNSIGNED	1	INITSUB#	SUBPOOL FOR ACEE STORAGE
2	(2)	BITSTRING	1	INITFLG0	FLAG BYTE 0
		1...		INITBLW	1 => LOC=BELOW SPECIFIED
		.1..		INITANY	1 => LOC=ANY SPECIFIED

Table 114. Structure INITPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		..1.		INITPRAL	VERIFYX INTERNAL PROPAGATION
		...1		INITVFYX	RACINIT VERIFYX INDICATOR
		.... 1...		INITSYSN	PARAMETER SPECIFIED THAT IS NOT COMPATIBLE WITH SYSTEM=YES
		.... .1..		INITNLOG	LOG=NONE SPECIFIED
		11		*	Reserved
3	(3)	BITSTRING	1	INITFLG1	FLAG BYTE 1
		11..		INITENVR	ENVIR - 00 CREATE, 01 CHANGE, 10 DELETE, 111 VERIFY
		..1.		INITNSMC	1 => NO STEP MUST COMPLETE
		...1		INITSUBS	SUBPOOL VALUE SPECIFIED
		.... 1...		INITPCHK	1 => NO PASSWORD PROCESSING TO BE PERFORMED
		.... .1..		INITNSTA	1 => STAT=NO SPECIFIED
		.... ..1.		INITULOG	1 => LOG=ALL SPECIFIED
		1		INITENCR	1 => ENCRYPT=NO SPECIFIED
4	(4)	ADDRESS	4	INITUPTR	ADDR OF USERID BUFFER
8	(8)	ADDRESS	4	INITPPTR	ADDR OF PASSWORD BUFFER
12	(C)	ADDRESS	4	INITSPTR	ADDR OF START PROC NAME
16	(10)	ADDRESS	4	INITIPTR	ADDR OF INSTALLATION INFO
20	(14)	ADDRESS	4	INITGPTR	ADDR OF GROUP NAME BUFFER
24	(18)	ADDRESS	4	INITNPTR	ADDR OF NEW PASSWORD BUFFER
28	(1C)	CHARACTER	1	INITEND1	END PART1
INITPRM2					
28	(1C)	STRUCTURE	20	INITPRM2	VERSION 1 RELEASE 2
28	(1C)	ADDRESS	4	INITPGRP	ADDRESS OF PROGRAMMER NAME BUFFER
32	(20)	ADDRESS	4	INITACCP	ADDRESS OF ACCOUNT NUMBER BUFFER
36	(24)	ADDRESS	4	INITOIDP	ADDRESS OF MAGNETIC STRIPE CARD BUFFER
40	(28)	ADDRESS	4	INITTRMP	ADDRESS OF TERMINAL ID BUFFER
44	(2C)	ADDRESS	4	INITJOBP	ADDRESS OF JOB NAME
48	(30)	CHARACTER	1	INITEND2	END PART2
INITPRM3					

Table 114. Structure INITPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
48	(30)	STRUCTURE	8	INITPRM3	VERSION 1 RELEASE 3
48	(30)	ADDRESS	4	INITAPPP	ADDRESS APPLICATION NAME
52	(34)	ADDRESS	4	INITACEP	ADDRESS ACEE ANCHOR
56	(38)	CHARACTER	1	INITEND3	END PART3

INITPRM4

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
56	(38)	STRUCTURE	44	INITPRM4	RELEASE 1.9
56	(38)	UNSIGNED	1	INITSESN	SESSION TYPE - SEE TOKEN MAP FOR SPECIFIC VALUES (ICHRUTKN)
57	(39)	BITSTRING	1	INITFLG2	WORK UNIT IDENTITY FLAGS
		1...		INITRS	PART OF TRUSTED COMP BASE
		.1..		INITRMT	JOB THIS JOB FROM REMOTE NODE
		..1.		INITRSSP	TRUSTED KEYWORD SPECIFIED
		...1		INITRMSP	REMOTE KEYWORD SPECIFIED
		 1111		*	Reserved
58	(3A)	BITSTRING	1	INITFLG3	MISCELLANEOUS FLAGS
		1...		INITERRO	ERROROPT=NOABEND
		.1..		INITNSTY	NESTED=YES
		..1.		INITNSTC	NESTED=COPY
		...1		INITNMFA	PASSCHK=NOMFA was coded
		 1111		*	Reserved
59	(3B)	BITSTRING	1	*	RESERVED
60	(3C)	ADDRESS	4	INITSLBP	SECURITY LABEL ADDRESS
64	(40)	ADDRESS	4	INITXNDP	EXECUTION NODE ADDRESS
68	(44)	ADDRESS	4	INITSIDP	SUBMITTERS USERID ADDRESS
72	(48)	ADDRESS	4	INITSNDP	SUBMITTERS NODE ADDRESS
76	(4C)	ADDRESS	4	INITSGPP	SUBMITTERS GROUP ADDRESS
80	(50)	ADDRESS	4	INITPOEP	PORT OF ENTRY ADDRESS
84	(54)	ADDRESS	4	INITUTKP	INPUT TOKEN ADDRESS
88	(58)	ADDRESS	4	INITSTKP	SUBMITTERS TOKEN ADDRESS
92	(5C)	ADDRESS	4	INITLSRP	LOG STRING ADDRESS
96	(60)	ADDRESS	4	INITOTKP	OUTPUT TOKEN ADDRESS
100	(64)	CHARACTER	1	INITEND4	END OF 1.9 PARAMETER LIST

INITPRM5

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
100	(64)	STRUCTURE	8	INITPRM5	RELEASE 1.9.2

Table 114. Structure INITPARM (continued)

Offset	Offset	Type	Len	Name(Dim)	Description
Dec	Hex				
100	(64)	ADDRESS	4	INITENVI	ENVRIN ADDRESS
104	(68)	ADDRESS	4	INITENVO	ENVROUT ADDRESS
108	(6C)	CHARACTER	1	INITEND5	END OF 1.9.2 PARAMETER LIST
INITPRM6					
108	(6C)	STRUCTURE	8	INITPRM6	RELEASE 2.6
108	(6C)	ADDRESS	4	INITPNP	POE NETWORK NAME ADDRESS
112	(70)	CHARACTER	1	INITEND6	END OF 2.6 PARAMETER LIST
INITPRM7					
112	(70)	STRUCTURE	8	INITPRM7	X500 NAME EXTENSION
112	(70)	ADDRESS	4	INITX5PR	X500NAME PAIR ADDRESS
116	(74)	CHARACTER	1	INITEND7	END OF X500 NAME EXT
INITPRM8					
116	(74)	STRUCTURE	4	INITPRM8	SERVAUTH EXTENSION
116	(74)	ADDRESS	4	*	RESERVED
120	(78)	ADDRESS	4	INITSRVA	SERVAUTH ADDRESS
124	(7C)	CHARACTER	1	INITEND8	END SERVAUTH EXTENSION
INITPRM9					
124	(7C)	STRUCTURE	12	INITPRM9	RELEASE HRF7730
124	(7C)	ADDRESS	4	INITPHRA	PHRASE ADDRESS
128	(80)	ADDRESS	4	INITNPHA	NEW PHRASE ADDRESS
132	(84)	ADDRESS	4	INITICTX	ICTX ADDRESS
136	(88)	CHARACTER	*	INITEND9	END OF HRF7730 PARM LIST
INITPRMA					
136	(88)	STRUCTURE	8	INITPRMA	RELEASE HRF7760
136	(88)	ADDRESS	4	INITIDID	IDID ADDRESS
140	(8C)	ADDRESS	4	INITICRX	ICRX ADDRESS
144	(90)	CHARACTER	1	INITENDA	END OF HRF7760 PLIST
INITPLV0001					
144	(90)	CHARACTER	*	INITPLV0001	RELEASE PLV0001
144	(90)	ADDRESS	4	INITIDTA	IDTA ADDRESS
148	(94)	CHARACTER	*	INITENDPLV0001	END OF PLV0001
INITPHRB					
0	(0)	STRUCTURE	*	INITPHRB	PASSWORD PHRASE BUFFER
0	(0)	UNSIGNED	1	INITPHRL	PASSWORD PHRASE LENGTH
1	(1)	CHARACTER	*	INITPHRS	PASSWORD PHRASE

Table 114. Structure INITPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
INITNPHB					
0	(0)	STRUCTURE	*	INITNPHB	NEW PASSWORD PHRASE BUFFER
0	(0)	UNSIGNED	1	INITNPHL	NEW PASSWORD PHRASE LENGTH
1	(1)	CHARACTER	*	INITNPHS	NEW PASSWORD PHRASE
INITPAS					
0	(0)	STRUCTURE	9	INITPAS	INITPPTR PASSWORD BUFFER
0	(0)	ADDRESS	1	INITPASL	PASSWORD LENGTH
1	(1)	CHARACTER	8	INITPASS	PASSWORD
INITGRP					
0	(0)	STRUCTURE	9	INITGRP	GROUP NAME BUFFER
0	(0)	ADDRESS	1	INITGRPL	GROUP NAME LENGTH
1	(1)	CHARACTER	8	INITGRPN	GROUP NAME
INITNPA					
0	(0)	STRUCTURE	9	INITNPA	NEW PASSWORD BUFFER
0	(0)	ADDRESS	1	INITNPAL	NEW PASSWORD LENGTH
1	(1)	CHARACTER	8	INITNPAS	NEW PASSWORD
INITOIDB					
0	(0)	STRUCTURE	256	INITOIDB	OID BUFFER
0	(0)	ADDRESS	1	INITOIDL	OID LENGTH
1	(1)	CHARACTER	255	INITOID	OID VALUE
INITENOD					
0	(0)	STRUCTURE	9	INITENOD	EXECUTION NODE KEYWORD
0	(0)	UNSIGNED	1	INITENLN	LENGTH OF EXEC NODE DATA
1	(1)	CHARACTER	8	INITENNM	NAME OF EXECUTION NODE
INITSERV					
0	(0)	STRUCTURE	9	INITSERV	SERVAUTH BUFFER
0	(0)	UNSIGNED	1	INITSRVL	SERVAUTH LENGTH
1	(1)	CHARACTER	*	INITSRVN	SERVAUTH NAME
INITSUID					
0	(0)	STRUCTURE	9	INITSUID	SUBMITTERS USERID KEYWORD
0	(0)	UNSIGNED	1	INITSILN	LENGTH OF SUBMIT USERID
1	(1)	CHARACTER	8	INITSINM	NAME OF SUBMITTERS ID
INITSNOD					
0	(0)	STRUCTURE	9	INITSNOD	SUBMITTERS NODE KEYWORD
0	(0)	UNSIGNED	1	INITSNLN	SUBMIT NODE DATA LENGTH

Table 114. Structure INITPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
1	(1)	CHARACTER	8	INITSNNM	NAME OF SUBMITTERS NODE
INITSGRP					
0	(0)	STRUCTURE	9	INITSGRP	SUBMITTERS GROUP KEYWORD
0	(0)	UNSIGNED	1	INITSGLN	SUBMIT GROUP DATA LENGTH
1	(1)	CHARACTER	8	INITSGNM	NAME OF SUBMIT GROUP
INITLGST					
0	(0)	STRUCTURE	256	INITLGST	LOG STRING KEYWORD MAPPING
0	(0)	UNSIGNED	1	INITLSLN	LENGTH OF LOG STRING DATA
1	(1)	CHARACTER	8	INITLGSD	LOG STRING DATA
INITENVD					
0	(0)	STRUCTURE	14	INITENVD	ENVR OBJECT DATA STRUCTURE
0	(0)	UNSIGNED	4	INITELEN	ENVR OBJECT LENGTH
4	(4)	UNSIGNED	4	INITESLN	ENVR OBJECT STORAGE AREA LENGTH
8	(8)	ADDRESS	4	INITESAD	ENVR OBJECT STORAGE AREA ADDRESS
12	(C)	UNSIGNED	1	INITESSP	ENVR OBJECT STORAGE AREA SUBPOOL
13	(D)	UNSIGNED	1	INITESKY	ENVR OBJECT STORAGE AREA KEY
INITPON					
0	(0)	STRUCTURE	*	INITPON	POE NETWORK NAME KEYWORD
0	(0)	ADDRESS	1	INITPONL	NETWORK NAME LENGTH
1	(1)	CHARACTER	*	INITPOEN	NETWORK NAME
INITX500					
0	(0)	STRUCTURE	*	INITX500	X500 NAME PAIR
0	(0)	ADDRESS	4	INITXLEN	LENGTH OF ENTIRE NAME PAIR DATA STRUCTURE
4	(4)	ADDRESS	2	INITIDNL	ISSUERS NAME LENGTH
6	(6)	ADDRESS	2	INITSDNL	SUBJECTS NAME LENGTH
8	(8)	CHARACTER	*	INITIDN	ISSUERS NAME
*	(*)	CHARACTER	*	INITSDN	SUBJECTS NAME

RIPL cross reference

Table 115. Cross Reference for RIPL

Name	Offset	Hex Value
INITACCP	20	

Table 115. Cross Reference for RIPL (continued)

Name	Offset	Hex Value
INITACEP	34	
INITANY	2	40
INITAPPP	30	
INITBLW	2	80
INITELEN	0	
INITENCR	3	1
INITENDA	90	
INITEND1	1C	
INITEND2	30	
INITEND3	38	
INITEND4	64	
INITEND5	6C	
INITEND6	70	
INITEND7	74	
INITEND8	78	
INITENLN	0	
INITENNM	1	
INITENOD	0	
INITENVD	0	
INITENVI	64	
INITENVO	68	
INITENVR	3	80
INITERRO	3A	80
INITESAD	8	20
INITESKY	D	40
INITESLN	4	
INITESSP	C	
INITFLG0	2	
INITFLG1	3	
INITFLG2	39	
INITFLG3	3A	
INITGPTR	14	
INITGRP	0	
INITGRPL	0	
INITGRPN	1	
INITICRX	8C	

Table 115. Cross Reference for RIPL (continued)

Name	Offset	Hex Value
INITIDID	88	
INITIDN	8	
INITIDNL	4	
INITIPTR	10	
INITJOBP	5C	
INITLEN	0	
INITLGSD	1	
INITLGST	0	
INITLSLN	0	
INITLSRP	5C	
INITNLOG	2	04
INITNMFA	3A	10
INITNPA	0	
INITNPAL	0	
INITNPAS	1	
INITNPHA	80	
INITNPTR	18	
INITNSMC	3	20
INITNSTA	3	04
INITNSTC	3A	20
INITNSTY	3A	40
INITOID	1	
INITOIDB	0	
INITOIDL	0	
INITOIDP	24	
INITOTKP	60	
INITPARM	0	
INITPAS	0	
INITPASL	0	
INITPASS	1	
INITPCHK	3	08
INITPGRP	1C	
INITPHRA	7C	
INITPNP	6C	
INITPOEN	1	
INITPOEP	50	

Table 115. Cross Reference for RIPL (continued)

Name	Offset	Hex Value
INITPON	0	
INITPONL	0	
INITPPTR	8	
INITPRAL	2	20
INITPRMA	88	
INITPRM2	1C	
INITPRM3	30	
INITPRM4	38	
INITPRM5	64	
INITPRM6	6C	
INITPRM7	70	
INITPRM8	74	
INITPRM9	7C	
INITRMSP	39	10
INITRMT	39	40
INITRS	39	80
INITRSSP	39	20
INITSDN	*	
INITSDNL	6	
INITSESN	38	
INITSERV	0	
INITSGLN	0	
INITSGNM	1	
INITSGPP	4C	
INITSGRP	0	
INITSIDP	44	
INITSILN	0	
INITSINM	1	
INITSLBP	3C	
INITSNDP	48	
INITSNLN	0	
INITSNNM	1	
INITSNOD	0	
INITSPTR	C	
INITSRVA	78	
INITSRVL	0	

Table 115. Cross Reference for RIPL (continued)

Name	Offset	Hex Value
INITSRVN	1	
INITSTKP	58	
INITSUB#	1	
INITSUBS	3	10
INITSUID	0	
INITSYSN	2	08
INITTRMP	28	
INITULOG	3	02
INITUPTR	4	
INITUSR	0	
INITUSRI	1	
INITUSRL	0	
INITUTKP	54	
INITVFYX	2	10
INITXLEN	0	
INITXNDP	40	
INITX5PR	70	
INITX500	0	

Chapter 48. RIXP: RACROUTE REQUEST=VERIFY/ VERIFYX Exit Parameter List

RIXP programming interface information

RIXP is a programming interface.

RIXP heading information

Common name: RACROUTE REQUEST=VERIFY or VERIFYX exit parameter list mapping

Macro ID: ICHRIXP

DSECT name: RIXPL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
229

Key
0

Size: 200 bytes

Created by: RACROUTE REQUEST=VERIFY or VERIFYX processing

Pointed to by: R1 at entry to ICHRIX01 and ICHRIX02

Serialization: None

Function: Contains the list of addresses passed to RACROUTE REQUEST=VERIFY or VERIFYX pre- and postprocessing installation exits

Note: If you are using VLF ACEE caching, be aware that some information normally retrieved from the profile and put into the RIXP parameter list is not available when an ACEE is retrieved from the cache, since this would create a performance impact that would negate the VLF performance enhancement. Therefore, an indicator (RIXCACHE) is set on to indicate that this invocation represents a call using VLF cached data. If you find that a RIXP parameter is zero, you should check the ACEE for the data if the RIXCACHE indicator is on.

RIXP mapping

Table 116. Structure RIXPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RIXPL	
0	(0)	ADDRESS	4	RIXLEN	Length address: points to a fullword containing the number of fullwords in this parameter list.
4	(4)	ADDRESS	4	RIXFLAG	Flag byte address: points to a 1-byte area of the following format:

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		....		RIXENVCR	B'00000000' 00.. .... ENVIR=CREATE was specified (or assumed).
		.1..		RIXENVCH	B'01000000' 01.. .... ENVIR=CHANGE was specified.
		1...		RIXENVDE	B'10000000' 10.. .... ENVIR=DELETE was specified.
		....		RIXSMCYS	B'00000000' ..0. .... SMC=YES was specified.
		..1.		RIXSMCNO	B'00100000' ..1. .... SMC=NO was specified.
		...1		RIXSUBPP	B'00010000' ...1 .... SUBPOOL parameter specified.
		....		RIXPSCKY	B'00000000' .... 0... PASSCHK=YES was specified.
		.... 1...		RIXPSCKN	B'00001000' .... 1... PASSCHK=NO was specified (bypass password checking). Both the preprocessing and postprocessing exit can set this option. This option causes RACINIT to: • Bypass checking that the old password is correct and has not expired • Bypass checking that the new password is valid • Bypass updating the old password with the new • Bypass incrementing the password revoke count or resetting it to zero. If PASSCHK=NO, the postprocessing exit must issue a return code of 4 to re-invoke the RACINIT function to allow the option to take effect. An installation can use this procedure to bypass enforcing password expiration.
		.... .1..		RIXSTANO	B'00000100' .... .1.. STAT=NO was specified.
		.... ..1.		RIXLOGAL	B'00000010' .... ..1. LOG=ALL was specified.
		1		RIXENCNO	B'00000001' ....1 ENCRYPT=NO was specified.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	RIXUID	USERID address: points to an area of the following format: Offset 0, length 1: Length of user identification. Offset 1, length 8: User identification. If no userid was supplied, the length is zero and the value is blanks, so that an exit routine can supply a value. If a started procedure name was supplied, then the userid might have come from the started procedure table (ICHRIN03). See Chapter on "RACF Options" in SPL: RACF.
12	(C)	ADDRESS	4	RIXPWD	PASSWORD address: points to an area of the following format: Offset 0, length 1: Length of password. Offset 1, length 8: Password. If ENCRYPT=NO was specified, the password is treated as if it were already encrypted. If no password was supplied, the length is zero and the value is blanks, so that an exit routine can supply a value.
16	(10)	ADDRESS	4	RIXSTART	START address: points to an 8-byte area containing the PROC name of the started task. If no started procedure name was supplied, the value is blanks so that an exit routine can supply a value. However, RACINIT will not use the value.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	RIXINSTL	INSTLN address: points to an area containing the installation parameters. This address is zero if INSTLN was not specified. No system modules specify the INSTLN parameter. INSTLN is intended for use by installation-written routines that invoke RACINIT to communicate with the RACINIT preprocessing exit routine. Do not confuse this value with the terminal data value (pointed to by RIXTRMDA, offset 68 in the parm list) or the user data value (pointed to by RIXUSRDA, offset 72 in the parm list), which are taken from fields in the RACF profiles for the user entering the system and the terminal being used.
24	(18)	ADDRESS	4	RIXGROUP	GROUP address: points to an area of the following format: Offset 0, length 1: Length of group name. Offset 1, length 8: Group name. If no group name was supplied, the length is zero and the value is blanks, so that an exit routine can supply a value.
28	(1C)	ADDRESS	4	RIXNEWPA	NEWPASS address: points to an area of the following format: Offset 0, length 1: Length of new password. Offset 1, length 8: New password. If no new password was supplied, the length is zero and the value is blanks, so that an exit routine can supply a value.
32	(20)	ADDRESS	4	RIXACEE	ACEE address: points to an area containing the access control environment element. At entry to the RACINIT preprocessing exit, this address points to the area of storage where the ACEE will be built. At entry to the RACINIT postprocessing exit, this address points to the actual ACEE built by RACINIT. Any changes made by the postprocessing routine remain in effect for the duration of the session or job.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
36	(24)	ADDRESS	4	RIXPGMNM	PGMNAME address: points to a 20-byte area containing the programmer name information (or blanks if no programmer name exists). This address is zero if PGMNAME was not specified.
40	(28)	ADDRESS	4	RIXACTIN	ACTINFO address: points to a 144-byte area containing accounting information (or zeroes if no accounting information exists). The 144-byte area is consistent with similar accounting information in the SMF (type 20) job initiation record: • The first byte contains the number (in binary) of accounting fields. • The following bytes contain accounting fields, where each entry for an accounting field contains a 1-byte length field followed by the field (in EBCDIC). A length indicator of 0 indicates an omitted field. This address is zero if ACTINFO was not supplied.
44	(2C)	ADDRESS	4	RIXOIDCD	OIDCARD address: points to an area containing a 1-byte length field followed by a field containing the OIDCARD identification number. The length byte is 0 if OIDCARD was not specified.
48	(30)	ADDRESS	4	RIXTRMID	TERMID address: points to an 8-byte area containing the terminal identifier. The name is left-justified and padded on the right with blanks. This address is 0 if TERMID was not specified. If this value is altered in the exit the POE value will also be altered and its class will be TERMINAL. (since a terminal is the same as a port of entry in the TERMINAL class)

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
52	(34)	ADDRESS	4	RIXWA	Work area address: points to a fullword of zeroes on the initial entry to the preprocessing routine. Because this field is set to zeroes before entry to the preprocessing exit, the preprocessing and postprocessing exits can use this work area to communicate with each other.
56	(38)	ADDRESS	4	RIXCOMP	ABEND code address: points to a 4-byte field containing the ABEND code that RACINIT is going to issue. The low-order 12 bits of the field contain the ABEND code. The address points to an area containing zeroes for the postprocessing routine if RACINIT is not going to issue an ABEND code. This address points to an area containing zeroes for the preprocessing routine. (If ABEND processing is to be bypassed, the exit routine can set the ABEND code to zero. In this case, the return code should also be set to zero; otherwise the ABEND reason code will be passed to the RACINIT caller as a return code). Do not confuse an ABEND issued by RACINIT with one issued by an invoker of RACINIT. For example, if a user is not defined to RACF, RACINIT will not issue an ABEND, but the invoker of RACHECK may. A batch job might fail with a JCL error in this case, although RACINIT completed without an ABEND.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
60	(3C)	ADDRESS	4	RIXRCODE	Return code address: points to a 4-byte field containing either 1. the return code to be passed back to the RACINIT caller in response to the identification request. For the meanings of these return codes, see *z/OS Security Server RACROUTE Macro Reference*. 2. the reason code used to cause the ABEND to be issued (if nonzero). For meanings of these ABEND codes, see *z/OS Security Server RACF Messages and Codes*. This address points to an area containing zeroes for the preprocessing routine. **Note:** Do not confuse this return code with the return code from the RACINIT pre/postprocessing exit routines, the meanings of which are documented in SPL: Supervisor and SPL: System Macros and Facilities.
64	(40)	ADDRESS	4	RIXFLAG2	Flag byte address: points to a 1-byte area of the following format:
		1...		RIXBYPOI	B'10000000' 1... .... Bypass OIDCARD processing. RACINIT will ignore any OIDCARD information and any user profile indication that an OIDCARD is required.
		.1..		RIXCACHE	B'01000000' .1.. .... For the post-processing exit, data from ACEE VLF cache. Data from cache may not be current if not cache sensitive. Data normally retrieved from profile may not be available. Use data from the ACEE.
		..1.		RIXPTAUT	B'00100000' ..1. .... For post-processing exit, indicates user is being authenticated by a PassTicket.
		...1		RIXPTRPY	B'00010000' ...1 .... For post-processing exit, indicates that PassTicket evaluation failed due to attempted replay of a PassTicket.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... 1...		RIXERRO	B'00001000' .... 1... ERROROPT=NOABEND
		.... .1..		RIXNSTY	B'00000100' .... .1.. NESTED=YES was coded. This means the ACEE compression exit will get control for the address space ACEE.
		.... ..1.		RIXIDPRD	B'00000010' .... ..1. NESTED=COPY was coded.
		1		RIXNICR	B'00000001' ....1 ID not resolved from ICR in ICRX and RACINIT redriven
68	(44)	ADDRESS	4	RIXTRMDA	Terminal data address: points to a 1-byte length field followed by the installation terminal data, as specified in the DATA parameter of the RDEFINE or RALTER commands. The length field includes the 1-byte length of the length field. This address is zero if (1) no data is present, (2) the profile could not be retrieved, (3) the preprocessing routine indicated bypassing of RACINIT, or (4) the NOTERMINAL system option is in effect.
72	(48)	ADDRESS	4	RIXUSRDA	User data address: points to a 1-byte length field followed by the installation data specified on the ADDUSER and ALTUSER commands for the user specified on RACINIT. The length field includes the 1-byte length of the length field. The address points to an area containing zeroes for the preprocessing routine. In addition, the address is zero if (1) no data is present, (2) the profile could not be retrieved, or (3) the preprocessing routine indicated bypassing of RACINIT.
76	(4C)	ADDRESS	4	RIXTRMLV	Terminal level number address: points to a 1-byte field containing the LEVEL value from the terminal profile as set by the RDEFINE or RALTER commands. This address is zero if (1) the profile could not be retrieved, or (2) the preprocessing routine indicated bypassing of RACINIT.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
80	(50)	ADDRESS	4	RIXJOBNM	Jobname address: points to an 8-byte area containing the job name of a background job. The area contains blanks if no job name information is available.
84	(54)	ADDRESS	4	RIXAPPLN	APPL name address: points to an 8-byte field containing the application name, if supplied on the RACINIT macro instruction. The name is left-justified and padded with blanks. If the APPL parameter was not specified, the field contains blanks.
88	(58)	ADDRESS	4	RIXSUBPL	SUBPOOL address: points to a 1-byte field containing the subpool (as specified on the RACINIT macro) from which the ACEE and its storage will be obtained. This field has meaning only when the appropriate bit is set in the flag byte, pointed to from offset 4 in the parameter list. Because the storage has already been obtained when the preprocessing exit gains control, there is no effect if the exit changes this value.
92	(5C)	ADDRESS	4	RIXACEEA	ACEE address: points to a fullword containing the address specified on the ACEE parameter of the RACINIT macro instruction. If the ACEE parameter was not specified on the RACINIT macro instruction, this parameter is zero. When specified, the fullword has the following meanings: • For ENVIR=CREATE, RACF will place the address of the ACEE to be built in the fullword and not into the ASXBSENV. This address is identical to the contents of the field at offset 32 in the parameter list. • For ENVIR=CHANGE or ENVIR=DELETE, the fullword contains the address of the ACEE as specified on the RACINIT macro instruction. This address is identical to the contents of RIXACEE at offset 32 in the parameter list.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
96	(60)	ADDRESS	4	RIXAPPLD	Application data pointer: points to a 1-byte field containing the length of the application data followed by the application data as specified by the DATA operand on the RDEFINE or RALTER commands. The pointer is zero if (1) the application name was not supplied, (2) the profile could not be retrieved, or (3) the preprocessing exit indicated bypassing of RACINIT. The pointer is always zero on entry to the preprocessing exit.
100	(64)	ADDRESS	4	RIXAPPLP	Application level pointer: points to a 1-byte field containing the level value for the application, as specified by the LEVEL operand on the RDEFINE or RALTER commands. The pointer is zero if: • The application name was not supplied • The profile could not be retrieved • The preprocessing exit indicated bypassing of RACINIT The address is always zero on entry to the preprocessing exit. **Note:** If the application identified by RIXAPPLN at offset 84 is IMS, you should not use this field becuse IMS uses this field when IMS is active.
104	(68)	ADDRESS	4	RIXPCIA	Password Change Interval Address: points to a 4-byte area that contains a 31-bit fixed binary integer that represents the password change interval value found in the user's profile. NOTE: Upon initial entry to exit ICHRIX01 the four byte field will contain zeros. Upon entry to the ICHRIX02 exit, the four byte field will contain the value from the user entry. Changes to this value are ignored by RACINIT processing.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
108	(6C)	ADDRESS	4	RIXPLCDA	Password Last Change Date Address: points to a 3-byte area that contains the date of the last password change. The format of this area is: yyddds where: 'yy' is the year, 'ddd' is the day, and 's' is the packed decimal sign. NOTE: Upon initial entry to exit ICHRIX01 the three byte field will contain zeros. Upon entry to the ICHRIX02 exit, the three byte field will contain the value from the user entry.
112	(70)	ADDRESS	4	RIXPSESN	"Session Type" Data Address: points to a 1-byte area that contains the session type. These are defined in the macro - ICHRUTKN.
116	(74)	ADDRESS	4	RIXWUIDF	Work Unit Identity Token Flag Byte Address: Mirrors the TOKFLG2 field in the token mapping macro ICHRUTKN. When the final individual fields are set on a RACINIT, this area is copied to the token associated with the ACEE. Points to a 1-byte area of the following format:
		1...		RIXDFTKN	B'10000000' 1... .... Default Token bit
		.1..		RIXUDUSR	B'01000000' .1.. .... Undefined User bit
		..1.		*	Reserved.
		...1		RIXERRTK	B'00010000' ...1 .... Token in error
		.... 1...		RIXTRUST	B'00001000' .... 1... Part of Trusted Computer Base
		.... .1..		RIXSURGU	B'00000100' .... .1.. Surrogate Userid
		.... ..1.		RIXREMOT	B'00000010' .... ..1. REMOTE Keyword specified
		1		RIXPRIV	B'00000001' ....1 Privileged user indicator
120	(78)	ADDRESS	4	RIXPPOEX	"Port of Entry Index" Address: points to a 1-byte area that contains the class determined by the port of entry and session type. These are defined in the macro - ICHRUTKN.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
124	(7C)	ADDRESS	4	RIXPSLBP	SECLABEL Address: points to an 8-byte character SECLABEL field.
128	(80)	ADDRESS	4	RIXPXNDP	EXECUTION NODE Address: points to a 1-byte field containing the length of the EXECUTION NODE followed by an 8-byte EXECUTION NODE name.
132	(84)	ADDRESS	4	RIXPSIDP	SUBMITTERS USERID Address: points to a 1-byte field containing the length of the SUBMITTERS ID followed by an 8-byte SUBMITTERS ID.
136	(88)	ADDRESS	4	RIXPSNDP	SUBMITTERS NODE Address: points to a 1-byte field containing the length of the SUBMITTERS NODE followed by an 8-byte SUBMITTERS NODE name.
140	(8C)	ADDRESS	4	RIXPSGPP	SUBMITTERS GROUP ID Address: points to a 1-byte field containing the length of the SUBMITTERS GROUP followed by an 8-byte SUBMITTERS GROUP name.
144	(90)	ADDRESS	4	RIXPPOE	PORT OF ENTRY Address: points to an 8-byte character PORT OF ENTRY field.
148	(94)	ADDRESS	4	RIXPUTKP	TOKNIN Address: points to a Users TOKEN which can be mapped by macro ICHRUTKN.
152	(98)	ADDRESS	4	RIXPSTKP	STOKEN Address: points to a Submitters TOKEN which can be mapped by macro ICHRUTKN.
156	(9C)	ADDRESS	4	RIXPLSRP	LOG STRING Address: points to a 1-byte length field followed by up to 255 bytes of LOG data used in auditing.
160	(A0)	ADDRESS	4	RIXPOTKP	TOKNOUT Address: points to a Users TOKEN which can be mapped by macro ICHRUTKN.
164	(A4)	ADDRESS	4	RIXPOENP	PORT OF ENTRY NETWORK NAME address: points to a 1-byte length followed by a 1 to 8-byte character PORT OF ENTRY NETNAME field. Zero if not specified.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
168	(A8)	ADDRESS	4	RIXPLCD4	Password Last Change Date Address: points to a 4-byte area that contains the date of the last password change. The format of this area is: yyyyddds where: 'yyyy' is the year, 'ddd' is the day, and 's' is the packed decimal sign. **Note:** Upon initial entry to exit ICHRIX01 the four byte field will contain zeros. Upon entry to the ICHRIX02 exit, the four byte field will contain the value from the user entry.
172	(AC)	ADDRESS	4	RIXX5PRP	X500 NAME PAIR ADDRESS: points to a 4-byte length of structure, followed by 2 2-byte lengths followed by up to 255 bytes of issuers name and up to 255 bytes of subjects name. RIXX5PRP will be 0 if X500NAME not specified
176	(B0)	ADDRESS	4	RIXSRVA	SERVAUTH address: points to a 1-byte length followed by up to 64 bytes of SERVAUTH resource name. Zero if not specified
180	(B4)	ADDRESS	4	RIXPHRA	PHRASE address: points to a 1-byte length (0 if not specified), followed by a 100 byte buffer containing the phrase
184	(B8)	ADDRESS	4	RIXNPHA	NEWPHRASE address: points to a 1-byte length (0 if not specified) followed by a 100 byte buffer containing the new pass phrase
188	(BC)	ADDRESS	4	RIXICTX	ICTX address: Points to an ICTX block as mapped by IRRPICTX
192	(C0)	ADDRESS	4	RIXIDID	IDID address: Points to an IDID block as mapped by IRRPIDID
196	(C4)	ADDRESS	4	RIXICRX	ICRX address: Points to an ICRX block as mapped by IRRPICRX
200	(C8)	ADDRESS	4	RIXFLAG3	Flag byte address: points to a 1-byte area of the following format:
		1...		RACPNMFA	PASSCHK=NOMFA was coded
		.111 1111		*	Reserved.
204	(CC)	ADDRESS	4	RIXIDTA	IDTA ADDRESS: points to an IDTA data area as mapped by IRRPIDTA.

Table 116. Structure RIXPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
208	(D0)	ADDRESS	4	RIXPHIA	Password Phrase Change Interval Address: points to a 4-byte area that contains a 31-bit fixed binary integer that represents the password phrase change interval value found in the user's profile. **Note:** Upon initial entry to exit ICHRIX01 the four byte field contains zeros. Upon entry to the ICHRIX02 exit, the four byte field contains the value from the user entry. Changes to this value are ignored by RACINIT processing.

RIXP cross reference

Table 117. Cross Reference for RIXP

Name	Offset	Hex Value
RACPNMFA	CE	80
RIXACEE	20	
RIXACEEA	5C	
RIXACTIN	28	
RIXAPPLD	60	
RIXAPPLN	54	
RIXAPPLP	64	
RIXBYPOI	40	80
RIXCACHE	40	40
RIXCOMP	38	
RIXDFTKN	74	80
RIXENCNO	4	1
RIXENVCH	4	40
RIXENVCR	4	0
RIXENVDE	4	80
RIXERRO	40	8
RIXERRTK	74	10
RIXFLAG	4	
RIXFLAG2	40	
RIXFLAG3	CE	
RIXGROUP	18	

Table 117. Cross Reference for RIXP (continued)

Name	Offset	Hex Value
RIXICRX	C4	
RIXICTX	BC	
RIXIDID	C0	
RIXIDPRD	40	
RIXINSTL	14	
RIXJOBNM	50	
RIXLEN	0	
RIXLOGAL	4	2
RIXNEWPA	1C	
RIXNPHA	B8	
RIXNSTC	02	
RIXNSTY	04	
RIXOIDCD	2C	
RIXPCIA	68	
RIXPGMNM	24	
RIXPHRA	B4	
RIXPLCDA	6C	
RIXPLCD4	A8	
RIXPLSRP	9C	
RIXPOENP	A4	
RIXPOTKP	A0	
RIXPPOE	90	
RIXPPOEX	78	
RIXPRIV	74	1
RIXPSCKN	4	8
RIXPSCKY	4	0
RIXPSESN	70	
RIXPSGPP	8C	
RIXPSIDP	84	
RIXPSLBP	7C	
RIXPSNDP	88	
RIXPSTKP	98	
RIXPTAUT	40	20
RIXPTRPY	40	10
RIXPUTKP	94	
RIXPWD	C	

Table 117. Cross Reference for RIXP (continued)

Name	Offset	Hex Value
RIXPXNDP	80	
RIXRCODE	3C	
RIXREMOT	74	2
RIXSMCNO	4	20
RIXSMCYS	4	0
RIXSRVA	4	
RIXSTANO	4	4
RIXSTART	10	
RIXSUBPL	58	
RIXSUBPP	4	10
RIXSURGU	74	4
RIXTRMDA	44	
RIXTRMID	30	
RIXTRMLV	4C	
RIXTRUST	74	8
RIXUDUSR	74	40
RIXUID	8	
RIXUSRDA	48	
RIXWA	34	
RIXWUIDF	74	
RIXX5PRP	AC	

Chapter 49. RLST: RACROUTE REQUEST=LIST Parameter List (Request Section)

RLST programming interface information

RLST is a programming interface.

RLST heading information

Common name: Request-specific portion of the RACROUTE REQUEST=LIST parameter list

Macro ID: IRRPRLST

DSECT name: RLSTPARM

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Varies with RELEASE= parameter specified

Created by: RACROUTE REQUEST=LIST macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=LIST routine

RLST mapping

Table 118. Structure RLSTPARM

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE		RLSTPARM	LIST parameters
0	(0)	ADDRESS	2	RLSTSPNS(0)	Subpool Specifications:
0	(0)	ADDRESS	1	RLSTPSPN	Profile subpool number
1	(1)	ADDRESS	1	RLSTNSPN	Tree node subpool number
2	(2)	ADDRESS	1	RLSTCODE	Always set to 2
3	(3)	BITSTRING	1	RLSTFLAG	Flags:

Table 118. Structure RLSTPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		11..		RLSTOPT	X'C0' Type of Request: B'00' for CREATE B'10' for DELETE
		..1.		RLSTOWN	X'20' 1 = Add OWNER to access list with ALTER authority
		...1		RLSTLOC	X'10' 1 = LOC=ABOVE specified
		.... 1...		RLSTREL	X'08' 1 = RELEASE=1.8 specified
		.... .1..		RLSTR19	X'04' 1 = RELEASE=1.9 specified
		.... ..1.		RLSTR192	X'02' 1 = RELEASE=1.9.2 specified
		1		*	Reserved
4	(4)	ADDRESS	4	RLSTLIST	Address of resource name list
8	(8)	ADDRESS	4	RLSTACEE	Address of ACEE to use
12	(C)	ADDRESS	4	RLSTINST	Address of installation exit data field
16	(10)	ADDRESS	4	RLSTAPPL	Address of application name
20	(14)	ADDRESS	4	RLSTCLAS	Address of class name
24	(18)	ADDRESS	4	RLSTFLTP	Address of filter string
28	(1C)	CHARACTER	4	RLSTXTND	This extension exists when RELEASE=2.1 or greater is specified
28	(1C)	BITSTRING	1	RLSTFLG2	2nd flag byte
		1111 11..		*	Reserved
		.... ..1.		RLSTR21	X'02' 1 = RELEASE=2.1 specified
		1		RLSTGLBL	X'01' 1 = GLOBAL=YES specified
29	(1D)	CHARACTER	1	*	Reserved for future use
30	(1E)	SIGNED	2	RLSTLEN	Length of entire plist
30	(1E)	..1.	2	RLSTPLEN	End of plist

RLST cross reference

Table 119. Cross Reference for RLST

Name	Offset	Hex Value
RLSTACEE	8	
RLSTAPPL	10	
RLSTCLAS	14	
RLSTCODE	2	
RLSTFLAG	3	

Table 119. Cross Reference for RLST (continued)

Name	Offset	Hex Value
RLSTFLG2	1C	
RLSTFLTP	18	
RLSTGLBL	1C	1
RLSTINST	C	
RLSTLEN	1E	
RLSTLIST	4	
RLSTLOC	3	10
RLSTNSPN	1	
RLSTOPT	3	C0
RLSTOWN	3	20
RLSTPLEN	1E	20
RLSTPSPN	0	
RLSTREL	3	8
RLSTR19	3	4
RLSTR192	3	2
RLSTR21	1C	2
RLSTSPNS	0	
RLSTXTND	1C	

Chapter 50. RLX1P: RACROUTE REQUEST=LIST Exit Parameter List

RLX1P programming interface information

RLX1P is a programming interface.

RLX1P heading information

Common name: RACROUTE REQUEST=LIST exit parameter list

Macro ID: ICHRLX1P

DSECT name: RLX1PL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
229

Key
0

Size: 60 bytes

Created by: RACROUTE REQUEST=LIST processor

Pointed to by: R1 at entry to ICHRLX01

Serialization: None

Function: Contains the list of addresses passed to RACROUTE REQUEST=LIST pre- and postprocessing installation exits

RLX1P mapping

Table 120. Structure RLX1PL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RLX1PL	
0	(0)	ADDRESS	4	RLX1LEN	Number address: points to a fullword containing the number of parameters in this list, including itself.
4	(4)	ADDRESS	4	RLX1FLAG	Flag byte address: points to a 1-byte area with the following format:
		....		RLX1ENVC	B'00000000' 00.. .... ENVIR=CREATE
		1...		RLX1ENVD	B'10000000' 10.. .... ENVIR=DELETE

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		....		RLX1OWNN	B'00000000' ..0. .... OWNER=NO
		..1.		RLX1OWNY	B'00100000' ..1. .... OWNER=YES ...0 0000 Reserved
8	(8)	ADDRESS	4	RLX1FUNC	Points to a 1-byte area with the following format:
		....		RLX1PRE	B'00000000' 0... .... Call is for preprocessing
		1...		RLX1POST	B'10000000' 1... .... Call is for postprocessing
12	(C)	ADDRESS	4	RLX1INST	INSTLN address: points to an area containing the data specified by the INSTLN parameter on the RACLIST macro. This address is 0 if INSTLN was not specified on RACLIST. No system modules specify the INSTLN parameter; it is intended for use by installation-written routines that invoke RACLIST to communicate with the RACLIST pre/postprocessing exit routine.
16	(10)	ADDRESS	4	RLX1CLAS	CLASS address: points to an 8-byte field containing the class name. The class name is left-justified and padded with blanks if necessary.
20	(14)	ADDRESS	4	RLX1ACEE	ACEE address: points to a fullword that contains the address of the ACEE as specified on the RACLIST macro. If the ACEE parameter was not specified on the RACLIST macro, the fullword contains zeroes and RACLIST uses the ACEE pointed to by TCBSENV in the current task control block (TCB) or ASXBSENV in the address space extension block (ASXB).
24	(18)	ADDRESS	4	RLX1APPL	APPL address: points to an 8-byte area containing the application name as specified on the RACLIST macro. If not specified on the RACLIST macro, the 8-byte area contains blanks.

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
28	(1C)	ADDRESS	4	RLX1SUBP	SUBPOOL address: points to a 2-byte area containing subpool information. The first byte identifies the subpool from which the in-storage profile index will be obtained. The second byte identifies the subpool from which the profiles will be obtained. The subpool values are taken from the SUBPOOL parameter on the RACLIST macro instruction. These values can be changed if the exit has been invoked for preprocessing; if the exit is invoked for postprocessing, changes will have no effect, because the storage has already been obtained.
32	(20)	ADDRESS	4	RLX1LIST	LIST address: specifies the address of a fullword containing the address of the list of resource names specified on the RACLIST macro. The fullword contains zeroes if LIST was not specified on the RACLIST macro. The first halfword of the list of resource names contains the number of resource names in the list. This count field is followed by the resource name entries. Each resource name entry consists of a 1-byte length field giving the length of the resource name followed by the resource name itself.

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
36	(24)	ADDRESS	4	RLX1RULE	Rule flags address: points to eleven contiguous 1-byte fields. RACLIST processing builds a working profile for each resource name and, for successive occurrences of the resource name, merges the new information with the information in the working profile. A resource name can appear in more than one resource group and can also have a profile of its own on ths RACF data set. These rule flags determine how conflicts are resolved between multiple occurrences of resource names within resource groups or between resource groups and a resource profile. The fields in each of the eleven flag bytes have the following significance:
		1...		RLX1GGL	B'10000000' 1000 .... Least restrictive when resolving conflicts between occurances in groups. (For example, a profile with UACC=UPDATE would be selected over one with UACC=NONE.)
		.1..		RLX1GGM	B'01000000' 0100 .... Most restrictive when resolving conflicts between occurences in groups. (In this case, UACC=NONE would be selected over UACC=UPDATE.)
		..1.		RLX1GGX	B'00100000' 0010 .... Use value from external profile when resolving conflicts between occurrences in groups.
		...1		RLX1GGW	B'00010000' 0001 .... Use value from working profile when resolving conflicts between occurences in groups. This rule means that the first value encountered is used. Note that for multiple occurrences of a resource name, the order in which they appear is dependent on the alphanumeric sequence of the resource group names and the individual profile name (if any).

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.... 1...		RLX1GIL	B'00001000' .... 1000 Least restrictive when resolving conflicts between groups and an individual occurence.
		.... .1..		RLX1GIM	B'00000100' .... 0100 Most restrictive when resolving conflicts between groups and an individual occurrence.
		.... ..1.		RLX1GIX	B'00000010' .... 0010 Use value from external profile when resolving conflists between groups and an individual occurrence.
		1		RLX1GIW	B'00000001' .... 0001 Use value from working profile when resolving conflicts between groups and an individual occurrence. This rule means that the first value encountered is used. Note that, for multiple occurrences of a resource name, the order in which they appear is dependent on the alphanumeric sequence of the resource group names and the individual profile name (if any). The nine flag bytes are initialized as follows:
		.1.. .1..		RLX1UACC	X'44' UACC: initialized to X'44', meaning use the most restrictive of the profile UACC authorizations.
		.1.. .1..		RLX1AUDF	X'44' AUDIT flags: initialized to X'44', meaning OR the flag bytes. This causes an audit option to be in effect in the final profile if it was in effect in any of the profiles being merged. RACLIST uses the most-encompassing audit qualifiers. If changed to X'88' by the exit, it would mean AND the flag bytes. This causes an audit option to be in effect in the final profile only if it was on in all of the profiles being merged. RACLIST uses the least-encompassing audit qualifiers.
		.1.. .1..		RLX1GLAU	X'44' GLOBALAUDIT flags: initialized to X'44' with meaning and effects identical to AUDIT flags.

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1.. .1..		RLX1RESL	X'44' Resource level: initialized to X'44', meaning use the higher level. If changed to X'88' by the exit, it would mean use the lower level. See the description of the LEVEL operand in the Command Language Reference.
		..1. ..1.		RLX1IDTA	X'22' Installation data: initialized to X'22', meaning use the value from the external profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.)
		..1. ..1.		RLX1ADTA	X'22' Application data: initialized to X'22', meaning use the value from the external profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.)
		1... 1...		RLX1ACCL	X'88'"Access list entries: initialized to X'88', meaning use the least restrictive of the entries.
		..1. ..1.		RLX1OWNX	X'22' OWNER: initialized to X'22', meaning use the value from the external profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.)
		..1. ..1.		RLX1NTFY	X'22' NOTIFY: initialized to X'22', meaning use the value from the external profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.)
		...1 ...1		RLX1TERM	X'11' TERMINAL: initialized to X'11', meaning use the value from the working profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.)

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		...1 ...1		RLX1SLBL	X'11' SECLABEL: initialized to X'11', meaning use the value from the working profile. (Bit settings 1000.... and 0100.... are treated as 0001....; bit settings ....1000 and ....0100 are treated as ....0001.) If a RACF selection exit is active, RACLIST uses the value from the working profile for all fields except the access list entries. For the other values, processing proceeds as if X'11' had been specified. The RACLIST selection can override this processing. See the "RACF Installation Exits" chapter of the SPL: RACF for more information.
40	(28)	ADDRESS	4	RLX1XAMT	Profile expansion amount address: points to a fullword initialized to zeroes, which is the minimum amount of expansion space to provide at the end of the working profile passed to the processing exit. This parameter allows an exit routine to control how much data it can store in the working profiles that RACLIST builds and passes to the selection exit.
44	(2C)	ADDRESS	4	RLX1CODE	Return code address: specifies the address of a fullword to be used as a return code by RACLIST if the exit issues return code 4. On entry to the preprocessing exit, the return code is 0. On entry to the postprocessing exit, the return code is the value RACLIST would return to the caller. Note that this field allows the exit to terminate RACLIST with a 0 or any other return code. Because the exit routine can build resident profiles and an index structure of its own, it might terminate RACLIST in a non-error case with a normal return code.

Table 120. Structure RLX1PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
48	(30)	ADDRESS	4	RLX1WA	Work area address: points to a fullword of zeroes. The exit can use this field for any purpose. Because this field is initialized to zeroes before entry to the pre/postprocessing exit, it can be used for communication between invocations of the exit for preprocessing and postprocessing.
52	(34)	ADDRESS	4	RLX1TREE	Class tree anchor element address: points to a fullword that contains the address of the class tree anchor element that is added to the class tree anchor element chain pointed to by the effective ACEE (see RIXAPPLN field at offset 20). Each class chained off the ACEE has one class tree anchor element, containing the classname for the in-storage profiles and a pointer to the in-storage profile structure. This field has meaning only for the RACLIST postprocessing exit and is 0 if a class tree is not encountered. See the "Data Areas" chapter of the SPL: RACF for the description of the RACF in-storage profile (ISP).
56	(38)	ADDRESS	4	RLX1FLTR	Filter string pointer: points to a one byte length follwed by a filter string. For the format of the filter string see the the description of the RACLIST macro FILTER keyword.

RLX1P cross reference

Table 121. Cross Reference for RLX1P

Name	Offset	Hex Value
RLX1ACCL	24	88
RLX1ACEE	14	
RLX1ADTA	24	22
RLX1APPL	18	
RLX1AUDF	24	44
RLX1CLAS	10	
RLX1CODE	2C	
RLX1ENVC	4	0

Table 121. Cross Reference for RLX1P (continued)

Name	Offset	Hex Value
RLX1ENVD	4	80
RLX1FLAG	4	
RLX1FLTR	38	
RLX1FUNC	8	
RLX1GGL	24	80
RLX1GGM	24	40
RLX1GGW	24	10
RLX1GGX	24	20
RLX1GIL	24	8
RLX1GIM	24	4
RLX1GIW	24	1
RLX1GIX	24	2
RLX1GLAU	24	44
RLX1IDTA	24	22
RLX1INST	C	
RLX1LEN	0	
RLX1LIST	20	
RLX1NTFY	24	22
RLX1OWNN	4	0
RLX1OWNX	24	22
RLX1OWNY	4	20
RLX1POST	8	80
RLX1PRE	8	0
RLX1RESL	24	44
RLX1RULE	24	
RLX1SLBL	24	11
RLX1SUBP	1C	
RLX1TERM	24	11
RLX1TREE	34	
RLX1UACC	24	44
RLX1WA	30	
RLX1XAMT	28	

Chapter 51. RLX2P: RACROUTE REQUEST=LIST Selection Exit Parameter List

RLX2P programming interface information

RLX2P is a programming interface.

RLX2P heading information

Common name: RACROUTE REQUEST=LIST selection exit parameter list

Macro ID: ICHRLX2P

DSECT name: RLX2PL

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
229

Key
0

Size: 120 bytes

Created by: RACROUTE REQUEST=LIST processor

Pointed to by: R1 at entry to ICHRLX02

Serialization: None

Function: Contains the list of addresses passed to RACROUTE REQUEST=LIST selection installation exit

RLX2P mapping

Table 122. Structure RLX2PL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RLX2PL	
0	(0)	ADDRESS	4	RLX2LEN	Number address: points to a fullword containing the number of parameters in this list, including itself.

Table 122. Structure RLX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	RLX2PRPA	Preparameters address: address of the parameter list passed to the RACLIST preprocessing exit routine. This parameter is passed to allow communication between the RACLIST pre/postprocessing exit routines and the RACLIST selection exit routine, because processing logic in one exit routine might require corresponding processing in the other exit routine.
8	(8)	ADDRESS	4	RLX2FLAG	Flag byte address: points to a 1-byte area with the following format:
		1...		RLX2NPRV	B'10000000' 1... .... The resource was not encountered
		....		RLX2PRVE	B'00000000' 0... .... The resource was encountered
		....		RLX2DGRP	B'00000000' .0.. .... Data comes from a group profile
		.1..		RLX2DRES	B'01000000' .1.. .... Data comes from a resource profile ..00 0000 Reserved
		..1.		RLX2MAPN	B'00100000' ..1. .... Indicates the presence of APAR OA49204. If the bit is "ON", RLX2RNAM points to a 1-byte length field followed by the name of the resource that RACLIST is currently processing.
12	(C)	ADDRESS	4	RLX2RNAM	Resource name address: points to a 1-byte length field followed by the name of the resource that RACLIST is currently processing.
16	(10)	ADDRESS	4	RLX2RGRP	Resource group name address: points to a 1-byte length field followed by the name of the resource group from which the current resource name was selected. This address is 0 if the resource name is not from a resource group. The exit should not change this value. Do not confuse the name of the resource being processed with the name of the resource group from which it was selected.

Table 122. Structure RLX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	RLX2RNCL	Resource name class address: address of an 8-byte class name for the currently selected resource. The exit should not change this value.
24	(18)	ADDRESS	4	RLX2RGCA	Resource group class address: address of an 8-byte class name for the resource group from which the current resource was selected. This address is 0 if the resource profile was not built because it is a member of a resource group. The exit should not change this value. Do not confuse the class of the resource being processed with the class of the resource group from which the resource was selected.
28	(1C)	ADDRESS	4	RLX2UACC	UACC address: points to a 1-byte field containing the universal access flags from the resource profile. The possible values are:
		1...		RLX2ALTE	X'80' - ALTER
		.1..		RLX2CONT	X'40' - CONTROL
		..1.		RLX2UPDA	X'20' - UPDATE
		...1		RLX2READ	X'10' - READ
		1		RLX2NONE	X'01' - NONE
32	(20)	ADDRESS	4	RLX2AUD	Audit flag address: points to a 1-byte field containing the audit indicators and qualifiers from the resource profile.
36	(24)	ADDRESS	4	RLX2GLAU	Global audit flags address: points to a 1-byte field containing the global audit indicators and qualifiers from the resource profile.
40	(28)	ADDRESS	4	RLX2RLVL	Resource level number address: points to a 1-byte field containing the LEVEL value from the resource profile.
44	(2C)	ADDRESS	4	RLX2DATA	DATA address: points to a 1-byte field followed by the installation data that was specified on the RDEFINE or RALTER command for the resource. The length field is zero if no data is present.

Table 122. Structure RLX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
48	(30)	ADDRESS	4	RLX2ACCL	Access list address: points to the access list retrieved from the source profile. The first two bytes of the list contain the number of entries in the list. Each entry is nine bytes long: an 8-character userid or group name followed by a 1-byte access authority. The possible values for the access authority are: X'80' - ALTER X'40' - CONTROL X'20' - UPDATE X'10' - READ X'01' - NONE
52	(34)	ADDRESS	4	RLX2ANCH	Profile anchor address: points to a working copy of the profile. On the first encounter with a resource, the profile is filled in with the data taken from the external profile, which is also passed in the preceding five parameters. On subsequent encounters with the resource, the profile is not updated to reflect the data taken from the external profile. It is the responsibility of the exit to modify, if desired, the UACC, audit, global audit, resource level, installation, and application data fields. These modifications have the effect of propagating the first value encountered. On return from this exit, RACF merges access lists according to the value of the rule flags for access list entries.
56	(38)	ADDRESS	4	RLX2OWNN	Owner name address: points to an 8-byte field containing the owner value from the resource profile. If OWNER=YES was specified on the RACLIST request, the owner field has been added to the access list (pointed to by offset 48 (X'30 or label RLX2ACCL)) with ALTER authority. Once the owner is added to the access list, this information is treated as if it were originally part of the list. This parameter allows the exit to selectively override the effect of the OWNER parameter.

Table 122. Structure RLX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
60	(3C)	ADDRESS	4	RLX2APPL	APPLDATA address: points to a 1-byte length field followed by the application data that was specified on the RDEFINE or RALTER command for the resource. The length field is zero if no data is present. RACLIST does not use this field in its processing. This field is intended for use by installation-written routines managing installation-defined resource classes and resources, to provide additional information to the exit routine.
64	(40)	ADDRESS	4	RLX2NTFY	NOTIFY address: points to an 8-byte area containing the userid of the user to be notified when RACF detects an unauthorized attempt to access a resource protected by this profile.
68	(44)	ADDRESS	4	RLX2SLVL	SECLEVEL address: points to an 2-byte length field followed by the SECLEVEL value from the resource profile.
72	(48)	ADDRESS	4	RLX2CTGY	CATEGORIES address: points to the address of a 2-byte field that contains the length of the 2-byte count of categories contained in the second field. The third 2-byte field is the length of the list of categories. The last field is the list of categories from the resource profile.
76	(4C)	ADDRESS	4	RLX2TERM	TERMINAL address: points to an area containing data for logon times allowed a terminal.
80	(50)	ADDRESS	4	RLX2WARN	WARNING address: points to an 2-byte field containing the warning flag from the resource profile. Possible values are: X'80' - Warning in effect X'00' - Warning not in effect
84	(54)	ADDRESS	4	RLX2KEYP	SESSION KEY address: points to an 8-byte area containing the session key used for establishing a session level security session.

Table 122. Structure RLX2PL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
88	(58)	ADDRESS	4	RLX2SLSF	SESSION LEVEL SECURITY FLAG address: points to a 1-byte field containing the session level security flag with the following format:
		1...		RLX2SLCK	B'10000000' 1... .... The resource has been locked to prevent a session from being established. .111 1111 Reserved
92	(5C)	ADDRESS	4	RLX2LKYD	LAST CHANGE DATE address: points to a 4-byte area containing the last change date for the session key.
96	(60)	ADDRESS	4	RLX2KYIN	SESSION KEY INTERVAL address: points to a 2-byte area containing the maximum number of days for which the key is valid.
100	(64)	ADDRESS	4	RLX2MXFL	MAXIMUM FAILURES address: points to a 2-byte area containing the maximum number of failed attempts permitted before the resource is locked.
104	(68)	ADDRESS	4	RLX2SLBL	SECLABEL address: points to an 8-byte area containing the SECLABEL of the resource profile.
108	(6C)	ADDRESS	4	RLX2ACL2	SECOND ACCCESS LIST address: Points to a variable length field containing a 2 byte length followed by the second access list entries. Each entry in the second access list conforms to the following structure: 8 Byte Program name of which the first byte is considered a flag byte. 8 Byte User or Group Id. 1 Byte Access Authority. 2 Byte Count Field. 1 Byte Length Field describing the length of the following. 8 Byte Class Id. 2 Byte Reserved. 1 Byte Length Field describing the variable legth data.
112	(70)	ADDRESS	4	RLX2AL2C	SECOND ACCESS LIST COUNT address: Points to a 2 byte field containing the number of entries in the second access list.
116	(74)	ADDRESS	4	RLX2SCON	Conversation Security

RLX2P cross reference

Table 123. Cross Reference for RLX2P

Name	Offset	Hex Value
RLX2ACCL	30	
RLX2ACL2	6C	
RLX2ALTE	1C	80
RLX2AL2C	70	
RLX2ANCH	34	
RLX2APPL	3C	
RLX2AUD	20	
RLX2CONT	1C	40
RLX2CTGY	48	
RLX2DATA	2C	
RLX2DGRP	8	0
RLX2DRES	8	40
RLX2FLAG	8	
RLX2GLAU	24	
RLX2KEYP	54	
RLX2KYIN	60	
RLX2LEN	0	
RLX2LKYD	5C	
RLX2MAPN	8	
RLX2MXFL	64	
RLX2NONE	1C	1
RLX2NPRV	8	80
RLX2NTFY	40	
RLX2OWNN	38	
RLX2PL	0	
RLX2PRPA	4	
RLX2PRVE	8	0
RLX2READ	1C	10
RLX2RGCA	18	
RLX2RGRP	10	
RLX2RLVL	28	
RLX2RNAM	C	
RLX2RNCL	14	
RLX2SCON	74	
RLX2SLBL	68	

Table 123. Cross Reference for RLX2P (continued)

Name	Offset	Hex Value
RLX2SLCK	58	80
RLX2SLSF	58	
RLX2SLVL	44	
RLX2TERM	4C	
RLX2UACC	1C	
RLX2UPDA	1C	20
RLX2WARN	50	

Chapter 52. RNG: RACF Database Range Table

RNG programming interface information

RNG is **NOT a programming interface.**

RNG heading information

Common name: RACF database range table

Macro ID: None

DSECT name: None

Owning component: Resource Access Control Facility (XXH00)

Eye-catcher ID: None

Storage attributes:

Subpool
LPA

Key
0

Size: 4 bytes plus 45 bytes per range (49 bytes minimum)

Created by: RACF initialization based on RACF defaults or installation-provided ICHRRNG module

Pointed to by: RCVTRNGP field of the RCVT data area

Serialization: None

Function: Describes the alphabetic range of profiles contained in each RACF database

RNG mapping

Table 124. Structure RNGPARD

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	ICHPRNG	RANGE TABLE MAPPING
0	(0)	SIGNED	4	RNGNUM	NUMBER OF ARRAY ELEMENTS
4	(4)	CHARACTER	45	RNGVALS (*)	ARRAY OF RANGE/DS-NUMBER PAIRS
4	(4)	CHARACTER	44	RNGSTART	LOWER BOUND OF RANGE
48	(30)	UNSIGNED	1	RNGDSNUM	DATA SET SEQUENCE NUMBER

Chapter 53. RRPF: Resident Profile Map

RRPF programming interface information

RRPF is a programming interface.

RRPF heading information

Common name: Resident profile map

Macro ID: ICHRRPF

DSECT name: RRPF, DSPVOLS, DSPACCES, DSPINSTD, DSPDPTD, DSP2ACCS

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes: When a CSA profile is requested:

Subpool
231

Key
0

When a private profile is requested:

Subpool
229

Key
0

Size:

Section	Size
1	136 bytes
2	2 bytes plus an unknown number of 6-byte fields at offset 2
3	2 bytes plus an unknown number of 9-byte fields at offset 2
4	2 bytes plus a variable of unknown length at offset 2
5	2 bytes plus an unknown number of 2-byte fields at offset 2
6	35 bytes plus a variable of unknown length at offset 35
7	2 bytes plus a variable of unknown length at offset 2
8	2 bytes plus a variable of unknown length at offset 2

Created by: RACROUTE REQUEST=AUTH processing when CSA or private option is specified

Pointed to by: ACEEAMP field of the ACEE data area or returned in Register 1 after RACROUTE REQUEST=AUTH request

Serialization: None

Function: Maps a profile for general resource used for authorization checking

RRPF mapping

Table 125. Structure RRPF

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	140	RRPF	RESOURCE PROFILE ELEMENT
0	(0)	UNSIGNED	4	DSPCORE	
0	(0)	UNSIGNED	1	RRPSP	AREA SUBPOOL NUMBER
1	(1)	ADDRESS	3	RRPLEN	TOTAL AREA LENGTH
4	(4)	CHARACTER	132	RRPVDATA	PROFILE DATA
4	(4)	CHARACTER	132	DSPSUB	
4	(4)	CHARACTER	44	DSPDSNM	RESOURCE NAME This name is also located in new structure below. This mapping maintained for compatibility for earlier releases
48	(30)	BITSTRING	1	DSPUACC	UNIVERSAL ACCESS
49	(31)	BITSTRING	1	DSPAUDIT	AUDIT FLAGS
50	(32)	BITSTRING	1	DSPTYPE	D.S. TYPE FLAGS
		1...		DSPTP	1 VSAM, 0 NON-VS
		.1..		DSPMDL	1 MODEL
		..1.		DSPTAPE	1 TAPE
		...1 1111		*	Reserved
51	(33)	ADDRESS	1	DSPLEVEL	RESOURCE LEVEL
52	(34)	SIGNED	4	DSPVOLOF	OFFSET TO VOLSER LIST
56	(38)	SIGNED	4	DSPACCOF	OFFSET TO ACCESS LIST
60	(3C)	CHARACTER	8	DSPCLASS	RESOURCE CLASS
68	(44)	BITSTRING	1	DSPGAUD	GLOBAL AUDIT FLG
69	(45)	UNSIGNED	1	DSPVRSN	VERSION = 1
70	(46)	BITSTRING	1	DSPWARN	WARNING FLAG - BIT 7 BIT 7 = 1 - RESOURCE HAS WARNING ATTRIBUTE
71	(47)	BITSTRING	1	DSPEOS	ERASE-ON-SCRATCH FLAG - BIT 0=1 - DATASET WILL BE ERASED WHEN SCRATCHED
72	(48)	SIGNED	4	DSPINST	OFFSET TO INSTALLATION DATA
76	(4C)	ADDRESS	4	DSPNEXTP	ADDR NEXT MODEL

Table 125. Structure RRPF (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
80	(50)	BITSTRING	1	DSPFNF	MODEL FOUND INDICATOR
81	(51)	UNSIGNED	1	DSPSLVL	SECURITY LEVEL
82	(52)	SIGNED	2	DSPRTPD	RETENTION PERIOD
84	(54)	CHARACTER	8	DSPOWNER	MODEL OWNER
92	(5C)	CHARACTER	8	DSPNOTFY	USERID TO NOTIFY WHEN THIS PROFILE DENIES ACCESS
100	(64)	SIGNED	4	DSPDPTOF	OFFSET TO CATEGORY LIST
104	(68)	SIGNED	4	DSPPGMOF	OFFSET TO CONDITIONAL ACCESS LIST
108	(6C)	BITSTRING	1	DSPRESF	RESOURCE FLAG (ONLY FOR TAPE VOLUMES - BIT 0 = 1 VOLUME MAY ONLY CONTAIN ONE DATA SET - BIT 1 = 1 VOLUME CAN CONTAIN A TVTOC)
109	(6D)	BITSTRING	1	DSPTDAYS	DAYS THAT A TERMINAL MAY NOT BE USED (BIT 0 SUNDAY BIT 1 MONDAY,....)
110	(6E)	CHARACTER	3	DSPLOGNT	EARLIEST TIME A TERMINAL MAY BE USED (HHMM)
113	(71)	CHARACTER	3	DSPLOGFT	LATEST TIME A TERMINAL MAY BE USED (HHMM)
116	(74)	CHARACTER	3	DSPTZONE	TIME OFFSET OF TERMINAL FROM CPU (+ = EAST, - = WEST).
119	(77)	BITSTRING	1	DSPFLAGS	Flags
		1...		DSPGEN	1 = DATASET profile is generic (note this bit is not on if a generic entity name is specified)
		.1..		DSPGRPDS	1 = DATASET high level qualifier is a group
120	(78)	CHARACTER	8	DSPSLABL	Security label
128	(80)	CHARACTER	4	DSPDSNBF(0)	Character form of offset to resource name
128	(80)	SIGNED	4	DSPDSNOF	Offset to resource name in extended format
132	(84)	CHARACTER	4	DSPAPOFF(0)	Character form of offset to application data only if class is RACLISTed or the profile is a model pointed to by ACEEAMP
132	(84)	SIGNED	4	DSPAPPOF	Offset to application data only if class is RACLISTed or the profile is a model pointed to by ACEEAMP
136	(88)	SIGNED	4	DSPIPOFF	Offset to IP lookup value

Table 125. Structure RRPF (continued)

Offset	Offset	Type	Len	Name(Dim)	Description
Dec	Hex				
DSPVOLS					
0	(0)	STRUCTURE	*	DSPVOLS	VOLSER LIST
0	(0)	UNSIGNED	2	DSPVOLCT	NUMBER OF ENTRIES
2	(2)	CHARACTER	1	DSPVOLSR	VOLSERS
DSPACCES					
0	(0)	STRUCTURE	*	DSPACCES	ACCESS LIST
0	(0)	UNSIGNED	2	DSPACT	NUMBER OF ENTRIE
2	(2)	CHARACTER	9	DSPACCLE	ACCESS LIST ENTRIES
2	(2)	CHARACTER	8	DSPAUSER	USERID/GRPNAME
10	(A)	BITSTRING	1	DSPACS	ACCESS AUTHORITY
DSPINSTD					
0	(0)	STRUCTURE	*	DSPINSTD	INSTALLATION DATA
0	(0)	SIGNED	2	DSPLINST	LENGTH OF INSTALLATION DATA
2	(2)	CHARACTER	*	DSPIDATA	INSTALLATION DATA
DSPDPTD					
0	(0)	STRUCTURE	*	DSPDPTD	CATEGORY LIST
0	(0)	SIGNED	2	DSPDPTCT	NUMBER OF CATEGORIES
2	(2)	SIGNED	2	DSPDEPT	CATEGORY VALUE
DSP2ACCS					
0	(0)	STRUCTURE	*	DSP2ACCS	Second Access List
0	(0)	UNSIGNED	2	DSP2GCT	Entry count
2	(2)	UNSIGNED	2	DSP2GLN	Access List Length
4	(4)	CHARACTER	20	DSP2ACCL(0)	Access list Entries
4	(4)	CHARACTER	8	DSP2ENT(0)	Resource name or flag
4	(4)	CHARACTER	1	DSPPGFLG	Flag byte
5	(5)	CHARACTER	7	DSPA2RST	Rest of name or flag
12	(C)	CHARACTER	8	DSP2USR	USER/GROUP ID
20	(14)	BITSTRING	1	DSP2ACS	ACCESS AUTHORITY
21	(15)	UNSIGNED	2	DSP2GACS	Access Count
23	(17)	UNSIGNED	1	DSP2GVRL	Variable Area Length
24	(18)	CHARACTER	*	DSP2GVAR	Variable Area
24	(18)	CHARACTER	8	DSP2CLID	Class ID
32	(20)	CHARACTER	2	DSP2RSVD	Reserved
34	(22)	UNSIGNED	1	DSP2VENL	Entity Length
35	(23)	CHARACTER	*	DSP2VENT	Entity

Table 125. Structure RRPF (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
DSPBUF					
0	(0)	STRUCTURE	*	DSPBUF	Resource name in extended format
0	(0)	CHARACTER	2	DSPDLEN	Character form of resource length
0	(0)	SIGNED	2	DSPDSNML	Resource length
2	(2)	CHARACTER	*	DSPDSNME	Resource name
DSPAPPL					
0	(0)	STRUCTURE	*	DSPAPPL	Structure of the application data.
0	(0)	SIGNED	2	DSPAPPLN	Length of the application data.
2	(2)	CHARACTER	*	DSPAPLDT	Application Data
DSPILOOK					
0	(0)	STRUCTURE	0	DSPILOOK	IP lookup data structure
0	(0)	CHARACTER	16	DSPIPLOK	IP lookup value for SERVAUTH class profiles

RRPF constants

Table 126. Constants for RRPF

Len	Type	Value	Name	Description
1	DECIMAL	0	DSPA2DAT	Conditional data is present.
1	DECIMAL	0	DSPVR00	Version 0 profile present.
1	DECIMAL	1	DSPVR01	Version 1 profile present.
1	DECIMAL	1	DSPCURV	Version 1 profile is current version.

RRPF cross reference

Table 127. Cross Reference for RRPF

Name	Offset	Hex Value
DSPACCES	0	
DSPACCLE	2	
DSPACCOF	38	
DSPACS	A	
DSPACT	0	
DSPAPLDT	2	
DSPAPOFF	84	
DSPAPPL	0	
DSPAPPLN	0	

Table 127. Cross Reference for RRPF (continued)

Name	Offset	Hex Value
DSPAPPOF	84	
DSPIPOFF	88	
DSPAUDIT	31	
DSPAUSER	2	
DSPA2RST	5	
DSPBUF	0	
DSPCLASS	3C	
DSPCORE	0	
DSPDEPT	2	
DSPDLEN	0	
DSPDPTCT	0	
DSPDPTD	0	
DSPDPTOF	64	
DSPDSNBF	80	
DSPDSNM	4	
DSPDSNME	2	
DSPDSNML	0	
DSPDSNOF	80	
DSPEOS	47	
DSPFNF	50	
DSPGAUD	44	
DSPIDATA	2	
DSPINST	48	
DSPINSTD	0	
DSPIPLOK	0	
DSPLEVEL	33	
DSPLINST	0	
DSPLOGFT	71	
DSPLOGNT	6E	
DSPMDL	32	40
DSPNEXTP	4C	
DSPNOTFY	5C	
DSPOWNER	54	
DSPPGFLG	4	
DSPPGMOF	68	
DSPRESF	6C	

Table 127. Cross Reference for RRPF (continued)

Name	Offset	Hex Value
DSPRTPD	52	
DSPSLABL	78	
DSPSLVL	51	
DSPSUB	4	
DSPTAPE	32	20
DSPTDAYS	6D	
DSPTP	32	80
DSPTYPE	32	
DSPTZONE	74	
DSPUACC	30	
DSPVOLCT	0	
DSPVOLOF	34	
DSPVOLS	0	
DSPVOLSR	2	
DSPVRSN	45	
DSPWARN	46	
DSP2ACCL	4	
DSP2ACCS	0	
DSP2ACS	14	
DSP2CLID	18	
DSP2ENT	4	
DSP2GACS	15	
DSP2GCT	0	
DSP2GLN	2	
DSP2GVAR	18	
DSP2GVRL	17	
DSP2RSVD	20	
DSP2USR	C	
DSP2VENL	22	
DSP2VENT	23	
RRPF	0	
RRPLEN	1	
RRPSP	0	
RRPVDATA	4	

Chapter 54. RSMXP: RACF Report Writer Selection Exit Parameter List

RSMXP programming interface information

RSMXP is a programming interface.

RSMXP heading information

Common name:	RACF report writer selection exit parameter list
Macro ID:	ICHRSMXP
DSECT name:	RSMXPL
Owning component:	Resource Access Control Facility (XXH00)
Eye-catcher ID:	None
Storage attributes:	**Subpool** Varies **Key** 8
Size:	24 bytes
Created by:	RACF report writer
Pointed to by:	R1 at entry to ICHRSMFE
Serialization:	None
Function:	Contains the list of addresses passed to the RACF report writer installation exit ICHRSMFE

RSMXP mapping

Table 128. Structure RSMPL

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	RSMPL	
0	(0)	ADDRESS	4	RSMCALLR	Reason code address: points to a fullword containing the reason for this call:
		....		RSMPRES	X'00000000' - Prerecord selection
		.... .1..		RSMPOSTS	X'00000004' - Postrecord selection
		.... 1...		RSMEOF	X'00000008' - End-of-file

Table 128. Structure RSMPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
4	(4)	ADDRESS	4	RSMDSTRA	Data string address: points to an area having the following format: 0 4 Address of DATA string 4 2 Length of DATA string 6 1 1-byte flag field. If bit 0 is on, DATA was specified on the RACFRW command.
8	(8)	ADDRESS	4	RSMSWTCH	Switch address: points to a 1-byte field containing switches indicating whether the RACF report writer will select or reject the record, based on the existing selection/ rejection criteria. The format of these switches is:
		....		RSMSLECT	B'00000000' 0... .... If 0, the record is selected. For RACF records with reason code = 0, this bit is 0 because no selection criterion has yet been applied.
		1...		RSMRJECT	B'10000000' 1... .... If 1, the record is rejected. For non-RACF records, this bit is 1 because non-RACF records are passed to this exit routine for inspection only.
		....		RSMREFRS	B'00000000' .0.. .... If 0, the record is a reformatted RACF SMF record. For more information about these records, see "Reformatted RACF SMF Records" in Chapter 11 of SPL: RACF.
		.1..		RSMNONRS	B'01000000' .1.. .... If 1, the record is a non-RACF SMF record. ..00 0000 Reserved.
12	(C)	ADDRESS	4	RSMRCD	Record address: points to the non-RACF SMF record or the reformatted RACF SMF record under inspection. For reason code X'00000008', this address is set to zero.
16	(10)	ADDRESS	4	RSMPRDCB	SYSPRINT DCB address: points to an area containing the SYSPRINT DCB that has been opened. The SYSPRINT DCB parameters are: DSORG=PS, RECFM=FA, MACRF=PM, and LRECL=133.

Table 128. Structure RSMPL (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	RSMCOMM	Communication area address: points to a fullword communication area that can be used by the exit routine. Initially, this field is set to zero.

RSMXP cross reference

Table 129. Cross Reference for RSMXP

Name	Offset	Hex Value
RSMCALLR	0	
RSMCOMM	14	
RSMDSTRA	4	
RSMEOF	0	8
RSMNONRS	8	40
RSMPOSTS	0	4
RSMPRDCB	10	
RSMPRES	0	0
RSMRCD	C	
RSMREFRS	8	0
RSMRJECT	8	80
RSMSLECT	8	0
RSMSWTCH	8	

Chapter 55. RUTKN: Resource/User Security Token

RUTKN programming interface information

RUTKN is a programming interface.

RUTKN heading information

Common name: Resource/user security token

Macro ID: ICHRUTKN

DSECT name: TOKEN

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by user

Key
Determined by user

Residency
Determined by user

Size: 80 bytes

Created by: RACROUTE REQUEST=TOKENBLD, VERIFY, or VERIFYX

Pointed to by: ACEETOKP. Also returned as an output parameter from RACROUTE REQUEST=TOKENBLD, VERIFY, or VERIFYX

Serialization: None

Function: Maps the RACF resource security token and the RACF user security token

RUTKN mapping

Table 130. Structure TOKEN

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	TOKEN	TOKPTR UTOKEN / RTOKEN MAPPING
0	(0)	BITSTRING	1	TOKLEN	UTOKEN / RTOKEN LENGTH
1	(1)	BITSTRING	1	TOKVERS	UTOKEN / RTOKEN VERSION #
2	(2)	BITSTRING	1	TOKFLG1	MISCELLANEOUS FLAGS
		1...		TOKENCR	Set to '1'B if the token is in an internal format (masked) and set to '0'B (external format) if it is not. TOKENCR must be set correctly regardless of the format of the token (internal or external).

Table 130. Structure TOKEN (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		.1..		*	Reserved
		..1.		TOKLT19	TOKEN CREATED BY PRE RACF 1.9 CALL
		...1		TOKVXPRP	VERIFYX PROPAGATION OCCURRED
		.... 1...		TOKUNUSR	NJE UNKNOWN USER
		.... .1..		TOKLOGU	LOG USER INDICATOR
		.... ..1.		TOKRSPEC	RACF SPECIAL INDICATOR
		1		*	Reserved
3	(3)	UNSIGNED	1	TOKSTYP	SESSION TYPE, DEFINED BELOW
4	(4)	BITSTRING	1	TOKFLG2	MISCELLANEOUS FLAGS
		1...		TOKDFLT	DEFAULT TOKEN
		.1..		TOKUDUS	UNDEFINED USER
		..1.		*	Reserved
		...1		TOKERR	TOKEN IN ERROR
		.... 1...		TOKTRST	PART OF TRUSTED COMPUTER BASE
		.... .1..		TOKSUS	SURROGATE USERID
		.... ..1.		TOKREMOT	REMOTE JOB INDICATOR
		1		TOKPRIV	PRIVILEDGED USER INDICATOR
5	(5)	UNSIGNED	1	TOKPOEX	PORT OF ENTRY CLASS INDEX
6	(6)	BITSTRING	1	TOKFLG3	MISCELLANEOUS FLAGS
		1...		TOKDGRP	DEFAULT GROUP ASSIGNED
		.1..		TOKDSEC	DEFAULT SECLABEL ASSIGNED
		..1.		TOKNETF	Network name specified
		...1		TOKIPV	IP Value present for SERVAUTH POE
		.... 1...		TOKWDWN	If "ON", when MLS is Active, Write-Down is allowed
		...1 1111		*	Reserved
7	(7)	CHARACTER	1	*	Reserved
8	(8)	CHARACTER	8	TOKSCL	SECLABL
16	(10)	CHARACTER	8	TOKXNOD	EXECUTION NODE
24	(18)	CHARACTER	8	TOKSUSR	SUBMITTING USERID
32	(20)	CHARACTER	8	TOKSNOD	SUBMITTER NODE
40	(28)	CHARACTER	8	TOKSGRP	SUBMITTING GROUPID

Table 130. Structure TOKEN (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
48	(30)	CHARACTER	8	TOKPOE	PORT OF ENTRY(CONS ID,TERM. ID)
56	(38)	CHARACTER	8	TOKNETW	Remote network name, if TOKNETF is on
64	(40)	CHARACTER	8	TOKUSER	SESSION OWNER USERID
72	(48)	CHARACTER	8	TOKGRUP	SESSION OWNER GROUPID

RUTKN constants

Table 131. Constants for RUTKN

Len	Type	Value	Name	Description
TOKSTYP SESSION TYPE DEFINITIONS				
1	DECIMAL	1	TOKSAS	SYSTEM ADDRESS SPACE
1	DECIMAL	2	TOKCMND	COMMAND
1	DECIMAL	3	TOKCONS	CONSOLE OPERATOR
1	DECIMAL	4	TOKSTP	STARTED PROCEDURE
1	DECIMAL	5	TOKMNT	MOUNT
1	DECIMAL	6	TOKTSO	TSO LOGON
1	DECIMAL	7	TOKBCH	INTERNAL READER BATCH JOB
1	DECIMAL	8	TOKXBM	INTERNAL READER EXECUTION BATCH MONITOR
1	DECIMAL	9	TOKRJE	RJE OPERATOR
1	DECIMAL	10	TOKNJE	NJE OPERATOR
1	DECIMAL	11	TOKNJEUS	VERIFYX UNKNOWN USER ID TOKEN
1	DECIMAL	12	TOKEBCH	EXTERNAL READER BATCH JOB
1	DECIMAL	13	TOKRBCH	RJE BATCH JOB
1	DECIMAL	14	TOKNBCH	NJE BATCH JOB
1	DECIMAL	15	TOKNSYS	NJE SYSOUT
1	DECIMAL	16	TOKEXBM	EXTERNAL XBM
1	DECIMAL	17	TOKRXBM	RJE XBM
1	DECIMAL	18	TOKNXBM	NJE XBM
1	DECIMAL	19	TOKAPPC	APPC SESSION
1	DECIMAL	20	TOKOSRV	OMVSSRV SESSION
1	DECIMAL	21	TOKIP	IP SESSION
1	DECIMAL	21	TOKLSESS	LAST CURRENTLY DEFINED SESSION

Table 131. Constants for RUTKN (continued)

Len	Type	Value	Name	Description
TOKPOEX CLASS INDEX DEFINITIONS				
1	DECIMAL	1	TOKTERM	TERMINAL CLASS
1	DECIMAL	2	TOKCON	CONSOLE CLASS
1	DECIMAL	3	TOKJESI	JESINPUT CLASS
1	DECIMAL	4	TOKAPORT	APPCPORT CLASS
1	DECIMAL	5	TOKSERV	SERVAUTH CLASS
1	DECIMAL	5	TOKPLAST	LAST CLASS DEFINITION
TOKVERS VERSION LEVEL DEFINITIONS				
1	DECIMAL	1	TOKVER01	VERSION 1 TOKEN
1	DECIMAL	1	TOKCVER	LAST CURRENTLY DEFINED VERSION

RUTKN cross reference

Table 132. Cross Reference for RUTKN

Name	Offset	Hex Value
TOKDFLT	4	80
TOKDGRP	6	80
TOKDSEC	6	40
TOKEN	0	
TOKENCR	2	80
TOKERR	4	10
TOKFLG1	2	
TOKFLG2	4	
TOKFLG3	6	
TOKGRUP	48	
TOKIPV	6	10
TOKLEN	0	
TOKLOGU	2	04
TOKLT19	2	20
TOKNETF	6	20
TOKNETW	38	
TOKPOE	30	
TOKPOEX	5	
TOKPRIV	4	01
TOKREMOT	4	02
TOKRSPEC	2	02

Table 132. Cross Reference for RUTKN (continued)

Name	Offset	Hex Value
TOKSCL	8	
TOKSGRP	28	
TOKSNOD	20	
TOKSTYP	3	
TOKSUS	4	04
TOKSUSR	18	
TOKTRST	4	08
TOKUDUS	4	40
TOKUNUSR	2	08
TOKUSER	40	
TOKVERS	1	
TOKVXPRP	2	10
TOKWDWN	6	08
TOKXNOD	10	

Chapter 56. RXTL: RACROUTE REQUEST=EXTRACT Parameter List (Request Section)

RXTL programming interface information

RXTL is a programming interface.

RXTL heading information

Common name:	Request-specific portion of the RACROUTE REQUEST=EXTRACT parameter list
Macro ID:	IRRPRXTL
DSECT name:	None
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	None
Storage attributes:	**Subpool** Determined by caller **Key** Determined by caller **Residency** Determined by caller
Size:	Varies
Created by:	RACROUTE REQUEST=EXTRACT macro
Pointed to by:	Address of SAFP plus offset in SAFPRACP
Serialization:	None
Function:	Maps the request-specific portion of the RACROUTE REQUEST=EXTRACT parameter list

RXTL mapping

Table 133. Structure EXTLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	12	EXTLIST	EXTRACT parametr list
0	(0)	SIGNED	2	EXTLEN	Parameter list length
2	(2)	BITSTRING	1	EXTFUNCT	Function code = x'82'
3	(3)	UNSIGNED	1	EXTTYPE	Request type 1=Extract 2=Encrypt 3=Extractn 4=Replace 5=Envrxtr
4	(4)	UNSIGNED	1	EXTVER	Version number 0 or 1

Table 133. Structure EXTLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
5	(5)	BITSTRING	1	EXTFLAGS	Flag byte
		1...		EXTBRNCH	Branch entry requested
		.1..		EXTENX	0 => Entity is specified 1 => Entityx is specified
		..1.		EXTPROP	1 => RACROUTE REQUEST=EXTRACT issued as a result of automatic direction
		...1 1111		*	Reserved
6	(6)	SIGNED	2	EXTOFF	Offset to variable part of list
8	(8)	ADDRESS	4	EXTENT	Address of ENTITY
8	(8)	ADDRESS	4	EXTENTX	Address of ENTITYX
12	(C)	CHARACTER		EXTEND	End of fixed part of parm
EXTEXT					
12	(C)	STRUCTURE	0	EXTEXT	TYPE=Extract parm list for release 1.6 and 1.7
12	(C)	ADDRESS	4	EXTCLAS	Address of CLASS
16	(10)	SIGNED	4	EXTSP	Subpool for workarea
20	(14)	ADDRESS	4	EXTFLDS	Address of data to be extracted Data prefixed by 4-byte length
24	(18)	CHARACTER	4	EXTEND1	End of fixed part of parm
EXTENB					
24	(18)	STRUCTURE	0	EXTENB	
24	(18)	ADDRESS	4	EXTSEGM	Address of SEGMENT
28	(1C)	ADDRESS	4	EXTSEGDT	Address of SEGDATA
32	(20)	ADDRESS	4	EXTACEE	Address of ACEE
36	(24)	ADDRESS	4	EXTVOL	Address of VOLSER
40	(28)	BITSTRING	4	EXTSPR	Special processing flags
				EXTRES1	Reserved
		.... 1...		EXTMATCH	Match entity to generic
		.... .1..		EXTGEN	GENERIC flag
		.... ..1.		EXTDRV	DFP flag
		1		EXTFLAC	FLDACC flag
44	(2C)	SIGNED	4	EXTENDX	End of fixed part of parameter
EXTENC					
12	(C)	STRUCTURE	8	EXTENC	TYPE=ENCRYPT parameter list
12	(C)	ADDRESS	4	EXTDATA	Address of data to be encrypted. Data prefixed by 1-byte length

Table 133. Structure EXTLIST (continued)

Offset	Offset	Type	Len	Name(Dim)	Description
Dec	Hex				
16	(10)	SIGNED	4	EXTMETH	Encryption method: 1 = RACF DES method 2 = RACF hashing method 3 = installation defined method 4 = NBS DES method
EXTENVX					
12	(C)	STRUCTURE	8	EXTENVX	TYPE=ENVRXTR parameter list
12	(C)	ADDRESS	4	EXTENVO	Address of the data structure to retrieve the security environment
16	(10)	ADDRESS	4	EXTACE2	Address of ACEE
FIELDS					
0	(0)	STRUCTURE	*	FIELDS	Map of FIELDS parameter
0	(0)	SIGNED	4	FLDCNT	Number of field names
4	(4)	CHARACTER	8	FLDNAME (*)	Individual field names
0	(0)	STRUCTURE	*	SEGDATS	Map SEGDATA parameter
0	(0)	SIGNED	4	SEGFLEN	Size of data
4	(4)	CHARACTER	*	SEGFDTA	Segment data

RXTL constants

Table 134. Constants for RXTL

Len	Type	Value	Name	Description
2	DECIMAL	24	EXTEXTL	Length of release 1.6 or 1.7 parameters
2	DECIMAL	44	EXTRL	Length of release 1.8 or 1.9 parameters
2	DECIMAL	20	EXTENCL	Length of encrypt parameters

RXTL cross reference

Table 135. Cross Reference for RXTL

Name	Offset	Hex Value
EXTACEE	20	
EXTACE2	10	
EXTBRNCH	5	80
EXTCLAS	C	
EXTDATA	C	
EXTDRV	28	02

Table 135. Cross Reference for RXTL (continued)

Name	Offset	Hex Value
EXTENB	18	
EXTENC	C	
EXTEND	C	
EXTENDX	2C	
EXTEND1	18	
EXTENT	8	
EXTENTX	8	
EXTENVO	C	
EXTENVX	C	
EXTENX	5	40
EXTEXT	C	
EXTFLAC	28	01
EXTFLAGS	5	
EXTFLDS	14	
EXTFUNCT	2	
EXTGEN	28	04
EXTLEN	0	
EXTLIST	0	
EXTMATCH	28	08
EXTMETH	10	
EXTOFF	6	
EXTPROP	5	20
EXTRES1	28	
EXTSEGDT	1C	
EXTSEGM	18	
EXTSP	10	
EXTSPR	28	
EXTTYPE	3	
EXTVER	4	
EXTVOL	24	
FIELDS	0	
FLDCNT	0	
FLDNAME	4	
SEGDATS	0	
SEGFDTA	4	
SEGFLEN	0	

Chapter 57. RXTW: RACROUTE REQUEST=EXTRACT Result Area Mapping

RXTW programming interface information

RXTW is a programming interface.

RXTW heading information

Common name: RACROUTE REQUEST=EXTRACT result area mapping

Macro ID: IRRPRXTW

DSECT name: EXTWKEA, EXTWANM, EXTWABG, EXTWADP, EXTWARM, EXTWAS1, EXTWAS2, EXTWAS3, EXTWAS4, EXTWAAC

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
229 (or subpool supplied by issuer of RACROUTE REQUEST=EXTRACT)

Key
0 (or as determined by the subpool of the issuer of RACROUTE REQUEST=EXTRACT)

Size:

Section
Size

1
72 bytes

2
For Release 1.7 - 8 bytes For Release 1.8 and subsequent releases - varies

3 through 11
Work attributes data - 4-byte field followed by variable data

Created by: RACROUTE REQUEST=EXTRACT

Pointed to by: Register 1 after RACROUTE REQUEST=EXTRACT has been issued

Serialization: None

Function: Maps the fixed portion of the results from RACROUTE REQUEST=EXTRACT and the work attributes data that is extracted from the ACEE

RXTW mapping

Table 136. Structure EXTWKEA

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	72	EXTWKEA	
0	(0)	SIGNED	4		
0	(0)	UNSIGNED	1	EXTWSP	Area subpool

Table 136. Structure EXTWKEA (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
1	(1)	UNSIGNED	3	EXTWLN	Area length
4	(4)	SIGNED	2	EXTWOFF	Offset from EXTWKEA to start of optional returned fields
6	(6)	CHARACTER	1	EXTFLAG	Flag Byte
		1...		EXTGENRC	Generic profile retrieved for TYPE=EXTRACTN
		.111 1111		*	Reserved
7	(7)	UNSIGNED	1	EXTWVERS	Version of Result Area
8	(8)	SIGNED	2	EXTWAOFF	Offset from EXTWKEA to start of optional Work Attributes area when extracting from the ACEE
10	(A)	UNSIGNED	2	EXTWRTAS	Caller's ASID
12	(C)	ADDRESS	4	EXTWRTAD	Caller's return address
16	(10)	CHARACTER	3	EXTWPRLN	USER'S OR DEFAULT PRIMARY LANGUAGE
19	(13)	CHARACTER	3	EXTWSELN	USER'S OR DEFAULT SECONDARY LANGUAGE
22	(16)	CHARACTER	1	EXTWPRUS	WHETHER THE REPORTED PRIMARY LANGUAGE IS DEFINED IN THE USER PROFILE(U) OR IS THE INSTALLATION DEFAULT(S)
23	(17)	CHARACTER	1	EXTWSEUS	WHETHER THE REPORTED SECONDARY LANGUAGE IS DEFINED IN THE USER PROFILE(U) OR IS THE INSTALLATION DEFAULT(S)
24	(18)	CHARACTER	8	EXTWUID	SPECIFIED OR CURRENT USER'S ID
32	(20)	CHARACTER	8	EXTWGRP	SPECIFIED USER'S DEFAULT GROUP OR CURRENT USER'S CONNECT GROUP
40	(28)	CHARACTER	32	*	RESERVED
72	(48)	CHARACTER	4	EXTWEND(0)	END OF FIXED PART

EXTWOPT

72	(48)	STRUCTURE	8	EXTWOPT	Optional fields for TYPE=EXTRACT or EXTRACTN and Release 1.8 or later
72	(48)	CHARACTER	8	EXTWPSWD	Encoded password for TYPE=EXTRACT and Release 1.7 or earlier

EXTWANM

0	(0)	STRUCTURE	0*	EXTWANM	WORKATTR - User name

Table 136. Structure EXTWKEA (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	SIGNED	4	EXTWNMLN	Length of user name
4	(4)	CHARACTER	*	EXTWNAME	User name for SYSOUT
EXTWABG					
0	(0)	STRUCTURE	*	EXTWABG	WORKATTR - Building name
0	(0)	SIGNED	4	EXTWBDLN	Length of building name
4	(4)	CHARACTER	*	EXTWBLDG	Building name for delivery
EXTWADP					
0	(0)	STRUCTURE	*	EXTWADP	WORKATTR - Department name
0	(0)	SIGNED	4	EXTWDTLN	Length of department name
4	(4)	CHARACTER	*	EXTWDEPT	Department name for delivery
EXTWARM					
0	(0)	STRUCTURE	*	EXTWARM	WORKATTR - Room name
0	(0)	SIGNED	4	EXTWRMLN	Length of room name
4	(4)	CHARACTER	*	EXTWROOM	Room for delivery
EXTWAS1					
0	(0)	STRUCTURE	*	EXTWAS1	WORKATTR - SYSOUT line 1
0	(0)	SIGNED	4	EXTWS1LN	Length of SYSOUT line 1
4	(4)	CHARACTER	*	EXTWSYS1	SYSOUT delivery line 1
EXTWAS2					
0	(0)	STRUCTURE	*	EXTWAS2	WORKATTR - SYSOUT line 2
0	(0)	SIGNED	4	EXTWS2LN	Length of SYSOUT line 2
4	(4)	CHARACTER	*	EXTWSYS2	SYSOUT delivery line 2
EXTWAS3					
0	(0)	STRUCTURE	*	EXTWAS3	WORKATTR - SYSOUT line 3
0	(0)	SIGNED	4	EXTWS3LN	Length of SYSOUT line 3
4	(4)	CHARACTER	*	EXTWSYS3	SYSOUT delivery line 3
EXTWAS4					
0	(0)	STRUCTURE	*	EXTWAS4	WORKATTR - SYSOUT line 4
0	(0)	SIGNED	4	EXTWS4LN	Length of SYSOUT line 4
4	(4)	CHARACTER	*	EXTWSYS4	SYSOUT delivery line 4
EXTWAAC					
0	(0)	STRUCTURE	*	EXTWAAC	WORKATTR - Account number
0	(0)	SIGNED	4	EXTWATLN	Length of account number
4	(4)	CHARACTER	*	EXTWACCT	Account number
EXTWAEML					

Table 136. Structure EXTWKEA (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	*	EXTWAEML	WORKATTR - E-mail address
0	(0)	SIGNED	4	EXTWAELN	Length of e-mail address
4	(4)	CHARACTER	*	EXTWMAIL	E-mail address

RXTW cross reference

Table 137. Cross Reference for RXTW

Name	Offset	Hex Value
EXTFLAG	6	
EXTGENRC	6	80
EXTWAAC	0	
EXTWABG	0	
EXTWACCT	4	
EXTWADP	0	
EXTWAELN	0	
EXTWAEML	0	
EXTWANM	0	
EXTWAOFF	8	
EXTWARM	0	
EXTWAS1	0	
EXTWAS2	0	
EXTWAS3	0	
EXTWAS4	0	
EXTWATLN	0	
EXTWBDLN	0	
EXTWBLDG	4	
EXTWDEPT	4	
EXTWDTLN	0	
EXTWEND	48	
EXTWGRP	20	
EXTWKEA	0	
EXTWLN	1	
EXTWMAIL		
EXTWNAME	4	
EXTWNMLN	0	
EXTWOFF	4	

Table 137. Cross Reference for RXTW (continued)

Name	Offset	Hex Value
EXTWOPT	48	48
EXTWPRLN	10	
EXTWPRUS	16	
EXTWPSWD		
EXTWRMLN	0	
EXTWROOM	4	
EXTWRTAD	C	
EXTWRTAS	A	
EXTWSELN	13	
EXTWSEUS	17	
EXTWSP	0	
EXTWSYS1	4	
EXTWSYS2	4	
EXTWSYS3	4	
EXTWSYS4	4	
EXTWS1LN	0	
EXTWS2LN	0	
EXTWS3LN	0	
EXTWS4LN	0	
EXTWUID	18	
EXTWVERS	7	

Chapter 58. SAFP: SAF Router Parameter List

SAFP programming interface information

SAFP is a programming interface.

SAFP heading information

Common name: System authorization facility (SAF) router parameter list

Macro ID: ICHSAFP

DSECT name: SAFP

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: 104 bytes

Created by: RACROUTE macro

Pointed to by: RACROUTE MF=E or MF=S places the address in R1 before invoking SAF

Serialization: None

Function: Serves as the descriptor for data passed to the SAF router by the RACROUTE macro

SAFP mapping

Table 138. Structure SAFP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	104	SAFP	
0	(0)	SIGNED	4	SAFPRRET	RACF or installation exit Return Code
4	(4)	SIGNED	4	SAFPRREA	RACF or installation exit Reason Code
8	(8)	SIGNED	2	SAFPPLN	Length of SAFP parameter list (in bytes)
10	(A)	UNSIGNED	1	SAFPVRRL	RACF Version/Release list indicator (values defined below)
11	(B)	CHARACTER	1		Reserved
12	(C)	SIGNED	2	SAFPREQT	Request number (values defined below)

Table 138. Structure SAFP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
14	(E)	BITSTRING	1	SAFPFLGS	Flags
		1...		SAFPMSGR	1 - message return requested
		.1..		SAFPR18	1 - Release 1.8 or higher function was requested
		..1.		SAFPSUPP	1 - message suppression on
		...1		SAFPDCPL	1 - DECOUPL=yes
		.... 1...		SAFPSYST	1 - SYSTEM=yes
		.... .111		*	Reserved
15	(F)	UNSIGNED	1	SAFPMSPL	Subpool to be used for messages to be returned, if SAFPMSGR on
16	(10)	ADDRESS	4	SAFPREQR	Requestor name address (points to an 8-byte character field)
20	(14)	ADDRESS	4	SAFPSUBS	Subsystem name address (points to an 8-byte character field)
24	(18)	ADDRESS	4	SAFPWA	SAF work area address
28	(1C)	ADDRESS	4	SAFPMSAD	Upon return, will contain the address of the message(s) returned from the invoked function
32	(20)	ADDRESS	4	*	Reserved
36	(24)	UNSIGNED	4	SAFPRACP	Offset to RACF related parameter list from base address of SAFP
40	(28)	SIGNED	4	SAFPSFRC	SAF Return Code
44	(2C)	SIGNED	4	SAFPSFRS	SAF Reason Code
48	(30)	SIGNED	2	SAFPPLNX	Length of SAFP extension parameter list (in bytes)
50	(32)	SIGNED	2	SAFPOLEN	Length of Original Plist
52	(34)	ADDRESS	4	SAFPRETD	Address of returned data
56	(38)	ADDRESS	4	SAFPFLAT	Address of flat parameter list
60	(3C)	ADDRESS	4	SAFPECB1	Address of ECB1
64	(40)	ADDRESS	4	SAFPECB2	Address of ECB2
68	(44)	ADDRESS	4	SAFPPREV	Address of previous flat list
72	(48)	ADDRESS	4	SAFPNEXT	Address of next flat list
76	(4C)	ADDRESS	4	SAFPORIG	Address of origional list
80	(50)	SIGNED	4	SAFPFLEN	Flat parameter list length
84	(54)	SIGNED	4	SAFPUSRW	User Word - identifier
88	(58)	ADDRESS	4	SAFPPREE	Address of Pre-Processing Exit
92	(5C)	ADDRESS	4	SAFPPOST	Address of Post-Processing Exit
96	(60)	ADDRESS	4	SAFPSYNC	Address of Synchronous ECB

Table 138. Structure SAFP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
100	(64)	UNSIGNED	1	SAFPSKEY	Requestors Storage Key
101	(65)	UNSIGNED	1	SAFPMODE	Requestors Addressing mode
102	(66)	UNSIGNED	1	SAFPSBYT	Status byte
		1...		SAFPGCS	1 - Request came from GCS
		.1..		SAFPSFSU	1 - SFS user accessing own file or directory (used for SFSAUTOACCESS processing)
103	(67)	UNSIGNED	1	*	Reserved
104	(68)	CHARACTER		*	

SAFP constants

Table 139. Constants for SAFP

Len	Type	Value	Name	Description
4	DECIMAL	104	SAFPLEN	
CONSTANTS FOR REQUEST NUMBER VALUES				
1	DECIMAL	1	SAFPAU	RACHECK - authorization function
1	DECIMAL	2	SAFPFAU	FRACHECK - Fast authorization function
1	DECIMAL	3	SAFPLIS	RACLIST - in-storage list building function
1	DECIMAL	4	SAFPDEF	RACDEF - definition function
1	DECIMAL	5	SAFPVER	RACINIT - verification function
1	DECIMAL	6	SAFPEXT	RACXTRT - extract function
1	DECIMAL	7	SAFPDIR	RACDAUTH - directed authorization function
1	DECIMAL	8	SAFPTKMP	RACTKSRV - security token services
1	DECIMAL	9	SAFPVERX	RACROUTE REQUEST=VERIFYX
1	DECIMAL	10	SAFPTKXT	RACTKSRV - extract token services
1	DECIMAL	11	SAFPTBLD	RACINIT - token build services
1	DECIMAL	12	SAFPEXTB	RACXTRT - branch entry
1	DECIMAL	13	SAFPAUD	RACAUDIT - audit service
1	DECIMAL	14	SAFPSTAT	RACSTAT - status service
1	DECIMAL	15	SAFPSIGN	SIGNON request
1	DECIMAL	16	SAFPTXMP	Token map request for XMREQ=YES is specified.

Table 139. Constants for SAFP (continued)

Len	Type	Value	Name	Description
1	DECIMAL	17	SAFPTXXT	Token extract request for XMREQ=YES is specified.
Constants for TOKENBLD request				
1	DECIMAL	8	SAFPTBRC	SAFPTBLD request SAF r.c
The following reason codes are used:				
1	DECIMAL	0	SAFPTBUT	TOKNOUT missing - 9C7 SAF abend
1	DECIMAL	4	SAFPTBUL	TOKNOUT length too small: on return the length field in the token has the correct length - 9C7 SAF abend
1	DECIMAL	8	SAFPTBTK	Invalid token data - 9C7 SAF abend
1	DECIMAL	12	SAFPTBSL	STOKEN length too small: on return the length field in the token has the correct length - 9C7 SAF abend
1	DECIMAL	16	SAFPTBUB	TOKNOUT length too large: on return the length field in the token has the correct length
1	DECIMAL	20	SAFPTBSB	STOKEN length too large: on return the length field in the token has the correct length
1	DECIMAL	24	SAFPTBV0	A token passed in did not have its version set - 9C7 SAF abend
1	DECIMAL	32	SAFPTBIL	TOKNIN length too small: on return the length field in the token has the correct length - 9C7 SAF abend
1	DECIMAL	28	SAFPTBIB	TOKNIN length too large: on return the length field in the token has the correct length
Constants for VERIFYX request				
1	DECIMAL	60	SAFPVXRC	SAFPVERX request SAF r.c
The following reason codes are used:				
1	DECIMAL	0	SAFPVXNR	RACF not available
1	DECIMAL	4	SAFPVXOP	Old Password required
1	DECIMAL	8	SAFPVXUS	Userid required
1	DECIMAL	12	SAFPVXEF	ESTAE failed
1	DECIMAL	16	SAFPVXUT	TOKNOUT keyword missing - 9C7 SAF abend

Table 139. Constants for SAFP (continued)

Len	Type	Value	Name	Description
1	DECIMAL	20	SAFPVXUL	TOKNOUT length too small: on return the length field in the token has the right length - 9C7 SAF abend
1	DECIMAL	24	SAFPVXTK	Invalid token data - 9C7 SAF abend
1	DECIMAL	28	SAFPVXSL	STOKEN length too small: on return the length field in the token has the right length - 9C7 SAF abend
1	DECIMAL	32	SAFPVXUB	TOKNOUT length too large: on return the length field in the token has the right length
1	DECIMAL	36	SAFPVXSB	STOKEN length too large: on return the length field in the token has the right length
1	DECIMAL	40	SAFPVXV0	A token passed in did not have its version set - 9C7 SAF abend
1	DECIMAL	44	SAFPVXIL	TOKNIN length too small: on return the length field in the token has the correct length - 9C7 SAF abend
1	DECIMAL	48	SAFPVXIB	TOKNIN length too large: on return the length field in the token has the correct length
Constants for VERIFY request				
1	DECIMAL	64	SAFPVYRC	SAFPVER request SAF r.c
The following reason codes are used:				
1	DECIMAL	0	SAFPVYTK	Invalid token data - 9C7 SAF abend
1	DECIMAL	4	SAFPVYUL	TOKNIN length too small: on return the length field in the token has the right length - 9C7 SAF abend
1	DECIMAL	8	SAFPVYSL	STOKEN length too small: on return the length field in the token has the right length - 9C7 SAF abend
1	DECIMAL	12	SAFPVYUB	TOKNIN length too large: on return the length field in the token has the right length
1	DECIMAL	16	SAFPVYSB	STOKEN length too large: on return the length field in the token has the right length

Table 139. Constants for SAFP (continued)

Len	Type	Value	Name	Description
1	DECIMAL	20	SAFPVYV0	A token passed in did not have its version set - 9C7 SAF abend
Constants for SIGNON request				
1	DECIMAL	12	SAFPSGRC	SAFPSIGN request SAF r.c
The following reason codes are used:				
1	DECIMAL	0	SAFPSGNE	Environment error: No ESA support for SIGNON
1	DECIMAL	12	SAFPSGOL	TOKNOUT length too small: on return the length field in the token has the right length - 9C7 SAF abend
1	DECIMAL	4	SAFPSGOT	TOKNOUT length too large: on return the length field in the token has the right length
Constants for version/release				
1	DECIMAL	25	SAFPCURR	Current level of RACF
1	DECIMAL	0	SAFPRL19	Indicates RACF 1.9.0
1	DECIMAL	2	SAFPR192	Indicates RACF 1.9.2
1	DECIMAL	3	SAFPRL21	Indicates RACF 2.1.0
1	DECIMAL	4	SAFPRL22	Indicates RACF 2.2.0
1	DECIMAL	5	SAFPRL23	Indicates RACF 2.3.0
1	DECIMAL	6	SAFPRL24	Indicates RACF 2.4.0
1	DECIMAL	7	SAFPRL26	Indicates RACF 2.6.0
1	DECIMAL	8	SAFPRL28	Indicates RACF 2.6.8
1	DECIMAL	9	SAFPRL73	Indicates HRF7703
1	DECIMAL	10	SAFPRL75	Indicates HRF7705
1	DECIMAL	11	SAFPRL76	Indicates HRF7706
1	DECIMAL	12	SAFPRL77	Indicates HRF7707
1	DECIMAL	13	SAFPRL78	Indicates HRF7708
1	DECIMAL	14	SAFPRL79	Indicates HRF7709
1	DECIMAL	15	SAFPRL70	Indicates HRF7720
1	DECIMAL	16	SAFPRL30	Indicates HRF7730
1	DECIMAL	17	SAFPRL40	Indicates HRF7740
1	DECIMAL	18	SAFPRL50	Indicates HRF7750
1	DECIMAL	19	SAFPRL60	Indicates HRF7760
1	DECIMAL	20	SAFPRL7A	Indicates HRF7770
1	DECIMAL	21	SAFPRL80	Indicates HRF7780
1	DECIMAL	22	SAFPRL90	Indicates HRF7790

Table 139. Constants for SAFP (continued)

Len	Type	Value	Name	Description
1	DECIMAL	23	SAFPRLA0	Indicates HRF77A0
1	DECIMAL	24	SAFPRLB0	Indicated HRF77B0
1	DECIMAL	25	SAFPRLC0	Indicates HRF77C0
1	DECIMAL	25	SAFPRLD0	Indicates HRF77D0

SAFP cross reference

Table 140. Cross Reference for SAFP

Name	Offset	Hex Value
SAFP	0	
SAFPDCPL	E	10
SAFPECB1	3C	
SAFPECB2	40	
SAFPFLAT	38	
SAFPFLEN	50	
SAFPFLGS	E	
SAFPGCS	66	80
SAFPMODE	65	
SAFPMSAD	1C	
SAFPMSGR	E	80
SAFPMSPL	F	
SAFPNEXT	48	
SAFPOLEN	32	
SAFPORIG	4C	
SAFPPLN	8	
SAFPPLNX	30	
SAFPPOST	5C	
SAFPPREE	58	
SAFPPREV	44	
SAFPRACP	24	
SAFPREQR	10	
SAFPREQT	C	
SAFPRETD	34	
SAFPRREA	4	
SAFPRRET	0	
SAFPR18	E	40
SAFPSBYT	66	

Table 140. Cross Reference for SAFP (continued)

Name	Offset	Hex Value
SAFPSFRC	28	
SAFPSFRS	2C	
SAFPSFSU	66	40
SAFPSKEY	64	
SAFPSUBS	14	
SAFPSUPP	E	20
SAFPSYNC	60	
SAFPSYST	E	8
SAFPUSRW	54	
SAFPVRRL	A	
SAFPWA	18	

Chapter 59. SAFR: Number of z/OS UNIX Systems Services Callable Services

SAFR programming interface information

SAFR is a programming interface.

SAFR heading information

Common name:	Number of z/OS UNIX System Services callable services
Macro ID:	IRRPSAFR
DSECT name:	SAFR
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	None
Storage attributes:	N/A
Size:	4 bytes
Created by:	SAF initialization
Pointed to by:	CVTSAF
Serialization:	None
Function:	Maps the number of z/OS UNIX callable services

SAFR mapping

Table 141. Structure SAFR

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	4	SAFR	
0	(0)	UNSIGNED	2	SAFR_SAFSERV	Number of callable services supported by SAF.
2	(2)	UNSIGNED	2	SAFR_SERVICES	Number of callable services available for use.

SAFR constants

Table 142. Constants for SAFR

Len	Type	Value	Name	Description
2	DECIMAL	4	SAFR#LEN	Length of SAFR

Chapter 60. SAFV: SAF Router Vector Table

SAFV programming interface information

The following fields are not programming interface information:

- SAFVIDEN
- SAFVVRSN
- SAFVEXIT
- SAFVSAFR
- SAFVSECT

SAFV heading information

Common name:	SAF router vector table
Macro ID:	ICHSAFV
DSECT name:	SAFV
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	SAFV (Offset: 0, Length: 4)
Storage attributes:	**Subpool** 245 **Key** 0 **Residency** Below 16M
Size:	68 bytes
Created by:	SAF initialization
Pointed to by:	CVTSAF
Serialization:	None
Function:	Maps the SAF router vector table

SAFV mapping

Table 143. Structure SAFV

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	64	SAFV	SAF VECTOR TABLE
0	(0)	CHARACTER	4	SAFVIDEN	IDENTIFYING LITERAL FOR DUMPS 'SAFV'
4	(4)	UNSIGNED	1	SAFVVRSN	TABLE VERSION NUMBER - '02'X
5	(5)	CHARACTER	3	*	RESERVED

Table 143. Structure SAFV (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
8	(8)	ADDRESS	4	SAFVEXIT	ADDRESS OF THE SAF ROUTER EXIT ROUTINE (ICHRTX00)
12	(C)	ADDRESS	4	SAFVSAFR	ADDRESS OF THE SAF ROUTER ROUTINE (ICHSFR00)
16	(10)	ADDRESS	4	SAFVRACR	ADDRESS OF THE RACF FRONT END ROUTINE (ICHRFR00), SET BY RACF INITIALIZATION (ICHSEC00) IF RACF IS INSTALLED
20	(14)	ADDRESS	4	SAFVRACT	ADDRESS OF THE RACF FRONT END TABLE (ICHRFR01), SET BY RACF INITIALIZATION (ICHSEC00) IF RACF IS INSTALLED
24	(18)	ADDRESS	4	SAFVEXUS	RESERVED FOR INSTALLATION EXIT USE
28	(1C)	ADDRESS	4	SAFVRAC2	ADDRESS OF THE RACF FRONT END ROUTINE FOR OMVS (IRRRFR10)
32	(20)	ADDRESS	4	SAFVXIT2	ADDRESS OF THE SAF ROUTER EXIT ROUTINE FOR OMVS (IRRSXT00)
36	(24)	ADDRESS	4	SAFVSECT	ADDRESS OF THE SAF SECURITY PRODUCT TRACE AREA (IRRSAFT)
40	(28)	ADDRESS	4	SAFVZPDR	ADDRESS OF THE PD ROUTER
44	(2C)	ADDRESS	4	SAFVZXIT	ADDRESS OF THE SAF ROUTER EXIT ROUTINE FOR SAF3 (IRRSZT00)
48	(30)	ADDRESS	4	SAFVZCVT	ADDRESS OF THE PD COMMUNICATION VECTOR TABLE
52	(34)	SIGNED	4	* (3)	Reserved
64	(40)	CHARACTER		*	ENSURE DOUBLE WORD LENGTH

SAFV constants

Table 144. Constants for SAFV

Len	Type	Value	Name	Descriptions
4	DECIMAL	64	SAFVLEN	Length of the SAF router vector table
4	CHARACTER	SAFV	SAFVIDC	Literal value to be stored in SAFVIDEN
1	DECIMAL	1	SAFVVNC	Current value of SAF version number stored in SAFVVRSN
1	DECIMAL	0	SAFVVN0	Original value of SAF version number stored in SAFVVRSN

Table 144. Constants for SAFV (continued)

Len	Type	Value	Name	Descriptions
1	DECIMAL	1	SAFVVN1	Value of SAF version number for UNIX System Services stored in SAFVVRSN
1	DECIMAL	2	SAFVVN2	Value of SAF version number for PD stored in SAFVVRSN

SAFV cross reference

Table 145. Cross Reference for SAFV

Name	Offset	Hex Value
SAFV	0	
SAFVEXIT	8	
SAFVEXUS	18	
SAFVIDEN	0	
SAFVRACR	10	
SAFVRACT	14	
SAFVRAC2	1C	
SAFVSAFR	C	
SAFVVRSN	4	
SAFVXIT2	20	
SAFVXITX	28	
SAFVSECT	24	
SAFVZCVT	30	
SAFVZPDR	28	
SAFVZXIT	2C	

Chapter 61. SECUR: OS/390 Security Context

SECUR programming interface information

SECUR is a programming interface.

SECUR heading information

Common name: OS/390 Security Context

Macro ID: IRRSECUR

DSECT name: IRR_SECURITY_CONTEXT

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by invoker

Key
Any

Residency
Any

Size: Version dependent

Created by: RRS context services

Pointed to by: N/A

Serialization: Through RRS context services APIs

Function: Maps the identity of the user associated with a particular transaction

SECUR mapping

Table 146. Structure SECURE

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	SECUR	
0	(0)	UNSIGNED	4	SECUR_Version	Version number
4	(4)	CHARACTER	8	SECUR_Creator	Name of creating component (such as RACF)
12	(C)	UNSIGNED	4	SECUR_ACEE_Alet	Alet for ACEE
16	(10)	ADDRESS	4	SECUR_ACEE_Address	Pointer to user's ACEE
20	(14)	CHARACTER	8	SECUR_userid	User ID padded with blanks

Constants for SECUR

Table 147. Constants for SECUR

Len	Type	Value	Name	Description
32	CHARACTER	IRR_SECURITY_CON TEXT	SECUR_CONTEXT_KEY	Context key identifier.
4	DECIMAL	1	SECUR_VERSION_1	Initial version
4	DECIMAL	28	SECUR_VERSION_1_LENGTH	Length of area
4	DECIMAL	1	SECUR_CURRENT_VERSION	Current version

Chapter 62. SGNPL: RACROUTE REQUEST=SIGNON Parameter List (Request Section)

SGNPL programming interface information

SGNPL is a programming interface.

SGNPL heading information

Common name: Request-specific portion of the RACROUTE REQUEST=SIGNON parameter list

Macro ID: IRRSGNPL

DSECT name: SIGENVD, SIGPLIST

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: 48

Created by: RACROUTE REQUEST=SIGNON macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=SIGNON routine

SGNPL mapping

Table 148. Structure SIGPLIST

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	48	SIGPLIST	
0	(0)	SIGNED	2	SIGPLLEN	SIGNON parm list length
2	(2)	UNSIGNED	1	SIGTYPE	Subfunction Type Index
3	(3)	UNSIGNED	1	SIGLTYPE	List Type Index
4	(4)	ADDRESS	4	SIGAPPLP	Address of APPL name buffer
8	(8)	ADDRESS	4	SIGPOEP	Addr. of POE name buffer
12	(C)	ADDRESS	4	SIGACEPP	Address of ACEE
16	(10)	ADDRESS	4	SIGGRUPP	Address of GROUP name buffer

Table 148. Structure SIGPLIST (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
20	(14)	ADDRESS	4	SIGSECLP	Address of SECLABEL buffer
24	(18)	ADDRESS	4	SIGENVRO	Address of output ENVR data block
28	(1C)	ADDRESS	4	SIGOTKP	Address of TOKNOUT
32	(20)	ADDRESS	4	SIGUSRDP	Address of USERID name buffer
36	(24)	ADDRESS	4	SIGVEXIT	Address of VERBEXIT routine 21
40	(28)	ADDRESS	4	SIGENVRI	Address of input ENVR data block
44	(2C)	ADDRESS	4	SIGPOENP	Address POENET name buffer. 1 byte length plus 1-8 byte network name, or zero if not specified.
0	(0)	STRUCTURE	14	SIGENVD	ENVR OBJECT DATA STRUCTURE
0	(0)	UNSIGNED	4	SIGENLEN	ENVR OBJECT LENGTH
4	(4)	UNSIGNED	4	SIGENSLN	ENVR OBJECT STORAGE AREA LENGTH
8	(8)	ADDRESS	4	SIGENSAD	ENVR OBJECT STORAGE AREA ADDRESS
12	(C)	UNSIGNED	1	SIGENSSP	ENVR OBJECT STORAGE AREA SUBPOOL
13	(D)	UNSIGNED	1	SIGENSKY	ENVR OBJECT STORAGE AREA KEY

SGNPL cross reference

Table 149. Cross Reference for SGNPL

Name	Offset	Hex Value
SIGACEPP	C	
SIGAPPLP	4	
SIGENLEN	0	
SIGENSAD	8	
SIGENSKY	D	
SIGENSLN	4	
SIGENSSP	C	
SIGENVD	0	
SIGENVRI	28	
SIGENVRO	18	
SIGGRUPP	10	
SIGLTYPE	3	
SIGOTKP	1C	
SIGPLIST	0	

Table 149. Cross Reference for SGNPL (continued)

Name	Offset	Hex Value
SIGPLLEN	0	
SIGPOENP	2C	
SIGPOEP	8	
SIGSECLP	14	
SIGTYPE	2	
SIGUSRDP	20	
SIGVEXIT	24	

Chapter 63. SGX1P: RACROUTE REQUEST=SIGNON Parameter List Mapping

SGX1P programming interface information

SGX1P is a programming interface.

SGX1P heading information

Common name: RACROUTE REQUEST=SIGNON verbexit parameter list mapping

Macro ID: ICHSGX1P

DSECT name: None

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
230

Key
2

Residency
Above 16MB

Size: 20

Created by: RACROUTE REQUEST=SIGNON macro

Pointed to by: Register 1

Serialization: None

Function: Maps the necessary fields to be passed to the verbexit routine so the verbexit routine can issue the ALLOCATE for the SIGNOFF TP to remove the corresponding user entries from the signed_on_to list at the partner LU

SGX1P mapping

Table 150. Structure SGNFP

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	20	SGNFP	
0	(0)	ADDRESS	4	SNGFREQC_ADDR	ADDRESS OF REQUEST CODE
4	(4)	ADDRESS	4	SNGFLLU_ADDR	ADDRESS OF LOCAL_LU_ NAME
8	(8)	ADDRESS	4	SNGFPLU_ADDR	ADDRESS OF PARTNER_LU_NAME UP TO 17 CHARACTERS, NETNAME.LUNAME
12	(C)	ADDRESS	4	SNGFGDLN_ADDR	ADDRESS OF GDS VARIABLE LENGTH

Table 150. Structure SGNFP (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
16	(10)	ADDRESS	4	SNGFGDVA_ADDR	ADDRESS OF GDS VARIABLE
		1...		SNGF_LAST_PARM	END OF PARM LIST INDICATOR

SGX1P constants

Table 151. Constants for SGX1P

Len	Type	Value	Name	Description
4	DECIMAL	20	SNGFPLEN	

Chapter 64. STAT: RACROUTE REQUEST=STAT Parameter List (Request Section)

STAT programming interface information

STAT is a programming interface.

STAT heading information

Common name:	Request-specific portion of the RACROUTE REQUEST=STAT parameter list
Macro ID:	IRRPSTAT
DSECT name:	STATPARM
Owning component:	Resource Access Control Facility (SC1BN)
Eye-catcher ID:	None
Storage attributes:	**Subpool** Determined by caller **Key** Determined by caller **Residency** Determined by caller
Size:	Varies
Created by:	RACROUTE REQUEST=STAT macro
Pointed to by:	Address of SAFP plus offset in SAFPRACP
Serialization:	None
Function:	Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=STAT routine

STAT mapping

Table 152. Structure STATPARM

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	0	STATPARM	STAT parameter list
0	(0)	ADDRESS	4	STATCLAS	Address of class name
4	(4)	ADDRESS	4	STATCDTP	Pointer to the address of the class entry in the static CDT
8	(8)	SIGNED	0	*	Present only if RACROUTE is used
8	(8)	STRUCTURE	0	STATEXT1	RELEASE=1.9 extension
8	(8)	UNSIGNED	2	STATLEN	Length of this parameter list
10	(A)	CHARACTER	2	*	Reserved

Table 152. Structure STATPARM (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
12	(C)	STRUCTURE	4	STATEXT2	RELEASE=2.2 extension
12	(C)	ADDRESS	4	STATCPYA	Address of the area to hold the CDT entry (COPY)
16	(10)	UNSIGNED	4	STATCPYL	Length of the COPY area (COPYLEN)
20	(14)	STRUCTURE	4	STATEXT3	RELEASE=7709 extension
20	(14)	ADDRESS	4	STATNEXT	Address of the class name for the NEXT= keyword

STAT constants

Table 153. Constants for STAT

Len	Type	Value	Name	Description
2	DECIMAL	12	STATLEN0	Length of parameter list for RELEASE=1.9 through RELEASE=2.1
2	DECIMAL	20	STATLEN1	Length of parameter list for RELEASE=2.2 through RELEASE=7708
2	DECIMAL	24	STATLEN2	Length of parameter list for RELEASE=7709

STAT cross reference

Table 154. Cross Reference for STAT

Name	Offset	Hex Value
STATCDTP	4	
STATCLAS	0	
STATCPYA	C	
STATCPYL	10	
STATEXT1	8	
STATEXT2	C	
STATEXT3	14	
STATLEN	8	
STATNEXT	14	
STATPARM	0	

Chapter 65. TSRV: RACROUTE REQUEST=TOKENMAP/TOKENXTR Parameter List (Request Section)

TSRV programming interface information

TSRV is a programming interface.

TSRV heading information

Common name: Request-specific portion of the RACROUTE REQUEST=TOKENMAP/TOKENXTR parameter list

Macro ID: None

DSECT name: None

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
Determined by caller

Key
Determined by caller

Residency
Determined by caller

Size: Varies

Created by: RACROUTE REQUEST=TOKENMAP/TOKENXTR macro

Pointed to by: Address of SAFP plus offset in SAFPRACP

Serialization: None

Function: Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=TOKENMAP/TOKENXTR routine

TSRV mapping

Table 155. Structure TSRVPARD

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	44	TSRVPARM	TOKENMAP/TOKENXTR parameters
0	(0)	ADDRESS	4	TSRVTKIN	TOKNIN pointer
4	(4)	ADDRESS	4	TSRVACEE	ACEE pointer
8	(8)	ADDRESS	4	TSRVTKOT	TOKNOUT pointer
12	(C)	BITSTRING	1	TSRVFLGS	Flag byte

Table 155. Structure TSRVPARD (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
		1...		TSRVFMT	Format of output token for TOKENMAP: 1 = encrypt, 0 = decrypt
		.111 1111		*	Reserved
13	(D)	CHARACTER	1	*	Reserved
14	(E)	UNSIGNED	2	TSRVLEN	Length of this parameter list
16	(10)	CHARACTER	8	*	Reserved

TSRV cross reference

Table 156. Cross Reference for TSRV

Name	Offset	Hex Value
TSRVACEE	4	
TSRVFLGS	C	
TSRVFMT	C	80
TSRVLEN	E	
TSRVPARM	0	
TSRVTKIN	0	
TSRVTKOT	8	

Chapter 66. WORK: z/OS UNIX System Services Work Area for SAF and RACF

WORK programming interface information

WORK is a programming interface.

WORK heading information

Common name: z/OS UNIX work area for SAF and RACF

Macro ID: IRRPWORK

DSECT name: WORK

Owning component: Resource Access Control Facility (SC1BN)

Eye-catcher ID: None

Storage attributes:

Subpool
N/A

Key
0

Residency
N/A

Size: 1024 bytes

Created by: Invoker of z/OS UNIX security functions

Pointed to by: Common SAF/RACF parameter list for z/OS UNIX security functions (IRRPCOMP)

Serialization: None

Function: Maps the 1024-byte work area for use by the callable services routers and the IRRSXT00 exit

WORK mapping

Table 157. Structure WORK

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
0	(0)	STRUCTURE	1024	WORK	z/OS UNIX SAF/RACF work area.
The following 16 bytes are reserved for use by the User Exit and can only be stored into once by system code. That store is SAF storing zero in the WORK_USER_SIGNAL field to tell the User Exit that it is being invoked before the security product (RACF). After the first call, the User Exit should make that field non-zero (SAF does not) so that it knows it is being invoked after RACF.					
0	(0)	CHARACTER	16	WORK_USER_PERM	For User Exit's use only.
0	(0)	UNSIGNED	4	WORK_USER_SIGNAL	0-first time User Exit called.
4	(4)	CHARACTER	12	*	Reserved for User Exit.

Table 157. Structure WORK (continued)

Offset Dec	Offset Hex	Type	Len	Name(Dim)	Description
The following 944 bytes are used by both RACF and the User Exit RACF can use the entire 944 bytes, the user the first 136.					
16	(10)	CHARACTER	944	WORK_RACF_WORKA	RACF work area.
16	(10)	CHARACTER	136	WORK_USER_WORKA	User work area.
152	(98)	CHARACTER	808	*	Reserved
The following 64 bytes are reserved for the SAF Router for future use.					
960	(3C0)	CHARACTER	64	*	Reserved for future use.

WORK constants

Table 158. Constants for WORK

Len	Type	Value	Name	Description
4	DECIMAL	1024	WORK_LEN	Constant WORK length

WORK cross reference

Table 159. Cross Reference for WORK

Name	Offset	Hex Value
WORK	0	
WORK_RACF_WORKA	10	
WORK_USER_PERM	0	
WORK_USER_SIGNAL	0	
WORK_USER_WORKA	10	

Appendix A. Accessibility

Accessible publications for this product are offered through IBM Documentation (www.ibm.com/docs/en/zos).

If you experience difficulty with the accessibility of any z/OS information, send a detailed message to the Contact the z/OS team web page (www.ibm.com/systems/campaignmail/z/zos/contact_z) or use the following mailing address.

IBM Corporation
Attention: MHVRCFS Reader Comments
Department H6MA, Building 707
2455 South Road
Poughkeepsie, NY 12601-5400
United States

Notices

This information was developed for products and services that are offered in the USA or elsewhere.

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 may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, 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.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for IBM Documentation. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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
Site Counsel
2455 South Road

Poughkeepsie, NY 12601-5400
USA

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user’s name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at ibm.com®/privacy and IBM’s Online Privacy Statement at ibm.com/privacy/details in the section entitled “Cookies, Web Beacons and Other Technologies,” and the “IBM Software Products and Software-as-a-Service Privacy Statement” at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMSdfp, JES2, JES3, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at Copyright and Trademark information (www.ibm.com/legal/copytrade.shtml).

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle, its affiliates, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Index

Product Number: 5650-ZOS

GA32-0885-50