

z/OS



# Security Server RACF Data Areas



z/OS



# Security Server RACF Data Areas

**Note**

Before using this information and the product it supports, read the information in Appendix A, "Notices" on page 249

This edition applies to version 1, release 13, modification 0 of IBM z/OS (product number 5694-A01) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces GA22-7680-12.

© Copyright International Business Machines Corporation 1994, 2011. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

# Contents

<b>About this document</b>	v
Intended audience	v
How to use this document	v
The header	v
Data area map	vi
Cross reference	vii
Where to find more information	vii
Softcopy documents	vii
RACF courses	vii
IBM Systems Center publications	viii
Other sources of information	viii
Internet sources	viii
The z/OS Basic Skills Information Center	ix
To request copies of IBM publications	x
<b>How to send your comments to IBM</b>	xi
If you have a technical problem	xi
<b>Summary of changes</b>	xiii
<b>RACF Data Areas</b>	1
ACEE: Accessor Environment Element	3
ACHKL: RACROUTE REQUEST=AUTH Parameter List (Request Section)	9
ACXP: ACEE Expansion/Compression Exit Parameter List	12
AFC: z/OS UNIX System Services Audit Function Codes	14
AUL: RACROUTE REQUEST=AUDIT Parameter List (Request Section)	17
CCXP: Command Preprocessing Exit (ICHCCX00) Parameter List	18
CDXP: Notify/Verify/List Exit Routines Parameter List	19
CGRP: Connect Group Name Table Definition	21
CNST/CNSX (RACF): Class Name and Syntax Table	23
CNST/CNSX (SAF): Class Name and Syntax Table	26
CNXP: Command Preprocessing Exit (ICHCNX00) Parameter List	27
COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services	30
COMX: 64-BIT common SAF/RACF parameter list for z/OS UNIX System Services security functions	75
COMY: 64-BIT enabled SAF callable services	92
CRED: z/OS UNIX System Services Credential Structure	94
CREI: z/OS UNIX System Services Credential Structure for IPC	97
DAUT: RACROUTE REQUEST=DIRAUTH Parameter List (Request Section)	99
DEXP: Data Encryption Exit Parameter List	101
DSDT: Data Set Descriptor Table	103
ENFP: RACF-Parameter List for ENF event code 62 listen exits	110
ENF2: RACF-Parameter List for ENF event code 71 listen exits	111
EVXP: RACF Command Exit Parameter List	112
FACL: z/OS UNIX System Services access control list	115
FAST: RACROUTE REQUEST=FASTAUTH Parameter List (Request Section)	118
FC: z/OS UNIX System Services Security Function Code Table	120
FXAP: RACROUTE REQUEST=FASTAUTH Extended Function Exit Parameter List	122
GANC: GENERICANCHOR settings mapping	124
GAPL: Generic anchor table entry	126
GPRFL: 64-bit Storage profile list mapping	129
GRPF: In-Storage Generic Profile Map	132
ICRX: Extended identity context reference	136
ICTX: Distributed Identity Support SAF	138
IDID: Distributed identity data	140
IFSP: z/OS UNIX System Services File Security Packet	143
IISP: z/OS UNIX System Services IPC Security Packet	145

ISP: RACF In-Storage Profile	146
OUSP: initUSP Output Parameter List	152
PWX2: New Password Phrase Exit Parameter List	154
PWXP: Password Exit Parameter List	156
RCVI: Identity cache communication vendor.	159
RCVT: RACF Communication Vector Table	161
RCXP: RACROUTE REQUEST=AUTH Exit Parameter List	171
RDDFL: RACROUTE REQUEST=DEFINE Parameter List (Request Section)	178
RDXP: RACROUTE REQUEST=DEFINE Exit Parameter List	181
RFXP: RACROUTE REQUEST=FASTAUTH Exit Parameter List	189
RIPL: RACROUTE REQUEST=TOKENBLD/VERIFY/VERIFYX Parameter List (Request Section)	192
RIXP: RACROUTE REQUEST=VERIFY/VERIFYX Exit Parameter List	198
RLST: RACROUTE REQUEST=LIST Parameter List (Request Section)	206
RLX1P: RACROUTE REQUEST=LIST Exit Parameter List	208
RLX2P: RACROUTE REQUEST=LIST Selection Exit Parameter List	213
RNG: RACF Database Range Table	217
RRPF: Resident Profile Map	218
RSMXP: RACF Report Writer Selection Exit Parameter List	222
RUTKN: Resource/User Security Token	224
RXTL: RACROUTE REQUEST=EXTRACT Parameter List (Request Section)	227
RXTW: RACROUTE REQUEST=EXTRACT Result Area Mapping	230
SAFP: SAF Router Parameter List	233
SAFR: Number of z/OS UNIX System Services Callable Services	237
SAFV: SAF Router Vector Table	238
SECUR: OS/390 Security Context	240
SGNPL: RACROUTE REQUEST=SIGNON Parameter List (Request Section)	241
SGX1P: RACROUTE REQUEST=SIGNON Parameter List Mapping	243
STAT: RACROUTE REQUEST=STAT Parameter List (Request Section)	244
TSRV: RACROUTE REQUEST=TOKENMAP/TOKENXTR Parameter List (Request Section)	246
WORK: z/OS UNIX System Services Work Area for SAF and RACF	247
<b>Appendix A. Notices</b>	249
Programming interface information	250
Policy for unsupported hardware	250
Trademarks	250
<b>Index</b>	251

---

## About this document

This document supports z/OS (5694-A01) and contains information about Resource Access Control Facility (RACF), which is part of z/OS Security Server for z/OS.

---

## Intended audience

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:

<b>Common Name:</b>	The descriptive name of the data area.
<b>Macro ID:</b>	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.
<b>DSECT Name:</b>	The name of the DSECT (dummy control section) created by the mapping macro.
<b>Owning Component:</b>	The component name and component identifier in parentheses.
<b>Eye-Catcher ID:</b>	The character-string identifier of the eye catcher (sometimes called the <b>control block ID</b> ) within the mapping macro. The offset and length of the eye catcher are also included.
<b>Storage Attributes:</b>	The storage attributes of the data area, including the following: <ul style="list-style-type: none"><li><b>Main Storage:</b> The central storage attributes of the data area.</li><li><b>Virtual Storage:</b> The virtual storage attributes of the data area.</li><li><b>Auxiliary Storage:</b> The spool storage attributes of the data area.</li><li><b>Subpool and Key:</b> 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.</li></ul>
<b>Size:</b>	The size of the data area in decimal bytes.
<b>Created by:</b>	The module, macro, or component whose use creates the data area.
<b>Pointed to by:</b>	The registers or data area fields that contain the address of the data area.
<b>Serialization:</b>	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: <ul style="list-style-type: none"><li>• Lock or locks</li><li>• ENQ and DEQ macros</li><li>• Compare and Swap (CS) instruction.</li></ul>
<b>Function:</b>	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:

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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:

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

**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:

<b>Name</b>	The name of the field, bit, or mask.
<b>Hex Offset</b>	The hexadecimal offset of the field into the data area. For bits, the hexadecimal offset of the field containing the bit.
<b>Hex Value</b>	Hexadecimal values are shown only for bits. The hexadecimal value shown implies the position of the bit in the field containing the bit.
<b>Level</b>	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:

<u>Name</u>	<u>Hex Offset</u>	<u>Hex Value</u>	<u>Level</u>
TCBACTIV	F0	80	2

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

```
240 (F0) FIXED 4 TCBXSCT Dispatcher intersect control word
240 (F0) BITSTRING 1 TCBXSCT1 Flag byte (MDC323)
```

<u>Offsets</u>			<u>Len</u>		
<u>Dec</u>	<u>Hex</u>	<u>Type</u>		<u>Name (Dim)</u>	<u>Description</u>
		1... ..		TCBACTIV	X'80' bit on means this TCB...

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

Where necessary, this document references information in other documents. For complete titles and order numbers for all elements of z/OS, see *z/OS Information Roadmap*.

## Softcopy documents

The RACF library is available on the following DVD softcopy collection in both BookManager and Portable Document Format (PDF) files. The collection includes Softcopy Reader, which is a program that enables you to view the BookManager files. You can view or print the PDF files with an Adobe Reader.

**SK3T-4271** *z/OS Version 1 Release 13 and Software Products DVD Collection*

This collection contains the documents for z/OS Version Release 13 and the libraries for multiple releases of more than 400 z/OS-related software products, on DVD.

The following CD softcopy collection includes books related to RACF:

**SK3T-7876** *IBM System z Redbooks Collection*

This softcopy collection contains a set of documents called IBM Redbooks that pertain to System z subject areas ranging from e-business application development and enablement to hardware, networking, Linux, solutions, security, Parallel Sysplex, and many others.

## RACF courses

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

<b>H3917</b>	<i>Basics of z/OS RACF Administration</i>
<b>H3927</b>	<i>Effective RACF Administration</i>
<b>ES885</b>	<i>Exploiting the Advanced Features of RACF</i>
<b>ES840</b>	<i>Implementing RACF Security for CICS</i>

IBM provides a variety of 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)

---

## IBM Systems Center publications

IBM systems centers produce documents known as IBM Redbooks that can help you set up and use RACF. These documents have not been subjected to any formal review nor have they been checked for technical accuracy, but they represent current product understanding (at the time of their publication) and provide valuable information on a wide range of RACF topics. They are not shipped with RACF; you must order them separately. A selected list of these documents follows. Other documents are available but they are not included in this list, either because the information they present has been incorporated into IBM product manuals or because their technical content is outdated.

GG24-4282	<i>Secured Single Signon in a Client/Server Environment</i>
GG24-4453	<i>Enhanced Auditing Using the RACF SMF Data Unload Utility</i>
GG26-2005	<i>RACF Support for Open Systems Technical Presentation Guide</i>
SG24-4820	<i>OS/390 Security Server Audit Tool and Report Application</i>
SG24-5158	<i>Ready for e-business: OS/390 Security Server Enhancements</i>
SG24-6840	<i>Communications Server for z/OS V1R2 TCP/IP Implementation Guide Volume 7: Security</i>

## 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:

- **Online library**

To view and print online versions of the z/OS publications, use this address:

<http://www.ibm.com/systems/z/os/zos/bkserv/>

- **Redbooks**

The redbooks that are produced by the International Technical Support Organization (ITSO) are available at the following address:

<http://www.ibm.com/redbooks/>

- **Enterprise system security**

For more information about security on the S/390 platform, OS/390, and z/OS, including the elements that comprise the Security Server, use this address:

<http://www.ibm.com/systems/z/advantages/security/>

- **RACF home page**

You can visit the RACF home page on the World Wide Web using this address:

<http://www.ibm.com/systems/z/os/zos/features/racf/>

- **RACF-L discussion list**

Customers and IBM participants may also discuss RACF on the RACF-L discussion list. RACF-L is not operated or sponsored by IBM; it is run by the University of Georgia.

To subscribe to the RACF-L discussion and receive postings, send a note to:

listserv@listserv.uga.edu

Include the following line in the body of the note, substituting your name and last name as indicated:

subscribe racf-1 *first\_name last\_name*

To post a question or response to RACF-L, send a note, including an appropriate Subject: line, to:

racf-1@listserv.uga.edu

- **Sample Code**

You can get sample code, internally-developed tools, and exits to help you use RACF. This code works in our environment, at the time we make it available, but is not officially supported. Each tool or sample has a README file that describes the tool or sample and any restrictions on its use.

To access this code from a web browser, go to <http://www.ibm.com/systems/z/os/zos/features/racf/> and select the "Resources" file tab, then select "Downloads" from the list, or go to <http://www.ibm.com/systems/z/os/zos/features/racf/goodies.html>.

The code is also available from **ftp.software.ibm.com** through **anonymous ftp**. To get access:

1. Log in as user **anonymous**.
2. Change the directory, as follows, to find the subdirectories that contain the sample code or tool you want to download:  
`cd eserver/zseries/zos/racf/`

An announcement is posted on the RACF-L discussion list whenever something is added.

**Note:** Some web browsers and some FTP clients (especially those using a graphical interface) might have problems using **ftp.software.ibm.com** because of inconsistencies in the way they implement the FTP protocols. If you have problems, you can try the following:

- Try to get access by using a web browser and the links from <http://www.ibm.com/systems/z/os/zos/features/racf/>.
- Use a different FTP client. If necessary, use a client that is based on command line interfaces instead of graphical interfaces.
- If your FTP client has configuration parameters for the type of remote system, configure it as UNIX instead of MVS.

#### **Restrictions**

Because the sample code and tools are not officially supported,

- There are no guaranteed enhancements.
- No APARs can be accepted.

---

## The z/OS Basic Skills Information Center

The z/OS Basic Skills Information Center is a Web-based information resource intended to help users learn the basic concepts of z/OS, the operating system that runs most of the IBM mainframe computers in use today. The Information Center is designed to introduce a new generation of Information Technology professionals to basic concepts and help them prepare for a career as a z/OS professional, such as a z/OS system programmer.

Specifically, the z/OS Basic Skills Information Center is intended to achieve the following objectives:

- Provide basic education and information about z/OS without charge
- Shorten the time it takes for people to become productive on the mainframe
- Make it easier for new people to learn z/OS.

To access the z/OS Basic Skills Information Center, open your Web browser to the following Web site, which is available to all users (no login required): <http://publib.boulder.ibm.com/infocenter/zos/basics/index.jsp>.

---

## **To request copies of IBM publications**

Direct your request for copies of any IBM document to your IBM representative or to the IBM branch office serving your locality.

There is also a toll-free customer support number (1-800-879-2755) available Monday through Friday from 8:30 a.m. through 5:00 p.m. Eastern time. You can use this number to:

- Order or inquire about IBM documents
- Resolve any software manufacturing or delivery concerns
- Activate the program reorder form to provide faster and more convenient ordering of software updates

---

## How to send your comments to IBM

We appreciate your input on this publication. Feel free to comment on the clarity, accuracy, and completeness of the information or give us any other feedback that you might have.

Use one of the following methods to send us your comments:

1. Send an e-mail to [mhvrcfs@us.ibm.com](mailto:mhvrcfs@us.ibm.com)
2. Visit the Contact z/OS Web page at <http://www.ibm.com/servers/eserver/zseries/zos/webqs.html>
3. Mail the comments to the following address:

IBM Corporation  
Attention: MHVRCFS Reader Comments  
Department H6MA, Building 707  
2455 South Road  
Poughkeepsie, NY 12601-5400  
US

4. Fax the comments to us as follows:

From the United State and Canada: 1+845+432-9405  
From all other countries: Your international access code +1+845+432-9405

Include the following information:

- Your name and address
- Your e-mail address
- Your telephone or fax number
- The publication title and order number:  
**z/OS V1R13.0 Security Server RACF Data Areas.**  
**GA22-7680-13**
- The topic and page number related to your comment
- The text of your comment

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

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

---

## If you have a technical problem

Do not use the feedback methods listed above. Instead, do one of the following:

- Contact your IBM service representative.
- Call IBM technical support.
- Visit the IBM zSeries support Web page at <http://www.ibm.com/servers/eserver/support/zseries/>.



---

## Summary of changes

This document contains terminology, maintenance, and editorial changes to improve consistency and retrievability. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

### Changes made in z/OS Version 1 Release 13

This document contains information previously presented in *Security Server RACF Data Areas, GA22-7680-12*, which supports z/OS Version 1 Release 12.

#### New information

- **IRRPAFC (AFC)**
  - New constants, AFC\_MOUNT\_NA, AFC\_MOUNT\_U, AFC\_MOUNT\_UNA, AFC\_UNMOUNT\_U, and AFC\_UNMOUNT\_UNA have been added.
- **IRRPCOMP (COMP)**
  - New constants, UMAP\_ID\_PROPAGATION\_LEN and UMAP\_TOTAL\_LEN have been added.
  - New function codes, UMAP\_DID\_TO\_R, CACH\_EXT\_STORE\_MULT, and CACH\_EXT\_ICRX\_VAL have been added.
  - New fields, UMAP\_DISTINGUISHED\_NAME and UMAP\_DISTINGUISHED\_NAME\_LEN have been added for distinguished name mapping.
  - New fields, UMAP\_REGISTRY\_NAME and UMAP\_REGISTRY\_NAME\_LEN have been added for registry or realm name mapping.
- **ICHRFXAP (FXAP)**
  - New fields, ARFXPVER, ARFXPLEN, ARFXUSED, ARFXNORM, ARFXOPER, ARFXTRST, and ARFXPRIV have been added.
- **IRRPICRX (ICRX)**
  - A new flag, ICRXFLGS, has been added.
- **ICHPISP (ISP)**
  - New fields, RPEDIDCT, RPEDIDLN, and RPEDIDOF, have been added for RACRPE.
- **ICHPRCVT (RCVT)**
  - New programming interfaces, RCVTIDPV and RCVTAUTU, have been added.
- **ICHRDXP (RDXP)**
  - New fields, RDXAUSED, RDXANORM, RDXASPEC, RDXAOPER, and RDXAEXIT have been added.

#### Changed information

- **ICHPRCVT (RCVT)**
  - RCVTVRMC has been changed to 7780 and a new constant RCVTVR72 has been added with that value.
- **ICHSAFP (SAFP)**
  - A new constant, SAFPR80 (with a value of 21) has been added. SAFPCURR has been changed to the same value (21).

### Changes made in z/OS Version 1 Release 12

This document contains information previously presented in *Security Server RACF Data Areas, GA22-7680-11*, which supports z/OS Version 1 Release 11.

#### New information

- **ICHGAPL (GAPL)**

- New fields, ATEMOBJS, ATEPRF64, and ATETMSTP have been added.
- IRRPCOMP (COMP)
  - A new constant, CDDL\_ATT\_SET\_MIN\_SERIAL, has been added.
- IRRPCOMX (COMX)
  - A new constant, CDDLX\_ATT\_SET\_MIN\_SERIAL, has been added.
- IRRPGANC(GANC)
  - A new macro, IRRPGANC, has been added.
- IRRGPRFL(GPRFL)
  - A new macro, IRRGPRFL, has been added.
- ICHPISP (ISP)
  - New field, RPEFSCPW, is added to RACRPE.
- ICHPRCVT (RCVT)
  - A new field, RCVTGANC, has been added.
- IRRPRXTW (RXTW)
  - New fields, EXTWRTAS and EXTWRTAD, are added to EXTWKEA.

#### Changed information

- ICHGAPL (GAPL)
  - The constant value for ATEVCON has been updated to 2.
- ICHPRCVT (RCVT)
  - RCVTVRMC has been changed to 7770 and a new constant RCVTVR71 has been added with that value.
- ICHSAFP (SAFP)
  - A new constant, SAFPRL7A (with a value of 20) has been added. SAFPCURR has been changed to the same value (20).

#### Changes made in z/OS Version 1 Release 11

This document contains information previously presented in *Security Server RACF Data Areas, GA22-7680-10*, which supports z/OS Version 1 Release 10.

#### New information

- IHAACEE (ACEE)
  - A new flag, ACEEDALY, has been added.
  - New fields, ACEEIDID and ACEETIME have been added.
- IRRPAFC (AFC)
  - A new constant, AFC\_UNAVAILABLE, has been added.
- IRRPENF2 (ENF2)
  - A new macro, IRRPENF2, has been added to support z/OS identity propagation.
- IRRPFC (FC)
  - New constant, IRRSPS00#, has been added for new R\_PgmSignVer callable service.
- IRRPICRX (ICRX)
  - A new macro, IRRPICRX, has been added to support z/OS identity propagation.
- IRRPIDID (IDID)
  - A new macro, IRRPIDID has been added to support z/OS identity propagation.
- IRRPCOMP (COMP)
  - New PGSN constants have been added for new R\_PgmSignVer callable service.



- New constants, ADMN\_XTR\_RESOURCE and ADMN\_XTR\_NEXT\_RESOURCE have been added.
- New parameter, INTAIDID\_AREA@ and a new flag, INTA\_LAST\_PARM\_IDID, have been added.
- New constant CACH\_LEN has been added.
- New constant, PKIS\_QRECOVER, has been added for R\_PKIServ callable service.
- New R\_PKIServ function specific parameter lists for QREC PKIS\_QREC\_EYECATCH, PKIS\_QREC\_RESULTL\_LEN, PKIS\_QREC\_RESULTL@, PKIS\_QREC\_NUMENTRIES, PKIS\_QREC\_CRIT\_EMAIL@, and PKIS\_QREC\_CRIT\_PASS@ have been added.
- PKIS\_EXP\_KEYID@ has been added to support PKI key generation.
- PKIS\_MODC\_REQUESTOREMAIL@ has been added to support PKI key generation.
- ICHPRCVT (RCVT)
  - New programming interfaces, RCVTDNL and RCVTRL, have been added.
  - A new field, RCVTGENT, has been added.
- IRRPOUSP (OUSP)
  - New segment fields have been added to support general resource profiles.
- IRRPRIPL (RIPL)
  - New HRF7760 parameters for INIT, INITPRMA, INITIDID, INITICRX, and INITENDA have been added.
- ICHRCXP (RCXP)
  - A new field, RCXAPROF, has been added.
- ICHRFXP (FXAP)
  - New fields, ARFXAPRT, ARFXAPRL, and ARFXAPRF, have been added.
- ICHRIXP (RIXP)
  - New parameters, RIXIDID and RIXICRX have been added.

#### Changed information

- ICHPRCVT (RCVT)
  - RCVTVRMC has been changed to 7760 and a new constant RCVTVR60 has been added with that value.
- ICHSAFP (SAFP)
  - A new constant, SAFPRL60 (with a value of 19) has been added. SAFPCURR has been changed to the same value (19).



## RACF Data Areas

ACEE: Accessor Environment Element	3
ACHKL: RACROUTE REQUEST=AUTH Parameter List (Request Section)	9
ACXP: ACEE Expansion/Compression Exit Parameter List	12
AFC: z/OS UNIX System Services Audit Function Codes	14
AUL: RACROUTE REQUEST=AUDIT Parameter List (Request Section)	17
CCXP: Command Preprocessing Exit (ICHCCX00) Parameter List	18
CDXP: Notify/Verify/List Exit Routines Parameter List	19
CGRP: Connect Group Name Table Definition	21
CNST/CNSX (RACF): Class Name and Syntax Table	23
CNST/CNSX (SAF): Class Name and Syntax Table	26
CNXP: Command Preprocessing Exit (ICHCNX00) Parameter List	27
COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services	30
COMX: 64-BIT common SAF/RACF parameter list for z/OS UNIX System Services security functions	75
COMY: 64-BIT enabled SAF callable services	92
CRED: z/OS UNIX System Services Credential Structure	94
CREI: z/OS UNIX System Services Credential Structure for IPC	97
DAUT: RACROUTE REQUEST=DIRAUTH Parameter List (Request Section)	99
DEXP: Data Encryption Exit Parameter List	101
DSDT: Data Set Descriptor Table	103
ENFP: RACF-Parameter List for ENF event code 62 listen exits	110
ENF2: RACF-Parameter List for ENF event code 71 listen exits	111
EVXP: RACF Command Exit Parameter List	112
FACL: z/OS UNIX System Services access control list	115
FAST: RACROUTE REQUEST=FASTAUTH Parameter List (Request Section)	118
FC: z/OS UNIX System Services Security Function Code Table	120
FXAP: RACROUTE REQUEST=FASTAUTH Extended Function Exit Parameter List	122
GANC: GENERICANCHOR settings mapping	124
GAPL: Generic anchor table entry	126
GPRFL: 64-bit Storage profile list mapping	129
GRPF: In-Storage Generic Profile Map	132
ICRX: Extended identity context reference	136
ICTX: Distributed Identity Support SAF.	138
IDID: Distributed identity data	140
IFSP: z/OS UNIX System Services File Security Packet	143
IISP: z/OS UNIX System Services IPC Security Packet	145
ISP: RACF In-Storage Profile	146
OUSP: initUSP Output Parameter List	152
PWX2: New Password Phrase Exit Parameter List	154
PWXP: Password Exit Parameter List	156
RCVI: Identity cache communication vendor.	159
RCVT: RACF Communication Vector Table	161
RCXP: RACROUTE REQUEST=AUTH Exit Parameter List	171
RDDFL: RACROUTE REQUEST=DEFINE Parameter List (Request Section)	178
RDXP: RACROUTE REQUEST=DEFINE Exit Parameter List	181
RFXP: RACROUTE REQUEST=FASTAUTH Exit Parameter List	189
RIPL: RACROUTE REQUEST=TOKENBLD/VERIFY/VERIFYX Parameter List (Request Section)	192
RIXP: RACROUTE REQUEST=VERIFY/VERIFYX Exit Parameter List	198
RLST: RACROUTE REQUEST=LIST Parameter List (Request Section)	206
RLX1P: RACROUTE REQUEST=LIST Exit Parameter List	208
RLX2P: RACROUTE REQUEST=LIST Selection Exit Parameter List	213
RNG: RACF Database Range Table	217
RRPF: Resident Profile Map	218
RSMXP: RACF Report Writer Selection Exit Parameter List	222
RUTKN: Resource/User Security Token	224
RXTL: RACROUTE REQUEST=EXTRACT Parameter List (Request Section)	227
RXTW: RACROUTE REQUEST=EXTRACT Result Area Mapping	230

SAFP: SAF Router Parameter List . . . . .	233
SAFR: Number of z/OS UNIX System Services Callable Services . . . . .	237
SAFV: SAF Router Vector Table . . . . .	238
SECUR: OS/390 Security Context . . . . .	240
SGNPL: RACROUTE REQUEST=SIGNON Parameter List (Request Section) . . . . .	241
SGX1P: RACROUTE REQUEST=SIGNON Parameter List Mapping . . . . .	243
STAT: RACROUTE REQUEST=STAT Parameter List (Request Section) . . . . .	244
TSRV: RACROUTE REQUEST=TOKENMAP/TOKENXTR Parameter List (Request Section) . . . . .	246
WORK: z/OS UNIX System Services Work Area for SAF and RACF . . . . .	247

## ACEE: Accessor Environment Element

NOT Programming Interface Information

The following fields are Not Programming Interface information:

ACEEAMP  
 ACEEMDLS  
 ACEECGRP  
 ACEECLCP  
 ACEEGATA  
 ACEEPADS  
 ACEE3PTY  
 ACEEOCOX  
 ACEEPTDS

End of NOT Programming Interface 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 TCBSSENV. ACEEs pointed to by ASXBSENV or TCBSSENV always reside below 16M.  
**Serialization:** None  
**Function:** Maps the ACEE; represents the authorities of a single accessor in the address space.

**Notes:**

1. If you use ACEEIEP, it must point to an area of storage you obtained using a GETMAIN. 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 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.

## ACEE

- 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*.

2. Within an IMS address space, ACEEAPTR is reserved for use by IMS during IMS initialization and signon.
3. 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.

4. 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.

5. The acronym at offset 0 is changed from ACEE to acee prior to freeing the ACEE storage.

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
0	(0)	STRUCTURE	192	ACEE	Accessor environment element
0	(0)	CHARACTER	4	ACEEACEE	Acronym in EBCDIC -ACEE-
4	(4)	ADDRESS	4	ACEECORE	ACEE subpool and length
4	(4)	ADDRESS	1	ACEESP	ACEE subpool number
5	(5)	ADDRESS	3	ACEELEN	Length of ACEE
8	(8)	UNSIGNED	1	ACEEVRSN	Version = 1.
9	(9)	CHARACTER	3	*	Reserved
12	(C)	ADDRESS	4	ACEEIEP	Reserved for installation. If used, it must point to a 1-byte subpool followed by a 3-byte length.
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	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	Group name information
29	(1D)	ADDRESS	1	ACEEGRPL	Group name length

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
30	(1E)	CHARACTER	8	ACEEGRPN	Valid connect group unless ACEEUSRI is '*' or '*BYPASS*'. For these two cases, ACEEGRPN is '*'.
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..		*	Reserved
		.... ..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
		.1.. ....		ACEERASP	1 - RACF address space
		.1.. ....		ACEECLNT	1 - Unauthenticated client
		...1 ....		ACEEACTL	1 - Authenticated client
		.... 1...		ACEETSKP	1 - Task level process
		.... .1..		ACEEIUSP	1 - INITUSP has been done
		.... ..1.		ACEEUID	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
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 - 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

# ACEE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
80	(50)	ADDRESS	4	ACEECLCP	Anchor for in-storage profile trees built by the RACLIST function
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 REQUEST=VERIFY macro, it is zero.
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 ccyyddF 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)	UNSIGNED	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 mask
		..1. ....		AfEED4OK	1 - ACEEPLCL points to a 1024-bit mask
		...1 ....		ACEEXNVR	1 - ACEEDAT4 contains data
		.... 1...		ACEESTOK	0 - ACEEDAT4 not used
		.... .1..		ACEENSTE	ENVR object created by another system
		.... ..1.		ACEEDALY	1 - An ACEE was built from a TOKEN with a SERVAUTH port of entry, and the SERVAUTH resource name is no longer available.
		.... ...1		*	On if nested ENVR object (in ACEENSTA field) should be used in auth check
134	(86)	BITSTRING	8	ACEEFLG6	1 - User logged on to an application which only records daily logon statistics.
		1... ....		ACEERAUI	Reserved
		.1.. ....		ACEERUAA	More miscellaneous flags
		..1. ....		ACEERUAV	Restricted access user ID
		...1 1111		*	"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)
135	(87)	CHARACTER	1	*	A check was made to RESTRICTED.FILESYS.ACCESS for this process so the value of ACEERUAA can be used
136	(88)	ADDRESS	4	ACEE3PTY	Reserved
140	(8C)	ADDRESS	4	ACEEPLCL	Reserved
144	(90)	CHARACTER	8	ACEESUID	Address of ACEE created by third-party RACHECK SVC processing
152	(98)	ADDRESS	4	ACEEOCOX	Pointer to extended class authorization mask, or 0
156	(9C)	CHARACTER	4	ACEEPTDS	Surrogate user ID (AUDIT)



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
160	(A0)	CHARACTER	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

**Constants**

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.

**Cross Reference**

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
ACEE	0		1	ACEEGRP	1D		2
ACEEACEE	0		2	ACEEGRPA	28	80	3
ACEEACLT	28	10	3	ACEEGRPL	1D		3
ACEEADSP	26	40	3	ACEEGRPN	1E		3
ACEEALTR	27	80	3	ACEEIEP	C		2
ACEEAMP	48		2	ACEEINST	10		2
ACEEAPDA	60		2	ACEEIUSP	28	04	3
ACEEAPLN	58		2	ACEELEN	5		3
ACEEAPLV	3A		2	ACEELOGU	26	08	3
ACEEAPTR	54		2	ACEEMDLS	68		2
ACEEAUDT	26	10	3	ACEEMODE	85	80	3
ACEECGRP	6C		2	ACEENONE	27	01	3
ACEECLCP	50		2	ACEENPWR	28	01	3
ACEECLNT	28	20	3	ACEENSTE	85	04	3
ACEECLTH	4C		2	ACEENSTA	B0		2
ACEECNTL	27	40	3	ACEEICTX	B4		2
ACEECORE	4		2	ACEEIDID	B8		2
ACEEDALY	85		3	ACEEOCOX	98		2
ACEEDASD	38	08	3	ACEEOPER	26	20	3
ACEEDATE	29		2	ACEEPADS	80		2
ACEEDAT4	7C		2	ACEEPLCL	8C		2
ACEEDSLP	78		2	ACEEPRIV	26	02	3
ACEEDUID	28	02	3	ACEEPROC	2C		2
ACEED4OK	85	20	3	ACEEPTDS	9C		2
ACEEDAT4	7C		2	ACEERACF	26	01	3
ACEEFCGP	74		2	ACEERASP	28	40	3
ACEEFLG1	26		2	ACEERAUI	86	80	3
ACEEFLG2	27		2	ACEEREAD	27	10	3
ACEEFLG3	28		2	ACEERUAA	86	40	3
ACEEFLG4	38		2	ACEERUAV	86	20	3
ACEEFLG5	85		2	ACEESLVL	84		2
ACEEFLG6	86		2	ACEESP	4		3
ACEEGATA	70		2	ACEESPEC	26	80	3

## ACEE

Name	Hex Offset	Hex Value	Level
ACEESRVA	A8		2
ACEESRVP	AC		2
ACEESTOK	85	08	3
ACEESUID	90		2
ACEETAPE	38	04	3
ACEETERM	38	02	3
ACEETIME	BC		2
ACEETOKP	A4		2
ACEETRDA	3C		2
ACEETRID	40		2
ACEETRLV	3B		2
ACEETRMP	34		2
ACEETSKP	28	08	3
ACEEUATH	38	20	3
ACEEUNAM	64		2
ACEEUPDT	27	20	3
ACEEUSER	14		2
ACEEUSRI	15		3
ACEEUSRL	14		3
ACEEVMASK	85	40	3
ACEEVRSN	8		2
ACEEXNVR	85		3
ACEEX5PR	A0		2
ACEE3PTY	88		2

## ACHKL: RACROUTE REQUEST=AUTH Parameter List (Request Section)

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	ACHKLIST	RACHECK PARAMETER LIST
0	(0)	ADDRESS	4	ACHKINSW	ADDRESS INSTALLATION DATA
0	(0)	UNSIGNED	1	ACHKLENG	LENGTH OF PARAMETER LIST
1	(1)	ADDRESS	3	ACHKINST	ADDRESS INSTALLATION DATA
4	(4)	SIGNED	4	ACHKENTW	ENTITY ADDRESS WORD
4	(4)	UNSIGNED	1	ACHKFLG1	FIRST FLAGS BYTE
		1... ....		ACHKRFI	RACFIND PARAMETER GIVEN
		.1.. ....		ACHKRFIY	RACFIND=YES
		..1. ....		ACHKENX	ENTITYX IS SPECIFIED
		...1 ....		ACHKDSTV	DSTYPE=V
		.... 1...		ACHK31IN	31-BIT-ADDRESS LIST INDICATOR
		.... .11.		ACHKLOGS	LOG=NOSTAT (BOTH ON)
		.... .1..		ACHKLOGF	LOG=NOFAIL
		.... ..1.		ACHKLOGN	LOG=NONE
		.... ...1		ACHKCSA	ENTITY=(ADDR,CSA)
5	(5)	ADDRESS	3	ACHKENT	ENTITY NAME ADDRESS
8	(8)	SIGNED	4	ACHKCLNW	CLASS NAME ADDRESS WORD
8	(8)	UNSIGNED	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)	UNSIGNED	1	ACHKFLG3	THIRD FLAGS BYTE
		1... ....		ACHKTAPE	DSTYPE=T
		.1.. ....		ACHKMDEL	DSTYPE=M
		..1. ....		ACHKPRF	PROFILE ADDR 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

## ACHKL

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
32	(20)	STRUCTURE	16	ACHK31	31-BIT-ADDRESS SAF EXTENSION
32	(20)	ADDRESS	4	ACHKIN31	31-BIT INSTALLATION DATA 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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
48	(30)	STRUCTURE	8	ACHK15	RACF 1.5 EXTENSION
48	(30)	ADDRESS	4	ACHKACC1	ACCLVL ADDRESS (1ST PART)
52	(34)	ADDRESS	4	ACHKACC2	ACCLVL ADDRESS (2ND PART)
56	(38)	CHARACTER		ACHK15EN	END OF V1.5 EXTENSION

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
56	(38)	STRUCTURE	4	ACHK17	RACF 1.7 EXTENSION
56	(38)	UNSIGNED	2	ACHKFSEQ	FILE SEQUENCE NO
58	(3A)	UNSIGNED	1	ACHKFLGT	TAPE FLAG BYTE
		11.. ....		ACHKTLBL	TAPELBL SPECIFIED B'00'=STD B'10'=BLP B'01'=NL
		..11 1111		*	RESERVED
59	(3B)	UNSIGNED	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		ACHK17EN	END OF RACF 1.7 EXTENSION

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
60	(3C)	STRUCTURE	8	ACHK18	RACF 1.8 EXTENSION
60	(3C)	ADDRESS	4	ACHKUSID	USERID POINTER
64	(40)	ADDRESS	4	ACHKGPID	GROUP NAME POINTER
68	(44)	CHARACTER		ACHK18EN	END OF 1.8 EXTENSION

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
68	(44)	STRUCTURE	4	ACHK18X	RACF 1.8X EXTENSION
68	(44)	ADDRESS	4	ACHKDDPR	DDNAME POINTER
72	(48)	CHARACTER		ACHK8XEN	END OF 1.8X EXTENSION

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
ACHKACCS	3B	10	3	ACHKUSID	3C		2
ACHKACC1	30		2	ACHKUTOK	4C		2
ACHKACC2	34		2	ACHKVOL	C	08	4
ACHKACEE	18		2	ACHKVOLS	D		3
ACHKAPPL	14		2	ACHKVOLW	C		2
ACHKCLN	9		3	ACHKVS31	2C		2
ACHKCLNW	8		2	ACHKWRON	3B	20	3
ACHKCL31	28		2	ACHK15	30		1
ACHKCSA	4	01	4	ACHK15EN	38		2
ACHKDDPR	44		2	ACHK17	38		1
ACHKDSTV	4	10	4	ACHK17EN	3C		2
ACHKEND	20		2	ACHK18	3C		1
ACHKENT	5		3	ACHK18EN	44		2
ACHKENTW	4		2	ACHK18X	44		1
ACHKENTX	24		2	ACHK19	48		1
ACHKENX	4	20	4	ACHK19EN	5C		2
ACHKEN31	24		3	ACHK31	20		1
ACHKEOS	3B	80	3	ACHK31EN	30		2
ACHKEVD	3B	40	3	ACHK31IN	4	08	4
ACHKFLGT	3A		2	ACHK8XEN	48		2
ACHKFLG1	4		3				
ACHKFLG2	8		3				
ACHKFLG3	C		3				
ACHKFLG4	3B		2				
ACHKFSEQ	38		2				
ACHKGEN	C	04	4				
ACHKGPID	40		2				
ACHKINST	1		3				
ACHKINSW	0		2				
ACHKIN31	20		2				
ACHKLENG	0		3				
ACHKLIST	0		1				
ACHKLOGF	4	04	5				
ACHKLOGN	4	02	5				
ACHKLOGS	4	06	4				
ACHKLSTR	54		2				
ACHKMDEL	C	40	4				
ACHKOVOL	10		2				
ACHKOWNR	1C		2				
ACHKPRF	C	20	4				
ACHKPRI	C	02	4				
ACHKRCVR	58		2				
ACHKRFI	4	80	4				
ACHKRFIY	4	40	4				
ACHKRTOK	50		2				
ACHKTALT	8	80	4				
ACHKTAPE	C	80	4				
ACHKTCTL	8	08	4				
ACHKTLBL	3A	C0	3				
ACHKTRD	8	02	4				
ACHKTUPD	8	04	4				

## ACXP: ACEE Expansion/Compression Exit Parameter List

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		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

Format of the range table supplied to or to be supplied by exit IRRACX01:

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		RANGE_TABLE	
0	(0)	SIGNED	4	RANGE_COUNT	Number of ranges in the table
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:

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		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)	

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
FIRST_RANGE	8		2	NEXT_RANGE	8		2
IRRACX01_ACEEPTR	4		2	RANGE_COUNT	0		2
IRRACX01_OPERATION	0		2	RANGE_END	4		2
IRRACX01_TABLE_PTR	8		2	RANGE_START	0		2

Name	Hex Offset	Hex Value	Level
RANGE_SUBPOOL	4		2
RECALL_OPERATION	0	2	2
STASH_OPERATION	0	1	2

---

**AFC: z/OS UNIX System Services Audit Function Codes**

**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

**Constants**

Len	Type	Value	Name	Description
2	DECIMAL	1	AFC_ACCESS	Check file accessibility
2	DECIMAL	2	AFC_CHAUDIT_U	Change user audit options
2	DECIMAL	3	AFC_CHDIR	Change current working directory
2	DECIMAL	4	AFC_CHMOD	Change file modes
2	DECIMAL	5	AFC_CHOWN	Change owner and group of file
2	DECIMAL	6	AFC_DUB	Initialize a process
2	DECIMAL	7	AFC_EXEC	Execute a file with NEWJOBNAME
2	DECIMAL	8	AFC_FCHAUDIT_U	Change user audit options when file is open
2	DECIMAL	9	AFC_FCHMOD	Change file modes when file is open
2	DECIMAL	10	AFC_FCHOWN	Change owner and group of file when open
2	DECIMAL	11	AFC_GETCWD	Get current working directory
2	DECIMAL	12	AFC_GETPSENT	Get process entry
2	DECIMAL	13	AFC_KILL	Signal a process
2	DECIMAL	14	AFC_LINK	Link to a file
2	DECIMAL	15	AFC_LSTAT	Get file status; do not resolve ending symlink
2	DECIMAL	16	AFC_MKDIR	Make a directory
2	DECIMAL	17	AFC_MKNOD	Make a file node
2	DECIMAL	18	AFC_MOUNT	Mount a file system (using the NOSETUID operand)
2	DECIMAL	19	AFC_OPEN	Open a file
2	DECIMAL	20	AFC_OPENDIR	Open a directory
2	DECIMAL	21	AFC_PATHCONF	Get configurable path name variables
2	DECIMAL	22	AFC_PTRACE	Debug a process
2	DECIMAL	23	AFC_READLINK	Read a symbolic link
2	DECIMAL	24	AFC_RENAME	Rename a file
2	DECIMAL	25	AFC_RMDIR	Remove a directory
2	DECIMAL	26	AFC_SETEGID	Set effective GID
2	DECIMAL	27	AFC_SETEUID	Set effective UID
2	DECIMAL	28	AFC_SETGID	Set real/saved and/or effective GID
2	DECIMAL	29	AFC_SETUID	Set real/saved and/or effective UID
2	DECIMAL	30	AFC_STAT	Get file status
2	DECIMAL	31	AFC_SYMLINK	Create a symbolic link
2	DECIMAL	32	AFC_UNLINK	Remove directory entries (Delete a file)
2	DECIMAL	33	AFC_UNMOUNT	Unmount a file system (with the NOSETUID operand)
2	DECIMAL	34	AFC_UTIME	Set file access/modification times
2	DECIMAL	35	AFC_UNDUB_EXIT	Terminate a process
2	DECIMAL	36	AFC_WRITE	Write to a file (Clear setid bits)



Len	Type	Value	Name	Description
2	DECIMAL	37	AFC_CHAUDIT_A	Change auditor audit options
2	DECIMAL	38	AFC_FCHAUDIT_A	Change auditor audit options when file is open
2	DECIMAL	39	AFC_LOOKUP	Path name resolution
2	DECIMAL	40	AFC_TTYNAME	Get path name of terminal
2	DECIMAL	41	AFC_IOCTL	Get path name
2	DECIMAL	42	AFC_GETMNT	Get mount entry
2	DECIMAL	43	AFC_QUIESCE	Quiesce a file system (with the NOSETUID operand)
2	DECIMAL	44	AFC_UNQUIESCE	Unquiesce a file system (with the NOSETUID operand)
2	DECIMAL	45	AFC_VREGISTER	Server registration
2	DECIMAL	46	AFC_VRESOLVEPN	Server resolve path name
2	DECIMAL	47	AFC_VLOOKUP	Server lookup
2	DECIMAL	48	AFC_VREADWRITE	Server read write
2	DECIMAL	49	AFC_VREADDIR	Server read directory
2	DECIMAL	50	AFC_SIGACTION	Change Osigset action
2	DECIMAL	51	AFC_VCREATE	Server create
2	DECIMAL	52	AFC_VMAKEDIR	Server make directory
2	DECIMAL	53	AFC_VSYMLINK	Server symbolic link
2	DECIMAL	54	AFC_VSETATTR	Server set file attributes
2	DECIMAL	55	AFC_VLINK	Server link
2	DECIMAL	56	AFC_VREMOVEDIR	Server remove directory
2	DECIMAL	57	AFC_VREMOVE	Server remove
2	DECIMAL	58	AFC_VRENAME	Server rename
2	DECIMAL	59	AFC_CHATTR	Change file attributes
2	DECIMAL	60	AFC_FCHATTR	Change file attributes for open file
2	DECIMAL	61	AFC_THLMT	Set thread limit
2	DECIMAL	62	AFC_MSGCTL	Message control
2	DECIMAL	63	AFC_MSGGET	Message obtain
2	DECIMAL	64	AFC_MSGRCV	Message receive
2	DECIMAL	65	AFC_MSGSND	Message send
2	DECIMAL	66	AFC_SEMCTL	Semaphore control
2	DECIMAL	67	AFC_SEMGET	Get set of semaphores
2	DECIMAL	68	AFC_SEMOP	Semaphore operations
2	DECIMAL	69	AFC_SHMAT	Shared memory attach
2	DECIMAL	70	AFC_SHMCTL	Shared memory control
2	DECIMAL	71	AFC_SETREGID	Set real and/or effective GID
2	DECIMAL	72	AFC_SHMGET	Shared memory get
2	DECIMAL	73	AFC_WGETIPC	Query IPC status
2	DECIMAL	74	AFC_REMOVE	Remove
2	DECIMAL	75	AFC_SET_MODE	Set mode
2	DECIMAL	76	AFC_SET_MSGQB	Set message queue maximum bytes
2	DECIMAL	77	AFC_SET_GID	Set supplementary groups
2	DECIMAL	78	AFC_PASSWORD	Verify password
2	DECIMAL	79	AFC_LCHOWN	Change owner and group of a symbolic link
2	DECIMAL	80	AFC_TRUNCATE	Truncate a file
2	DECIMAL	81	AFC_PFSCTL	Control function for the physical file system.
2	DECIMAL	82	AFC_SETRLIMIT	Set maximum resource consumption
2	DECIMAL	83	AFC_SETPRIORITY	Set process scheduling priority
2	DECIMAL	84	AFC_NICE	Change priority of a process
2	DECIMAL	85	AFC_SETREUID	Set real and/or effective UID
2	DECIMAL	86	AFC_WRITEV	Write on a file
2	DECIMAL	87	AFC_FCHDIR	Change working directory
2	DECIMAL	88	AFC_CHROOT	Change root directory
2	DECIMAL	89	AFC_REALPATH	Resolve path name
2	DECIMAL	90	AFC_STATVFS	Get file system information
2	DECIMAL	91	AFC_BIND	Bind a name to a socket
2	DECIMAL	92	AFC_SOCKET	Create an endpoint for communication
2	DECIMAL	93	AFC_THREAD_SEC	Thread level security
2	DECIMAL	94	AFC_AUTHCHECK	Authority check
2	DECIMAL	95	AFC_ACC_SEND	Send access rights
2	DECIMAL	96	AFC_ACC_RECV	Receive access rights
2	DECIMAL	97	AFC_ACC_DISC	Discard access rights
2	DECIMAL	98	AFC_NEWGRP	Newgrp shell utility
2	DECIMAL	99	AFC_CONSOLE	Console communication service
2	DECIMAL	100	AFC_SERV_INIT	WLM service
2	DECIMAL	101	AFC_SPAWN	SpawnUSERID
2	DECIMAL	102	AFC_SWAP_SERV	Swap services

## AFC

Len	Type	Value	Name	Description
2	DECIMAL	103	AFC_WLMC	WLM C and C++
2	DECIMAL	104	AFC_LOGIN	__LOGIN SYSTEM CALL
2	DECIMAL	105	AFC_MOUNT_SETUID	Mount a file system (using the SETUID operand)
2	DECIMAL	106	AFC_UNMOUNT_SETUID	Unmount a file system (using the SETUID operand)
2	DECIMAL	107	AFC_QUIESCE_SETUID	Quiesce a file system (using the SETUID operand)
2	DECIMAL	108	AFC_UNQUIESCE_SETUID	Unquiesce a file system (using the SETUID operand)
2	DECIMAL	109	AFC_CHMOUNT	Change mount
2	DECIMAL	110	AFC_CHMOUNT_SETUID	Change mount (setuid)
2	DECIMAL	111	AFC_SETFACL	Add, alter, or delete an access control list
2	DECIMAL	112	AFC_SHUTDOWN_REG	Shutdown registration
2	DECIMAL	113	AFC_EACCESS	Check effective access
2	DECIMAL	114	AFC_SETFSECL	Set security label of a UNIX file or directory
2	DECIMAL	115	AFC_POE	System port of entry address
2	DECIMAL	116	AFC_LCHATTR	Change file attributes
2	DECIMAL	117	AFC_UNAVAILABLE	Audit function code not available
2	DECIMAL	118	AFC_MOUNT_NA	Mount no audit
2	DECIMAL	119	AFC_MOUNT_U	User mount
2	DECIMAL	120	AFC_MOUNT_UNA	User mount no audit
2	DECIMAL	121	AFC_UNMOUNT_U	User unmount
2	DECIMAL	122	AFC_UNMOUNT_UNA	User unmount no audit
2	DECIMAL	123	AFC_ENDOF_TAB	End of table

## AUL: RACROUTE REQUEST=AUDIT Parameter List (Request Section)

**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 SAFR plus offset at SAFPRACP

**Serialization:** None

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	AUDLIST	RACAUDIT parameter list
0	(0)	UNSIGNED	2	AUDVERS	Parameter list version
2	(2)	UNSIGNED	2	AUDLEN	Parameter list length
4	(4)	ADDRESS	4	AUDEVENT	Address of event name
8	(8)	UNSIGNED	2	AUDEQUAL	Event code qualifier
10	(A)	UNSIGNED	2	*	Reserved
12	(C)	ADDRESS	4	AUDCLASS	Address of class name
16	(10)	ADDRESS	4	AUDENTYX	Address of entity name
20	(14)	ADDRESS	4	AUDACEE	Address of ACEE
24	(18)	ADDRESS	4	AUDLOGST	Address of LOGSTR data
28	(1C)	UNSIGNED	1	AUDRESUL	Result byte
29	(1D)	UNSIGNED	3	*	Reserved
32	(20)	ADDRESS	4	* (4)	Reserved

### Cross Reference

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

## CCXP: Command Preprocessing Exit (ICHCCX00) Parameter List

**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

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

## CDXP: Notify/Verify/List Exit Routines Parameter List

### NOT Programming Interface Information

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

CDXCLPT	CDXENPT	CDXOPTP	CDXSGNM
CDXCPPL	CDXKYNM	CDXPFNM	CDXUWPT
CDXDWPT	CDXNTRYX		

### End of NOT Programming Interface 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 the IRRVAF1A exit.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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... .... Add .1... .... Alter ..1... .... Delete ...1... .... List .... 1111 Reserved
38	(26)	BITSTRING	1	CDXFLAGS	Miscellaneous flags

## CDXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
				CDXNTRYX	.... 1... 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)	CHARACTER	2	CDXFLAG2	Reserved
92	(5C)			CDXADDL	Additional parameter list if any. See mappings below for different parameter lists.
Additional parameter list for validation (verify) exit only					
92	(5C)	SIGNED	4	CDXVMSK	Verify 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

## Cross Reference

Name	Hex Offset	Hex Value	Level
CDXADDL	5A	5C	2
CDXANMSK	64		2
CDXCLTP	25		2
CDXCPPL	4		2
CDXDAPT	8		2
CDXDWPT	C		2
CDXENPT	14		2
CDXFLAGS	26		2
CDXFLAG2	5A		2
CDXCLPT	27		2
CDXKYNM	30		2
CDXLFPT	60		2
CDXLHPT	5C		2
CDXNXPT	0		2
CDXOPTP	26		2
CDXORMSK	68		2
CDXPFNM	1C		2
CDXPFTP	24		2
CDXSGNM	28		2
CDXTMLN	58		2
CDXTMNM	50		2
CDXUWPT	10		2
CDXVMSK	5C		2

## CGRP: Connect Group Name Table Definition

NOT 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.

End of NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	CGRP	Connect group name table
0	(0)	CHARACTER	32	CGRPHADR	CGRP header
0	(0)	CHARACTER	4	CGRPID	Table ID
4	(4)	CHARACTER	4	CGRPCORE	CGRP subpool and length
4	(4)	UNSIGNED	1	CGRPSP	Subpool number
5	(5)	ADDRESS	3	CGRPLEN	Length of CGRP
8	(8)	SIGNED	2	CGRPNUM	Maximum entries in table
10	(A)	UNSIGNED	1	CGRPVRSN	Version = 1
11	(B)	CHARACTER	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
24	(18)	CHARACTER	8	*	Reserved
32	(20)	CHARACTER	24	CGRPENT (*)	Group name entry
32	(20)	CHARACTER	8	CGRPNAME	Group name
40	(28)	BITSTRING	1	CGRPIND	Indicators for this entry
		1... ..		CGRPCHK	Always zero, was revoke indicator
		.1... ..		CGRPPREFR	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.

## CGRP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
41	(29)	.... 1111 BITSTRING	1	* CGRPAUTH	Reserved Group authority indicators
		1... .... .1.. ....		CGRPSPEC *	On if group-special authority Reserved
		..1. ....		CGRPOPER	On if group-operations authority
		...1 ....		CGRPAUDT	On if group-auditor authority
42	(2A)	.... 1111 SIGNED	2	* CGRPGPNM	Reserved Number of entries in group authority table related to this connect group
44	(2C)	SIGNED	4	CGRPGPTE	Address of first group authority table entry related to this connect group
48	(30)	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.
50	(32)	CHARACTER	6	*	Reserved
56	(38)	CHARACTER		*	End of entry

## Constants

Len	Type	Value	Name	Description
4	CHARACTER	CGRP	CGRPTID	Table ID

## Cross Reference

Name	Hex Offset	Hex Value	Level
CGRP	0		1
CGRPAUDT	29	10	4
CGRPAUTH	29		3
CGRPCHK	28	80	4
CGRPCOMP	28	20	4
CGRPCORE	4		3
CGRPENT	20		2
CGRPGCHG	14		3
CGRPGPAT	10		3
CGRPGPNM	2A		3
CGRPGPTE	2C		3
CGRPHADR	0		2
CGRPID	0		3
CGRPIND	28		3
CGRPLEN	5		4
CGRPNAME	20		3
CGRPNUM	8		3
CGRPOPER	29	20	4
CGRPPROP	28	10	4
CGRPPREFR	28	40	4
CGRPSP	4		4
CGRPSPEC	29	80	4
CGRPSUPG	30		3
CGRPSYNC	C		3
CGRPVRSN	A		3



## CNST/CNSX (RACF): Class Name and Syntax Table

NOT Programming Interface Information

The following fields are not programming interface information:

- CNSTGNLP
- CNSTRCLP
- CNSTSTKN

End of NOT Programming Interface 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	CNST	Class name/syntax table
0	(0)	SIGNED	2	CNSTLGT	Length of CNST portion of entry
2	(2)	UNSIGNED	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	Class member name syntax
19	(13)	UNSIGNED	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
		.... 1111		*	Reserved
21	(15)	BITSTRING	1	CNSTREMN	Syntax of remaining character
		1... ..		CNSTRALP	1 => Alphabetic char allowed
		.1.. ..		CNSTRNAT	1 => National char allowed

## CNST/CNSX (RACF)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		..1. ....		CNSTRNUM	1 => Numeric char allowed
		...1 ....		CNSTRSPE	1 => Special char allowed
		.... 1111		*	Reserved
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.		*	Reserved
		.... ...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		*	Force alignment

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	152	CNSX	Class entry extension
0	(0)	BITSTRING	128	CNSTVMSK	Longer mask for option flag
0	(0)	BITSTRING	16	CNSTLMSK	Long mask for option flag
0	(0)	BITSTRING	4	CNSTMASK	Mask for old option flags
4	(4)	BITSTRING		CNSTMSKS	Used by ICHSEC00 to find old mapping length
128	(80)	ADDRESS	4	CNSTGNLP	Pointer to GENLISTed CDT profiles
132	(84)	ADDRESS	4	CNSTRCLP	Pointer to RACLISTed CDT profiles
136	(88)	UNSIGNED	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)	UNSIGNED	1	CNSTKEYQ	KEYQUAL ==> Number of significant qualifiers
139	(8B)	UNSIGNED	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 0 ==> This is a real CDT
		...1 ....		CNSTNGEN	1 ==> SETR GENERIC is not allowed for the class
		.... 1111		*	Reserved
141	(8D)	CHARACTER	3	*	Reserved (align to doubleword boundary)
144	(90)	CHARACTER	8	CNSTSTKN	STOKEN for data space used to RACLIST profiles
152	(98)	CHARACTER		*	Force alignment

## Cross Reference

Name	Hex Offset	Hex Value	Level
CNST	0		1
CNSTACEE	17	40	3
CNSTALTR	16	80	3
CNSTCASE	89	06	3
CNSTCNSX	18		2
CNSTCNTL	16	40	3
CNSTCOPY	8C	20	3
CNSTDDUP	8C	40	3
CNSTNGEN	8C	10	3
CNSTDFRC	88		2
CNSTDSPC	17	04	3
CNSTDUP	89	08	3
CNSTDYN	8C	80	3
CNSTEXEC	16	08	3
CNSTEMAC	89	04	3
CNSTFALP	14	80	4
CNSTFLG0	89		2
CNSTFLG1	8C		2
CNSTFNAT	14	40	4
CNSTFNUM	14	20	4
CNSTFRST	14		3
CNSTFSPE	14	10	4
CNSTGENL	17	08	3
CNSTGNLP	80		2
CNSTID	2		2
CNSTKEYQ	8A		2
CNSTLGT	0		2
CNSTLMSK	0		3
CNSTMASK	0		4
CNSTMAXL	13		3
CNSTMFLG	17		2
CNSTMSKS	4		4
CNSTNAME	3		2
CNSTNONE	16	01	3
CNSTOPER	17	20	3
CNSTORML	8B		2
CNSTOWNR	17	01	3
CNSTPRDF	89	40	3
CNSTRACL	17	10	3
CNSTRALP	15	80	4
CNSTRCLP	84		2
CNSTREAD	16	10	3
CNSTREMN	15		3
CNSTRGRP	17	80	3
CNSTRLRQ	89	80	3
CNSTRMAC	89	10	3
CNSTRNAT	15	40	4
CNSTRNUM	15	20	4
CNSTRSPE	15	10	4
CNSTSGNL	89	02	3
CNSTSNTX	13		2
CNSTSTKN	90		2
CNSTUACC	16		2
CNSTUPDT	16	20	3
CNSTUSLB	89	20	3
CNSTVMSK	0		2
CNSTXFLG	17	02	3
CNSTXREF	B		2
CNSX	0		1

**CNST/CNSX (SAF): Class Name and Syntax Table**

**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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	CNST	Class name/syntax table
0	(0)	SIGNED	2	CNSTLGT	Length of CNST portion of entry
3	(3)	CHARACTER	8	CNSTNAME	Class name
8	(8)	CHARACTER	11	*	Reserved
19	(13)	UNSIGNED	1	CNSTMAXL	Maximum length of member name
20	(14)	BITSTRING	4	*	Reserved
24	(18)	ADDRESS	4	CNSTCNSX	Address of the rest of the class entry fields in CNSX
28	(1C)	CHARACTER		*	Force alignment

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	152	CNSX	Class entry extension
0	(0)	CHARACTER	139	*	Reserved
139	(8B)	UNSIGNED	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

**CNXP: Command Preprocessing Exit (ICHCNX00) Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		CNXPL	
Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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		CNXRACHK	X'0100' RACHECK
		BITSTRING		CNXRDDEF	X'0201' RACDEF DEFINE
		BITSTRING		CNXRDREN	X'0202' RACDEF RENAME
		BITSTRING		CNXRDADV	X'0203' RACDEF ADDVOL
		BITSTRING		CNXRDDEL	X'0204' RACDEF DELETE
		BITSTRING		CNXRDCHV	X'0205' RACDEF CHGVOL
		BITSTRING		CNXADSET	X'0301' ADDSD SET
		BITSTRING		CNXADNOS	X'0302' ADDSD NOSET
		BITSTRING		CNXADMOD	X'0303' ADDSD MODEL
		BITSTRING		CNXALSET	X'0401' ALTDSD SET
		BITSTRING		CNXALNOS	X'0402' ALTDSD NOSET
		BITSTRING		CNXDLSET	X'0501' DELDSD SET
		BITSTRING		CNXDLNOS	X'0502' DELDSD NOSET
		BITSTRING		CNXLDPRE	X'0601' LISTDSD prelocate call
		BITSTRING		CNXLDDS	X'0602' LISTDSD DATASET
		BITSTRING		CNXLDIDP	X'0603' LISTDSD ID or PREFIX
		BITSTRING		CNXPERTO	X'0701' PERMIT TO-resource
		BITSTRING		CNXPERFR	X'0702' PERMIT FROM-resource
		BITSTRING		CNXSRCPR	X'0801' SEARCH prelocate
		BITSTRING		CNXSRCPO	X'0802' SEARCH postlocate
		BITSTRING		CNXUT100	X'0900' IRRUT100
		BITSTRING		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

# CNXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		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-aligned.
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-aligned.
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-aligned, 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-aligned 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). For information about the prelocate call to ICHCNX00 from LISTDSD and SEARCH, see the discussion of ICHCNX00 processing in the <i>z/OS Security Server RACF System Programmer's Guide</i> .
32	(20)	ADDRESS	4	CNXQUAL	Qualifier address: points to an 8-byte field containing the data set qualifier. The qualifier is left-aligned, 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 discussion of ICHCNX00 processing in the <i>z/OS Security Server RACF System Programmer's Guide</i> . 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 user ID or a group defined to RACF. For the other commands and IRRUT100, RACF determines if the value is a user ID.
36	(24)	ADDRESS	4	CNXDSTYP	Data set type address: points to a 1-byte flag field indicating the type of data set. The use of this field is explained in more detail in the discussion of ICHCNX00 processing in the <i>z/OS Security Server RACF System Programmer's Guide</i> .
		.... ...1		CNXUNKWN	X'01' unknown
		.1.. ....		CNXGRPDS	X'40' group data set
		1... ....		CNXUSRDS	X'80' user data set
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 discussion of ICHCNX00 processing in the <i>z/OS Security Server RACF System Programmer's Guide</i> , 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
44	(2C)	ADDRESS	4	CNXCPL	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 discussion of ICHCNX00 processing in the <i>z/OS Security Server RACF System Programmer's Guide</i> . The address is zero in non-TSO cases.

## Cross Reference

Name	Hex Offset	Hex Value	Level
CNXADMOD	4	303	2
CNXADNOS	4	302	2
CNXADSET	4	301	2
CNXALNOS	4	402	2
CNXALSET	4	401	2
CNXALTCR	8	80	2
CNXALTER	28	80	2
CNXAUTH	28		2
CNXCALLR	4		2
CNXCPL	2C		2
CNXDLNOS	4	502	2
CNXDLSET	4	501	2
CNXDSTYP	24		2
CNXFLAG	8		2
CNXGRPDS	24	40	2
CNXLDDS	4	602	2
CNXLDIDP	4	603	2
CNXLDPRE	4	601	2
CNXLEN	0		2
CNXNONE	28	1	2
CNXOLDNM	10		2
CNXOLVOL	18		2
CNXPERFR	4	702	2
CNXPERTO	4	701	2
CNXQUAL	20		2
CNXRACHK	4	100	2
CNXRDADV	4	203	2
CNXRDCHV	4	205	2
CNXRDDEF	4	201	2
CNXRDDEL	4	204	2
CNXRDREN	4	202	2
CNXREAD	8	8	2
CNXRESCL	1C		2
CNXRESNM	C		2
CNXRXTRT	4	D00	2
CNXSRCPO	4	802	2
CNXSRCPR	4	801	2
CNXUNKWN	24	1	2
CNXUSRDS	24	80	2
CNXUT100	4	900	2
CNXVOL	14		2

---

**COMP: Common SAF/RACF Parameter List for z/OS UNIX System Services**

<b>Common Name:</b>	Common SAF/RACF parameter list for z/OS UNIX System Services security functions	
<b>Macro ID:</b>	IRRPCOMP	
<b>DSECT Name:</b>	COMP, IUSP, CSID, EXID, GETG, CHKP, GMAP, CKPO, QRYS, CMOD, CLID, CAUD, COWN, UMSK, KACC, QRYF, KFOW, MKRT, PTRC, MFSP, RAUD, GUGP, FORK, MISP, IACC, IOWN, CKO2, GETE, DKEY, DINF, DRUR, DAUT, GINF, RAUX, INTA, ADMN, UMAP, CDDL, KERB, TKTS, PKIS, CACH, PRXY, RACL, PGSN, WPRV, SECL	
<b>Owning Component:</b>	Resource Access Control Facility (SC1BN)	
<b>Eye-Catcher ID:</b>	None	
<b>Storage Attributes:</b>	Subpool	N/A
	Key	Any
	Residency	Invoker's primary address space
<b>Size:</b>	<b>Section</b>	<b>Size</b>
	COMP	28 bytes
	IUSP	8 bytes
	CSID	16 bytes
	EXID	40 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
	GINF	28 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



<b>Size (continued):</b>	<b>Section</b>	<b>Size</b>
	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	12 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	COMP	Common SAF/RACF plist.
0	(0)	ADDRESS	4	COMP_WORKA_STOR@	Addr of 1024-byte work area.
4	(4)	ADDRESS	4	COMP_SAFRC_ALET@	Addr of ALET for SAF RC.
8	(8)	ADDRESS	4	COMP_SAFRC_STOR@	Addr of SAF Return Code.
12	(C)	ADDRESS	4	COMP_RACRC_ALET@	Addr of ALET for RACF Return Code.
16	(10)	ADDRESS	4	COMP_RACRC_STOR@	Addr of RACF Return Code.
20	(14)	ADDRESS	4	COMP_RACSC_ALET@	Addr of ALET for RACF Reason Code.
24	(18)	ADDRESS	4	COMP_RACSC_STOR@	Addr of RACF Reason Code.
28	(1C)	CHARACTER	0	COMP_SERVR_PARAMS	Service routine parameter. lists begin here.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	8	IUSP	
0	(0)	ADDRESS	4	IUSP_WORKA_ALET@	Addr of ALET for the output area.
4	(4)	ADDRESS	4	IUSP_WORKA_STOR@	Addr of the 4-byte output area. The output area contains an addr that points to data about a user.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	CSID	
0	(0)	ADDRESS	4	CSID_ID_ALET@	Addr of ALET for the UID/GID.
4	(4)	ADDRESS	4	CSID_ID@	Addr of the UID/GID.
8	(8)	ADDRESS	4	CSID_IDOA_ALET@	Addr of ALET for output area.
12	(C)	ADDRESS	4	CSID_IDOA@	Addr of a 3 word output area. This area contains the new real, effective, and saved UIDs/GIDs.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	EXID	
0	(0)	ADDRESS	4	EXID_FLAG_ALET@	Address of ALET for a byte flag.
4	(4)	ADDRESS	4	EXID_FLAG@	Address of a 1-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@	Addr of the UID to be set.
16	(10)	ADDRESS	4	EXID_GID_ALET@	Addr of ALET for GID to be set.
20	(14)	ADDRESS	4	EXID_GID@	Addr of the GID to be set.
24	(18)	ADDRESS	4	EXID_UIDO_ALET@	Addr of ALET for UID output area.
28	(1C)	ADDRESS	4	EXID_UIDO@	Addr 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@	Addr of ALET for GID output area.
36	(24)	ADDRESS	4	EXID_GIDO@	Addr of GID output area, which is 3 words filled in by IRRREX00 with the real, effective and saved GIDs.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	GINF	
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	GETG	
0	(0)	ADDRESS	4	GETG_KEY_ALET@	Addr 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@	Addr of ALET for the Group Count.
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@	Addr of ALET for the Group List.
20	(14)	ADDRESS	4	GETG_GLST@	Addr of the Group List area in which supplemental GIDs are returned.
24	(18)	ADDRESS	4	GETG_GNUM_ALET@	Addr of ALET for number of groups.
28	(1C)	ADDRESS	4	GETG_GNUM@	Addr of the Number of Groups. (A one word area in which the number of supplemental group GIDs is returned.)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	8	CHKP	
0	(0)	ADDRESS	4	CHKP_AUFC_ALET@	Addr of ALET for audit function code.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
4	(4)	ADDRESS	4	CHKP_AUFC@	Addr of the audit function code. (A fullword containing the function code that identifies the system call being processed.)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	GMAP	
0	(0)	ADDRESS	4	GMAP_FLAG_ALET@	Address of ALET for 1-byte flag
4	(4)	ADDRESS	4	GMAP_FLAG@	Address of the 1-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 or ID left-aligned and padded with blanks.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	CKPO	
0	(0)	ADDRESS	4	CKPO_REQT_ALET@	Address of ALET for 1-byte Request type.
4	(4)	ADDRESS	4	CKPO_REQT@	Address of the 1-byte request type: request from KILL used when SIGCONT signal is being sent.
8	(8)	ADDRESS	4	CKPO_UIDS_ALET@	request.
12	(C)	ADDRESS	4	CKPO_UIDS@	request. Address of ALET for UIDs area.
16	(10)	ADDRESS	4	CKPO_PID_ALET@	Address of a 3 word area containing real, effective, and saved UIDs (in that order) for the target process, followed by an 8-byte security label when request type is 5.
20	(14)	ADDRESS	4	CKPO_PID@	Address of ALET for target PID.
24	(18)	ADDRESS	4	CKPO_CODE_ALET@	Address of the words containing the target process's PID.
28	(1C)	ADDRESS	4	CKPO_CODE@	Address of ALET for signal code.
					Address of word containing a code identifying signal being sent. This word is ignored for request type 3.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	QRYS	
0	(0)	ADDRESS	4	QRYS_OPTN_CODE_ALET@	Address of ALET for 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@P8C x'00000002' - POSIX_SAVED_IDS op@P8C
8	(8)	ADDRESS	4	QRYS_OUTP_VAL_ALET@	Address of ALET for one word output value

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
12	(C)	ADDRESS	4	QRY5_OUTP_VAL@	Address of one word output value.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	CMOD	
0	(0)	ADDRESS	4	CMOD_MODE_ALET@	Address of the ALET for the mode structure.
4	(4)	ADDRESS	4	CMOD_MODE@	Address of a one word mode parameter.
8	(8)	ADDRESS	4	CMOD_FSP_ALET@	Address of the ALET for the FSP structure.
12	(C)	ADDRESS	4	CMOD_FSP@	Address of the FSP structure.
16	(10)	ADDRESS	4	CMOD_FID_ALET@	Address of the 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 the ALET for the CRED structure.
28	(1C)	ADDRESS	4	CMOD_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	CLID	
0	(0)	ADDRESS	4	CLID_FSP_ALET@	Address of the 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 the 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 the ALET for the CRED structure.
20	(14)	ADDRESS	4	CLID_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	CAUD	
0	(0)	ADDRESS	4	CAUD_AO_ALET@	Address of the ALET for the Audit Options.
4	(4)	ADDRESS	4	CAUD_AO@	Addr 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 3 bytes contain an Audit Option: x'00' - do not 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 the 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 the ALET for the file ID.
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 the ALET for the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
28	(1C)	ADDRESS	4	CAUD_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	COWN	
0	(0)	ADDRESS	4	COWN_UID_ALET@	Address of ALET for UID to be set.
4	(4)	ADDRESS	4	COWN_UID@	Addr of the UID to be set as the file owner UID.
8	(8)	ADDRESS	4	COWN_GID_ALET@	Addr of ALET for GID to be set.
12	(C)	ADDRESS	4	COWN_GID@	Addr of the GID to be set as the file owner GID.
16	(10)	ADDRESS	4	COWN_FSP_ALET@	Address of the 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 the 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 the ALET for the CRED structure.
36	(24)	ADDRESS	4	COWN_CRED@	Address of the CRED structure.

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

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

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	QRYF	
0	(0)	ADDRESS	4	QRYF_OCODE_ALET@	Address of the ALET for the option code.
4	(4)	ADDRESS	4	QRYF_OCODE@	Address of a 1 word option code that contains: x'00000001
8	(8)	ADDRESS	4	QRYF_OUTV_ALET@	'-POSIX_CHOWN_RESTRICTED
12	(C)	ADDRESS	4	QRYF_OUTV@	Address of the ALET for the output value. Address of the output value: 0 - POSIX_CHOWN_RESTRICTED. -1 - Not POSIX_CHOWN_RESTRICTED.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	KFOW	
0	(0)	ADDRESS	4	KFOW_FSP_ALET@	Address of the ALET for the FSP structure.
4	(4)	ADDRESS	4	KFOW_FSP@	Address of the FSP structure.
8	(8)	ADDRESS	4	KFOW_FID_ALET@	Address of the 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 the ALET for the CRED structure.
20	(14)	ADDRESS	4	KFOW_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	MKRT	
0	(0)	ADDRESS	4	MKRT_MODE_ALET@	Address of the ALET for the mode structure.
4	(4)	ADDRESS	4	MKRT_MODE@	Address of the mode parameter.
8	(8)	ADDRESS	4	MKRT_OFSP_ALET@	Address of the ALET for the output FSP.
12	(C)	ADDRESS	4	MKRT_OFSP@	Address of the output FSP.
16	(10)	ADDRESS	4	MKRT_FID_ALET@	Address of the 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 the ALET for the data set name.
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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	PTRC	
0	(0)	ADDRESS	4	PTRC_TUID_ALET@	Address of ALET for the target UIDs area.
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 security label when the high-order bit of the target 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	MFSP	
0	(0)	ADDRESS	4	MFSP_MODE_ALET@	Address of the ALET for the mode structure.
4	(4)	ADDRESS	4	MFSP_MODE@	Address of the mode parameter.
8	(8)	ADDRESS	4	MFSP_OFSP_ALET@	Address of the ALET for the output FSP.
12	(C)	ADDRESS	4	MFSP_OFSP@	Address of the output FSP.
16	(10)	ADDRESS	4	MFSP_ODFSP_ALET@	Address of the 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 the 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 the ALET for the CRED structure.
36	(24)	ADDRESS	4	MFSP_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	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.
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.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	GUGP	
0	(0)	ADDRESS	4	GUGP_KEY_ALET@	Addr 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.
24	(18)	ADDRESS	4	GUGP_GCNT_ALET@	Addr 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@	Addr of ALET for the Group List.
36	(24)	ADDRESS	4	GUGP_GLST@	Addr of the Group List area in which supplemental GIDs are returned.
40	(28)	ADDRESS	4	GUGP_GNUM_ALET@	Addr of ALET for number of groups.
44	(2C)	ADDRESS	4	GUGP_GNUM@	Addr of the Number of Groups. (A one word area in which the number of supplemental group GIDs is returned.)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	FORK	
0	(0)	ADDRESS	4	FORK_FLAG_ALET@	Address of ALET for 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 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 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's ALET.
28	(1C)	ADDRESS	4	FORK_DATA@	Address of the data being passed from the parent to the child by way of FORK.
32	(20)	ADDRESS	4	FORK_DATASP_ALET@	Address of ALET for data subpool.
36	(24)	ADDRESS	4	FORK_DATASP@	Address of a word that contains the subpool from which to obtain the storage to hold the data.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	MISP	
0	(0)	ADDRESS	4	MISP_MODE_ALET@	Address of the ALET for the mode structure.
4	(4)	ADDRESS	4	MISP_MODE@	Address of the mode parameter.@L1A
8	(8)	ADDRESS	4	MISP_OISP_ALET@	Address of the ALET for the output ISP.
12	(C)	ADDRESS	4	MISP_OISP@	Address of the output ISP.
16	(10)	ADDRESS	4	MISP_OIPC_ALET@	Address of the ALET for the output IPCP.
20	(14)	ADDRESS	4	MISP_OIPC@	Address of the output IPCP.
24	(18)	ADDRESS	4	MISP_CREI_ALET@	Address of the ALET for the CREDIPC structure.
28	(1C)	ADDRESS	4	MISP_CREI@	Address of the CREDIPC structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	IACC	
0	(0)	ADDRESS	4	IACC_ACODE_ALET@	Address of the 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 the ALET for the ISP structure.
12	(C)	ADDRESS	4	IACC_ISP@	Address of the ISP structure.
16	(10)	ADDRESS	4	IACC_CREI_ALET@	Address of the ALET for the CREDIPC structure.
20	(14)	ADDRESS	4	IACC_CREI@	Address of the CREDIPC structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	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 to be set.
12	(C)	ADDRESS	4	IOWN_UID@	Addr of the UID to be set as the ISP owner UID.
16	(10)	ADDRESS	4	IOWN_GID_ALET@	Addr of ALET for GID to be set.@L1A
20	(14)	ADDRESS	4	IOWN_GID@	Addr of the GID to be set as the ISP owner GID.
24	(18)	ADDRESS	4	IOWN_MODE_ALET@	Address of the ALET for the mode structure.
28	(1C)	ADDRESS	4	IOWN_MODE@	Address of the mode parameter.@L1A
32	(20)	ADDRESS	4	IOWN_ISP_ALET@	Address of the 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 the ALET for the CREDIPC structure.
44	(2C)	ADDRESS	4	IOWN_CREI@	Address of the CREDIPC structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	CKO2	
0	(0)	ADDRESS	4	CKO2_FSP1_ALET@	Address of ALET for the first FSP structure.
4	(4)	ADDRESS	4	CKO2_FSP1@	Addr of the first FSP structure.
8	(8)	ADDRESS	4	CKO2_FSP2_ALET@	Addr of ALET for the second FSP structure.
12	(C)	ADDRESS	4	CKO2_FSP2@	Addr of the second FSP structure.
16	(10)	ADDRESS	4	CKO2_FID1_ALET@	Address of the ALET for the first file ID.
20	(14)	ADDRESS	4	CKO2_FID1@	Address of the first file ID.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
24	(18)	ADDRESS	4	CKO2_FID2_ALET@	Address of the 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 the ALET for the CRED structure.
36	(24)	ADDRESS	4	CKO2_CRED@	Address of the CRED structure.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	56	GETE	
0	(0)	ADDRESS	4	GETE_WORKKB_ALET@	Address of ALET for 1024 byte work area.
4	(4)	ADDRESS	4	GETE_WORKKB@	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 Count.
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 the 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 the GIDs.
52	(34)	ADDRESS	4	GETE_GID@	Address of a 3 word output area. This area contains the real, effective, and saved GIDs.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	DKEY	
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)	ADDRESS	4	DKEY_USERID_ALET@	Address of ALET for the RACF userid
8	(8)	ADDRESS	4	DKEY_ENTITY_ALET@	Address of ALET for the RACF entity
12	(C)	ADDRESS	4	DKEY_USERID@	Address of 9 byte area that contains a 1-byte length followed by userid of up to 8 chars
12	(C)	ADDRESS	4	DKEY_ENTITY@	Address of 247 byte area that contains a 1-byte length followed by entity name up to 246 chars
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 users DCE key
24	(18)	ADDRESS	4	DKEY_KEYL_ALET@	Address of ALET for the DCE key length
28	(1C)	ADDRESS	4	DKEY_KEYL@	Address of fullword containing the DCE key area length

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	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 user ID of up to 8 chars
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 the 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 data obtained
32	(20)	ADDRESS	4	DINF_OUTPUTL_ALET@	Address of ALET for the Output area length
36	(24)	ADDRESS	4	DINF_OUTPUTL@	Address of fullword containing the Output area length

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	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 principal UUID

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
The following 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 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 user ID (1-byte for the length followed by up to 8 bytes for the RACF user ID itself)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	36	DAUT	
0	(0)	ADDRESS	4	DAUT_ACEEP@	Address of a fullword 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 36 byte area for the DCE principle UUID
12	(C)	ADDRESS	4	DAUT_HOME_UUID@	Address of 36 byte area for the DCE home cell UUID.
16	(10)	ADDRESS	4	DAUT_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)
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-aligned 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	68	INTA	
0	(0)	ADDRESS	4	INTA_FUNC@	Address of 1-byte function code
4	(4)	ADDRESS	4	INTA_ATTRIBUTES@	Address of a fullword area which contains the attribute flags for the service
8	(8)	ADDRESS	4	INTA_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)
12	(C)	ADDRESS	4	INTA_ACEEP@	Address of a fullword 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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)
		1... ..		INTA_LAST_PARM_VER1	Variable length parameter list. This was the last parameter for plist version 1
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)
		1... ..		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.
		1... ..		INTA_LAST_PARM_ENVR_IN	Variable length parameter list This is a valid last parameter for the plist version 3.
36	(24)	ADDRESS	4	INTA_ENVR_OUT@	Address of an area containing the data structure to retrieve the security environment.
		1... ..		INTA_LAST_PARM_ENVR_OUT	Variable length parameter list. This is a valid last parameter for plist version 3.
40	(28)	ADDRESS	4	INTA_OUTA_STOR@	Address of an area containing an OUSP
		1... ..		INTA_LAST_PARM_OUTA_STOR	Variable length parameter list. This is a valid last parameter for the plist version 3.
44	(2C)	ADDRESS	4	INTA_X500_NAME_PTR	A word in which to store the Address of 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 to up to 255 bytes of subjects name.
		1... ..		INTA_LAST_X500_NAME	Variable length parameter list. This could be the last parameter for the plist version 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.
		1... ..		INTA_LAST_PARM_VAR_LIST	Variable length parameter list. This could be the last parameter for the plist version 4.
52	(34)	ADDRESS	4	INTA_SECLABEL@	Address of 9 byte area for the security label. (1 byte for the length followed by up to 8 bytes for the security label itself)
56	(38)	ADDRESS	4	INTA_SERVAUTH@	Address of variable length area for the SERVAUTH resource name. (1 byte for the length followed by up to 64 bytes for the SERVAUTH name)
		1... ..		INTA_LAST_PARM_SERVAUTH	Variable length parameter list. This could be the last parameter for plist version 5.
60	(3C)	ADDRESS	4	INTA_PHRASE@	Address of a variable length area containing the user's password phrase (1 byte length of password phrase followed by password phrase).
		1... ..		INTA_LAST_PARM_PHRASE	Variable length parameter list. 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)

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		1... ..		INTA_LAST_PARM_IDID	Variable length parameter list. This could be the last parameter.
		1... ..		INTA_LAST_PARM	Variable length parameter list. This is the last parameter.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE 1... ..	4	INTA_ATTRIBUTES_MAP INTA_MANAGED	ACEE should be managed (CREATE only)
		.1.. ..		INTA_USP	Initialize a USP for the ACEE (CREATE only)
		..1. ....		INTA_TASK_LVL	For CREATE,ON indicates chain ACEE to the TCB.OFF indicates return ACEE address by way of the Acee_Ptr field. For DELETE, ON indicates get ACEE address from the TCB. OFF indicates ACEE address passed through Acee_Ptr
		...1 1..		INTA_CLIENT	ACEE client flags (CREATE only)
		...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 - 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 pair (QUERY only)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	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 structure
8	(8)	ADDRESS	4	ADMN_USERID@	Address of 9-byte area for the RACF userid under whose authority this service executes (1 byte for the length followed by up to 8 bytes for the userid itself)@04A
12	(C)	ADDRESS	4	ADMN_ACEEP@	Address of a fullword containing the ACEE address under the authority of which this service executes
16	(10)	ADDRESS	4	ADMN_OUTPUT_SP@	Address of a caller-supplied 1-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 output, 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 (for alignment)
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_SEGSTR	First segment entry

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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_FLDSTR	First field entry for segment

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 (for alignment)
10	(A)	UNSIGNED	2	ADMN_GRPADM_ERROFF	Offset to segment/field in error
12	(C)	UNSIGNED	2	ADMN_GRPADM_SEG_NUM	Number of segments to update
14	(E)	CHARACTER	1	ADMN_GRPADM_SEGSTR	First segment entry

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	15	ADMN_RESADM_MAP	Mapping for add/alt/del/list resource, add/alt/del/list data set, and permit
0	(0)	CHARACTER	9	ADMN_RESADM_CLASS	Class: not required for data set functions
0	(0)	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 (for alignment)
10	(A)	UNSIGNED	2	ADMN_RESADM_ERROFF	Offset to segment/field in error
12	(C)	UNSIGNED	2	ADMN_RESADM_SEG_NUM	Number of segments to update
14	(E)	CHARACTER	1	ADMN_RESADM_SEGSTR	First segment entry

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	14	ADMN_SETRADM_MAP	Mapping for alter SETROPTS information

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	14	ADMN_XTRUNL_MAP	R_admin SETROPTS output
0	(0)	CHARACTER	4	ADMN_XTRUNL_EYE	Eye catcher: "RXTR" or "RUNL"
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	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 user ID. The first byte is the length followed by 8 bytes for the RACF user ID



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
16	(10)	ADDRESS	4	UMAP_CERTIFICATE@	Address of input area for a digital certificate. First 4 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. First 2 bytes contain the length followed by 246 bytes for the Application ID. If not specified on input, length must be zero.
		1... ..		UMAP_LAST_PARM	Variable length parameter list (for function codes 1-6).
24	(18)	STRUCTURE	8	UMAP_ID_PROPAGATION	
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.
		1... ..		UMAP_IDPROP_LAST_PARM	Variable length parameter list (for function code 8).

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	248	UMAP_DISTINGUISHED_NAME_DS	Distinguished name mapping
0	(0)	UNSIGNED	2	UMAP_DISTINGUISHED_NAME_LEN	Distinguished name length
2	(2)	CHARACTER	246	UMAP_DISTINGUISHED_NAME	Distinguished name string

## COMP

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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 fullword 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 CheckStatus, AbortQuery, and DataGetNext.
16	(10)	ADDRESS	4	CDDL_VERSION@	Address of a 4 byte area containing the input version number for the 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.

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	4	CDDL_ATTRIBUTES_MAP	Attribute flags to cause different behaviors in different functions in R_Datalib.
		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 (CERTUSER) is to be incremented to at least the input serial number.
		1... ..		CDDL_ATT_TRUST	For function DataPut to add certificate with TRUST status. When this flag is off, it indicates RACF determines the status.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		.1.. ....		CDDL_ATT_HIGHTRUST	For function DataPut to add certificate under CERTAUTH with HIGHTRUST 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_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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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, non-zero 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.
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.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	8	CDDL_CRT_STATUS	Parameter list for CheckStatus
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	4	CDDL_GET_UPDATE	Parm list for GetUpdateCode
0	(0)	UNSIGNED	4	CDDL_RING_SEQN	Ring sequence number

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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, non-zero 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.
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	20	CDDL_REMOVE_CERT	Parameter list for DataRemove.
0	(0)	UNSIGNED	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.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	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 240 byte area for the Kerberos Principal. x'00' in the first byte if not specified.@LEA
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	4	KERB_FLDDATA_MAP	Mapping for Field entry
0	(0)	UNSIGNED	2	KERB_FLDDATA_LEN	Length of field struct
2	(2)	UNSIGNED	2	KERB_FLD_NUM	Number of fields to update or list
4	(4)	CHARACTER	0	KERB_FLDSTRT	First field entry

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	Field data

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
12	(C)	ADDRESS	4	TKTS_CRED_AREA@	Address of an area consisting of a 2 byte length field followed by a ticket.
16	(10)	ADDRESS	4	TKTS_OPTIONS@	Address of an address containing a binary bit string which identifies the credential specific processing to be performed. This parameter is only used when the function code is x'0003'.
20	(14)	ADDRESS	4	TKTS_PRIN_USERID@	Address of a storage area containing up to 242 bytes (depending on function code). This area consists of a 2 byte length field followed by either an input userid for the PassTicket function or storage area in which the output principal name is saved.
24	(1C)	ADDRESS	4	TKTS_APPLICATION@	Address of an area consisting of a 2 byte length followed by up to 8 bytes containing the PassTicket application id.
		1... ....		TKTS_LAST_PARM	Variable length parameter list. This is the last parameter

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	12	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 <i>z/OS Security Server RACF Callable Services</i> for function specific parameter lists for callable service R_PgmSignVer.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)
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@)

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)
		1... ..		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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	4	PKIS_ATTRIBUTES_MAP	Certificate Request Attribute mapping
		1... ..		PKIS_SYNC_CREATE	Synchronous generation flag
0	(0)	BITSTRING	3	*	Reserved

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	PKIS_EXPORT_MAP	Function Specific Parameter List (FSPL) for EXPORT
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



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 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.
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
24	(18)	UNSIGNED	4	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 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	PKIS_RESL_ENTRY	Results List Entry for QUERYREQS or QUERCERTS ( 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	Results list entry value

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	PKIS_REQDETAILS_MAP	Function Specific Parameter List (FSPL) for REQDETAILS

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	PKIS_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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	PKIS_SUML_ENTRY	Summary List Entry for REQDETAILS CERTDETAILS, or VERIFY ( a single entry )
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	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 be 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	36	PKIS_QUERYCERTS_MAP	Function Specific Parameter List (FSPL) for QUERYCERTS

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	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'

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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 requestor. The first byte of this area contains the length of the email address.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 requestor'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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 '
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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'

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		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
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		PKIS_PREREG_MAP	Mapping for PREREGISTRATION 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	88	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 a fullword containing the number of remaining parms, including this one
8	(8)	ADDRESS	4	CACH_FUNC@	Address of 2 byte function code Constants for the function codes codes are supplied below
12	(C)	ADDRESS	4	CACH_OPTION@	Address of 2 byte field containing an option value that further defines the function

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	44	PRXY	
0	(0)	ADDRESS	4	PRXY_PARM_ALET@	Address of ALET for remaining parameters
4	(4)	ADDRESS	4	PRXY_FUNC@	Address of 2 byte function code Constants for the function codes codes are supplied below
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 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	(1C)	ADDRESS	4	PRXY_RESULTS@	Address of a pointer to the results

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		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 parameter 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 the first update of the variable length parameter list.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	PRXY_F3_PLIST	Function specific parameter list for PRXY function 3
0	(0)	UNSIGNED	1	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.
		.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.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	40	RACL	Mapping for callable service R_SETFACL
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 1... ..	4	RACL_CRED@ RACL_LAST_PARM	Address of the CRED structure Variable length parameter list This is the last parameter

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	76	RAUX	Mapping for callable service R_AUDITX
0	(0)	ADDRESS	4	RAUX_NUM_PARAMS@	Address of 4-byte area containing the number of parameters in the parameter list.
4	(4)	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.

## COMP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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 even 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@	Name of area that consists of 4-byte length field followed by character data.
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 an array.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	12	WPRV	Mapping for callable service R_WRITEPRIV
0	(0)	ADDRESS	4	WRPV_FUNC@	Address of the function byte
4	(4)	ADDRESS	4	WPRV_ACEE_Alet@	Address of the ALET for the ACEE
8	(8)	ADDRESS	4	WPRV_ACEE@	Address of the ACEE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	SECL	Mapping for callable service R_SETFSECL
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

## Constants

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_KILL	KILL Verification.
4	DECIMAL	3	CKPO_GETPSENT	Caller is GETPSENT.
4	DECIMAL	4	CKPO_OPEN_STTY	Open Slave 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.
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	CONTROL 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

# COMP

Len	Type	Value	Name	Description
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
ADMN 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	SETROPTS 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
4	DECIMAL	32	ADMN_XTR__NEXT_RESOURCE	Extract the next general resource profile
ADMN eyecatchers				
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
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

Len	Type	Value	Name	Description
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 recreate 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
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
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
	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 non-zero 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
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

## COMP

Len	Type	Value	Name	Description
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.
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 requestor'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				
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 <i>z/OS Security Server RACF Callable Services</i> 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	

Len	Type	Value	Name	Description
RACL Operation Type Constants				
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	

## Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
ADMN	0		ADMN_USRADM_ERROFF	A	
ADMN_ACEEP@	C		ADMN_USRADM_FLD_DATA	B	
ADMN_FUNC@	0		ADMN_USRADM_FLD_FLAG	8	
ADMN_GRPADM_ERROFF	A		ADMN_USRADM_FLD_LEN	9	
ADMN_GRPADM_GROUP	0		ADMN_USRADM_FLD_NAME	0	
ADMN_GRPADM_GRP_ID	1		ADMN_USRADM_FLD_NUM	9	
ADMN_GRPADM_GRP_LEN	0		ADMN_USRADM_FLDENTRY	0	
ADMN_GRPADM_MAP	0		ADMN_USRADM_FLDSTRT	B	
ADMN_GRPADM_SEG_NUM	C		ADMN_USRADM_MAP	0	
ADMN_GRPADM_SEGSTRT	E		ADMN_USRADM_SEG_FLAG	8	
ADMN_LAST_PARM	14	80	ADMN_USRADM_SEG_NAME	0	
ADMN_OUT_ENTRY	0		ADMN_USRADM_SEG_NUM	C	
ADMN_OUT_EYE	4		ADMN_USRADM_SEGENTRY	0	
ADMN_OUT_LEN	9		ADMN_USRADM_SEGSTRT	E	
ADMN_OUT_MLEN	0		ADMN_USRADM_USER	0	
ADMN_OUT_MSTR	2		ADMN_USRADM_USER_ID	1	
ADMN_OUT_NEXT	0		ADMN_USRADM_USER_LEN	0	
ADMN_OUT_OFF	C		ADMN_XTR_CONNECT	29	
ADMN_OUT_SPID	8		ADMN_XTR_GROUP	27	
ADMN_OUT_STRT	10		ADMN_XTR_NEXT_GROUP	28	
ADMN_OUTMSG_MAP	0		ADMN_XTR_NEXT_RESOURCE	32	
ADMN_OUTPUT_MSG@	14		ADMN_XTR_NEXT_USER	26	
ADMN_OUTPUT_SP@	10		ADMN_XTR_PPENV	30	
ADMN_PARMLIST@	4		ADMN_XTR_PWENV	24	
ADMN_RESADM_CLAS_LEN	0		ADMN_XTR_RESOURCE	31	
ADMN_RESADM_CLAS_NAM	1		ADMN_XTR_USER	25	
ADMN_RESADM_CLASS	0		ADMN_XTRUNL_ENTRY	E	
ADMN_RESADM_ERROFF	A		ADMN_XTRUNL_EYE	0	
ADMN_RESADM_MAP	0		ADMN_XTRUNL_MAP	0	
ADMN_RESADM_SEG_NUM	C		ADMN_XTRUNL_NUM	C	
ADMN_RESADM_SEGSTRT	E		ADMN_XTRUNL_OUTLEN	4	
ADMN_RUNCOMD_CMD	2		CACH	0	
ADMN_RUNCOMD_LEN	0		CACH_ACEE@	4C	
ADMN_RUNCOMD_MAP	0		CACH_ACEE_ALET@	48	
ADMN_SETRADM_ERROFF	A		CACH_DATA_LEN@	28	
ADMN_SETRADM_MAP	0		CACH_DATA_PTR@	24	
ADMN_SETRADM_SEG_NUM	C		CACH_DATA_TIMEOUT@	2C	
ADMN_SETRADM_SEGSTRT	E		CACH_FUNC@	8	
ADMN_UNL_ENTRY	0		CACH_NAME@	18	
ADMN_UNL_LEN	8		CACH_NUM_PARMS@	4	
ADMN_UNL_NUM	C		CACH_OPTION@	C	
ADMN_UNL_RECSTRT	18		CACH_PARM_ALET@	0	
ADMN_UNL_SETR	23		CACH_RECNAME_LEN@	20	
ADMN_UNL_TYPE	0		CACH_RECNAME_PTR@	1C	
ADMN_USERID@	8		CACH_REF@	40	

# COMP

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
CACH_REF_TIMEOUT@	38		CDDL_RACF_USERID	2C	
CACH_REF_USERID@	3C		CDDL_RCERT_USERID	8	
CACH_SRC_LEN@	34		CDDL_RECID_LEN	40	
CACH_SRC_PTR@	30		CDDL_RECID_PTR	44	
CACH_SUBPOOL@	44		CDDL_REMOVE_CERT	0	
CACH_VERSION_LEN@	14		CDDL_RES_HANDLE	0	
CACH_VERSION@	10		CDDL_RING_SEQN	0	
CAUD	0		CDDL_RING@	C	
CAUD_AO_ALET@	0		CDDL_RLABEL_LEN	0	
CAUD_AO@	4		CDDL_RLABEL_PTR	4	
CAUD_CRED_ALET@	18		CDDL_SDN_LEN	38	
CAUD_CRED@	1C		CDDL_SDN_PTR	3C	
CAUD_FID_ALET@	10		CDDL_TOKEN	0	
CAUD_FID@	14		CDDL_USERID@	8	
CAUD_FSP_ALET@	8		CDDL_VERSION@	10	
CAUD_FSP@	C		CHKP	0	
CDDL	0		CHKP_AUFC_ALET@	0	
CDDL_ABORT	0		CHKP_AUFC@	4	
CDDL_ATT_ALL_KEYTYPES	0	80	CKO2	0	
CDDL_ATT_DEL_CERT_TOO	0	80	CKO2_CRED_ALET@	20	
CDDL_ATT_HIGHTRUST	0	40	CKO2_CRED@	24	
CDDL_ATT_REUSE_RING	0	80	CKO2_FID1_ALET@	10	
CDDL_ATT_SET_MIN_SERIAL	0	80	CKO2_FID1@	14	
CDDL_ATT_SKIPAUTH	0	80	CKO2_FID2_ALET@	18	
CDDL_ATT_TRUST	0	80	CKO2_FID2@	1C	
CDDL_ATTR_ID	8		CKO2_FSP1_ALET@	0	
CDDL_ATTR_LEN	C		CKO2_FSP1@	4	
CDDL_ATTR_PTR	10		CKO2_FSP2_ALET@	8	
CDDL_ATTRIBUTES@	4		CKO2_FSP2@	C	
CDDL_ATTRIBUTES_MAP	0		CKPO	0	
CDDL_CERT_DEFAULT	8		CKPO_CODE_ALET@	18	
CDDL_CERT_LEN	C		CKPO_CODE@	1C	
CDDL_CERT_PTR	10		CKPO_PID_ALET@	10	
CDDL_CERT_USAGE	4		CKPO_PID@	14	
CDDL_CRT_LEN	0		CKPO_REQT_ALET@	0	
CDDL_CRT_PTR	4		CKPO_REQT@	4	
CDDL_CRT_STATUS	0		CKPO_SIGQUEUE	5	
CDDL_FUNC@	0		CKPO_UIDS_ALET@	8	
CDDL_GET_CERT	0		CKPO_UIDS@	C	
CDDL_GET_UPDATE	0		CLID	0	
CDDL_HANDLE_MAP	0		CLID_CRED_ALET@	10	
CDDL_ISN	0		CLID_CRED@	14	
CDDL_ISN_CLEN	0		CLID_FID_ALET@	8	
CDDL_ISN_CPTR	4		CLID_FID@	C	
CDDL_ISN_SERNUM	8		CLID_FSP_ALET@	0	
CDDL_LABEL_LEN	24		CLID_FSP@	4	
CDDL_LABEL_PTR	28		CMOD	0	
CDDL_LAST_PARM	14	80	CMOD_CRED_ALET@	18	
CDDL_PARMLIST@	14		CMOD_CRED@	1C	
CDDL_PCERT_DEFAULT	4		CMOD_FID_ALET@	10	
CDDL_PCERT_LEN	8		CMOD_FID@	14	
CDDL_PCERT_PTR	C		CMOD_FSP_ALET@	8	
CDDL_PCERT_USAGE	0		CMOD_FSP@	C	
CDDL_PCERT_USERID	20		CMOD_MODE_ALET@	0	
CDDL_PK_BITSIZE	20		CMOD_MODE@	4	
CDDL_PK_LEN	14		COMP	0	
CDDL_PK_PTR	18		COMP_RACRC_ALET@	C	
CDDL_PK_TYPE	1C		COMP_RACRC_STOR@	10	
CDDL_PKEY_LEN	10		COMP_RACSC_ALET@	14	
CDDL_PKEY_PTR	14		COMP_RACSC_STOR@	18	
CDDL_PLABEL_LEN	18		COMP_SAFRC_ALET@	4	
CDDL_PLABEL_PTR	1C		COMP_SAFRC_STOR@	8	
CDDL_PREDICATES	4		COMP_SERVR_PARMS	1C	
CDDL_PREV_HANDLE	0		COMP_WORKA_STOR@	0	
CDDL_PUT_CERT	0		COWN	0	



Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
COWN_CRED_ALET@	20		DRUR_FUNC@	4	
COWN_CRED@	24		DRUR_HOME_UUID_ALET@	10	
COWN_FID_ALET@	18		DRUR_HOME_UUID@	14	
COWN_FID@	1C		DRUR_PRIN_UUID_ALET@	8	
COWN_FSP_ALET@	10		DRUR_PRIN_UUID@	C	
COWN_FSP@	14		DRUR_USERID_ALET@	18	
COWN_GID_ALET@	8		DRUR_USERID@	1C	
COWN_GID@	C		EXID	0	
COWN_UID_ALET@	0		EXID_FLAG_ALET@	0	
COWN_UID@	4		EXID_FLAG@	4	
CSID	0		EXID_GID_ALET@	10	
CSID_ID_ALET@	0		EXID_GID@	14	
CSID_ID@	4		EXID_GIDO_ALET@	20	
CSID_IDOA_ALET@	8		EXID_GIDO@	24	
CSID_IDOA@	C		EXID_UID_ALET@	8	
DAUT	0		EXID_UID@	C	
DAUT_ACC_REQ@	20		EXID_UIDO_ALET@	18	
DAUT_ACEEP@	0		EXID_UIDO@	1C	
DAUT_ALET@	4		FORK	0	
DAUT_CLASS_NAME@	14		FORK_DATA_ALET@	18	
DAUT_ENT_LEN@	1C		FORK_DATA@	1C	
DAUT_ENT_NAME@	18		FORK_DATAKEY_ALET@	8	
DAUT_HOME_UUID@	C		FORK_DATAKEY@	C	
DAUT_PRIN_UUID@	8		FORK_DATALEN_ALET@	10	
DAUT_USERID@	10		FORK_DATALEN@	14	
DINF	0		FORK_DATASP_ALET@	20	
DINF_FIELD_DATA	A		FORK_DATASP@	24	
DINF_FIELD_LEN	8		FORK_FLAG_ALET@	0	
DINF_FIELD_LIST	0		FORK_FLAG@	4	
DINF_FIELD_LIST_CNT	2		GETE	0	
DINF_FIELD_LIST_LEN	0		GETE_GCNT_ALET@	10	
DINF_FIELD_LIST_ST	4		GETE_GCNT@	14	
DINF_FIELD_NAME	0		GETE_GID_ALET@	30	
DINF_FIELDL_ALET@	10		GETE_GID@	34	
DINF_FIELDL@	14		GETE_GLST_ALET@	18	
DINF_FIELDS	0		GETE_GLST@	1C	
DINF_FUNC_ALET@	0		GETE_GNUM_ALET@	20	
DINF_FUNC@	4		GETE_GNUM@	24	
DINF_OUTPUT_AREA	0		GET_INFO@	28	
DINF_OUTPUT_AREA_CNT	2		GETE_KEY_ALET@	8	
DINF_OUTPUT_AREA_LEN	0		GETE_KEY@	C	
DINF_OUTPUT_AREA_ST	4		GETE_UID_ALET@	28	
DINF_OUTPUT_FLD	0		GETE_UID@	2C	
DINF_OUTPUT_FLD_DATA	A		GETE_WORKB_ALET@	0	
DINF_OUTPUT_FLD_LEN	8		GETE_WORKB@	4	
DINF_OUTPUT_FLD_NAME	0		GETG	0	
DINF_OUTPUTA_ALET@	18		GETG_GCNT_ALET@	8	
DINF_OUTPUTA@	1C		GETG_GCNT@	C	
DINF_OUTPUTL_ALET@	20		GETG_GLST_ALET@	10	
DINF_OUTPUTL@	24		GETG_GLST@	14	
DINF_USERID_ALET@	8		GETG_GNUM_ALET@	18	
DINF_USERID@	C		GETG_GNUM@	1C	
DKEY	0		GETG_KEY_ALET@	0	
DKEY_FUNC_ALET@	0		GETG_KEY@	4	
DKEY_FUNC@	4		GINF	0	
DKEY_KEY_ALET@	10		GINF_NUM_PARMS@	0	
DKEY_KEY@	14		GINF_PARM_ALET@	4	
DKEY_KEYL_ALET@	18		GINF_FUNC_CODE@	8	
DKEY_KEYL@	1C		GINF_OPTION@	C	
DKEY_USERID_ALET@	8		GINF_RACF_ENTITY@	10	
DKEY_ENTITY_ALET@	8		GINF_RACF_CLASS@	14	
DKEY_ENTITY@	12		GINF_RESULT_ENTRIES@	18	
DKEY_USERID@	C		GMAP	0	
DRUR	0		GMAP_FLAG_ALET@	0	
DRUR_FUNC_ALET@	0		GMAP_FLAG@	4	

# COMP

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
GMAP_ID_ALET@	8		IOWN_CMD_ALET@	0	
GMAP_ID@	C		IOWN_CMD@	4	
GMAP_NAME_ALET@	10		IOWN_CREI_ALET@	28	
GMAP_NAME@	14		IOWN_CREI@	2C	
GUGP	0		IOWN_GID_ALET@	10	
GUGP_GCNT_ALET@	18		IOWN_GID@	14	
GUGP_GCNT@	1C		IOWN_ISP_ALET@	20	
GUGP_GLST_ALET@	20		IOWN_ISP@	24	
GUGP_GLST@	24		IOWN_MODE_ALET@	18	
GUGP_GNUM_ALET@	28		IOWN_MODE@	1C	
GUGP_GNUM@	2C		IOWN_UID_ALET@	8	
GUGP_KEY_ALET@	0		IOWN_UID@	C	
GUGP_KEY@	4		IUSP	0	
GUGP_NAME_ALET@	10		IUSP_WORKA_ALET@	0	
GUGP_NAME@	14		IUSP_WORKA_STOR@	4	
GUGP_NLEN_ALET@	8		KACC	0	
GUGP_NLEN@	C		KACC_ACODE_ALET@	0	
IACC	0		KACC_ACODE@	4	
IACC_ACODE_ALET@	0		KACC_CRED_ALET@	18	
IACC_ACODE@	4		KACC_CRED@	1C	
IACC_CREI_ALET@	10		KACC_FID_ALET@	10	
IACC_CREI@	14		KACC_FID@	14	
IACC_ISP_ALET@	8		KACC_FLAG_ALET@	20	
IACC_ISP@	C		KACC_FLAG@	24	
INTA	0		KACC_FSP_ALET@	8	
INTA_ACEEP@	C		KACC_FSP@	C	
INTA_APPL_ID@	10		KERB	0	
INTA_ATTRIBUTES_MAP	0		KERB_DATA_AREA@	C	
INTA_ATTRIBUTES@	4		KERB_FLD_DATA	A	
INTA_AUTH_CLNT	0	08	KERB_FLD_LEN	8	
INTA_CERTIFICATE@	1C		KERB_FLD_NAME	0	
INTA_CLIENT	0	18	KERB_FLD_NUM	2	
INTA_ENVR_IN@	20		KERB_FLDDATA_LEN	0	
INTA_ENVR_OUT@	24		KERB_FLDDATA_MAP	0	
INTA_ENVR_RET	0	02	KERB_FLDENTRY	0	
INTA_FUNC@	0		KERB_FLDSTRT	4	
INTA_IDID_AREA@	40		KERB_FUNC@	0	
INTA_LAST_PARM	30	80	KERB_LAST_PARM	C	80
INTA_LAST_PARM_ENVR_IN	20	80	KERB_NAME@	8	
INTA_LAST_PARM_ENVR_OUT	24	80	KERB_RACF_NAME@	4	
INTA_LAST_PARM_OUTA_STOR	28	80	KFOW	0	
INTA_LAST_PARM_IDID	3C	80	KFOW_CRED_ALET@	10	
INTA_LAST_PARM_PHRASE	3C	80	KFOW_CRED@	14	
INTA_LAST_PARM_VAR_LIST	30	80	KFOW_FID_ALET@	8	
INTA_LAST_PARM_VER1	18	80	KFOW_FID@	C	
INTA_LAST_PARM_VER2	1C	80	KFOW_FSP_ALET@	0	
INTA_LAST_X500_NAME	2C	80	KFOW_FSP@	4	
INTA_LOGSTRING@	18		MFSP	0	
INTA_MANAGED	0	80	MFSP_CRED_ALET@	20	
INTA_MSG_SUPP	0	04	MFSP_CRED@	24	
INTA_NO_TIMEOUT	0	01	MFSP_FID_ALET@	18	
INTA_OUSP_RET	1	80	MFSP_FID@	1C	
INTA_OUTA_STOR@	28		MFSP_MODE_ALET@	0	
INTA_PASSWORD@	14		MFSP_MODE@	4	
INTA_PHRASE@	3C		MFSP_ODFSP_ALET@	10	
INTA_SECLABEL@	34		MFSP_ODFSP@	14	
INTA_SERVAUTH	38		MFSP_OFSP_ALET@	8	
INTA_TASK_LVL	0	20	MFSP_OFSP@	C	
INTA_UNAUTH_CLNT	0	10	MISP	0	
INTA_USERID@	8		MISP_CREI_ALET@	18	
INTA_USP	0	40	MISP_CREI@	1C	
INTA_VARIABLE_LIST@	30		MISP_MODE_ALET@	0	
INTA_X500_NAME_PTR	2C		MISP_MODE@	4	
INTA_X500_RET	1	40	MISP_OIPC_ALET@	10	
IOWN	0		MISP_OIPC@	14	

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
MISP_OISP_ALET@	8		PKIS_QREC_CRIT_PASS@	18	
MISP_OISP@	C		PKIS_QRYC_CRIT_DAYS	1C	
MKRT	0		PKIS_QRYC_CRIT_NAME@	20	
MKRT_DSN_ALET@	18		PKIS_QRYC_CRIT_STATUS	18	
MKRT_DSN@	1C		PKIS_QRYC_EYECATCH	0	
MKRT_FID_ALET@	10		PKIS_QRYC_NUMENTRIES	14	
MKRT_FID@	14		PKIS_QRYC_RESULTL_LEN	8	
MKRT_MODE_ALET@	0		PKIS_QRYC_RESULTL@	C	
MKRT_MODE@	4		PKIS_QRYC_SERIALNUM@	10	
MKRT_OFSP_ALET@	8		PKIS_QRYR_CERTID@	10	
MKRT_OFSP@	C		PKIS_QRYR_CRIT_DAYS	1C	
PKIS	0		PKIS_QRYR_CRIT_NAME@	20	
PKIS_ATTRIBUTES_MAP	0		PKIS_QRYR_CRIT_STATUS	18	
PKIS_ATTRIBUTES@	8		PKIS_QRYR_EYECATCH	0	
PKIS_CA_DOMAIN@	18		PKIS_QRYR_NUMENTRIES	14	
PKIS_CERTDETAILS_MAP	0		PKIS_QRYR_RESULTL_LEN	8	
PKIS_CPL_ENTRY	0		PKIS_QRYR_RESULTL@	C	
PKIS_CPL_FIELD_LEN	C		PKIS_QUERYCERTS_MAP	0	
PKIS_CPL_FIELD_NAME	0		PKIS_QUERYREQS_MAP	0	
PKIS_CPL_FIELD_VALUE	10		PKIS_RENEW_MAP	0	
PKIS_CRTD_CERTPL_LEN	10		PKIS_RENW_CERTID@	10	
PKIS_CRTD_CERTPL@	14		PKIS_RENW_CERTPL_LEN	8	
PKIS_CRTD_EYECATCH	0		PKIS_RENW_CERTPL@	C	
PKIS_CRTD_SERIALNUM@	18		PKIS_RENW_EYECATCH	0	
PKIS_CRTD_SUML_LEN	8		PKIS_RENW_SERIALNUM@	14	
PKIS_CRTD_SUML@	C		PKIS_RESPOND	14	
PKIS_EXP_CERTAN_LEN	8		PKIS_REQD_CERTID@	18	
PKIS_EXP_CERTAN@	C		PKIS_REQD_CERTPL_LEN	10	
PKIS_EXP_CERTID@	10		PKIS_REQD_CERTPL@	14	
PKIS_EXP_EYECATCH	0		PKIS_REQD_EYECATCH	0	
PKIS_EXPORT_MAP	0		PKIS_REQD_SUML_LEN	8	
PKIS_FUNC_PARML@	14		PKIS_REQD_SUML@	C	
PKIS_FUNC@	4		PKIS_REQDETAILS_MAP	0	
PKIS_GENC_CERTID@	10		PKIS_RESL_ENTRY	0	
PKIS_GENC_CERTPL_LEN	8		PKIS_RESL_ENTRY_LEN	0	
PKIS_GENC_CERTPL@	C		PKIS_RESL_ENTRY_VALUE	1	
PKIS_GENC_EYECATCH	0		PKIS_REVK_EYECATCH	0	
PKIS_EXP_KEYID@	14		PKIS_REVK_REASON	8	
PKIS_LAST_PARM	14	80	PKIS_REVK_SERIALNUM@	C	
PKIS_LOG_STRING@	C		PKIS_REVOKE_MAP	0	
PKIS_MODC_ACTION	8		PKIS_SCEP_EYECATCH	0	
PKIS_MODC_COMMENT@	C		PKIS_SNID_ENTRY	0	
PKIS_MODC_EYECATCH	0		PKIS_SNID_ENTRY_LEN	0	
PKIS_MODC_REASON	18		PKIS_SNID_ENTRY_VALUE	1	
PKIS_MODC_REQUESTOREMAIL@	1C		PKIS_SUML_ENTRY	0	
PKIS_MODC_SERIALNUMSL_LEN	10		PKIS_SUML_ENTRY_LEN	0	
PKIS_MODC_SERIALNUMSL@	14		PKIS_SUML_ENTRY_VALUE	1	
PKIS_MODIFYCERTS_MAP	0		PKIS_SYNCH_CREATE	0	80
PKIS_MODIFYREQS_MAP	0		PKIS_VERF_CERT_LEN	18	
PKIS_MODR_ACTION	8		PKIS_VERF_CERT@	1C	
PKIS_MODR_CERTIDL_LEN	10		PKIS_VERF_CERTPL_LEN	10	
PKIS_MODR_CERTIDL@	14		PKIS_VERF_CERTPL@	14	
PKIS_MODR_CERTPL_LEN	18		PKIS_VERF_EYECATCH	0	
PKIS_MODR_CERTPL@	1C		PKIS_VERF_SUML_LEN	8	
PKIS_MODR_COMMENT@	C		PKIS_VERF_SUML@	C	
PKIS_MODR_EYECATCH	0		PKIS_VERIFY_MAP	0	
PKIS_NUM_PARMS@	0		PRXY	0	
PKIS_PARM_VER@	10		PRXY_BIND_DN@	C	
PKIS_PREG_CERTID@	16		PRXY_BIND_PW@	10	
PKIS_QRECOVER_MAP	0		PRXY_F3_CLASS	2	
PKIS_QREC_EYECATCH	0		PRXY_F3_DATETIME	18	
PKIS_QREC_RESULTL_LEN	8		PRXY_F3_FLAGS	1	
PKIS_QREC_RESULTL@	C		PRXY_F3_INITIATOR	10	
PKIS_QREC_NUMENTRIES	10		PRXY_F3_OPTYPE	0	
PKIS_QREC_CRIT_EMAIL@	14		PRXY_F3_PLIST	0	

# COMP

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
PRXY_F3_PROFLEN	A		SECL_FID_Alet@	8	
PRXY_F3_PROFNAME@	C		SECL_FID@	C	
PRXY_F3_PWUPD	1	80	SECL_Cred_Alet@	10	
PRXY_F3_PWUPD2	1	40	SECL_Cred@	14	
PRXY_F3_PWUPD3	1		TKTS	0	
PRXY_FUNC@	4		TKTS_CRED_AREA@	C	
PRXY_FUNC_VERSION@	20		TKTS_FUNC@	4	
PRXY_FUNC_PARMLIST@	24		TKTS_LAST_PARM	14	80
PRXY_LAST_PARM0	1C	80	TKTS_OPTION_WORD@	8	
PRXY_LAST_PARM1	28	80	TKTS_OPTIONS@	10	
PRXY_LDAP_ERROR@	28		TKTS_PARM_ALET@	0	
PRXY_LDAP_HOST@	8		TKTS_PRIN_USERID@	14	
PRXY_PARM_ALET@	0		UMAP	0	
PRXY_RESULTS@	1C		UMAP_APPL_USERID	0	
PRXY_USERDN@	18		UMAP_APPL_USERID@	14	
PRXY_USERID@	14		UMAP_APPLID	2	
PTRC	0		UMAP_APPLID_LEN	0	
PTRC_TGID_ALET@	8		UMAP_CERTIFICATE	0	
PTRC_TGID@	C		UMAP_CERTIFICATE@	10	
PTRC_TPID_ALET@	10		UMAP_DIGCERT	4	
PTRC_TPID@	14		UMAP_DIGCERT_LEN	0	
PTRC_TUID_ALET@	0		UMAP_DISTINGUISHED_NAME	2	
PTRC_TUID@	4		UMAP_DISTINGUISHED_NAME@	18	
QRYF	0		UMAP_DISTINGUISHED_NAME_DS	0	
QRYF_OCODE_ALET@	0		UMAP_DISTINGUISHED_NAME_LEN	0	
QRYF_OCODE@	4		UMAP_FUNC@	4	
QRYF_OUTV_ALET@	8		UMAP_ID_PROPAGATION	18	
QRYF_OUTV@	C		UMAP_IDPROP_LAST_PARM	1C	80
QRYS	0		UMAP_LAST_PARM	14	80
QRYS_OPTN_CODE_ALET@	0		UMAP_OPTION_WORD@	8	
QRYS_OPTN_CODE@	4		UMAP_PARM_ALET@	0	
QRYS_OUTP_VAL_ALET@	8		UMAP_RACFID	1	
QRYS_OUTP_VAL@	C		UMAP_RACFID_LEN	0	
RACL	0		UMAP_REGISTRY_NAME	2	
RACL_CRED_ALET@	20		UMAP_REGISTRY_NAME@	1C	
RACL_CRED@	24		UMAP_REGISTRY_NAME_DS	0	
RACL_EDIT	0		UMAP_REGISTRY_NAME_LEN	0	
RACL_EDIT_ACL	4		UMAP_USERID	0	
RACL_EDIT_ACLTYPE	2		UMAP_USERID@	C	
RACL_EDIT_ALET@	0		UMSK	0	
RACL_EDIT_OPTYPE	0		UMSK_MODE_ALET@	0	
RACL_EDIT@	4		UMSK_MODE@	4	
RACL_EDITLEN_ALET@	8		WPRV	0	
RACL_EDITLEN@	C		WPRV_FUNC@	0	
RACL_FID_ALET@	18		WPRV_ACEE_Alet@	4	
RACL_FID@	1C		WPRV_ACEE@	8	
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_FSP_Alet@	0				
SECL_FSP@	4				

## COMX: 64-BIT common SAF/RACF parameter list for z/OS UNIX System Services security functions

<b>Common Name:</b>	64-bit common SAF/RACF parameter list for z/OS UNIX System Services security functions		
<b>Macro ID:</b>	IRRPCOMX		
<b>DSECT Name:</b>	COMX, CDDLX		
<b>Owning Component:</b>	Resource Access Control Facility (SC1BN)		
<b>Eye-Catcher ID:</b>	None		
<b>Storage Attributes:</b>	Subpool	N/A	
	Key	Any	
	Residency	Invoker's primary address space	
<b>Size:</b>	<b>Section</b>	<b>Size</b>	
	COMX	64 bytes	
	CDDLX	48 bytes	
	GSEC	240 bytes	
<b>Created by:</b>	Invoker of 64-bit z/OS UNIX System Services security functions		
<b>Pointed to by:</b>	Address of COMX is passed in register 1 when invoking 64-bit z/OS UNIX System Services security functions		
<b>Serialization:</b>	None		
<b>Function:</b>	Maps the common input parameter list for the 64-bit RACF and SAF callable services routers		

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	COMPX	z/OS UNIX SAF/RACF common parameter list.
0	(0)	ADDRESS	8	COMPX_PARMCNT@	Address of number of parameters in list.
8	(8)	ADDRESS	8	COMPX_WORKA_STOR@	Address of 1024 byte work area.
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		COMPX_SERVR_PARMS	Service routine parameter lists begin here.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	CDDLX	Mapping for CDDLX.
0	(0)	ADDRESS	8	CDDLX_FUNC@	Address of 1 byte input function code.
8	(8)	ADDRESS	8	CDDLX_ATTRIBUTES@	Address of full word of input attributes.
16	(10)	ADDRESS	8	CDDLX_USERID@	Address of a 9 byte input area for the RACF user ID. (1 byte for the length followed by 8 bytes for the RACF user ID itself). A length of 0 indicates the current user is the ring owner.

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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@. Must be set to 0.
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	4	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 the certificates from it. When this flag is off, it indicates the creation of a new key ring.
		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 to add certificate with TRUST status. When this flag is off, it indicates RACF determines the status.
		.1... ..		CDDLX_ATT_HIGHTRUST	For function DataPut to add certificate under CERTAUTH with HIGHTRUST 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_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 must be performed.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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' - Reserver bits must be set to 0
12	(C)	UNSIGNED	4	CDDLX_CERT_DEFAULT	Output default indicator. Non-zero value indicates this is the default certificate for the ring; zero value indicates this is not the default certificate for the ring.
16	(10)	UNSIGNED	4	CDDLX_CERT_LEN	On input, this contains the length of the certificate area pointed to by CDDLX_CERT_PTR. On output, this contains the actual size of the certificate returned or 0 if no certificate is 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, this contains the size of the private key area pointed to by CDDLX_PK_PTR. On output, this contains the length of the private key 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.
52	(34)	UNSIGNED	4	CDDLX_PK_BITSIZE	Output value indicating the size of the private key modules in bits.
56	(38)	UNSIGNED	4	CDDLX_LABEL_LEN	On input, this contains the length of the field pointed to by CDDLX_LABEL_PTR and must be at least 32. On output, this contains the length of the label returned at the address in CDDLX_LABEL_PTR and is 32 or less.
60	(3C)	UNSIGNED	4	*	Reserved for alignment.
64	(40)	ADDRESS	8	CDDLX_LABEL_PTR	Input value specifying the address of the output area to be used for the label name.
72	(48)		9	CDDLX_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.
81	(51)	CHARACTER	3	*	Reserved.

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
84	(54)	UNSIGNED	4	CDDLX_SDN_LEN	On input, this contains the length of the output buffer pointed to by CDDLX_SDN_PTR. On output, this 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 is 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.

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

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	CDDLX_ISN	Parameter list for IncSerialNum.
0	(0)	UNSIGNED	4	CDDLX_ISN_CLEN	Length of DER-encoded input Digital Certificate.
4	(4)		4	Reserved	Reserved.
8	(8)	ADDRESS	8	CDDLX_ISN_PTR	Address of DER-encoded input.
16	(10)	CHARACTER	8	CDDLX_ISN_SERNUM	Output-incremented serial number of Digital Certificate.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	68	CDDLX_PUT_CERT	Parameter list for DataPut.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	CHARACTER	4	CDDLX_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	CDDLX_PCERT_DEFAULT	A 4 byte input value for default certificate indicator. Zero value indicates this is not default certificate for the ring, non-zero indicates it is.
8	(8)	UNSIGNED	4	CDDLX_PCERT_LEN	A 4 byte input value contains the length of the certificate area pointed by CDDL_PCERT_PTR.
12	(C)	UNSIGNED	4	*	Reserved for alignment.
16	(10)	ADDRESS	8	CDDLX_PCERT_PTR	An input value specifying address of the input certificate.
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	*	Reserved for alignment.
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	*	Reserved for alignment.
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-aligned and padded blanks.
65	(41)	CHARACTER	3	*	Reserved.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	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 CDDL_RLABEL_PTR.
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.

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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-aligned and padded with blanks.
25	(19)	CHARACTER	3	*	Reserved.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	CDDLX_HANDLE_MAP	Mapping of area pointed to by CDDLX_RES_HANDLE and 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.
20	(14)	UNSIGNED	4	*	Reserved for alignment.
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC	Mapping for GSEC.
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.
24	(18)	ADDRESS	8	GSEC_PARMLIST@	Address of function specific parameter list.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_RETNAME	Parameter list for Return principal 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

  

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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 name
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_TOK@	Address of returned token

  

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_CONTSECCONTEXT	Parameter list for Continue initialization 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_CONTXT_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 token from context acceptor
56	(38)	ADDRESS	8	GSEC_CONT_RET_FLAGS@	Address of area for return flags
64	(40)	ADDRESS	8	GSEC_CONT_CNTXT_EXPIRE@	Address of area for actual context expiration

  

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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 length of token
40	(28)	ADDRESS	8	GSEC_ACC_TOKEN@	Address of context token from context initiator

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 name
64	(40)	ADDRESS	8	GSEC_ACC_RET_FLAGS@	Address of area for return flags
72	(48)	ADDRESS	8	GSEC_ACC_CONTXT_EXPIRE@	Address of area for context expiration
80	(50)	ADDRESS	8	GSEC_ACC_OUT_TOK@	Address of returned token
88	(58)	ADDRESS	8	GSEC_ACC_CRED_HNDL@	Address of returned credential handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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_CONTXT_HNDL@	Address of context handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_REL_CRED	Parameter list for Release 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_GETMIC	Parameter list for Get MIC for 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_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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_VERMIC	Parameter list for Verify MIC for 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_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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
64	(40)	ADDRESS	8	GSEC_VER_TOKEN@	Address of token

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_UNWRAPMSG	Parameter list for Unwrap 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_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 msg
64	(40)	ADDRESS	8	GSEC_UNWRAP_STATE@	Address of area for confidentiality state

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_IMPORTSECCONTEXT	Parameter list for Import 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_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

## COMX

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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_IMPCCRD_TOKLN@	Address of length of token
40	(28)	ADDRESS	8	GSEC_IMPCCRD_TOKEN@	Address of token
48	(30)	ADDRESS	8	GSEC_IMPCCRD_CRED_HNDL@	Address of area for credential handle

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	0	GSEC_ACQUIRECRED	Parameter list for Acquire initiator 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_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

Offsets			Len		
Dec	Hex	Type		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

Offsets			Len		
Dec	Hex	Type		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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_INITSECCONTEXT	Parameter list for initiate 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_INIT_TGT_PRIN@	Address of target principal name
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_TOK@	Address of returned token

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_CONTSECCONTEXT	Parameter list for Continue initialization 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
12	(C)	ADDRESS	4	GSEC31_GSS_ACEE@	Address of input ACEE
16	(10)	ADDRESS	4	GSEC31_CONXT_HNDL@	Address of context handle returned from INIT_SEC_CONTEXT
20	(14)	ADDRESS	4	GSEC31_CONT_TOKLN@	Address of token length
24	(18)	ADDRESS	4	GSEC31_CONT_TOKEN@	Address of token from context acceptor
28	(1C)	ADDRESS	4	GSEC31_CONT_RET_FLAGS@	Address of area for return flags
32	(20)	ADDRESS	4	GSEC31_CONT_CNTXT_EXPIRE@	Address of area for actual context expiration

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 length of token
20	(14)	ADDRESS	4	GSEC31_ACC_TOKEN@	Address of context token from context initiator
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 name
32	(20)	ADDRESS	4	GSEC31_ACC_RET_FLAGS@	Address of area for return flags
36	(24)	ADDRESS	4	GSEC31_ACC_CONTXT_EXPIRE@	Address of area for context expiration
40	(28)	ADDRESS	4	GSEC31_ACC_OUT_TOK@	Address of returned token
44	(2C)	ADDRESS	4	GSEC31_ACC_CRED_HNDL@	Address of returned credential handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	(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_DEL_CONTXT_HNDL@	Address of context handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_RELCRED	Parameter list for Release 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_REL_CRED_HNDL@	Address of credential handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_GETMIC	Parameter list for Get MIC for 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_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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_VERMIC	Parameter list for Verify MIC for a message
0	(0)	ADDRESS	4	GSEC31_GSS_FUNC@	Address of GSS-API function code



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_UNWRAPMSG	Parameter list for Unwrap 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_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 unwrapped msg
32	(20)	ADDRESS	4	GSEC31_UNWRAP_STATE@	Address of area for confidentiality state

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC31_IMPORTSECCONTEXT	Parameter list for Import security context

## COMX

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	GSEC_IMPSEC_CONTXT_HNDL@	Address of area for context handle

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	0	GSEC_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	GSEC_EXPCRD_CRED_HNDL@	Address of credential handle
20	(14)	ADDRESS	4	GSEC31_EXPCRD_OUT_TOK@	Address of returned token

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	GSEC31_PASSTICKET_PARMS	Parameter list for Passticket
0	(0)	ADDRESS	4	GSEC31_TICKET_OPTIONS	Address of Passticket function code

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
CDDLX	0		CDDLX_PARMLIST@	28	
CDDLX_ABORT	0		CDDLX_PK_BITSIZE	34	
CDDLX_ATT_ALL_KEYTYPES	0	80	CDDLX_PK_LEN	20	
CDDLX_ATT_DEL_CERT_TOO	0	80	CDDLX_PK_PTR	28	
CDDLX_ATT_HIGHTRUST	0	40	CDDLX_PK_TYPE	30	
CDDLX_ATT_REUSE_RING	0	80	CDDLX_PREDICATES	8	
CDDLX_ATT_SET_MIN_SERIAL	0	80	CDDLX_PREV_HANDLE	0	
CDDLX_ATT_SKIPAUTH	0	80	CDDLX_PCERT_DEFAULT	4	
CDDLX_ATT_TRUST	0	80	CDDLX_PCERT_LEN	8	
CDDLX_ATTR_ID	C		CDDLX_PCERT_PTR	10	
CDDLX_ATTR_LEN	10		CDDLX_PCERT_USAGE	0	
CDDLX_ATTR_PTR	18		CDDLX_PCERT_PTR	10	
CDDLX_ATTRIBUTES@	8		CDDLX_PCERT_USERID	38	
CDDLX_ATTRIBUTES_MAP	0		CDDLX_PKEY_LEN	18	
CDDLX_CERT_DEFAULT	C		CDDLX_PKEY_PTR	20	
CDDLX_CERT_LEN	10		CDDLX_PLABEL_LEN	28	
CDDLX_CERT_PTR	18		CDDLX_PLABEL_PTR	30	
CDDLX_CERT_USAGE	8		CDDLX_RACF_USERID	48	
CDDLX_CRT_LEN	0		CDDLX_RCERT_USERID	10	
CDDLX_CRT_PTR	8		CDDLX_RECID_LEN	60	
CDDLX_CRT_STATUS	0		CDDLX_RECID_PTR	68	
CDDLX_FUNC@	0		CDDLX_REMOVE_CERT	0	
CDDLX_GET_CERT	0		CDDLX_RES_HANDLE	0	
CDDLX_GET_UPDATE	0		CDDLX_RING_SEQN	0	
CDDLX_HANDLE_MAP	0		CDDLX_RING@	18	
CDDLX_ISN	0		CDDLX_RLABEL_LEN	0	
CDDLX_ISN_CLEN	0		CDDLX_RLABEL_PTR	8	
CDDLX_ISN_PTR	8		CDDLX_SDN_LEN	54	
CDDLX_ISN_SERNUM	10		CDDLX_SDN_PTR	58	
CDDLX_LABEL_LEN	38		CDDLX_TOKEN	0	
CDDLX_LABEL_PTR	40		CDDLX_USERID@	10	

## COMX

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
CDDLX_VERSION@	20		GSEC_INIT_CNTXT_HNDL@	40	
COMPX	0		GSEC_INIT_CRED_HNDL@	38	
COMPX_PARMCNT@	0		GSEC_INIT_OUT_TOK@	58	
COMPX_RACRC_ALET@	20		GSEC_INIT_REQ_EXPIRE@	30	
COMPX_RACRC_STOR@	28		GSEC_INIT_REQ_FLAGS@	28	
COMPX_RACSC_ALET@	30		GSEC_INIT_RET_FLAGS@	48	
COMPX_RACSC_STOR@	38		GSEC_INIT_TGT_PRIN@	20	
COMPX_SAFRC_ALET@	10		GSEC_INITSECCONTEXT	0	
COMPX_SAFRC_STOR@	18		GSEC_OPT@	0	
COMPX_SERVR_PARMS	40		GSEC_PARMLIST@	18	
COMPX_WORKA_STOR@	8		GSEC_PASSTICKET_PARMS	0	
GSEC	0		GSEC_REL_CRED_HNDL@	20	
GSEC_ACC_CONTXT_EXPIRE@	48		GSEC_RELCRED	0	
GSEC_ACC_CONTXT_HNDL@	30		GSEC_RET_CRED_LEN@	8	
GSEC_ACC_CRED_HNDL@	58		GSEC_RET_FUNC@	0	
GSEC_ACC_OUT_TOK@	50		GSEC_RET_INPUT_CRED@	10	
GSEC_ACC_PRIN@	38		GSEC_RET_OID@	18	
GSEC_ACC_RET_FLAGS@	40		GSEC_RET_RC@	28	
GSEC_ACC_TOKEN@	28		GSEC_RET_TICKET_PRIN@	20	
GSEC_ACC_TOKLN@	20		GSEC_RETNAME	0	
GSEC_ACCSECCONTEXT	0		GSEC_TICKET_OPTIONS	0	
GSEC_ACQCRD_CRED_EXPIRE@	38		GSEC_TICKET	8	
GSEC_ACQCRD_CRED_HNDL@	28		GSEC_UNWRAP_CONTXT_HNDL@	20	
GSEC_ACQCRD_PRIN@	30		GSEC_UNWRAP_MSG@	38	
GSEC_ACQCRD_REQ_EXPIRE@	20		GSEC_UNWRAP_STATE@	40	
GSEC_ACQUIRECRED	0		GSEC_UNWRAP_TOKEN@	30	
GSEC_APPLICATION	18		GSEC_UNWRAP_TOKLN@	28	
GSEC_CONT_CNTXT_EXPIRE@	40		GSEC_UNWRAPMSG	0	
GSEC_CONT_RET_FLAGS@	38		GSEC_USERID	10	
GSEC_CONT_TOKEN@	30		GSEC_VER_CONTXT_HNDL@	20	
GSEC_CONT_TOKLN@	28		GSEC_VER_MSG@	30	
GSEC_CONTSECCONTEXT	0		GSEC_VER_MSGLN@	28	
GSEC_CONTXT_HNDL@	20		GSEC_VER_TOKEN@	40	
GSEC_DEL_CONTXT_HNDL@	20		GSEC_VER_TOKLN@	38	
GSEC_DELSECCONTEXT	0		GSEC_VERMIC	0	
GSEC_EXPCRD_CRED_HNDL@	10		GSEC_WRAP_CONTXT_HNDL@	20	
GSEC_EXPCRD_CRED_HNDL@	20		GSEC_WRAP_MSG@	38	
GSEC_EXPCRD_OUT_TOK@	28		GSEC_WRAP_MSGLN@	30	
GSEC_EXPORTCRED	0		GSEC_WRAP_OUT_TOK@	48	
GSEC_EXPORTCRED	0		GSEC_WRAP_REQ_FLAG@	28	
GSEC_EXPORTSECCONTEXT	0		GSEC_WRAP_STATE@	40	
GSEC_EXPSEC_CONTXT_HNDL@	20		GSEC_WRAPMSG	0	
GSEC_EXPSEC_OUT_TOK@	28		GSEC31_ACC_CONTXT_EXPIRE@	24	
GSEC_FUNC_COUNT@	10		GSEC31_ACC_CONTXT_HNDL@	18	
GSEC_FUNC@	8		GSEC31_ACC_CRED_HNDL@	2C	
GSEC_GET_CONTXT_HNDL@	20		GSEC31_ACC_OUT_TOK@	28	
GSEC_GET_MSG@	30		GSEC31_ACC_PRIN@	1C	
GSEC_GET_MSGLN@	28		GSEC31_ACC_RET_FLAGS@	20	
GSEC_GET_OUT_TOK@	38		GSEC31_ACC_TOKEN@	14	
GSEC_GETMIC	0		GSEC31_ACC_TOKLN@	10	
GSEC_GSS_ACEE@	18		GSEC31_ACCSECCONTEXT	0	
GSEC_GSS_FUNC@	0		GSEC31_ACQCRD_CRED_EXPIRE@	1C	
GSEC_GSS_MAJOR@	8		GSEC31_ACQCRD_CRED_HNDL@	14	
GSEC_GSS_MINOR@	10		GSEC31_ACQCRD_PRIN@	18	
GSEC_GSSAPI_COMMON	0		GSEC31_ACQCRD_REQ_EXPIRE@	10	
GSEC_IMPCRD_CRED_HNDL@	30		GSEC31_ACQUIRECRED	0	
GSEC_IMPCRD_TOKEN@	28		GSEC31_APPLICATION	C	
GSEC_IMPCRD_TOKLN@	20		GSEC31_CONT_CNTXT_EXPIRE@	20	
GSEC_IMPORTCRED	0		GSEC31_CONT_RET_FLAGS@	1C	
GSEC_IMPORTSECCONTEXT	0		GSEC31_CONT_TOKEN@	18	
GSEC_IMPSEC_CONTXT_HNDL@	18		GSEC31_CONT_TOKLN@	14	
GSEC_IMPSEC_CONTXT_HNDL@	30		GSEC31_CONTSECCONTEXT	0	
GSEC_IMPSEC_TOKEN@	28		GSEC31_CONTXT_HNDL@	10	
GSEC_IMPSEC_TOKLN@	20		GSEC31_DEL_CONTXT_HNDL@	10	
GSEC_INIT_CNTXT_EXPIRE@	50		GSEC31_DELSECCONTEXT	0	

Name	Hex Offset	Hex Value
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_MSG@	18	
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	
GSEC31_INIT_CRED_HNDL@	1C	
GSEC31_INIT_OUT_TOK@	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_OPTIONS	0	
GSEC31_TICKET	4	
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	
GSEC31_WRAP_REQ_FLAG@	14	
GSEC31_WRAP_STATE@	20	
GSEC31_WRAPMSG	0	

## COMY: 64-BIT enabled SAF callable services

**Common Name:** 64-bit enabled SAF callable services

**Macro ID:** IRRPCOMY

**DSECT Name:** COMY, PGSN, RAUX

**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
PGSN	24 bytes
RAUX	152 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	56	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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	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 <i>z/OS Security Server RACF Callable Services</i> for function specific parameter lists for callable service R_PgmSignVer.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	152	RAUX64	

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	ADDRESS	8	RAUX64_NUM_PARMS@	
4	(4)	ADDRESS	8	RAUX64_ACEE_ALET@	
8	(8)	ADDRESS	8	RAUX64_ACEE@	
12	(C)	ADDRESS	8	RAUX64_PARM_ALET@	
16	(10)	ADDRESS	8	RAUX64_OPTION_WORD@	
20	(14)	ADDRESS	8	RAUX64_LINK_VALUE@	
24	(18)	ADDRESS	8	RAUX64_ATTRIBUTES@	
28	(1C)	ADDRESS	8	RAUX64_COMPONENT@	
32	(20)	ADDRESS	8	RAUX64_FMID@	
36	(24)	ADDRESS	8	RAUX64_SUBTYPE@	
40	(28)	ADDRESS	8	RAUX64_EVENT@	
44	(2C)	ADDRESS	8	RAUX64_QUALIFIER@	
48	(30)	ADDRESS	8	RAUX64_CLASS@	
52	(34)	ADDRESS	8	RAUX64_RESOURCE@	
56	(38)	ADDRESS	8	RAUX64_LOG_STRING@	
60	(3C)	ADDRESS	8	RAUX64_RELOCATE_COUNT@	
64	(40)	ADDRESS	8	RAUX64_RELOCATE_PTR@	
68	(44)	ADDRESS	8	RAUX64_MESSAGE_COUNT@	
72	(48)	ADDRESS	8	RAUX64_MESSAGE_PTR@	

## Constants

## Cross Reference

Name	Hex Offset	Hex Value
COMY	0	
COMY_WORKA_STOR@	0	
COMY_SAFRC_ALET@	8	
COMY_SAFRC_STOR@	10	
COMY_RACRC_ALET@	18	
COMY_RACRC_STOR@	20	
COMY_RACSC_ALET@	28	
COMY_RACSC_STOR@	30	
PGSN64	0	
PGSN64_NUM_PARMS@	0	
PGSN64_FUNC@	8	
PGSN64_FUNC_PARML@	10	
RAUX64	0	
RAUX64_NUM_PARMS@	0	
RAUX64_ACEE_ALET@	8	
RAUX64_ACEE@	10	
RAUX64_PARM_ALET@	18	
RAUX64_OPTION_WORD@	20	
RAUX64_LINK_VALUE@	28	
RAUX64_ATTRIBUTES@	30	
RAUX64_COMPONENT@	38	
RAUX64_FMID@	40	
RAUX64_SUBTYPE@	48	
RAUX64_EVENT@	50	
RAUX64_QUALIFIER@	58	
RAUX64_CLASS@	60	
RAUX64_RESOURCE@	68	
RAUX64_LOG_STRING@	70	
RAUX64_RELOCATE_COUNT@	78	
RAUX64_RELOCATE_PTR@	80	
RAUX64_MESSAGE_COUNT@	88	
RAUX64_MESSAGE_PTR@	90	

## CRED: z/OS UNIX System Services Credential Structure

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	128	CRED	
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
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	CREFUNCTION	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	First path name: The path name specified by the user on syscall. For rename and link, this is the OLD path name.
					Reserved
16	(10)	UNSIGNED	2	*	
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	CREFDN1	First file name: The name of the part of the first path name that is being checked on the current OPEN/MVS-RACF call.
					Reserved
28	(1C)	UNSIGNED	2	*	
30	(1E)	UNSIGNED	2	CREFDN1LEN	Length of first file name
32	(20)	ADDRESS	4	CREFDN1ALET	ALET for first file name
36	(24)	ADDRESS	4	CREFDN1ADDR	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 or unmount.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
40	(28)	UNSIGNED	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
52	(34)	CHARACTER	12	CREFDN2	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	CREFDN2LEN	Length of second file name
56	(38)	ADDRESS	4	CREFDN2ALET	ALET for second file name
60	(3C)	ADDRESS	4	CREFDN2ADDR	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 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)	CHARACTER	28	*	End of Cred
128	(80)	CHARACTER	0	*	End of the Cred

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	80	CredAclInfo	
0	(0)	UNSIGNED	4	CredAccAclLen	Length of ACL buffer
4	(4)	UNSIGNED	4	CredAccAclAlet	ALET of ACL buffer
8	(8)	CHARACTER	4	*	Reserved
12	(0C)	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
28	(1C)	ADDRESS	4	CredFileModelAcl	Address of file model ACL
32	(20)	UNSIGNED	4	CredDirModelAclLen	Length of ACL buffer
36	(24)	UNSIGNED	4	CredDirModelAclAlet	ALET of 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
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	*	End of the CredAclInfo

## CRED

### Cross Reference

Name	Hex Offset	Hex Value	Level
CRED	0		1
CREFN1	1C		2
CredAccAcl	C		2
CredAccAclAlet	4		2
CredAccAclLen	0		2
Cred_ACEEptr	58		2
CredAclAlet	44		2
CredAclInfo	0		1
CredAclInfoPtr	4C		2
CredAclPtr	48		2
CredDirModelAcl	2C		2
CredDirModelAclAlet	24		2
CredDirModelAclLen	20		2
CredFileModelAcl	1C		2
CredFileModelAclAlet	14		2
CredFileModelAclLen	10		2
CREFN1ADDR	24		3
CREFN1ALET	20		3
CREFN1LEN	1E		3
CREFN2	34		2
CREFN2ADDR	3C		3
CREFN2ALET	38		3
CREFN2LEN	36		3
CREFUNCTION	A		2
CREDID	0		2
CREDLEN	5		2
CREDLFSFLAGS	D		2
CREDNAMEFLG	C		2
CredPDirModelAcl	4C		2
CredPDirModelAclAlet	44		2
CredPFileModelAcl	3C		2
CredPFileModelAclAlet	34		2
CREDPN1	10		2
CREDPN1ADDR	18		3
CREDPN1ALET	14		3
CREDPN1LEN	12		3
CREDPN2	28		2
CREDPN2ADDR	30		3
CREDPN2ALET	2C		3
CREDPN2LEN	2A		3
Cred_ROSeclabel	5C		2
Cred_Seclabel	50		2
CREDSP	4		2
CREDUTYPE	9		2
CREDVER	8		2

## CREI: z/OS UNIX System Services Credential Structure for IPC

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
					User type X'02' - System function caller (treated like superuser)
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 ID of the IPC service that is being checked
20	(14)	CHARACTER	44	*	Reserved
64	(40)	CHARACTER		*	End of the CREI

### Constants

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

Name	Hex Offset	Hex Value	Level
CREI	0		1
CREIFUNCTION	A		2
CREIID	0		2
CREIIPCID	10		2
CREIIPCKEY	C		2
CREILEN	5		2
CREISP	4		2
CREIUTYPE	9		2
CREIVER	8		2

## DAUT: RACROUTE REQUEST=DIRAUTH Parameter List (Request Section)

**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 SAFR plus offset in SAFPRACP  
**Serialization:** None  
**Function:** Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=DIRAUTH routine

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	36	DAUTPARM	DIRAUTH parameters
0	(0)	BITSTRING	1	DAUTLOGP	Auditing option flags
		1... ..		DAUTASIS	1 = ASIS
		.1.. ..		DAUTNFAI	1 = NOFAIL
		..1. ....		DAUTNONE	1 = NONE
		...1 1111		*	Reserved for LOG keyword
1	(1)	UNSIGNED	1	DAUTPVER	Parameter list version number
					00-original version, length 8
					01-variable length
2	(2)	UNSIGNED	2	DAUTPLEN	Length of parameter list if version 01
4	(4)	ADDRESS	4	DAUTRTOK	Message RTOKEN address
8	(8)	BITSTRING	1	DAUTFLG1	Option flags
		1... ..		DAUTMAC	1=MAC type checking requested
		.1.. ..		DAUTRVRS	1=Reverse MAC type checking requested
		..1. ....		DAUTEQUM	1=Equal MAC type checking requested
		...1 1111		*	Reserved
9	(9)	BITSTRING	1	DAUTFLG2	Option flags
		1... ..		DAUTREAD	1=Read access requested
		.1.. ..		DAUTRDWR	1=Readwrite access requested
		..1. ....		DAUTWRIT	1=Write access requested
		...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

Name	Hex Offset	Hex Value	Level
DAUTACEE	18		2
DAUTALET	1C		2
DAUTASIS	0	80	3
DAUTCLAS	C		2
DAUTEQUM	8	20	3
DAUTFLG1	8		2
DAUTFLG2	9		2
DAUTLOGP	0		2
DAUTLSTR	20		2
DAUTMAC	8	80	3
DAUTNFAI	0	40	3
DAUTNONE	0	20	3
DAUTPARM	0		1
DAUTPLEN	2		2
DAUTPVER	1		2
DAUTRDWR	9	40	3
DAUTREAD	9	80	3
DAUTRSEC	10		2
DAUTRVRS	8	40	3
DAUTRTOK	4		2
DAUTUSEC	14		2
DAUTWRIT	9	20	3

## DEXP: Data Encryption Exit Parameter List

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		DEXPL	
Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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: X'00000000' 0 means encrypt the data. 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: X'01' 1 - Group X'02' 2 - User X'03' 3 - Connect X'04' 4 - Data set X'05' 5 - General
24	(18)	ADDRESS	4	DEXGENRL 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

Name	Hex Offset	Hex Value	Level
DEXCONNECT	14	3	3
DEXCOMP	4	4	3
DEXCRYPT	4	0	3
DEXCTEXT	C		2
DEXDLGTH	8		2
DEXDS	14	4	3
DEXETEXT	10		2
DEXFUNC	4		2
DEXGENRL	14	5	3
DEXGROUP	14	1	3
DEXLEN	0		2
DEXPL	0		1
DEXPROF	1C		2
DEXTMPLC	14		2
DEXTMPLN	18		2
DEXUSER	14	2	3



## DSDT: Data Set Descriptor Table

NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	DSDT	Located through RCVT
0	(0)	CHARACTER	4	DSDTID	EBCDIC 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
		.1.. ..		DSDTBFMT	Flag for backup format 0 - Indicates OLD format or no BACKUP data sets specified at IPL. 1 - Indicates RDS format
		..1. ....		DSDTPXST	Existence of primary databases 0 - No primary data sets specified at IPL. 1 - At least one primary data set specified at IPL.
		...1 ....		DSDTBXST	Existence of backup databases 0 - No backup data sets specified at IPL. 1 - At least one backup data set specified at IPL.
		.... 1111		*	
14	(E)	CHARACTER	2	*	Doubleword alignment
16	(10)	CHARACTER	8	*	Reserved for expansion
<b>GRS latch sets:</b>					
Each set contains 1 latch for each data set in the RACF database for a total for DSDTNUM latches.					
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:					
136	(88)	BITSTRING	1	DSDTDSFL	Data Sharing Flags
		1... ..		DSDTDSRQ	

# DSDT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		.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:					
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
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	
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 suffix (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.. ..		DSDPD SVM	IXLVCTR message issued
		..11 1111		*	Reserved
291	(123)	CHARACTER	1	*	Boundary Alignment
292	(124)	CHARACTER	8	*	Reserved 2 words
300	(12C)	CHARACTER	8	DSDPDDNM	Dynamic allocation DDNAME
308	(134)	CHARACTER	12	*	Reserved 3 words
Entry for Backup Data Set:					
320	(140)	CHARACTER	176	DSDTBACK	ENTRY FOR BACKUP DATA SET

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 ....		DSDBRF5H	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
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 database
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	DSDBCND5	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:

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

# DSDT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
8	(8)	ADDRESS	4	DSDEINDX	PTR TO IN-STORAGE BUFFERS OR RESIDENT INDEX TABLE. ZERO IF NO RESIDENT BLOCKS FOR THE DATA SET
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 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	DSDESDDS	
80	(50)	CHARACTER	16	DSDESNAM	XES Structure Name for this database
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
120	(78)	CHARACTER	8	DSDECNSN	XES Connection Name suffix (System Name)
128	(80)	CHARACTER	12	DSDEVTK	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... ....		DSDEDSM	IXLCACHE message issued
		.1.. ....		DSDEDSVM	IXLVCTR 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

## Constants

Len	Type	Value	Name	Description
1	DECIMAL	2	DSBTCVSN	CURRENT VERSION NUMBER
4	CHARACTER	DSDT	DSDTIDC	EBCDIC ID
Constants for RACF Sysplex Data Sharing:				
48	CHARACTER	See	DSDTPRLS	IRRXC00.LATCHSET.PRIMARY
48	CHARACTER	Desc.	DSDTBKLS	IRRXC00.LATCHSET.BACKUP
Constants for DSDTDSMO:				
1	DECIMAL	0	DSDTDSNO	DSDTDSMO=DSDTDSNO - NO DATA SHARING ACROSS IRRXC00
1	DECIMAL	1	DSDTDSHR	DSDTDSMO=DSDTDSHR - IN DATA SHARING MODE, ALL OTHER MEMBERS OF IRRXC00 IN DATASHARING OR READONLY
1	DECIMAL	2	DSDTDSRO	DSDTDSMO=DSDTDSRO - IN READONLY MODE, ALL OTHER MEMBERS OF IRRXC00 IN DATASHARING OR READONLY
1	DECIMAL	3	DSDTTRAN	DSDTDSMO=DSDTTRAN - IN TRANSITION MODE, IS EITHER IN READONLY OR DATASHARING AFTER CONNECT SERVICES CALLED.
Constants for CONSTAT fields:				
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.

8	CHARACTER	SYSZRACF	DSDTQNAM
48	CHARACTER	See	DSDTRNAM
		note.	

**Note:** The value for DSDTRNAM is DSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDTDSDT.

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
DSDBACTV	15D	80	5	DSDBMSTR	15D	20	5
DSDBALTI	15D	04	5	DSDBNREC	15E		4
DSDBBAM	158		4	DSDBPRIM	15D	40	5
DSDBCACF	1D2		5	DSDBRBIP	1D1	80	6
DSDBCBLN	18E		4	DSDBRBLD	1D1		5
DSDBCNAM	1B0		5	DSDBRFSH	15D	10	5
DSDBCNDS	1B4		6	DSDBRUCB	150		4
DSDBCNP	1B0		6	DSDBRXNO	160		4
DSDBCNSL	1B7		6	DSDBSDS	190		4
DSDBCNSN	1B8		6	DSDBSHR	15D	08	5
DSDBCONS	1D0		5	DSDBSNAM	190		5
DSDBCTOK	1A0		5	DSDBSTAT	15D		4
DSDBDAT	15D	02	5	DSDBVLEN	1CC		5
DSDBDCB	140		4	DSDBVTOK	1C0		5
DSDBDDNM	1DC		4	DSDBXLEN	154		4
DSDBDEB	144		4	DSDE	0		1
DSDBDSCM	1D2	80	6	DSDEACTV	1D	80	3
DSDBDSN	161		4	DSDEALTI	1D	04	3
DSDBDSNL	15C		4	DSDEBAM	18		2
DSDBDSNO	18D		4	DSDECACF	92		3
DSDBDSVM	1D2	40	6	DSDECBLN	4E		2
DSDBHDR	14C		4	DSDECMS	1D	01	3
DSDBINDX	148		4	DSDECNAM	70		3

# DSDT

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
DSDECNDS	74		4	DSDPSNAM	E0		5
DSDECNP	70		4	DSDPSTAT	AD		4
DSDECNSL	77		4	DSDPVLEN	11C		5
DSDECNSN	78		4	DSDPVTOK	110		5
DSDECONS	90		3	DSDPXLEN	A4		4
DSDECTOK	60		3	DSDT	0		1
DSDEDAT	1D	02	3	DSDTBACK	140		3
DSDEDCB	0		2	DSDTBFMT	D	40	3
DSDEDDNM	9C		3	DSDTBLNM	50		2
DSDEDEB	4		2	DSDTBLTK	80		2
DSDEDSM	92	80	4	DSDTBXST	D	10	3
DSDEDSN	21		2	DSDTDSDX	8		2
DSDEDSNL	1C		2	DSDTDSFL	88		2
DSDEDSNO	4D		2	DSDTDSMO	89		2
DSDEDSVM	92	40	4	DSDTDSRQ	88	80	3
DSDEHDR	C		2	DSDTENTY	90		2
DSDEINDX	8		2	DSDTFLAG	D		2
DSDEMSTR	1D	20	3	DSDTID	0		2
DSDENREC	1E		2	DSDTNUM	4		2
DSDEPRIM	1D	40	3	DSDTPFMT	D	80	3
DSDERBIP	91	80	4	DSDTPLNM	18		2
DSDERBLD	91		3	DSDTPLTK	48		2
DSDERFSH	1D	10	3	DSDTPRIM	90		3
DSDERUCB	10		2	DSDTPXST	D	20	3
DSDERXNO	20		2	DSDTVRSN	C		2
DSDESDS	50		2				
DSDESHR	1D	08	3				
DSDESNAM	50		3				
DSDESTAT	1D		2				
DSDEVLEN	8C		3				
DSDEVTOK	80		3				
DSDEXLEN	14		2				
DSDPACTV	AD	80	5				
DSDPALT	AD	04	5				
DSDPBAM	A8		4				
DSDPCACF	122		5				
DSDPCBLN	DE		4				
DSDPCNAM	100		5				
DSDPCNDS	104		6				
DSDPCNP	100		6				
DSDPCNSL	107		6				
DSDPCNSN	108		6				
DSDPCONS	120		5				
DSDPCTOK	F0		5				
DSDPDAT	AD	02	5				
DSDPDCB	90		4				
DSDPDDNM	12C		4				
DSDPDEB	94		4				
DSDPDSCM	122	80	6				
DSDPDSN	B1		4				
DSDPDSNL	AC		4				
DSDPDSNO	DD		4				
DSDPDSVM	122	40	6				
DSDPHDR	9C		4				
DSDPINDX	98		4				
DSDPMSTR	AD	20	5				
DSDPNREC	AE		4				
DSDPPRIM	AD	40	5				
DSDPRBIP	121	80	6				
DSDPRBLD	121		5				
DSDPRFSH	AD	10	5				
DSDPRUCB	A0		4				
DSDPRXNO	B0		4				
DSDPSDS	E0		4				
DSDPSHR	AD	08	5				

\_\_\_\_\_ End of NOT Programming Interface Information \_\_\_\_\_

## ENFP

---

### ENFP: RACF-Parameter List for ENF event code 62 listen exits

**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:** 28 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

Offsets			Len		
Dec	Hex	Type		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 = X'01'
8	(8)	SIGNED	2	IRR_ENFLEN	Parameter list length
10	(A)	CHARACTER	2	*	Reserved
12	(C)	CHARACTER	4	IRR_ENFQ	Qualifier code
12	(C)	CHARACTER	1	IRR_ENFQ_LIST	SETR RACLIST signal when X'80'
13	(D)	CHARACTER	1	IRR_ENFQ_REFR	SETR RACLIST REFRESH signal when X'80'
14	(E)	CHARACTER	1	IRR_ENFQ_NOLIST	SETR NORACLIST signal when X'80'
15	(F)	CHARACTER	1	*	Reserved
16	(10)	CHARACTER	8	IRR_ENFCLASS	Class name

### Cross Reference

Name	Hex Offset	Hex Value	Level
ENFP	0		1
IRR_ENFCLASS	10		2
IRR_ENFID	0		2
IRR_ENFLEN	8		2
IRR_ENFQ	C		2
IRR_ENFQ_LIST	C		2
IRR_ENFQ_REFR	D		2
IRR_ENFQ_NOLIST	E		2
IRR_ENFVER	6		2



## ENF2: RACF-Parameter List for ENF event code 71 listen exits

**Common Name:** Mapping macro for RACF ENF event code 71  
**Macro ID:** IRRPEN2  
**DSECT Name:** ENF2  
**Owning Component:** SAF  
**Eye-Catcher ID:** IRRENF2  
**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 71 listen exits

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	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'01'
8	(8)	SIGNED	2	IRR_ENF2LEN	Parameter list length
10	(A)	CHARACTER	2	*	Reserved
12	(C)	CHARACTER	4	IRR_ENF2Q	Qualifier code
12	(C)	BITSTRING	1	IRR_ENF2Q_CON	CONNECT command when X'80'
				IRR_ENF2Q_REM	REMOVE command when X'40'
				IRR_ENF2Q_ALU_REVOKE	ALTUSER REVOKE command when X'20'
13	(D)	CHARACTER	3	*	Reserved
16	(10)	CHARACTER	8	IRR_ENF2USER	RACF userID

### Cross Reference

Name	Hex Offset	Hex Value	Level
ENFP2	0		1
IRR_ENF2ID	0		2
IRR_ENF2VER	6		2
IRR_ENF2LEN	8		2
IRR_ENF2Q	C		2
IRR_ENF2Q_CON	C		2
IRR_ENF2Q_REM	C		2
IRR_ENF2Q_ALU_REVOKE	C		2
IRR_ENF2USER	10		2

---

**EVXP: RACF Command Exit Parameter List**

**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:** 44 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		EVXPL	
Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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..		EVXALTDSD	X'04' ALTDSD
		.... ..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 flag space) 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		...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)
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-aligned 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 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-aligned 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 initialized to blanks. Can be used to supply message insert for IRRV0221 when the pre-processing call sets register 15 to a value other than 0 or 4.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Name	Hex Offset	Hex Value	Level
CMDBUFD	0	0	2
CMDBUFL	0	0	2
CMDBUFO	0	0	2
EVXABCD	1C	4	2
EVXABND	8		2
EVXACD	8		2
EVXACEE	10	4	2
EVXADDGR	4		2
EVXADDSD	4		2
EVXADDUS	4		2
EVXALTD	4		2
EVXALTGR	4		2
EVXALTUS	4		2
EVXAT	8		2
EVXCALLR	4	4	2
EVXCMBUF	C	0	2
EVXCMDRC	18	4	2
EVXCONNE	4		2
EVXDELDS	4		2
EVXDELGR	4		2
EVXDELUS	4		2
EVXFLAGS	8		2
EVXLEN	0		2
EVXLISTD	4		2
EVXLISTG	4		2
EVXLISTU	4		2
EVXMSSG	28	4	2
EVXOPER	8		2
EVXPARM	8		2
EVXPASSW	4		2
EVXPERMI	4		2
EVXPOST	8		2
EVXPRE	8		2
EVXRALTE	4		2
EVXRASP	8		2
EVXRDEFI	4		2
EVXRDELE	4		2
EVXREMOV	4		2
EVXRLIST	4		2
EVXSEARC	4		2
EVXSETRO	4		2
EVXSRCND	20	4	2
EVXSRCUS	24	4	2
EVXWORK	14	4	2

**FACL: z/OS UNIX System Services access control list**

**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 System Services file or directory

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 (header plus all entries)
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
		..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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

## FACL

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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_	RACF Return Code
				RETURN_CODE	
40	(28)	UNSIGNED	4	FACL_EDIT_RACF_	RACF Reason Code
				REASON_CODE	
44	(2C)	CHARACTER	20	*	Reserved
64	(40)	CHARACTER	0	FACL_EDIT_HEADER_EXTENSION	Reserved
64	(40)	CHARACTER	0	FACL_EDIT_ENTRIES	Start of entries

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	Key for Sorting ACL
3	(3)	UNSIGNED	1	FACL_ENTRY_TYPE	Type of ACL entry
4	(4)	SIGNED	4	FACL_ENTRY_ID	UID/GID value
8	(8)	CHARACTER	0	FACL_ENTRY_END	

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)	UNSIGNED	1	FACL_EDIT_TYPE	Type of ACL entry
4	(4)	SIGNED	4	FACL_EDIT_ID	UID/GID value
8	(8)	CHARACTER	0	FACL_EDIT_END	

## FACL Constants

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	

## Cross Reference

Name	Hex Offset	Hex Value	Name	Hex Offset	Hex Value
FACL	0		FACL_MIN_HEADER	0	
FACL_ACCESS	1D	80	FACL_NUM_ENTRY	8	
FACL_DEL_ENTRY	0	80	FACL_NUM_USERS	1E	
FACL_DIR_MODEL	1D	20	FACL_PERM_BITS	2	07
FACL_EDIT	0		FACL_PERM_BITS_MASK	1	07
FACL_EDIT_ACCESS	1D	80	FACL_PERMS_MASK	1	
FACL_EDIT_DIR_MODEL	1D	20	FACL_RACF_REASON_CODE	28	
FACL_EDIT_END	8		FACL_RACF_RETURN_CODE	24	
FACL_EDIT_ENTRIES	40		FACL_READ	2	04
FACL_EDIT_ENTRY	0		FACL_READ_MASK	1	04
FACL_EDIT_ERROFF	20		FACL_SORT_KEY	3	
FACL_EDIT_EXECUTE	2	01	FACL_VERS	1C	
FACL_EDIT_FILE_MODEL	1D	40	FACL_WRITE	2	02
FACL_EDIT_FLAG	1D		FACL_WRITE_MASK	1	02
FACL_EDIT_HDR_ID	0				
FACL_EDIT_HEADER_EXTENSION	40				
FACL_EDIT_ID	4				
FACL_EDIT_LEN	4				
FACL_EDIT_LEN_ENTRY	A				
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				

---

**FAST: RACROUTE REQUEST=FASTAUTH Parameter List (Request Section)**

**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 SAFR plus offset in SAFPRACP  
**Serialization:** None  
**Function:** Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=FASTAUTH routine

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	28	FASTPARG	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
		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
		.... 1111		*	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 Version: 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



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
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 ICHRF01 and ICHRF02. FASTALET exists only when FASTPVER is 1 or higher.
32	(20)	ADDRESS	4	FASTLOGS	LOGSTR address: If nonzero, it points to a 1-byte length field followed by character data that can be from 0 to 255 bytes long. FASTLOGS exists only when FASTPVER is 1 or higher.
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.

## Cross Reference

Name	Hex Offset	Hex Value	Level
FASTACEE	C		2
FASTACHK	1	10	3
FASTALET	1C		2
FASTALTR	0	80	3
FASTAPPL	10		2
FASTASIS	1	80	3
FASTATTR	0		2
FASTCLAS	8		2
FASTCNTL	0	08	3
FASTCRIT	28		2
FASTENTP	4		2
FASTENTX	1	20	3
FASTENVR	24		2
FASTINST	18		2
FASTLOGO	1		2
FASTLOGS	20		2
FASTNFAL	1	40	3
FASTPARM	0		1
FASTPLEN	2		2
FASTPVER	3		2
FASTREAD	0	02	3
FASTUPDT	0	04	2
FASTWKA	14		2

## FC: z/OS UNIX System Services Security Function Code Table

**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

### Constants

Len	Type	Value	Name	Description
1	DECIMAL	1	IRRSIU00#	Function code 1 - initUSP
1	DECIMAL	2	IRRSU00#	Function code 2 - deleteUSP
1	DECIMAL	3	IRRSF00#	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	IRRSYG00#	Function code 10 - get supplemental groups
1	DECIMAL	11	IRRSU00#	Function code 11 - set UID
1	DECIMAL	12	IRRSU00#	Function code 12 - set effective UID
1	DECIMAL	13	IRRSYG00#	Function code 13 - set GID
1	DECIMAL	14	IRRSYG00#	Function code 14 - set effective GID
1	DECIMAL	15	IRRSYG00#	Function code 15 - change owner group
1	DECIMAL	16	IRRSYG00#	Function code 16 - change file mode
1	DECIMAL	17	IRRSYG00#	Function code 17 - change file audit options
1	DECIMAL	18	IRRSYG00#	Function code 18 - exec_setUID/GID
1	DECIMAL	19	IRRSYG00#	Function code 19 - audit
1	DECIMAL	20	IRRSYG00#	Function code 20 - check process owner
1	DECIMAL	21	IRRSYG00#	Function code 21 - query security options
1	DECIMAL	22	IRRSYG00#	Function code 22 - query file options
1	DECIMAL	23	IRRSYG00#	Function code 23 - clear_setid
1	DECIMAL	24	IRRSYG00#	Function code 24 - check file owner
1	DECIMAL	25	IRRSYG00#	Function code 25 - make_root_FSP
1	DECIMAL	26	IRRSYG00#	Function code 26 - PTRACE authority check
1	DECIMAL	27	IRRSYG00#	Function code 27 - get users groups
1	DECIMAL	28	IRRSYG00#	Function code 28 - fork exit
1	DECIMAL	29	IRRSYG00#	Function code 29 - makeISP
1	DECIMAL	30	IRRSYG00#	Function code 30 - check IPC access
1	DECIMAL	31	IRRSYG00#	Function code 31 - R_IPC_ctl
1	DECIMAL	32	IRRSYG00#	Function code 32 - ck_owner_two_files
1	DECIMAL	33	IRRSYG00#	Function code 33 - get UIDs, GIDs, and supplemental groups
1	DECIMAL	34	IRRSYG00#	Function code 34 - R_dceinfo
1	DECIMAL	35	IRRSYG00#	Function code 35 - R_dcekey
1	DECIMAL	36	IRRSYG00#	Function code 36 - R_dceruid
1	DECIMAL	37	IRRSYG00#	Function code 37 - R_dceauth
1	DECIMAL	38	IRRSYG00#	Function code 38 - initACEE
1	DECIMAL	39	IRRSYG00#	Function code 39 - R_admin
1	DECIMAL	40	IRRSYG00#	Function code 40 - R_usermap

Len	Type	Value	Name	Description
1	DECIMAL	41	IRRSDL00#	Function code 41 - R_datalib
1	DECIMAL	44	IRRSXP00#	Function code 44 - R_PKIServ
1	DECIMAL	45	IRRSCH00#	Function code 45 - R_cacheserv
1	DECIMAL	46	IRRSYP00#	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	IRRSXS00#	Function code 50 - R_GenSec
1	DECIMAL	51	IRRSAX00#	Function code 51 - R_Auditx
1	DECIMAL	52	IRRSX00#	Function code 52 - R_GetInfo
1	DECIMAL	53	IRRSXS00#	Function code 53 - R_PgmSignVer

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
IRRSAX00#	0	33	2	IRRSWP00#	0	31	2
IRRSAU00#	0	13	2				
IRRSBA00#	0	11	2				
IRRSBF00#	0	10	2				
IRRSCH00#	0	2D	2				
IRRSCL00#	0	1F	2				
IRRSCL00#	0	2F	2				
IRRSCL00#	0	F	2				
IRRSCL00#	0	17	2				
IRRSCL00#	0	20	2				
IRRSCL00#	0	25	2				
IRRSCL00#	0	22	2				
IRRSCL00#	0	23	2				
IRRSCL00#	0	41	2				
IRRSCL00#	0	2	2				
IRRSCL00#	0	E	2				
IRRSCL00#	0	27	2				
IRRSCL00#	0	C	2				
IRRSCL00#	0	12	2				
IRRSCL00#	0	1C	2				
IRRSCL00#	0	21	2				
IRRSCL00#	0	A	2				
IRRSCL00#	0	34	2				
IRRSCL00#	0	9	2				
IRRSCL00#	0	32	2				
IRRSCL00#	0	26	2				
IRRSCL00#	0	28	2				
IRRSCL00#	0	1	2				
IRRSCL00#	0	6	2				
IRRSCL00#	0	18	2				
IRRSCL00#	0	1E	2				
IRRSCL00#	0	14	2				
IRRSCL00#	0	7	2				
IRRSCL00#	0	3	2				
IRRSCL00#	0	1D	2				
IRRSCL00#	0	5	2				
IRRSCL00#	0	19	2				
IRRSCL00#	0	35	2				
IRRSCL00#	0	1A	2				
IRRSCL00#	0	2C	2				
IRRSCL00#	0	2E	2				
IRRSCL00#	0	16	2				
IRRSCL00#	0	15	2				
IRRSCL00#	0	30	2				
IRRSCL00#	0	D	2				
IRRSCL00#	0	B	2				
IRRSCL00#	0	24	2				
IRRSCL00#	0	1B	2				
IRRSCL00#	0	8	2				

---

**FXAP: RACROUTE REQUEST=FASTAUTH Extended Function Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	ADDRESS	4	ARFXEXPL	Address of the current ICHRFX01 or ICHRFX02 parameter list which can be used by ICHRFX03 or ICHRFX04
4	(4)	ADDRESS	4	ARFXALET	ALET of the data space or address space containing the profile and profile name. It is set for post-processing exit only.
8	(8)	ADDRESS	4	ARFXPROF	Address of the profile being used in the data space or address space. It is set for post-processing 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 post-processing exit only.
14	(E)	UNSIGNED	2	ARFXAPRL	Length of authorization profile name. It is set for post-processing 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.
		.... 0000		*	Reserved
		0000 0000		*	Reserved
26	(1A)	UNSIGNED	2	*	Reserved

| **Constants**

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

## GANC: GENERICANCHOR settings mapping

NOT Programming Interface 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

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

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
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 number of 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)	CHARACTER	1	GENANC_JOBNAME_FLAGS	Flags
		1... ..		GENANC_JOBNAME_GENERIC	1- Jobname is generic
		.111 1111		*	RESERVED
22	(16)	CHARACTER	2	*	RESERVED

## Cross Reference

Name	Hex Offset	Hex Value	Level
GENANC_JOB_COUNT	10		3
GENANC_JOBNAME	8		3
GENANC_JOBNAME_DATA	0		1
GENANC_JOBNAME_DATAPTR	4		2
GENANC_JOBNAME_ENTRIES	8		2
GENANC_JOBNAME_FLAGS	15		3
GENANC_JOBNAME_GENERIC	15	80	4
GENANC_JOBNAME_HEADER	0		2
GENANC_JOBNAME_LENGTH	14		3
GENANC_TABLELEN	0		3
GENANC_TOTAL_JOB_NAMES	4		3
GENANC_SETTINGS	0		1
GENANC_SYSTEM_COUNT	0		2

End of NOT Programming Interface Information

**GAPL: Generic anchor table entry**

NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	GENATE	Generic anchor table entry (GATE)
0	(0)	CHARACTER	4	ATEID	Eyecatcher (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 (in 64-bit storage) 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 64-bit storage
48	(30)	BITSTRING	8	ATETMSTP	GATE creation timestamp
56	(38)	CHARACTER	*	ATEKEYQ	Key qualifier of profile list



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	PRFLFH	Length of header
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
12	(C)	ADDRESS	4	PLELMEMP	RACFVARS member list pointer

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	PLELMEM	RACFVARS member list
0	(0)	UNSIGNED	1	PLELMBRL	RACFVARS member length
1	(1)	CHARACTER	*	PLELMBR	RACFVARS member

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
ATEFLAG	14		2
ATEID	0		2
ATEKEYQ	28		2
ATEKQLEN	16		2
ATELN	9		3
ATEMOBJS	15		2
ATENEXT	24		2
ATENM	C		2
ATEPRF64	28		2
ATEPROFL	20		2
ATERCNT	1C		2
ATESP	8		3
ATESPLN	8		2
ATETMSTP	30		2
ATETYP	14	80	3
ATEVERSN	4		2
GENATE	0		1
GENPLEL	0		1
GENPLELH	0		2
GENPRFL	0		1
PLELFLGS	2		3
PLELMBR	1		2
PLELMBRL	0		2
PLELMEM	0		1
PLELMEMC	8		2
PLELMEML	A		2
PLELMEMP	C		2
PLELMLSZ	10		3
PLELNL	0		3
PLELNM	14		2
PLELPRF	4		3
PLELRTRV	2	80	4
PLELRVAR	0		2
PLELRVCT	A		3
PLELRVRC	8		3
PLELRVRP	C		3
PLELRVRS	0		1
PRFENT	1C		2
PRFINDX	18		4
PRFLID	0		2
PRFLLE	12		2
PRFLH	10		2
PRFLLN	5		3
PRFLNE	C		2
PRFLNEXT	8		2
PRFLSP	4		3
PRFLSPLN	4		2
PRFSRTPL	14		2

End of NOT Programming Interface Information

## GPRFL: 64-bit Storage profile list mapping

NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	GPRFL64	Profile list block header - 64-bit storage version of GENPRFL
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	Address of first memory object containing the profiles themselves - Set only in first memory object
32	(20)	ADDRESS	8	PR64DATA	Address pointing to start of data (PLELs, RVARs) in this memory object
40	(28)	ADDRESS	*	PR64ELT(*)	Array of addresses pointing to list elements

## GPRFL

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	GENPLE2	Profile list element - 64-bit storage version of GENPLEL
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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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 pointer
16	(10)	UNSIGNED	2	PLE2MEMC	RACFVARS member count
18	(12)	UNSIGNED	2	PLE2MEML	RACFVARS member list length

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	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 piece of storage in this memory object (for the next profile)

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
GENPLE2	0		1	PR64ID	0		2
GENPLE2H	0		2	PR64NE	4		2
GPRFL64	0		1	PR64NEXT	10		2
GP64HDR	0		1				
GP64NXTO	0		2				
GP64NXTP	8		2				
PLE2FLGS	2		3				
PLE2MEMC	10		2				
PLE2MEML	12		2				
PLE2MEMP	8		2				
PLE2NL	0		3				
PLE2NM	18		2				
PLE2PRF	8		3				
PLE2RTRV	2	80	4				
PLE2RVAR	0		2				
PLE2RVCT	6		3				
PLE2RVRC	4		3				
PLE2RVRP	10		3				
PLE2RVRS	0		1				
PR64DATA	20		2				
PR64ELT	28		2				
PR64GP64	18		2				

\_\_\_\_\_ End of NOT Programming Interface Information \_\_\_\_\_

## GRPF: In-Storage Generic Profile Map

NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	68	GRPF	GENERIC PROFILE MAP
0	(0)	UNSIGNED	4	GRPFSPLN	
0	(0)	UNSIGNED	1	GRPFS	AREA SUBPOOL NUMBER
1	(1)	UNSIGNED	3	GRPFL	TOTAL AREA LENGTH
4	(4)	CHARACTER	64	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	GRPFLEVEL	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	GRPFDOPOF	OFFSET TO RESOURCE CATEGORY LIST
52	(34)	SIGNED	4	GRPFA2OF	OFFSET TO CONDITIONAL ACCESS LIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
56	(38)	CHARACTER	4	GRPFSLBL	SECLABEL
64	(40)	SIGNED	4	GRPFIPOF	OFFSET TO IP LOOKUP VALUE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	GRPFACL	ACCESS LIST
0	(0)	UNSIGNED	2	GRPFACCT	NUMBER OF ENTRIES
2	(2)	CHARACTER	9	GRPFACLE (*)	ACCESS LIST ENTRIES
2	(2)	CHARACTER	8	GRPFACLU	USERID/GROUPNAME
10	(A)	BITSTRING	1	GRPFACLA	ACCESS AUTHORITY

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	GRPFINS	INSTALLATION DATA
0	(0)	SIGNED	2	GRPFINSL	LENGTH OF INSTALLATION
2	(2)	CHARACTER	*	GRPFINST	INSTALLATION DATA

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	GRPFDP	CATEGORY LIST
0	(0)	SIGNED	2	GRPFDP	NUMBER OF ENTRIES IN CATEGORY LIST
2	(2)	SIGNED	2	GRPFDEPT (*)	CATEGORY LIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	Program 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 ID / Group Name
20	(14)	BITSTRING	1	GRPFAC2A	Access Authority
21	(15)	UNSIGNED	2	GRPFAC2G	Access Count
23	(17)	UNSIGNED	1	GRPFAC2V	Variable Entity Length
24	(18)	CHARACTER	*	GRPFAC2E	Variable Entity
24	(18)	CHARACTER	8	GRPFAC2ID	Class ID
32	(20)	CHARACTER	2	GRPFAC2R	Reserved
34	(22)	UNSIGNED	1	GRPFAC2LV	Variable Length
35	(23)	CHARACTER	*	GRPFAC2V	Variable Entity

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	CHARACTER	16	GRPFIPLK	IP Lookup value for SERVAUTH class profiles

## GRPF

### Constants

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

### Cross Reference

Name	Hex Offset	Hex Value	Level
GRPAC2ID	18		3
GRPAC2LV	22		3
GRPA2RST	5		4
GRPF	0		1
GRPFACCT	0		2
GRPFACL	0		1
GRPFACLA	A		3
GRPFACLE	2		2
GRPFACLU	2		3
GRPFACL2	0		1
GRPFACOF	10		3
GRPFACVE	18		2
GRPFACVL	17		3
GRPFAC2A	14		3
GRPFAC2E	4		3
GRPFAC2L	4		2
GRPFAC2U	C		3
GRPFAC2V	23		3
GRPFAUDT	D		3
GRPFA2CT	0		2
GRPFA2LN	2		2
GRPFA2OF	34		3
GRPFDEPT	2		2
GRPFDPOF	30		3
GRPFDPT	0		1
GRPFDPTL	0		2
GRPFEOS	1C		3
GRPFGACS	15		3
GRPFGAUD	E		3
GRPFGPIN	18		3
GRPFINOF	14		3
GRPFINSO	0		1
GRPFINSL	0		2
GRPFINST	2		2
GRPFIPLK	0		1
GRPFIPOF	40		2
GRPFLL	1		3
GRPFLLDAY	1E		3
GRPFLEVL	F		3
GRPFLGFT	2A		3
GRPFLGNT	27		3
GRPFNTFY	1F		3
GRPFOWNR	4		3
GRPFRSVD	20		3
GRPFRTPD	1A		3
GRPFS	0		3
GRPFSLBL	38		3
GRPFSLVL	1D		3
GRPFSPLN	0		2
GRPFST	4		2
GRPFTZNE	2D		3
GRPFUACC	C		3
GRPFWARN	19		3
GRPPGFLG	4		4



\_\_\_\_\_ End of NOT Programming Interface Information \_\_\_\_\_

## ICRX: Extended identity context reference

<b>Common Name:</b>	Extended identity context reference (ICRX)		
<b>Macro ID:</b>	IRRPICRX		
<b>DSECT Name:</b>	ICRX		
<b>Owning Component:</b>	Resource Access Control Facility (SC1BN)		
<b>Eye-Catcher ID:</b>	ICRX (Offset 0, length 4)		
<b>Storage Attributes:</b>	Subpool	Determined by user	
	Key	Determined by user	
	Residency	Determined by user	
<b>Size:</b>	Variable		
<b>Created by:</b>	RACF or caller of RACF		
<b>Pointed to by:</b>	Output of R_cacheserv, input for RACROUTE REQUEST=VERIFY		
<b>Serialization:</b>	None		
<b>Function:</b>	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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
6	(6)	BITSTRING 1... ..	1	ICRXFLGS ICRXMULT	Flag bits Multi-use ICRX
7	(7)	UNSIGNED	1	ICRXSP	Subpool number
8	(8)	UNSIGNED	4	ICRXLEN	Total length of extended identity context reference
12	(C)	UNSIGNED	4	ICRXICRO	Offset to ICR (identity context reference ICRXICR) or 0, if omitted
16	(10)	UNSIGNED	4	ICRXDIDO	Offset to IDID (distributed identity data) or 0, if omitted. IDID is mapped by macro IRRPIDID.
20	(14)	UNSIGNED	4	ICRXUSRO	Offset to RACF user ID (ICRXUSER) or 0, if omitted.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	ICRXUSER	Userid buffer
0	(0)	UNSIGNED	1	ICRXUSRL	Userid length
1	(1)	CHARACTER	8	ICRXUSRI	Userid

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	16	ICRXICR	Identity context reference
0	(0)	CHARACTER	8	ICRXREFU	Reference part 1
8	(8)	CHARACTER	8	ICRXREFR	Reference part 2

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
ICRX	0		
ICRXDIDO	10		
ICRXFLGS	6		2
ICRXICR	0		
ICRXICRO	C		
ICRXID	0		
ICRXLEN	8		
ICRXMULT	6	80	3
ICRXOFFN	5		
ICRXREFR	8		
ICRXREFU	0		
ICRXSP	7		
ICRXUSER	0		
ICRXUSRI	1		
ICRXUSRL	0		
ICRXUSRO	14		
ICRXVERS	4		

---

**ICTX: Distributed Identity Support SAF.**

**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.

**Note:**

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 on
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.

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

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
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. Offset to authentication mechanism, or zero if omitted RESERVED
26	(1A)	UNSIGNED	2	ICTXMCH@	
28	(1C)	CHARACTER	12	*	

## Constants

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

## Cross Reference

Name	Hex 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	

**IDID: Distributed identity data**

<b>Common Name:</b>	Distributed identity data (IDID)		
<b>Macro ID:</b>	IRRPIDID		
<b>DSECT Name:</b>	IDID		
<b>Owning Component:</b>	Resource Access Control Facility (SC1BN)		
<b>Eye-Catcher ID:</b>	IDID (Offset 0, length 4)		
<b>Storage Attributes:</b>	Subpool	Same as ACEE	when pointed to by ACEE
	Key	Same as ACEE	when pointed to by ACEE
	Residency	Resides above 16M	
<b>Size:</b>	Variable		
<b>Created by:</b>	RACF or caller of RACF		
<b>Pointed to by:</b>	ACEEIDID		
<b>Serialization:</b>	None		
<b>Function:</b>	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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	32	IDID	
0	(0)	CHARACTER	4	IDIDID	Literal ID 'IDID'
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 REQUEST=VERIFY, otherwise, optional.
8	(8)	UNSIGNED	4	IDIDLEN	Total length of IDID
12	(C)	UNSIGNED	4	IDIDOFF1	Offset to section 1 (IDIDSEC1) or 0, if omitted.
16	(10)	UNSIGNED	4	IDIDOFF2	Offset to section 2 (IDIDSEC2) or 0, if omitted.
20	(14)	UNSIGNED	4	IDIDOFF3	Offset to section 3 (IDIDSEC3) or 0, if omitted.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
24	(18)	UNSIGNED	4	IDIDOFF4	Offset to section 4 (IDIDSEC4) or 0, if omitted.
28	(1C)	UNSIGNED	4	IDIDOFF5	Offset to section 5 (IDIDSEC5) or 0, if omitted.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	Binary value indicating that user name format has been determined and if determined, the format: 0=format is undetermined 1=format is straight string 2=format is x.500
4	(4)	UNSIGNED	4	IDID1OF1	Offset to user DN (IDID1USR) from start of section 1
8	(8)	UNSIGNED	4	IDID1OF2	Offset to registry name (IDID1REG) from start of section 1

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	IDID1USR	User's distinguished name
0	(0)	BINARY	2	IDID1UDL	Length of user's name up to RCVTDNL maximum
2	(2)	UTF8	*	IDID1UDN	Distributed client end-user's identity, within registry designated by IDID1REG, in UTF8 form, represented as one of the following: <ul style="list-style-type: none"> <li>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-8'92x). Note that this is identical to what is the result of a WAS <code>wscredential.getUniqueSecurityName()</code> method invocation run on a WAS server having authenticated the user by way of the LDAP registry.</li> <li>A simple character string such as a user-ID as defined within a registry.</li> </ul>

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 <code>wscredential.getRealmName()</code> method invocation run on a WAS server having authenticated the user by way of the LDAP registry.

## IDID

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

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

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

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

## Constants

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

## Cross Reference

Name	Hex 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	
IDIDSEC4	0	
IDIDSEC5	0	



## IFSP: z/OS UNIX System Services File Security Packet

**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 z/OS UNIX security functions  
**Serialization:** None  
**Function:** Contains z/OS UNIX-related information for an z/OS UNIX file

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 perm byte size bits
		1111 1...		*	Reserved
		.... .111		IFSP_OWNER	Owner perm bits
		.... .1..		IFSP_OWNER_READ	Owner read perm bit
		.... ..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

## IFSP

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

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
IFSP	0		1
IFSP_ADTR_AUDIT_EXEC_SCH	1E		3
IFSP_Access_Acl	13		3
IFSP_ADTR_AUDIT_OPTS	1C		2
IFSP_ADTR_AUDIT_READ	1C		3
IFSP_ADTR_AUDIT_WRITE	1D		3
IFSP_Dir_Model_Acl	13		3
IFSP_DIRECTORY	14	80	3
IFSP_File_Model_Acl	13		3
IFSP_FLAG	14		2
IFSP_GROUP	11	04	4
IFSP_GROUP_EXECUTE	11	01	5
IFSP_GROUP_READ	11	04	5
IFSP_GROUP_WRITE	11	02	5
IFSP_GROUP_8	11		3
IFSP_ID	0		2
IFSP_LEN	6		2
IFSP_OTHER	12	04	4
IFSP_OTHER_EXECUTE	12	01	5
IFSP_OTHER_READ	12	04	5
IFSP_OTHER_WRITE	12	02	5
IFSP_OTHER_8	12		3
IFSP_OWN_GID	C		2
IFSP_OWN_UID	8		2
IFSP_OWNER	10	04	4
IFSP_OWNER_EXECUTE	10	01	5
IFSP_OWNER_READ	10	04	5
IFSP_OWNER_WRITE	10	02	5
IFSP_OWNER_8	10		3
IFSP_PERMISSION	10		2
IFSP_S_ISGID	17	01	3
IFSP_S_ISUID	17	02	3
IFSP_S_ISVTX	17	04	3
IFSP_SECLABEL	20		2
IFSP_USER_AUDIT_EXEC_SCH	1A		3
IFSP_USER_AUDIT_OPTS	18		2
IFSP_USER_AUDIT_READ	18		3
IFSP_USER_AUDIT_WRITE	19		3
IFSP_VERS	4		2

## IISP: z/OS UNIX System Services IPC Security Packet

**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: makeIISP (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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	64	IISP	
0	(0)	CHARACTER	4	IISP_ID	LITERAL ID 'IISP'
4	(4)	UNSIGNED	1	IISP_VERS	Version Number for IISP
5	(5)	UNSIGNED	1	*	Reserved
6	(6)	UNSIGNED	2	IISP_LEN	Length of the IISP
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

### Constants

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

### Cross Reference

Name	Hex Offset	Hex Value	Level
IISP	0		1
IISP_ID	0		2
IISP_IPCID	14		2
IISP_IPCKEY	10		2
IISP_IPCP_ALET	8		2
IISP_IPCP_PTR	C		2
IISP_LEN	6		2
IISP_SECLABEL	18		2
IISP_VERS	4		2

---

**ISP: RACF In-Storage Profile**


---

 NOT Programming Interface Information
 

---

The following fields are Not Programming Interface information:

- RACRTE
- RACRSE
- RACRNE

---

 End of NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	44	RACRTE	RACLIST CLASS TREE ANCHOR ELEMENT
0	(0)	ADDRESS	4	RTENEXT	ADDRESS OF NEXT ANCHOR OR 0
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	PROFILE & 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
20	(14)	ADDRESS	4	RTEGENL	ADDRESS OF GENERIC PROFILE LIST OR 0
24	(18)	SIGNED	4	RTE SIZE	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 FOR GLOBAL=YES DATASPACE
40	(28)	UNSIGNED	4	RTEALET	PASN-AL ALET FOR DATASPACE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	RACRNE	RACLIST CLASS TREE NODE ELEMENT
0	(0)	ADDRESS	4	RNELEFT	ADDRESS OF LEFT DAUGHTER NODE OR 0
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)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)	CHARACTER	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

ISP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)
38	(26)	CHARACTER	8	RPENTFY	USERID TO NOTIFY WHEN THIS PROFILE DENIES ACCESS
46	(2E)	CHARACTER	3	RPEZTONE	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?
		..11 1111		*	RESERVED
50	(32)	CHARACTER	8	RPESCLBL	SECLABEL
58	(3A)	UNSIGNED	2	RPESESOFF	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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	RPEDID1	DIDLIST1 REPEAT GROUP, PART 1
0	(0)	UNSIGNED	1	RPEDIDLL	LABEL LENGTH
1	(1)	CHARACTER	*	RPEDIDLB	LABEL

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	RPEINST	INSTALLATION DATA VARIABLE LENGTH PORTION
0	(0)	UNSIGNED	1	RPEINSTL	INSTALLATION DATA LENGTH
1	(1)	CHARACTER	*	RPEINSTD	INSTALLATION DATA STRING

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	CHARACTER	16	RPEIPLOK	IP LOOKUP VALUE FOR SERVAUTH CLASS PROFILES

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	RPEAPPL	APPLICATION DATA VARIABLE LENGTH PORTION
0	(0)	UNSIGNED	1	RPEAPPLL	APPLICATION DATA LENGTH
1	(1)	CHARACTER	*	RPEAPPLD	APPLICATION DATA STRING

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	9	RPEACCLE (*)	ACCESS LIST
0	(0)	CHARACTER	8	RPEAUSR	USERID/GROUPNAME
8	(8)	BITSTRING	1	RPEACS	ACCESS AUTHORITY

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	2	RPEDPTD (*)	CATEGORY LIST
0	(0)	SIGNED	2	RPEDEPT	CATEGORY

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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	RPELSFG	SESSION FLAGS
9	(9)	UNSIGNED	1	RPESKYLN	LENGTH OF SESSION KEY
10	(A)	CHARACTER	*	RPESEVAR	VARIABLE LEN FIELDS
10	(A)	CHARACTER	*	RPESENKEY	SESSION KEY

Offsets			Len		
Dec	Hex	Type		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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	CHARACTER	9	RPEICSF	ICSF segment data
0	(0)	UNSIGNED	1	RPEICEXP	Symmetric key export option

## ISP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	RPEMEM	MEMBER LIST
0	(0)	UNSIGNED	1	RPEMEML	MEMBER LENGTH
1	(1)	CHARACTER	*	RPEMEMBR	MEMBER

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	1	RPESESS2	MORE SESSION
0	(0)	CHARACTER	1	RPESE2FX	MORE SESSION FIXED FIELDS
0	(0)	BITSTRING	1	RPESCONV	CONVERSATION SECURITY

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RACRNE	0		1	RPEACL2	0		1
RACRPE	0		1	RPEACS	8		2
RACRSE	0		1	RPEAC2LN	40		2
RACRTE	0		1	RPEAC2NO	3E		2
RNEBAL	C		2	RPEAC2OF	42		2
RNEKEY	10		2	RPEAPPL	0		1
RNELEFT	0		2	RPEAPPLD	1		2
RNEPROF	4		2	RPEAPPLL	0		2
RNERIGHT	8		2	RPEAPPOF	10		2
RNEUP	C		3	RPEATTR	6		2
RPEACCLE	0		1	RPEAUDIT	7		3
RPEACCNO	A		2	RPEAUSR	0		2
RPEACCOF	C		2	RPEA2ACA	10		3



Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RPEA2CLI	14		3	RPEPLEN	0		2
RPEA2CNT	11		3	RPESCLBL	32		2
RPEA2ELN	1E		3	RPESCLVL	1F		2
RPEA2ENT	1F		3	RPESCONV	0		3
RPEA2FIX	0		2	RPESFIX	0		2
RPEA2PGM	0		3	RPESSESLN	3C		2
RPEA2RST	1		4	RPESSESO	3A		2
RPEA2RSV	1C		3	RPESSESN	0		1
RPEA2USR	8		3	RPESSESS2	0		1
RPEA2VAR	14		2	RPESSEVAR	A		2
RPEA2VRL	13		3	RPSE2FX	0		2
RPECLABS	0		3	RPSE2LN	4A		2
RPECSFLN	50		3	RPSE2OF	4C		2
RPECSFOF	52		3	RPESKYL	9		3
RPEDEPT	0		2	RPESLSFG	8		3
RPEDID1	0		1	RPESNKEY	A		3
RPEDID2	0		1	RPETZONE	2E		2
RPEDIDCT	54		3	RPEUACC	6		3
RPEDIDL	1		2	RPEUCNT	4		2
RPEDIDLL	0		2	RSENEXT	0		2
RPEDIDLN	56		3	RSEPOOL	6		2
RPEDIDOF	58		3	RSESIZE	4		2
RPEDIDRL	8		2	RSESTORE	8		2
RPEDIDRG	9		2	RTEALET	28		2
RPEDIDUS	0		2	RTEASPN	12		2
RPEDPTD	0		1	RTECLASS	4		2
RPEDPTOF	1C		2	RTEGENL	14		2
RPEEND	4E		2	RTEGNUM	1C		2
RPEFLAGS	31		2	RTENEXT	0		2
RPEFSCPW	31	40	3	RTENSPN	11		3
RPEFWARN	31	80	3	RTEPSPN	10		3
RPEGAUD	8		3	RTESIZE	18		2
RPEICAUS	1		3	RTESPNS	10		2
RPEICEXP	0		3	RTESTOKN	20		2
RPEICLBS	1		3	RTESTORE	C		2
RPEICLCT	0		3	RTETREE	8		2
RPEICOFF	5		3				
RPEIKLBS	9		3				
RPEIKLCT	7		3				
RPEILLN	0		3				
RPEILABE	0		3				
RPEILABS	0		3				
RPEICSF	0		2				
RPEINSOF	E		2				
RPEINST	0		1				
RPEINSTD	1		2				
RPEINSTL	0		2				
RPEKYDAT	0		3				
RPEKYINT	4		3				
RPELDAYS	1E		2				
RPELEVEL	9		3				
RPELLEN	2		2				
RPELOGFT	23		2				
RPELOGNT	20		2				
RPEMEM	0		1				
RPEMEMBR	1		2				
RPEMEMCT	44		2				
RPEMEML	0		2				
RPEMEMLN	46		2				
RPEMEMOF	48		2				
RPEMFAIL	6		3				
RPENTFY	26		2				
RPENUMDP	1A		2				
RPEOWNER	12		2				
RPEPGFLG	0		4				

## OUSP: initUSP Output Parameter List

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	OUSPFIXED + 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 TSO userid.
14	(E)	CHARACTER	8	OUSP_TSOU	TSO userid.
22	(16)	SIGNED	2	OUSP_HDPNLEN	Length, home directory path name.
24	(18)	SIGNED	2	OUSP_IPPNLEN	Length, initial program path name.
26	(1A)	CHARACTER	*	OUSP_HDPN	Home directory path name, length up to 1023.
*	*	CHARACTER	*	OUSP_IPPN	Initial program path name, length up to 1023.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	OUSP3	Limits area.
0	(0)	SIGNED	2	OUSP_NUMLIMITS	Number of limits in the following array.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
4	(4)	CHARACTER	1	OUSP_UNITS	Units M, G, T, P (megabytes, gigabytes, terabytes, or petabytes).

### Constants

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	2	OUSP_ASSIZEMAX	Key for ASSIZEMAX.
1	DECIMAL	3	OUSP_FILEPROCMAX	Key for FILEPROCMAX.
1	DECIMAL	4	OUSP_PROCSERMAX	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.

### Cross Reference

Name	Hex Offset	Hex Value	Level
OUSP	0		1
OUSP2	0		1
OUSP3	0		1
OUSP_FLAGS	0		2
OUSP_GID	8		2
OUSP_HDPN	1A		2
OUSP_HDPNLEN	16		2
OUSP_IPPN	*		2
OUSP_IPPNLEN	18		2
OUSP_LEN	2		2
OUSP_LIMITKEY	0		2
OUSP_LIMIT_ENTRY	0		1
OUSP_LIMITSDONOTFIT	0	40	3
OUSP_LIMITSSET	0	80	3
OUSP_LIMITVALUE	1		2
OUSP_NUMLIMITS	0		2
OUSP_TSOU	E		2
OUSP_TSOULEN	C		2
OUSP_UID	4		2
OUSP_UNITS	4		2
OUSP_VRSN	0		2
OUSP_WASP	1		2

**PWX2: New Password Phrase Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		PWX2	
0	(0)	ADDRESS	4	PWX2LEN	Length address: Points to a fullword containing the number of fullwords in this parameter list.
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 ADDUSER, PASSWORD and ALTUSER commands. If the TSO command processor parameter list is absent, the address is zero.
12	(C)	ADDRESS	4	PWX2NEW	New Password Phrase address: Points to an area of the following format: Offset 0, length 1: Length of new password phrase. Offset 1, length 100: new password 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
20	(14)	ADDRESS	4	PWX2WA	Exit work area address: Points to a fullword whose contents are either: <ul style="list-style-type: none"> <li>• Zero, for ADDUSER, ALTUSER and PASSWORD commands</li> <li>• The contents of the user work area address that RACINIT processing passes to ICHRIX01 and ICHRiX02.</li> </ul>
24	(18)	ADDRESS	4	PWX2CURP	Current password phrase address: Points to an area of the following format: Offset 0, length 1: Length of current password phrase. Offset 1, variable length: Current password phrase.
28	(1C)	ADDRESS	4	PWX2LCDA	Password Phrase Last Change Date Address: Points to a 4-byte area that contains the date of the last password phrase change. The format of this area is: yyyyddd yyyy is the year ddd is the day 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.

### Cross Reference

Name	Hex 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	

---

**PWXP: Password Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
		.... ..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 specific 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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: <i>yydds</i> where: <i>yy</i> is the year, <i>ddd</i> is the day, and <i>s</i> 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	PWXPHWST	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.
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.) This parameter is available only if the caller is RACINIT. In all cases, if a parameter is not present, its address is zero.
		.... ..1.		PWXPTKT	X'02' Passticket passed as current password to the user exit
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.

## PWXP

### Cross Reference

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



**RCVI: Identity cache communication vendor.**

**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:**

1. 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.

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

The RCVI consists of 2 blocks: RCVISID1 and RCVISID2

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

## RCVI

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)	CHARACTER	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)	CHARACTER	32	*	RESERVED

## Constants

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

## Cross Reference

Name	Hex 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	
RCVIBDDN	92C	

## RCVT: RACF Communication Vector Table

NOT Programming Interface Information

The following fields are the only intended Programming Interfaces in RCVT:

RCVT	RCVTGLBL	RCVTML2F	RCVTSTA1
RCVTAPTR	RCVTID	RCVTPINV	RCVTTAPE
RCVTCDTL	RCVTINAC	RCVTPNL0	RCVTTDSN
RCVTDATP	RCVTISTL	RCVTPTGN	RCVTVERS
RCVTDNL	RCVTJALL	RCVTRELS	RCVTVRMC
RCVTENVP	RCVTJCHK	RCVTREXP	RCVTVRMF
RCVTFLGS	RCVTJSYS	RCVTRL	RCVTVRN
RCVTFLG1	RCVTJUND	RCVTRNA	RCVTVRMN
RCVTFLG3	RCVTJXAL	RCVTROFF	RCVTWARN
RCVTFRCP	RCVTMFLG	RCVTRVOK	RCVTWUID
RCVTGENT	RCVTMFL1	RCVTSTAT	

### 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.

**Note:** The 118th bit of the RCVTVCPTR 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.

**Note:** 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.

<b>Common Name:</b>	RACF communication vector table
<b>Macro ID:</b>	ICHPRCVT
<b>DSECT Name:</b>	RCVT
<b>Owning Component:</b>	Resource Access Control Facility (SC1BN)
<b>Eye-Catcher ID:</b>	RCVT (Offset: 0, Length: 4)
<b>Storage Attributes:</b>	Subpool    SQA Key        0
<b>Size:</b>	2308 bytes
<b>Created by:</b>	RACF initialization or equivalent
<b>Pointed to by:</b>	CVTRAC
<b>Serialization:</b>	None
<b>Function:</b>	Communication area for information global to RACF functions (or equivalent product functions)

## RCVT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	2308	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 IN RACF DATABASE
56	(38)	CHARACTER	44	RCVTDSN	DSN OF RACF DATA SET
100	(64)	CHARACTER	44	RCVTUADS	DSN OF UADS DATA SET OR ZERO
144	(90)	CHARACTER	6	RCVTUVOL	VOLID OF UADS DATA SET OR ZERO
150	(96)	BITSTRING	1	RCVTSTA1	
		1... ..		RCVTTAPE	TAPE VOLUME PROTECTION IN EFFECT
		.1.. ..		RCVTDASD	DASD VOLUME PROTECTION IN EFFECT
		..1. ....		RCVTDGEN	GENERIC PROFILE CHECKING IN EFFECT FOR DATASET CLASS
		...1 ....		RCVTDGCM	GENERIC COMMAND PROCESSING IN EFFECT FOR DATASET CLASS
		.... 1...		RCVTRDSN	INPUT DATA SET NAME IS USED FOR LOGGING AND MESSAGES
		.... .1..		RCVTJXAL	JES-XBMALLRACF IN EFFECT
		.... ..1.		RCVTJCHK	JES-EARLYVERIFY IN EFFECT
		.... ...1		RCVTJALL	JES-BATCHALLRACF IN EFFECT
151	(97)	BITSTRING	1	RCVTAUOP	RACF 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 DEACTIVATED BY RVARV COMMAND
		.1.. ..		RCVTRDHD	RACF WAS REACTIVATED BY RVARV AND REFRESH OF THE RESIDENT ICB IS REQUIRED
		..1. ....		RCVTSHR	AT SOME TIME DURING THIS IPL, THE RACF DATA SET 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
154	(9A)	BITSTRING	1	RCVTRMSG	RACF MESSAGE ICH412I WAS ISSUED
				RCVTRWUID	RACF WORK UNIT IDENTITY SUPPORT EXISTS.
				RCVTRGLBL	GLOBAL=YES SUPPORT EXISTS
				RCVTEROP	RACF TERMINAL OPTIONS
				RCVTTERP	TERMINAL AUTHORIZATION CHECKING
155	(9B)	ADDRESS	1	RCVTTUAC	DEFAULT UACC FOR TERMINALS NOT DEFINED TO RACF. IF ON - UACC = NONE, IF OFF - UACC = READ
				RCVTAVIO	DO NOT CREATE LOG RECORD FOR COMMAND VIOLATIONS ONLY
				RCV TSAUD	DO NOT AUDIT SPECIAL USER
				*	RESERVED
				RCVTPINV	GLOBAL MAX PASSWORD INTERVAL VALUE - VALID RANGE 1-254
156	(9C)	ADDRESS	4	RCVTRAU0	PTR TO AUDITING MODULE
160	(A0)	ADDRESS	4	RCVTRIXP	PTR RACROUTE REQUEST=VERIFY(X) POST PROCESSING EXIT
164	(A4)	ADDRESS	4	RCVTRCXP	PTR RACROUTE REQUEST=AUTH POST PROCESSING EXIT
168	(A8)	ADDRESS	4	RCVTRID0	PTR TO MISCELLANEOUS VERIFY ROUTINE
172	(AC)	ADDRESS	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
					(SEE FIELD RCVTVRMN FOR ACTUAL VERSION AND RELEASE NUMBER)
173	(AD)	CHARACTER	3	RCVTVRN	VERSION NUMBER IN HIGH NIBBLE
				RCVTRRELS	RELEASE NUMBER IN LOW NIBBLE
176	(B0)	ADDRESS	4	RCV TAPTR	RESERVED 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 DESCRIPTOR TABLE
192	(C0)	ADDRESS	4	RCVTRXP	PTR TO RACSTAT MODULE
196	(C4)	ADDRESS	4	RCVTRFRCP	PTR TO FRACHECK MODULE
200	(C8)	ADDRESS	4	RCVTRFRXP	PTR RACROUTE REQUEST=FASTAUTH EXIT
204	(CC)	ADDRESS	4	RCVTRLX	PTR RACROUTE REQUEST=LIST PREPROCESSING EXIT
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	RCVTENV	PTR TO ENVIRONMENT SERVICE (IRRENS00)
220	(DC)	ADDRESS	4	RCVTEIMR	PTR TO EIM REGISTRY NAME
224	(E0)	ADDRESS	4	RCVTDSDT	PTR TO DATA SET DESCRIPTOR TABLE
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	RCVTPWD	PTR TO RACF PASSWORD EXIT.
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	PASSWORD WARNING VALUE.

# RCVT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)	UNSIGNED	1	RCVTELEN	ENDING LENGTH VALUE.
246	(F6)	CHARACTER	8	RCVTRULS	CONTENT RULES.
246	(F6)	CHARACTER	1	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)	CHARACTER	4	RCVTMDEL	MODEL OPTIONS.
324	(144)	BITSTRING	1	*	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-GROUPS CHECKING IS ACTIVE.
		.111 1111		*	RESERVED
330	(14A)	UNSIGNED	1	RCVTALIS	Application Identity Mapping (AIM) Stage Enablement: 0=stage 0, not active, only alias mapping profiles. 1=stage 1, active, maintain alias index, but also maintain and use VLF and mapping profiles. Reject alias locates. 2=stage 2, active, maintain and use alias index, but also maintain VLF and the mapping profiles. Allow alias locates. 3=stage 3, 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 USER'S 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 - ICHRD02
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 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
		...1 ....		RCVTEOSL	ERASE-ON-SCRATCH BY SECLEVEL IN EFFECT (THIS FLAG IS IGNORED IF RCVTEOS HAS A VALUE OF'0'B)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		.... 1...		RCVTEOSA	ERASE-ON-SCRATCH FOR ALL DATASETS 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	RCVTSVLV	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 4 BYTE DATES ON PROGRAMMING INTERFACES IS AVAILABLE
		.... ...1		RCVTTLPS	TASK LEVEL ACEE SUPPORT AVAILABLE FOR CALLABLE SERVICE
411	(19B)	BITSTRING	8	RCVTMFL1	MORE MISCELLANEOUS FLAGS
		1... ....		RCVTXUSP	EXTENDED INITUSP SUPPORT AVAILABLE
		.1.. ....		RCVTXFAR	FASTAUTH ENVRIN/SETROPTS RACLST SUPPORT AVAILABLE
		..1. ....		RCVTENVS	ENVIRONMENT SERVICE (IRRENS00) AVAILABLE
		...1 ....		RCVTX500	X500NAME SUPPORT AVAILABLE
		.... 1...		RCVTPSEN	PGMSECURITY OPTION
		.... .1..		RCVTPSWR	0 - BASIC 1 - ENHANCED PGMSECURITY ENHANCED
		.... ..1.		RCVTLPAP	0 - FAILURE 1 - WARNING RACF REQ=F AE ATTCH1ST required for LPA
		.... ...1		RCVTFGSG	0 - NO 1 - YES 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 ENCRYPTION 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 RACROUTE REQUEST=FASTAUTH POST PROCESSING INSTALLATION EXIT (ICHRFX02)
432	(1B0)	CHARACTER	8	RCVTPROB	ADDRESSES OF CONTROLLED PROGRAMS LIST ANCHOR BLOCKS
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 R VARY SWITCH
448	(1C0)	CHARACTER	8	RCVTINPW	PASSWORD FOR R VARY STATUS
456	(1C8)	ADDRESS	4	RCVTLARP	PTR TO LINKAGE ASSIST ROUTINE FOR INSTALL EXITS (ICHLAR00)

## RCVT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
460	(1CC)	ADDRESS	4	RCVCTV0	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	PTR TO PASS TICKET EVAL ROUTINE
476	(1DC)	ADDRESS	4	RCVTPHRX	ADDRESS OF NEW PASSWORD PHRASE EXIT ICHPWX11
480	(1E0)	ADDRESS	4	RCVTGENT	Address of generic name translate routine
484	(1E4)	ADDRESS	4	RCVTGANC	Address of GENERICANCHOR keyword settings
488	(1E8)	ADDRESS	108	*	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	RCVTLAR0	ADDRESS OF ICHLAR02
608	(260)	ADDRESS	4	RCVTRFLT0	ADDRESS OF IRRFLT00
612	(264)	ADDRESS	4	RCVTRFLT1	ADDRESS OF IRRFLT01
616	(268)	CHARACTER	4	RCVTVRMN	LAST 4 CHARACTERS OF THE RACF FMID
620	(26C)	SIGNED	4	RCVTVMSP	ICB SYNC COUNT VM 370
624	(270)	SIGNED	4	RCVTVMXA	ICB SYNC COUNT VM XA
628	(274)	BITSTRING	1	RCVTFLG2	RACF 1.9.0 SETROPTS OPTIONS
		1... ..		RCVTSLCL	SETROPTS SECLABELCONTROL IS ACTIVE
		.1.. ..		RCVTCATD	SETROPTS CATDSNS IS ACTIVE
		..1. ....		RCVTMLQT	SETROPTS MLQUIET IS ACTIVE
		...1 ....		RCVTMLST	SETROPTS MLSTABLE IS ACTIVE
		.... 1...		RCVTMLS	SETROPTS MLS IS ACTIVE
		.... .1..		RCVTMLAC	SETROPTS MACTIVE IS ACTIVE
		.... ..1.		RCVTGNOW	SETROPTS GENERICOWNER IS ACTIVE
		.... ...1		RCVTAUSL	SETROPTS SECLABELAUDIT IS ACTIVE
629	(275)	BITSTRING	1	RCVTLOGD	LOGOPTIONS FOR DATASET IS ACTIVE
		1... ..		RCVTDLGA	LOGOPTIONS "ALWAYS" FOR THE DATASET CLASS IS ACTIVE
		.1.. ..		RCVTDLGN	LOGOPTIONS "NEVER" FOR THE DATASET CLASS IS ACTIVE
		..1. ....		RCVTDLGS	LOGOPTIONS "SUCCESSSES" FOR THE DATASET CLASS IS ACTIVE
		...1 ....		RCVTDLGF	LOGOPTIONS "FAILURES" FOR THE DATASET CLASS IS ACTIVE
		.... 1111		*	RESERVED
630	(276)	SIGNED	2	RCVTSINT	LU SESSION INTERVAL
632	(278)	BITSTRING	1	RCVTML2F	SETR FLAGS
		1... ..		RCVTMLFS	1 = SECLABELS REQUIRED FOR FILE SYSTEM OBJECTS
		.1.. ..		RCVTMLIP	1 = SECLABELS REQUIRED FOR IPC OBJECTS
		..1. ....		RCVTMLNM	1 = DISPLAY OF NAMES RESTRICTED BY SECLABEL
		...1 ....		RCVTSBYS	1 = SECLABEL BY SYSTEM REQUESTED
		.... 1111		*	RESERVED
633	(279)	BITSTRING	1	RCVTFLG3	MISCELLANEOUS FLAGS
		1... ..		RCVTDCDT	1-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
		.... 1111		*	RESERVED
634	(27A)	Signed	1	RCVTPMIN	Minimum days between password changes
635	(27B)	CHARACTER	61	*	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



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
728	(2D8)	ADDRESS	4	RCVTFRX4	ADDRESS OF THE 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
		.... ..11		*	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 mask
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	RCVTVGST	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

## Constants

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	7780	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
4	CHARACTER	2060	RCVTVR26	Indicates RACF 2.6.0
4	CHARACTER	2608	RCVTVR28	Indicates HRF2608
4	CHARACTER	7703	RCVTVR73	Indicates HRF7703

## RCVT

Len	Type	Value	Name	Description
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	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	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

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RCVT	0		1	RCVTDSN	38		2
RCVTAAPL	2FC	08	3	RCVTDSNL	34		2
RCVTADAS	97	08	3	RCVTDSPC	2F8		2
RCVTADAT	97	10	3	RCVTDX11	2DC		2
RCVTAGRO	97	40	3	RCVTDYNL	410	10	3
RCVTALIS	14A		2	RCVTD4OK	19A	08	3
RCVTAOPR	97	01	3	RCVTEGN	35	01	3
RCVTAPTR	B0		2	RCVTELEN	F5		3
RCVTATAP	97	04	3	RCVTENVS	19B	20	3
RCVTATER	97	02	3	RCVTEIMR	DC		2
RCVTAUOP	97		2	RCVTENVP	D8		2
RCVTAUSE	97	20	3	RCVTEOS	189	20	3
RCVTAUSL	274	01	3	RCVTEOSA	189	08	3
RCVTAUTP	E8		2	RCVTEOSL	189	10	3
RCVTAUTU	279	10	3	RCVTEROP	9A		2
RCVTAVIO	9A	20	3	RCVTEXTA	AD		2
RCVTAXTA	98		2	RCVTFGSG	19B		3
RCVTBAM	2C		2	RCVTFLGS	99		2
RCVTCATD	274	40	3	RCVTFLG1	188		2
RCVTCATF	2FC	10	3	RCVTFLG2	274		2
RCVTCCTL	900		2	RCVTFLG3	279		2
RCVTCCTP	BC		2	RCVTFLT0	260		2
RCVTCFLD	279	20	3	RCVTFLT1	264		2
RCVTCGSN	170		2	RCVTFL2X	2FC		2
RCVTCISP	1B0		3	RCVTFPB	180		2
RCVTCMPM	2FC	80	3	RCVTFPDS	188	80	3
RCVTCV0	1CC		2	RCVTFRCP	C4		2
RCVTDASD	96	40	3	RCVTFRXP	C8		2
RCVTDATP	D4		2	RCVTFRX2	1AC		2
RCVTDCEB	4		2	RCVTFRX3	14C		2
RCVTDNL	174		2	RCVTFRX4	2D8		2
RCVTDCTD	279	80	3	RCVTGANC	1E4		2
RCVTDEB	8		2	RCVTGENT	1E0		2
RCVTDESX	1A0		2	RCVTGLBL	99	01	3
RCVTDGCM	96	10	3	RCVTGLS1	254		2
RCVTDGEN	96	20	3	RCVTGLS6	2E4		2
RCVTDLGA	275	80	3	RCVTGNOW	274	02	3
RCVTDLGF	275	10	3	RCVTHDR	14		2
RCVTDLGN	275	40	3	RCVTHIST	F0		2
RCVTDLGS	275	20	3	RCVTID	0		2
RCVTDPTB	2E8		2	RCVTIDPV	178		1
RCVTDSDT	E0		2	RCVTINAC	F3		2

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RCVTINDX	C		2	RCVTRCX	1C		2
RCVTINPW	1C0		2	RCVTRCXP	A4		2
RCVTISTL	30		2	RCVTRDHD	99	40	3
RCVTJALL	96	01	3	RCVTRDSN	96	08	3
RCVTJCHK	96	02	3	RCVTRDX	20		2
RCVTJSYS	2B8		2	RCVTRDXP	17C		2
RCVTJUND	2C0		2	RCVTRELS	AC	0F	3
RCVTJWTO	2FD	80	3	RCVTREXP	C0		2
RCVTJXAL	96	04	3	RCVTRID0	A8		2
RCVTLARP	1C8		2	RCVTRIX	18		2
RCVTLAR2	25C		2	RCVTRIXP	A0		2
RCVTLGRP	149	80	3	RCVTRL	176		2
RCVTLOGD	275		2	RCVTRLX	CC		2
RCVTLPAP	19B		3	RCVTRLXP	D0		2
RCVTMDEL	144		2	RCVTRMSG	99	04	3
RCVTMFLG	19A		2	RCVTRNA	35	80	3
RCVTMFL1	19B		2	RCVTRNGP	E4		2
RCVTMGDG	144	80	4	RCVTROFF	99	80	3
RCVTMGRP	144	20	4	RCVTRTPD	18C		2
RCVTMLAC	274	04	3	RCVTRUCB	24		2
RCVTMLAF	2FC	20	3	RCVTRULS	F6		3
RCVTMLFS	278	80	3	RCVTRUL1	F6		4
RCVTMLIP	278	40	3	RCVTRUL2	F7		4
RCVTMLNM	278	20	3	RCVTRUL3	F8		4
RCVTML2F	278		2	RCVTRUL4	F9		4
RCVTMLQT	274	20	3	RCVTRUL5	FA		4
RCVTMLS	274	08	3	RCVTRUL6	FB		4
RCVTMLSF	2FC	40	3	RCVTRUL7	FC		4
RCVTMLST	274	10	3	RCVTRUL8	FD		4
RCVTMUSR	144	40	4	RCVTRVOK	F1		2
RCVTNADC	2FC	04	3	RCVTRX10	2F0		2
RCVTNADS	35	02	3	RCVTRX11	2F4		2
RCVTNCDX	B8		2	RCV TSAUD	9A	10	3
RCVTNCX	B4		2	RCVTSBYS	278	10	3
RCVTNDSS	35	20	3	RCVTSHR	99	20	3
RCVTNDUP	99	10	3	RCVTSINT	276		2
RCVTNDVS	35	08	3	RCVTSKGN	1D4		2
RCVTNJEF	2FD		2	RCV TSLAU	199		2
RCVTNLS	35	40	3	RCV TSLCL	274	80	3
RCVTNREC	36		2	RCV TSLFN	F4		3
RCVTNRTN	1A8		2	RCV TSLVL	18E		2
RCVTNTAB	1A4		2	RCV TSNTX	F4		2
RCVTNTMS	35	04	3	RCV TSPT	19C		2
RCVTNTVS	35	10	3	RCV TSTAT	35		2
RCVTOISP	1B4		3	RCV TSTA1	96		2
RCVTOPTX	149		2	RCV TSVC0	2D0		2
RCVTOPTX	149		2	RCV TSWPW	1B8		2
RCVTPHRX	1DC		2	RCV TTAPE	96	80	3
RCVTPLC	279	40	3	RCV TTDSN	188	40	3
RCVTPNL0	1D0		2	RCV TTEMP	10		2
RCVTPRO	189	80	3	RCV TTERP	9A	80	3
RCVTPROB	1B0		2	RCV TTLPS	19A	01	3
RCVTPROF	189	40	3	RCV TTMP2	2C8		2
RCVTPROG	18A	80	3	RCV TTUAC	9A	40	3
RCVTPTGN	2D4		2	RCV TUADS	64		2
RCVTPSEN	19B		3	RCV TUVOL	90		2
RCVTPSWR	19B		3	RCV TVCAU	5FE		2
RCVTPWDX	EC		2	RCV TVCGC	3FE		2
RCVTQLLN	18F		2	RCV TVCGE	37E		2
RCVTQUAL	190		2	RCV TVCPR	2FE		2
RCVTRAU0	9C		2	RCV TVCST	67E		2
RCVTRCK2	2EC		2	RCV TVERS	AC		2
RCVTRCK4	2CC		2	RCV TVFPT	57E		2
RCVTRCVI	184		2	RCV TVGNL	4FE		2
RCVTRCVX	258		2	RCV TVLGA	6FE		2

## RCVT

Name	Hex Offset	Hex Value	Level
RCVTVLGF	87E		2
RCVTVLGS	7FE		2
RCVTVLNV	77E		2
RCVTVMSP	26C		2
RCVTVMXA	270		2
RCVTVRCL	47E		2
RCVTVRMF	19A	80	3
RCVTVRMN	268		2
RCVTVRN	AC	F0	3
RCVTVSL	150		2
RCVTWARN	F2		2
RCVTWCNT	148		2
RCVTWUID	99	02	3
RCVTXFAR	19B	40	3
RCVTXLEN	28		2
RCVTXLT0	2E0		2
RCVTXMFR	19A	20	3
RCVTXRCO	19A	04	3
RCVTXUSP	19B	80	3
RCVTX500	19B	10	3
RCVT24MD	99	08	3
RCVT310U	19A	40	3
RCVT4INF	19A	02	3

End of NOT Programming Interface Information

**RCXP: RACROUTE REQUEST=AUTH Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
		11.. ....		RCXFIYES	B'11000000' 11.. .... RACFIND=YES was specified.
		..1. ....		RCXFLGNM	B'00100000' ..1. .... ENTITYX was specified.
		...1 ....		RCXDTPV	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 parameter 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... ....		RCXDTPYPT	B'10000000' 1... .... DSTYPE=T
		..1. ....		RCXDTPYPM	B'01000000' ..1. .... DSTYPE=M was specified.

# RCXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		.... ....		RCXENDSN	B'00000000' ..0. .... ENTITY=dsname or ENTITYX=dsname; tape volser or DASD volser address was specified.
		..1. ....		RCXPRPRA	B'00100000' ..1. .... PROFILE=profile address 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. .... ..0 Reserved.
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 RACROUTE REQUEST=AUTH preprocessing and postprocessing exit routines. Do not confuse this value with the DATA address (pointed to by RCXDATA, offset 32 in the parameter list) that comes from a field in the RACF profile for the resource being checked.
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-aligned 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 ICHCXP 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 might 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.
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 zeros on the initial entry to the preprocessing routine. An installation can use this field for any purpose. Because this field is set to zeros before entry to the preprocessing exit, the preprocessing and postprocessing exits can use this work area to communicate with each other.

Offsets		Type	Len	Name (Dim)	Description												
Dec	Hex																
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 zeros 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 is 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 does 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.												
44	(2C)	ADDRESS	4	RCXRCODE	Return code address: points to a 4-byte field containing either: <ol style="list-style-type: none"> <li>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 <i>z/OS Security Server RACROUTE Macro Reference</i>.</li> <li>the reason code used to cause the ABEND to be issued. For meanings of these ABEND reason codes, see <i>z/OS Security Server RACF Messages and Codes</i>.</li> </ol> <p>Changes to the field pointed to by RCXRCODE do not affect the values pointed to by RCXFLAG2 (attempted access) and RCXACC (allowed access).</p> <p><b>Note:</b> Do not confuse this code with the return code from the RACROUTE REQUEST=AUTH preprocessing and postprocessing exits described in <i>z/OS Security Server RACF System Programmer's Guide</i>.</p>												
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: <table border="0"> <tr> <td>1... ....</td> <td>RCXALTER</td> </tr> <tr> <td>.1.. ....</td> <td>RCXCONTR</td> </tr> <tr> <td>..1. ....</td> <td>RCXUPDAT</td> </tr> <tr> <td>...1 ....</td> <td>RCXREAD</td> </tr> <tr> <td>.... 1..1</td> <td>RCXEXEC</td> </tr> <tr> <td>.... ...1</td> <td>RCXNONE</td> </tr> </table> <p>X'80' - ALTER            X'40' - CONTROL            X'20' - UPDATE            X'10' - READ            X'09' - EXECUTE (both X'08' and X'01' are set)            X'01' - NONE. The area is zero if (1) the profile could not be retrieved, or (2) the preprocessing routine indicated bypassing RACHECK.</p>	1... ....	RCXALTER	.1.. ....	RCXCONTR	..1. ....	RCXUPDAT	...1 ....	RCXREAD	.... 1..1	RCXEXEC	.... ...1	RCXNONE
1... ....	RCXALTER																
.1.. ....	RCXCONTR																
..1. ....	RCXUPDAT																
...1 ....	RCXREAD																
.... 1..1	RCXEXEC																
.... ...1	RCXNONE																
52	(34)	ADDRESS	4	RCXRLLV	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 defined volume of a multivolume data set or tape volume set. This is blank if OLDVOL was not specified.												

## RCXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 IRRUT100 utility allows an installation to modify or eliminate the RACF DASD data set naming convention. Corresponding processing might be required in the RACROUTE REQUEST=AUTH 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 8-byte field containing the application name (if supplied on the RACHECK macro instruction). The name is left-aligned 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.
68	(44)	ADDRESS	4	RCXACEE	ACEE address: points to a fullword containing the address of the ACEE parameter of the RACHECK macro instruction, which is used during RACHECK processing. If the ACEE parameter was not specified on the RACHECK macro instruction, the fullword pointed to by TCBSENV in the current task control block (TCB) or ASXBSENV in the address space extension block (ASXB) is used during RACHECK processing. The ACEE being used for RACHECK processing is also used for authority checking unless user ID was specified (RCXUSRID is nonzero) or UTOKEN was specified (RCXUTOKN is nonzero). In this case, an ACEE is created for the user ID specified or found in the UTOKEN, as part of RACHECK processing.
72	(48)	ADDRESS	4	RCXOWNER	OWNER address: points to an 8-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 might 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 does 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.
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 2-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:



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		1... .... .1.. .... .... ....		RCXTFBLP RCXTFNL RCXTFSTD	B'10000000' 10.. .... TAPBL=BLP was specified. B'01000000' 01.. .... TAPBL=NL was specified. B'00000000' 00.. .... TAPBL=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: B'10000000' 1... .... STATUS=ERASE was specified B'01000000' .1.. .... STATUS=EVERDOM was specified B'00100000' ..1. .... STATUS=WRITEONLY was specified 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 z/OS Security Server RACROUTE Macro Reference for the meanings of the RACHECK reason codes. Do not confuse this reason code with the ABEND reason code.
104	(68)	ADDRESS	4	RCXNOTIF	Address of NOTIFY user ID: an 8-byte area containing the user ID 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 user ID of the user whose access authority is to be checked.
112	(70)	ADDRESS	4	RCXGRPID	Address of GROUPNAME for third party RACHECK: an 8-byte area containing the GROUPNAME 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 data set name specified as the ENTITY/ENTITYX name, or 8 blanks if DDNAME was not specified.
120	(78)	ADDRESS	4	RCXFLAG5	Flag byte 5 address: points to a 1-byte area of the following format: ACEE is a DCE unauthenticated client. ACEE is a DCE server. 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 1-byte length followed by the character data that the issuer of RACHECK wants to be appear in the SMF record.
136	(88)	ADDRESS	4	RCXRECVR	Address of RECVR for TSO RECEIVE command: an 8-byte area containing the user ID that matches the ID in ACEE in order to pass the authority check.
140	(8C)	ADDRESS	4	RCXCOTP	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 post-processing exit only.

# RCXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 post-processing exit only. The profile name type can be one of the following values: 1 = no profile used, 2 = discrete profile name in external format, 4 = generic profile name in external format, 5 = no profile name because global access profile was used, 6 = no profile was found.
148	(94)	ADDRESS	4	RCXAUSED	Address of authorization 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.
		.... .000		*	<b>Note:</b> When *BYPASS* is used, the final RACF return code is 4. This might allow access to a resource, depending on the application.
		0000 0000		*	Reserved

Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RCXACC	30		2	RCXSEQNO	58		2
RCXACCLP	54		2	RCXSSRVR	78	40	3
RCXACCLV	50		2	RCXSTACS	60	10	2
RCXACEE	44		2	RCXSTERA	60	80	2
RCXALTER	30	80	2	RCXSTEVD	60	40	2
RCXABYPS	94	10	2	RCXSTWRN	C	10	2
RCXAEXIT	94	20	2	RCXSTWRO	60	20	2
RCXANORM	94	80	2	RCXTFBLP	5C	80	2
RCXAOPER	94	40	2	RCXTFLAG	5C		2
RCXAPPLN	40		2	RCXTFNL	5C	40	2
RCXAPROF	90		2	RCXTFSTD	5C	0	2
RCXATTAL	8	80	2	RCXUPDAT	30	20	2
RCXATTCO	8	8	2	RCXUSRID	6C		2
RCXATTRE	8	2	2	RCXUTOKN	7C		2
RCXATTUP	8	4	2	RCXVSER	1C		2
RCXAUSED	94		2	RCXWA	24		2
RCXCDDTP	8C		2	RCX31BA	4	8	2
RCXCLASS	18		2				
RCXCNXPL	3C		2				
RCXCOMP	28		2				
RCXCONTR	30	40	2				
RCXDATA	20		2				
RCXDDNAM	74		2				
RCXDTPM	C	40	2				
RCXDTPPT	C	80	2				
RCXDTPPV	4	10	2				
RCXENDSN	C	0	2				
RCXENORP	14		2				
RCXENTTY	4	1	2				
RCXEXEC	30	9	2				
RCXFINO	4	80	2				
RCXFINOT	4	0	2				
RCXFIYES	4	C0	2				
RCXFLAG	4		2				
RCXFLAG2	8		2				
RCXFLAG3	C		2				
RCXFLAG4	60		2				
RCXFLAG5	120		2				
RCXFLGNM	4	20	2				
RCXGENYS	C	4	2				
RCXGRPID	70		2				
RCXINSTL	10		2				
RCXLCNTL	4C		2				
RCXLEN	0		2				
RCXLGNOA	4	0	2				
RCXLGNOF	4	4	2				
RCXLGNON	4	2	2				
RCXLGNOS	4	6	2				
RCXLOGST	84		2				
RCXNONE	30	1	2				
RCXNOTIF	68		2				
RCXOLVOL	38		2				
RCXOWNER	48		2				
RCXPAPRO	C	2	2				
RCXPRPRA	C	20	2				
RCXRCODE	2C		2				
RCXREAD	30	10	2				
RCXREASN	64		2				
RCXRECVR	88		2				
RCXRLLV	34		2				
RCXRTOKN	80		2				
RCXSCLNT	78	80	3				

## RDDFL: RACROUTE REQUEST=DEFINE Parameter List (Request Section)

**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 SAFR plus the offset in SAFPRACP

**Serialization:** None

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	RDDFLIST	RACDEF PARAMETER LIST
0	(0)	ADDRESS	4	RDDFINSW	ADDRESS OF INSTALLATION DATA
0	(0)	UNSIGNED	1	RDDFLENG	LENGTH OF PARAMETER LIST
1	(1)	ADDRESS	3	RDDFINST	INSTALLATION DATA
4	(4)	SIGNED	4	RDDFENTW	ENTITY NAME ADDRESS WORD
4	(4)	UNSIGNED	1	RDDFFLGS	FLAGS BYTE
		11.. ....		RDDFCHGV	TYPE=CHGVOL.
		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	ENTITY NAME ADDRESS
8	(8)	ADDRESS	4	RDDFOVOL	OLD VOLSER ADDRESS
8	(8)	ADDRESS	4	RDDFNMX	NEWNAMX ADDRESS
8	(8)	ADDRESS	4	RDDFNAM	NEWNAME ADDRESS
12	(C)	ADDRESS	4	RDDFVSER	NEW VOLSER ADDRESS
16	(10)	ADDRESS	4	RDDFCLNW	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)	BITSTRING	1	RDDFUACC	UACC FLAGS.
		1... ....		RDDFALTR	ALTER AUTHORITY.
		.1.. ....		RDDFCNTL	CONTROL AUTHORITY.
		..1. ....		RDDFUPD	UPDATE AUTHORITY.
		...1 ....		RDDFREAD	READ AUTHORITY.
		.... 1...		RDDFEXEC	EXEC AUTHORITY. (TURNED ON TOGETHER WITH NONE)
		.... .11.		*	RESERVED.
		.... ...1		RDDFNONE	NONE AUTHORITY.
37	(25)	UNSIGNED	1	RDDFLVL	LEVEL VALUE. 00 TO 99.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
38	(26)	BITSTRING	1	RDDFAUDT	AUDIT FLAGS.
		1... ..		RDDFALL	AUDIT ALL ACCESSES.
		.1.. ..		RDDFSUCC	AUDIT SUCCESSFUL ACCESS.
		..1. ....		RDDFFAIL	AUDIT ACCESSES THAT FAIL.
		...1 ....		RDDFANON	NO AUDITING.
		.... 11..		RDDFQS	SUCCESS QUALIFIER
		.... ..11		RDDFQF	FAILURE QUALIFIER
39	(27)	BITSTRING	1	RDDFFLG2	2ND FLAG BYTE
		1... ..		RDDFRFI	RACFIND PARAMETER GIVEN
		.1.. ..		RDDFRFIY	RACFIND=YES
		..1. ....		RDDFCHKA	CHKAUTH=YES
		...1 ....		RDDFTAPE	DSTYPE=T GIVEN
		.... 1...		RDDFEOS	ERASE=YES GIVEN
		.... .1..		RDDFMGEN	MGENER PARAMETER GIVEN 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 DATA ADDRESS
48	(30)	CHARACTER		RDDFEND	END OF V1.4 LIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 LIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	TAPE FLAG BYTE
		11.. ..		RDDFTLBL	TAPELBL SPECIFIED NL=B'01' STD=B'00' BLP=B'10'
		..11 11..		*	RESERVED
		.... ..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	B'1'=RACF COMMAND ISSUED RACDEF
		.111 111.		*	RESERVED
		.... ...1		RDDFPROP	B'1'=RACDEF ISSUED AS A RESULT OF AUTOMATIC DIRECTION.
80	(50)	ADDRESS	4	RDDFMCLS	ADDRESS OF MCLASS VALUE
84	(54)	ADDRESS	4	RDDFNOTF	ADDRESS OF NOTIFY ID
88	(58)	ADDRESS	4	RDDFSTCL	Address of STORCLAS data
92	(5C)	ADDRESS	4	RDDFMGCL	Address of MGMTCLAS data
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

## RDDFL

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
104	(68)	CHARACTER		RDD17END	END OF V1.7 LIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
104	(68)	STRUCTURE	20	RDDF18X	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

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RDDFACC1	38		2	RDDFMGEN	27	04	3
RDDFACC2	3C		2	RDDFMVOL	18		2
RDDFACEE	1C		2	RDDFNEWN	4	10	4
RDDFALL	26	80	3	RDDFNAM	8		4
RDDFALTR	24	80	3	RDDFNMX	8		3
RDDFANON	26	10	3	RDDFNONE	24	01	3
RDDFAUDT	26		2	RDDFNOTF	54		2
RDDFCATG	44		2	RDDFOLDV	4	20	4
RDDFCHGV	4	80	4	RDDFOVOL	8		2
RDDFCHKA	27	20	3	RDDFOWNR	28		2
RDDFCLNW	10		2	RDDFPROP	79	01	3
RDDFCNTL	24	40	3	RDDFQF	26	02	3
RDDFDATA	2C		2	RDDFQS	26	08	3
RDDFDDPR	68		2	RDDFREAD	24	10	3
RDDFDSTV	4	04	4	RDDFRFI	27	80	3
RDDFEND	30		2	RDDFRFIY	27	40	3
RDDFENT	5		3	RDDFRSOW	60		2
RDDFENTW	4		2	RDDFSLAB	6C		2
RDDFENTX	34		2	RDDFSLVL	40		2
RDDFENV	64		2	RDDFSPEC	4	01	4
RDDFEN31	34		3	RDDFSTCL	58		2
RDDFEOS	27	08	3	RDDFSUCC	26	40	3
RDDFEXDT	48		2	RDDFTADV	4	40	5
RDDFEXEC	24	08	3	RDDFTAPE	27	10	3
RDDFEXP	4E	01	3	RDDFTDEL	4	80	5
RDDFEXPX	4E	02	3	RDDFTLBL	4E	80	3
RDDFFAIL	26	20	3	RDDFUACC	24		2
RDDFFLGS	4		3	RDDFUNIT	20		2
RDDFFLGT	4E		2	RDDFUPD	24	20	3
RDDFFLG2	27		2	RDDFVRFY	64	80	3
RDDFFSEQ	4C		2	RDDFVSER	C		2
RDDFGEN	27	01	3	RDDFWARN	27	02	3
RDDFIENX	64	40	3	RDDF17	38		1
RDDFIMEX	64	20	3	RDDF18X	68		1
RDDFINMX	64	10	3	RDDF31	30		1
RDDFINST	1		3	RDDF31IN	4	08	4
RDDFINSW	0		2	RDD17END	68		2
RDDFIN31	30		2	RDD31END	38		2
RDDFISCM	4F	80	3	RDD8XEND	7C		2
RDDFISUR	4F		2				
RDDFLENG	0		3				
RDDFLIST	0		1				
RDDFLVL	25		2				
RDDFMCLS	50		2				
RDDFMDEL	4	02	4				
RDDFMENT	14		3				
RDDFMENX	14		2				
RDDFMGCL	5C		2				

**RDXP: RACROUTE REQUEST=DEFINE Exit Parameter List**

**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 ICHRD01 and ICHRD02  
**Serialization:** None  
**Function:** Contains the list of addresses passed to the RACROUTE REQUEST=DEFINE pre- and postprocessing installation exits.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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).
		.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 volume address 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 RACROUTE REQUEST=DEFINE 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 always starts with a 1-byte length field, followed by data.

# RDXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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-aligned and padded on the right with blanks. (See note at end of map on page 187.)
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 end of map on page 187.)
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 end of map on page 187.)
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 end of map on page 187.)
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 end of map on page 187.) 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.
		..1. ....		RDXADMCD	B'00100000' Add the MENTITY profile to the chain of profiles pointed to by the ACEEAMP field whether 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.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 IRRUT100 utility, allows an installation to modify or eliminate the RACF DASD data set naming convention. Corresponding processing might be required in the RACROUTE REQUEST=DEFINE 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:
		BYTE 0			
		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.
		.... 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 information from the installation-supplied profile.
		.1.. ....		RDXCNDAC	B'01000000' Use conditional access list from the installation-supplied profile.
		..11 1111		RDXPRESV	B'00111111' Reserved
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.

# RDXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
52	(34)	ADDRESS	4	RDXACEE	ACEE address: points to a fullword containing the address of an ACEE that is 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 zeros 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 end of map on page 187.)
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 end of map on page 187.) The UACC value has the following format: 1... .... RDXALTER B'10000000' ALTER authority .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. .... RDXAUAFF B'00100000' Audit all access attempts that fail. ...1 .... RDXNOAUD B'00010000' No auditing. .... 11.. RDXQSUC B'00001100' Qualifier for successful access attempts. The qualifier is of the following format: 00-READ, 01-UPDATE, 10-CONTROL, 11-ALTER .... ..11 RDXQFAL 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
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 user ID or group name. If there is no owner, this field contains blanks or zeros 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
80	(50)	ADDRESS	4	RDXFLAG2	Flag Byte 2 Address: points to a 1-byte area of the following format. (See note at end of map on page 187.)
		.... ....		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.
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 zeros 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 is 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 does 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: <ul style="list-style-type: none"> <li>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 <i>z/OS Security Server RACROUTE Macro Reference</i>.</li> <li>2. the reason code used to cause the ABEND to be issued. For meanings of these ABEND reason codes, see <i>z/OS Security Server RACF Messages and Codes</i>.</li> </ul> <p><b>Note:</b> Do not confuse this code with the return code from the RACROUTE REQUEST=DEFINE preprocessing and postprocessing exits described in <i>z/OS Security Server RACF System Programmer's Guide</i>.</p>
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 <i>z/OS Security Server RACROUTE Macro Reference</i> ).
96	(60)	ADDRESS	4	RDXWA	Exit work area: points to a fullword of zeros on the initial entry to the preprocessing routine. An installation can use this field for any purpose. Because this field is set to zeros 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.

# RDXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 always starts with a one byte length, followed by data.
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:
132	(84)	ADDRESS	4	RDXICMND RDXIRESV RDXIPROP RDXNOTIF	RACDEF issued by a command processor. Reserved. RACDEF issued as a result of automatic direction.
136	(88)	ADDRESS	4	RDXMCLAS	Address of NOTIFY userid: points to an 8-byte area containing the user ID of the user to be notified when RACF detects an unauthorized attempt to access a resource protected by this profile.
140	(8C)	ADDRESS	4	RDXSTCLA	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.
144	(90)	ADDRESS	4	RDXMGCLA	Address of storage class: points to a 2 byte length followed by a 1 to 8 byte storage class name.
148	(94)	ADDRESS	4	RDXRESOW	Address of management class: points to a 2 byte length followed by a 1 to 8 byte management class name.
152	(98)	ADDRESS	4	RDXRESOW	Address of resource owner: points to a 2 byte length followed by a 1 to 8 byte resource owner name.
156	(9C)	ADDRESS	4	RDXENVIR RDXEVRFY RDXERESV RDXDDNAM	Address of a 1-byte flag which indicates what ENVIR was coded. B'10000000' 1... .... ENVIR=VERIFY coded. B'01111111' .111 1111 Reserved.
160	(A0)	ADDRESS	4	RDXSLABL	Address of DDNAME: points to an 8-byte area containing the ddname associated with the data set name specified as the ENTITY name, or 8 blanks if DDNAME was not specified.
					Address of SECLABL; points to an eight byte field which contains the security label.

Offsets		Type	Len	Name (Dim)	Description	
Dec	Hex					
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 is sent to the remote system as INSTLN. If RDXIPROP is on, or RDXINSTL is zero, RDXINPTR is 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 is sent to the remote system as the second subparameter of the ACCLVL keyword. If RDXIPROP is on or RDXACCLP is zero, RDXACPTR is 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 post-processing exit only.	
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 post-processing exit only.	
180	(B4)	ADDRESS	4	RDXAUSED	Address of authorization used to grant access. It is set for post-processing exit only. Points to a 2-byte area of the following format:	
				1... ....	RDXANORM	Normal authority was used.
				.1.. ....	RDXASPEC	SPECIAL authority was used.
				..1. ....	RDXAOPER	OPERATIONS authority was used.
				...1 ....	RDXAEXIT	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

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RDXACCLP	6C		2	RDXMCDTP	B0		2
RDXACCLV	68		2	RDXMCLAS	88		2
RDXACEE	34		2	RDXMENTI	1C		2
RDXACLST	2C	2	2	RDXMGCLA	90		2
RDXACPTR	A8	4	2	RDXMGNAI	50	0	2
RDXADDVO	4	40	2	RDXMGNYS	50	4	2
RDXADMCD	24	20	2	RDXMVSER	20		2
RDXADMEN	24	10	2	RDXNDMNS	24	80	2
RDXADMCR	24	40	2	RDXNEWNA	4	10	2
RDXAEXIT	B4	10	2	RDXNOAUD	44	10	2
RDXALTER	3C	80	2	RDXNONEA	3C	1	2
RDXANORM	B4	80	2	RDXNOTFY	2C	10	2
RDXAOPER	B4	20	2	RDXNOTIF	84		2
RDXASPEC	B4	40	2	RDXOLDVO	4	20	2
RDXAUAAF	44	20	2	RDXOWNER	48		2
RDXAUACC	44	80	2	RDXOWNRI	2C	20	2
RDXAUDIT	44		2	RDXPRESV	2C	3F	2
RDXAUDTI	2C	10	2	RDXPROFA	30		2
RDXAUSAC	44	40	2	RDXPROFO	2C		2
RDXAUSED	B4		2	RDXQFAIL	44	3	2
RDXCATEG	74		2	RDXQSUC	44	C	2
RDXCATIN	2C	80	2	RDXRCODE	58		2
RDXCDTP	AC		2	RDXREADA	3C	10	2
RDXCHGVO	4	C0	2	RDXREAS	5C		2
RDXCKAYS	50	20	2	RDXRESFL	2C	4	2
RDXCLASS	18		2	RDXRESOW	94		2
RDXCNDAC	2C	40	2	RDXRESV7	24	F	2
RDXCNXPL	28		2	RDXRESV8	4	8	2
RDXCOMP	54		2	RDXRESV9	3C	E	2
RDXCONTR	3C	40	2	RDXRETPD	7C		2
RDXDATA	4C		2	RDXRFDNO	50	80	2
RDXDDNAM	9C		2	RDXRFDNS	50	0	2
RDXDEFIN	4	0	2	RDXRFDYS	50	C0	2
RDXDELET	4	80	2	RDXSALTE	44	C	2
RDXDSTYM	4	2	2	RDXSCONT	44	8	2
RDXDSTYT	50	10	2	RDXSECLP	70		2
RDXDSTYV	4	4	2	RDXSECLV	2C	40	2
RDXENTIT	C		2	RDXSEQNO	78		2
RDXENVIR	98		2	RDXSLABL	A0		2
RDXEOSCR	2C	20	2	RDXSPECY	4	1	2
RDXERAYS	50	8	2	RDXSREAD	44	0	2
RDXERESV	98	7F	2	RDXSTCLA	8C		2
RDXEVRFY	98	80	2	RDXSUPDA	44	4	2
RDXFLAG	4		2	RDXTLBLP	64	80	2
RDXFLAGA	24		2	RDXTLNL	64	40	2
RDXFLAG2	50		2	RDXTLSTD	64	0	2
RDXFLAG3	64		2	RDXTYPEV	4	C0	2
RDXGLAUD	2C	8	2	RDXUACC	3C		2
RDXGNRCY	50	1	2	RDXUACCV	2C	80	2
RDXICMND	80	80	2	RDXUALTE	44	3	2
RDXIDATA	2C	4	2	RDXUCONT	44	2	2
RDXINSTL	8		2	RDXUNIT	38		2
RDXINPTR	A4	4	2	RDXUPDAT	3C	20	2
RDXIPROP	80	1	2	RDXUREAD	44	0	2
RDXIRESV	80	7E	2	RDXURETP	2C	8	2
RDXISSUR	80		2	RDXUUPDA	44	1	2
RDXLEN	0		2	RDXVORN	10		2
RDXLEVEL	40		2	RDXVSER	14		2
RDXLEVLV	2C	40	2	RDXWA	60		2
RDXLGDAY	2C	2	2	RDXWNGYS	50	2	2
RDXLGTIM	2C	1	2	RDXWRNNG	2C	1	2
RDXLGZON	2C	80	2				

## RFXP: RACROUTE REQUEST=FASTAUTH Exit Parameter List

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	BITSTRING	1	RFXFLAGS	Authority flags: contains the requested access authority. Access authority requested is: X'02' - READ, X'04' - UPDATE, X'08' - CONTROL, and X'80' - ALTER. Note: These bit mappings are identical to the bit mappings for the RACHECK ATTR parameter pointed to by RCXFLAG2 at offset 8 in the RACROUTE REQUEST=AUTH preprocessing and postprocessing exit parameter list. See data area RCXP.
		.... ..1.		RFXREAD	
		.... ..1..		RFXUPDAT	
		.... 1...		RFXCONTR	
		1... ..		RFXALTER	
1	(1)	BITSTRING	1	RFXFLAG2	LOG flags: contains the types of access attempts to be recorded. LOG values are: X'80' - ASIS, X'40' - NOFAIL. X'20' - ENTITYX specified. If on, RFXENTIT points to a name in ENTITYX format. X'10' - AUTHCHKS=CRITONLY
		1... ..		RFXASIS	
		..1.. ..		RFXNFAIL	
		..1. ....		RFXNENTX	
		...1 ..		RFXNACHK	
		.... 1111		*	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 R6 44 - z/OS V1R8
3	(3)	HEX	1	RFXPVERS	Parameter List Version: 0 - OS/390 Security Server R3 or earlier 1 - OS/390 Security Server R4 2 - OS/390 Security Server R6 3 - z/OS V1R8

## RFXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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-aligned and padded with blanks if necessary. For ENTITYX (RFXNENTX=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	RFXCLASS	CLASS name address: points to an 8-byte field containing the class name; the name is left-aligned 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 uses 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-aligned 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 RACROUTE REQUEST=FASTAUTH 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 RACROUTE REQUEST=FASTAUTH preprocessing exit routine.
28	(1C)	ADDRESS	4	RFXALET	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. This field is only present when RFXPVERS is 1 or higher.
32	(20)	ADDRESS	4	RFXLOGS	LOGSTR address: If nonzero, it points to a 1-byte length field followed by character data that can be from 0 to 255 bytes long. This field is only present when RFXPVERS is 1 or higher.
36	(24)	ADDRESS	4	RFXPENVR	ENVR address: If nonzero, 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 nonzero, it points to an area containing a data structure to be used for authority checking. This field is only present if RFXPVERS is 3 or higher.



## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
RFXACEE	C		2
RFXALET	1C		2
RFXALTER	0	80	3
RFXANSTL	18		2
RFXAPPL	10		2
RFXASIS	1	80	3
RFXCLASS	8		2
RFXCONTR	0	8	3
RFXENTIT	4		2
RFXFLAGS	0		2
RFXFLAG2	1		2
RFXLOGS	20		2
RFXNACHK	1	10	3
RFXNENTX	1	20	3
RFXNFAIL	1	40	3
RFXPCRIT	28		2
RFXPENVR	24		2
RFXPLEN	2		2
RFXPVERS	3		2
RFXREAD	0	2	3
RFXUPDAT	0	4	3
RFXWA	14		2

**RIPL: RACROUTE REQUEST=TOKENBLD/VERIFY/VERIFYX Parameter List (Request Section)**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE		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
		..1. ....		INITPRAL	VERIFYX INTERNAL PROPAGATION
		...1 ....		INITVFX	RACINIT VERIFYX INDICATOR
		.... 1...		INITSYSN	1 - PARAMETER SPECIFIED THAT IS NOT COMPATIBLE WITH SYSTEM=YES
		.... .1..		INITNLOG	1 - LOG=NONE SPECIFIED
		.... ..11		*	RESERVED
3	(3)	BITSTRING	1	INITFLG1	FLAG BYTE 1
		11.. ....		INITENVR	ENVIR - 00 CREATE, 01 CHANGE, 10 DELETE, 11 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	ADDRESS OF USERID BUFFER
8	(8)	ADDRESS	4	INITPPTR	ADDRESS OF PASSWORD BUFFER
12	(C)	ADDRESS	4	INITSPTR	ADDRESS OF START PROC NAME
16	(10)	ADDRESS	4	INITPTR	ADDRESS OF INSTALLATION INFORMATION
20	(14)	ADDRESS	4	INITGPTR	ADDRESS OF GROUP NAME BUFFER
24	(18)	ADDRESS	4	INITNPTR	ADDRESS OF NEW PASSWORD BUFFER
28	(1C)	CHARACTER		INITEND1	END PART1

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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		INITEND2	END PART2

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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		INITEND3	END PART3

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
56	(38)	STRUCTURE	44	INITPRM4	RELEASE 1.9
56	(38)	UNSIGNED	1	INITSESN	SESSION TYPE - SEE TOKEN MAP FOR SPECIFIC VALUES
57	(39)	BITSTRING	1	INITFLG2	WORK UNIT IDENTITY FLAGS
		1... ..		INITRS	PART OF TRUSTED COMP BASE
		.1.. ..		INITRMT	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 was coded
		..1. ....		INITNSTC	NESTED=COPY was coded
		...1 1111		*	RESERVED
59	(3B)	SIGNED	1	*	RESERVED
60	(3C)	ADDRESS	4	INITSLBP	SECLABL ADDRESS
64	(40)	ADDRESS	4	INITXNDP	EXENODE ADDRESS
68	(44)	ADDRESS	4	INITSIDP	SUSERID ADDRESS
72	(48)	ADDRESS	4	INITSNDP	SNODE ADDRESS
76	(4C)	ADDRESS	4	INITSGPP	SGROUP ADDRESS
80	(50)	ADDRESS	4	INITPOEP	POE ADDRESS
84	(54)	ADDRESS	4	INITUTKP	INPUT TOKEN ADDRESS
88	(58)	ADDRESS	4	INITSTKP	STOKEN ADDRESS
92	(5C)	ADDRESS	4	INITLSRP	LOGSTR ADDRESS
96	(60)	ADDRESS	4	INITOTKP	OUTPUT TOKEN ADDRESS
100	(64)	CHARACTER		INITEND4	END OF 1.9 PLIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
100	(64)	STRUCTURE	8	INITPRM5	RELEASE 1.9.2
100	(64)	ADDRESS	4	INITENVI	ENVRIN ADDRESS
104	(68)	ADDRESS	4	INITENVO	ENVROUT ADDRESS
108	(6C)	CHARACTER		INITEND5	END OF 1.9.2 PLIST

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
108	(6C)	STRUCTURE	8	INITPRM6	RELEASE 2.6
108	(6C)	ADDRESS	4	INITPNP	POE NETWORK NAME ADDRESS
112	(70)	CHARACTER		INITEND6	END OF 2.6

# RIPL

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
112	(70)	STRUCTURE	8	INITPRM7	X500 NAME EXTENSION
112	(70)	ADDRESS	4	INITX5PR	X500 NAME PAIR ADDRESS
116	(74)	CHARACTER		INITEND7	END OF X500 NAME EXT

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
116	(74)	STRUCTURE	4	INITPRM8	SERVAUTH EXTENSION
116	(74)	ADDRESS	4	*	RESERVED
120	(78)	ADDRESS	4	INITSRVA	SERVAUTH ADDRESS
124	(7C)	CHARACTER		INITEND8	END SERVAUTH EXTENSION

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
124	(7C)	STRUCTURE	12	INITPRM9	RELEASE HRF7730
124	(7C)	ADDRESS	4	INITPHRA	PASSWORD PHRASE ADDRESS
128	(80)	ADDRESS	4	INITNPHA	NEW PASSWORD PHRASE ADDRESS
132	(84)	ADDRESS	4	INITICTX	ICTX ADDRESS
136	(88)	CHARACTER	*	INITEND9	END OF HRF7730 PLIST

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
136	(88)	STRUCTURE	8	INITPRMA	RELEASE HRF7760
136	(88)	ADDRESS	4	INITIDID	DISTRIBUTED IDENTITY DATA STRUCTURE (IDID) ADDRESS
140	(8C)	ADDRESS	4	INITICRX	ICRX ADDRESS
144	(90)	CHARACTER	*	INITENDA	END OF HRF7760 PLIST

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	INITPHRB	PASSWORD PHRASE BUFFER
0	(0)	UNSIGNED	1	INITPHRL	PASSWORD PHRASE LENGTH
1	(1)	CHARACTER	*	INITPHRS	PASSWORD PHRASE

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	INITNPHB	NEW PASSWORD PHRASE BUFFER
0	(0)	UNSIGNED	1	INITNPHL	NEW PASSWORD PHRASE LENGTH
1	(1)	CHARACTER	*	INITNPHS	NEW PASSWORD PHRASE

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	9	INITUSR	USERID BUFFER
0	(0)	ADDRESS	1	INITUSRL	USERID LENGTH
1	(1)	CHARACTER	8	INITUSRI	USERID

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	9	INITPAS	PASSWORD BUFFER
0	(0)	ADDRESS	1	INITPASL	PASSWORD LENGTH
1	(1)	CHARACTER	8	INITPASS	PASSWORD

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITGRP	GROUP NAME BUFFER
0	(0)	ADDRESS	1	INITGRPL	GROUP NAME LENGTH
1	(1)	CHARACTER	8	INITGRPN	GROUP NAME

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITNPA	NEW PASSWORD BUFFER
0	(0)	ADDRESS	1	INITNPAL	NEW PASSWORD LENGTH
1	(1)	CHARACTER	8	INITNPAS	NEW PASSWORD

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	256	INITOIDB	OID BUFFER
0	(0)	ADDRESS	1	INITOIDL	OID LENGTH
1	(1)	CHARACTER	255	INITOID	OID VALUE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITSERV	SERVAUTH BUFFER
0	(0)	UNSIGNED	1	INITSRVL	SERVAUTH LENGTH
1	(1)	CHARACTER	*	INITSRVN	SERVAUTH NAME

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITSUID	SUBMITTERS USERID KEYWORD
0	(0)	UNSIGNED	1	INITSILN	LENGTH OF SUBMIT USERID
1	(1)	CHARACTER	8	INITSINM	NAME OF SUBMITTER'S ID

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITSNOD	SUBMITTER'S NODE KEYWORD
0	(0)	UNSIGNED	1	INITSNLN	SUBMIT NODE DATA LENGTH
1	(1)	CHARACTER	8	INITSNNM	NAME OF SUBMITTER'S NODE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	9	INITSGRP	SUBMITTER'S GROUP KEYWORD
0	(0)	UNSIGNED	1	INITSGLN	SUBMIT GROUP DATA LENGTH
1	(1)	CHARACTER	8	INITSGNM	NAME OF SUBMIT GROUP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	256	INITLGST	LOG STRING KEYWORD MAP
0	(0)	UNSIGNED	1	INITLSLN	LENGTH OF LOG STRING DATA

## RIPL

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
1	(1)	CHARACTER	255	INITLGSD	LOG STRING DATA

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	14	INITENV	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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	INITPON	POE NETWORK NAME KEYWORD
0	(0)	ADDRESS	1	INITPONL	NETWORK NAME LENGTH
1	(1)	CHARACTER	*	INITPOEN	NETWORK NAME

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	INITX500	X500 NAME PAIR
0	(0)	UNSIGNED	4	INITXLEN	LENGTH OF ENTIRE NAME PAIR DATA STRUCTURE
4	(4)	UNSIGNED	2	INITIDNL	ISSUERS NAME LENGTH
6	(6)	UNSIGNED	2	INITSDNL	SUBJECTS NAME LENGTH
8	(8)	CHARACTER	*	INITIDN	ISSUERS NAME
*	(*)	CHARACTER	*	INITSDN	SUBJECTS NAME

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
INITACCP	20		2	INITESSP	C		2
INITACEP	34		2	INITFLG0	2		2
INITANY	2	40	3	INITFLG1	3		2
INITAPPP	30		2	INITFLG2	39		2
INITBLW	2	80	3	INITFLG3	3A		2
INITELEN	0		2	INITGPTR	14		2
INITENCR	3	01	3	INITGRP	0		1
INITENDA	90		2	INITGRPL	0		2
INITEND1	1C		2	INITGRPN	1		2
INITEND2	30		2	INITICRX	8C		2
INITEND3	38		2	INITIDID	88		2
INITEND4	64		2	INITIDN	8		2
INITEND5	6C		2	INITIDNL	4		2
INITEND6	70		2	INITIPTR	10		2
INITEND7	74		2	INITJOBP	5C		2
INITEND8	78		2	INITLEN	0		2
INITENLN	0		2	INITLGSD	1		2
INITENNM	1		2	INITLGST	0		1
INITENOD	0		1	INITLSLN	0		2
INITENV	0		1	INITLSRP	5C		2
INITENVI	64		2	INITNLOG	2	04	3
INITENVO	68		2	INITNPA	0		1
INITENVR	3	80	3	INITNPAL	0		2
INITERRO	3A	80	3	INITNPAS	1		2
INITESAD	8	20	2	INITNPHA	80		2
INITESKY	D	40	2	INITNPTR	18		2
INITESLN	4		2	INITNSMC	3	20	3

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
INITNSTA	3	04	3	INITUSRL	0		2
INITNSTC	3A		3	INITUTKP	54		2
INITNSTY	3A		3	INITVFX	2	10	3
INITOID	1		2	INITXLEN	0		2
INITOIDB	0		1	INITXNDP	40		2
INITOIDL	0		2	INITX5PR	70		2
INITOIDP	24		2	INITX500	0		1
INITOTKP	60		2				
INITPARG	0		1				
INITPAS	0		1				
INITPASL	0		2				
INITPASS	1		2				
INITPCHK	3	08	3				
INITPGRP	1C		2				
INITPHRA	7C		2				
INITPNP	6C		2				
INITPOEN	1		2				
INITPOEP	50		2				
INITPON	0		1				
INITPONL	0		2				
INITPPTR	8		2				
INITPRAL	2	20	3				
INITPRMA	88		1				
INITPRM2	1C		1				
INITPRM3	30		1				
INITPRM4	38		1				
INITPRM5	64		1				
INITPRM6	6C		1				
INITPRM7	70		1				
INITPRM8	74		1				
INITPRM9	7C		1				
INITRMSP	39	10	3				
INITRMT	39	40	3				
INITRS	39	80	3				
INITRSSP	39	20	3				
INITSDN	*		2				
INITSDNL	6		2				
INITSESN	38		2				
INITSERV	0		2				
INITSGLN	0		2				
INITSGNM	1		2				
INITSGPP	4C		2				
INITSGRP	0		1				
INITSIDP	44		2				
INITSILN	0		2				
INITSINM	1		2				
INITSLBP	3C		2				
INITSNDP	48		2				
INITSNLN	0		2				
INITSNNM	1		2				
INITSNOD	0		1				
INITSPTR	C		2				
INITSRVA	78		2				
INITSRVL	0		2				
INITSRVN	1		2				
INITSTKP	58		2				
INITSUB#	1		2				
INITSUBS	3	10	3				
INITSUID	0		1				
INITSYSN	2	08	3				
INITTRMP	28		2				
INITULOG	3	02	3				
INITUPTR	4		2				
INITUSR	0		1				
INITUSRI	1		2				

---

**RIXP: RACROUTE REQUEST=VERIFY/VERIFYX Exit Parameter List**

**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.

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
0	(0)	STRUCTURE		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:
		.... ....		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: <ul style="list-style-type: none"> <li>• Bypass checking that the old password is correct and has not expired</li> <li>• Bypass checking that the new password is valid</li> <li>• Bypass updating the old password with the new</li> <li>• Bypass incrementing the password revoke count or resetting it to zero.</li> </ul> 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.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
8	(8)	ADDRESS	4	RIXLOGAL RIXENCNO RIXUID	B'00000010' .... ..1. LOG=ALL was specified. B'00000001' .... ..1 ENCRYPT=NO was specified. 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 the user ID is less than 8 characters in length, the remainder of the field should be filled with blanks. If no user ID 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 user ID might have come from the started procedure table (ICHRIN03). For information about ICHRIN03, see <i>z/OS Security Server RACF System Programmer's Guide</i> .
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 does not use the value.
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 RACROUTE REQUEST=VERIFY(X) preprocessing exit. Do not confuse this value with the terminal data value (pointed to by RIXTRMDA, offset 68 in the parameter list) or the user data value (pointed to by RIXUSRDA, offset 72 in the parameter 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 RACROUTE REQUEST=VERIFY(X) preprocessing exit, this address points to the area of storage where the ACEE is built. At entry to the RACROUTE REQUEST=VERIFY(X) 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.
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.

## RIXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
40	(28)	ADDRESS	4	RIXACTIN	<p>ACTINFO address: points to a 144-byte area containing accounting information (or zeros if no accounting information exists). The 144-byte area is consistent with similar accounting information in the SMF (type 20) job initiation record:</p> <ul style="list-style-type: none"> <li>The first byte contains the number (in binary) of accounting fields.</li> <li>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.</li> </ul>
44	(2C)	ADDRESS	4	RIXOIDCD	<p>OIDCARD address: points to an area containing a 1-byte length field followed by a field containing the OIICARD identification number. The length byte is 0 if OIICARD was not specified.</p>
48	(30)	ADDRESS	4	RIXTRMID	<p>TERMID address: points to an 8-byte area containing the terminal identifier. The name is left-aligned 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 is also altered and its class is TERMINAL (since a terminal is the same as a port of entry in the TERMINAL class.)</p>
52	(34)	ADDRESS	4	RIXWA	<p>Work area address: points to a fullword of zeros on the initial entry to the preprocessing routine. Because this field is set to zeros before entry to the preprocessing exit, the preprocessing and postprocessing exits can use this work area to communicate with each other.</p>
56	(38)	ADDRESS	4	RIXCOMP	<p>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 zeros for the postprocessing routine if RACINIT is not going to issue an ABEND code. This address points to an area containing zeros 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 is 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 does 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.</p>

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
60	(3C)	ADDRESS	4	RIXRCODE	<p>Return code address: points to a 4-byte field containing either:</p> <ol style="list-style-type: none"> <li>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 <i>z/OS Security Server RACROUTE Macro Reference</i>.</li> <li>the reason code used to cause the ABEND to be issued (if nonzero). For meanings of these ABEND codes, see <i>z/OS Security Server RACF Messages and Codes</i>.</li> </ol> <p>This address points to an area containing zeros for the preprocessing routine.</p> <p><b>Note:</b> Do not confuse this code with the return code from the RACROUTE REQUEST=VERIFY(X) preprocessing and postprocessing exits documented in <i>z/OS Security Server RACF System Programmer's Guide</i>.</p>
64	(40)	ADDRESS	4	RIXFLAG2	<p>Flag byte address: points to a 1-byte area of the following format:</p> <p>B'10000000' 1... .... Bypass OIDCARD processing. RACINIT ignores any OIDCARD information and any user profile indication that an OIDCARD is required.</p> <p>B'01000000' .1.. .... 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.</p> <p>B'00100000' ..1. .... For post-processing exit, indicates user is being authenticated by a PassTicket.</p> <p>B'00010000' ...1 .... For post-processing exit, indicates a replay of previously used PassTicket.</p> <p>B'00001000' ....1 .... ERROROPT=NOABEND .... .... .111 Reserved.</p> <p>NESTED=YES was coded</p> <p>NESTED=COPY was coded</p> <p>B'00000001' Indicates that RACINIT is being internally redriven to build an ACEE or RACO based on information within section two of the IDID because ICR within the ICRX could not be used to locate a corresponding stored entry within the RACF Identity Propagation cache.</p>
		1... ....		RIXBYPOI	
		.1.. ....		RIXCACHE	
		..1. ....		RIXPTAUT	
		...1 ....		RIXPTRPY	
		.... 1...		RIXERRO	
		.... .1..		RIXNSTY	
		.... ..1.		RIXNSTC	
		.... ...1		RIXIDPRD	
68	(44)	ADDRESS	4	RIXTRMDA	<p>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.</p>
72	(48)	ADDRESS	4	RIXUSRDA	<p>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 zeros 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.</p>

## RIXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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-aligned 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 is 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: <ul style="list-style-type: none"> <li>For ENVIR=CREATE, RACF places 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.</li> <li>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.</li> </ul>
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 (1) the application name was not supplied, (2) the profile could not be retrieved, or (3) 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 because IMS uses this field when IMS is active.)

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 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.
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: yyddd 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 contains zeros. Upon entry to the ICHRIX02 exit, the three byte field contains the value from the user entry.
112	(70)	ADDRESS	4	RIXPSES	"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: B'10000000' 1... .... Default Token bit B'01000000' .1.. .... Undefined User bit Reserved B'00010000' ...1 .... Token in error B'00001000' .... 1... Part of Trusted Computer Base B'00000100' .... .1.. Surrogate User ID B'00000010' .... ..1. REMOTE Keyword specified 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.
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 NAME 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.

## RIXP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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: YYYDDDS where 'YYYY' is the year, 'DDD' is the day, and 'S' is the packed decimal sign. <b>Note:</b> Upon initial entry to exit ICHRIX01, the 4-byte field contains zeros. Upon entry to the ICHRIX02 exit the 4-byte field contains the value from the user entry.
192	(AC)	ADDRESS	4	RIXX5PRP	X500 NAME PAIR ADDRESS: points to 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. Zero if 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	PASSWORD PHRASE ADDRESS: points to a 1-byte length (0 if not specified) followed by a 100 byte buffer containing the password phrase.
184	(B8)	ADDRESS	4	RIXNPHA	NEW PASSWORD PHRASE ADDRESS: points to a 1-byte length (0 if not specified) followed by a 100 byte buffer containing the new password 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 data area as mapped by IRRPIDID.
196	(C4)	ADDRESS	4	RIXICRX	ICRX ADDRESS: Points to an ICRX data area as mapped by IRRPICRX.

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RIXACEE	20		2	RIXJOBNM	50		2
RIXACEEA	5C		2	RIXLEN	0		2
RIXACTIN	28	2	2	RIXLOGAL	4	2	2
RIXAPPLD	60		2	RIXNEWPA	1C		2
RIXAPPLN	54		2	RIXNPHA	B8		2
RIXAPPLP	64		2	RIXNSTC	02		3
RIXBYPOI	40	80	2	RIXNSTY	04		3
RIXCACHE	40	40	2	RIXOIDCD	2C		2
RIXCOMP	38		2	RIXPCIA	68		2
RIXDFTKN	74	80	2	RIXPGMNM	24		2
RIXENCNO	4	1	2	RIXPHRA	B4		2
RIXENVCH	4	40	2	RIXPLCDA	6C		2
RIXENVCR	4	0	2	RIXPLCD4	A8		2
RIXENVDE	4	80	2	RIXPLSRP	9C		2
RIXERRO	40	08	2	RIXPOENP	A4		2
RIXERRTK	74	10	2	RIXPOTKP	A0		2
RIXFLAG	4		2	RIXPPOE	90		2
RIXFLAG2	40		2	RIXPPOEX	78		2
RIXGROUP	18		2	RIXPRIV	74	1	2
RIXICRX	C4		2	RIXPSCKN	4	8	2
RIXICTX	BC		2	RIXPSCKY	4	0	2
RIXIDID	C0		2	RIXPSESN	70		2
RIXIDPRD	40		3	RIXPSGPP	8C		2
RIXINSTL	14		2	RIXPSIDP	84		2

Name	Hex Offset	Hex Value	Level
RIXPSLBP	7C		2
RIXPSNDP	88		2
RIXPSTKP	98		2
RIXPTAUT	40	20	2
RIXPTRPY	40	10	2
RIXPUTKP	94		2
RIXPWD	C		2
RIXPXNDP	80		2
RIXRCODE	3C		2
RIXREMOT	74	2	2
RIXSMCNO	4	20	2
RIXSMCYS	4	0	2
RIXSRVA	4		2
RIXSTANO	4	4	2
RIXSTART	10		2
RIXSUBPL	58		2
RIXSUBPP	4	10	2
RIXSURGU	74	4	2
RIXTRMDA	44		2
RIXTRMID	30		2
RIXTRMLV	4C		2
RIXTRUST	74	8	2
RIXUDUSR	74	40	2
RIXUID	8		2
RIXUSRDA	48		2
RIXWA	34		2
RIXWUIDF	74		2
RIX5PRP	AC		2

---

**RLST: RACROUTE REQUEST=LIST Parameter List (Request Section)**

**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 SAFR plus offset in SAFPRACP

**Serialization:** None

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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	Set to 2 or 3
3	(3)	BITSTRING	1	RLSTFLAG	Flags:
		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 (0)	This extension exists when RELEASE=2.1 or greater is specified or RLSTCODE=3
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



## Cross Reference

Name	Hex Offset	Hex Value	Level
RLSTACEE	8		2
RLSTAPPL	10		2
RLSTCLAS	14		2
RLSTCODE	2		2
RLSTFLAG	3		2
RLSTFLG2	1C		2
RLSTFLTP	18		2
RLSTGLBL	1C	1	2
RLSTINST	C		2
RLSTLEN	1E		2
RLSTLIST	4		2
RLSTLOC	3	10	2
RLSTNSPN	1		2
RLSTOPT	3	C0	2
RLSTOWN	3	20	2
RLSTPLEN	1E	20	2
RLSTPSPN	0		2
RLSTREL	3	8	2
RLSTR19	3	4	2
RLSTR192	3	2	2
RLSTR21	1C	2	2
RLSTSPNS	0		2
RLSTXTND	1C		2

---

**RLX1P: RACROUTE REQUEST=LIST Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
		.... ....		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 RACROUTE REQUEST=LIST pre/postprocessing exits.
16	(10)	ADDRESS	4	RLX1CLAS	CLASS address: points to an 8-byte field containing the class name. The class name is left-aligned 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 zeros 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.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 is obtained. The second byte identifies the subpool from which the profiles is 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 have no effect, because the storage is 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 zeros 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.
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 the 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 occurrences 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 occurrences 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 occurrences 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).
		.... 1...		RLX1GIL	B'00001000' .... 1000 Least restrictive when resolving conflicts between groups and an individual occurrence.
		.... .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 conflicts between groups and an individual occurrence.

## RLX1P

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
		.... ...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.
		.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 <i>z/OS Security Server RACF Command Language Reference</i> .
		..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.)
		...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. For more information, see the <i>z/OS Security Server RACF System Programmer's Guide</i> .

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
40	(28)	ADDRESS	4	RLX1XAMT	Profile expansion amount address: points to a fullword initialized to zeros, 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.
48	(30)	ADDRESS	4	RLX1WA	Work area address: points to a fullword of zeros. The exit can use this field for any purpose. Because this field is initialized to zeros 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 in the ISP data area. For a description of the use of the RACF in-storage profile (ISP) in RACROUTE REQUEST=LIST exit processing, see <i>z/OS Security Server RACF System Programmer's Guide</i> . This field has meaning only for the RACROUTE REQUEST=LIST postprocessing exit and is 0 if a class tree is not encountered.
56	(38)	ADDRESS	4	RLX1FLTR	Filter string pointer: points to a 1-byte length followed by a filter string. For the format of the filter string, see the description of the RACLIST macro FILTER keyword.

**Cross Reference**

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RLX1ACCL	24	88	2	RLX1GGX	24	20	2
RLX1ACEE	14		2	RLX1GIL	24	8	2
RLX1ADTA	24	22	2	RLX1GIM	24	4	2
RLX1APPL	18		2	RLX1GIW	24	1	2
RLX1AUDF	24	44	2	RLX1GIX	24	2	2
RLX1CLAS	10		2	RLX1GLAU	24	44	2
RLX1CODE	2C		2	RLX1IDTA	24	22	2
RLX1ENVC	4	0	2	RLX1INST	C		2
RLX1ENVD	4	80	2	RLX1LEN	0		2
RLX1FLAG	4		2	RLX1LIST	20		2
RLX1FLTR	38		2	RLX1NTFY	24	22	2
RLX1FUNC	8		2	RLX1OWNN	4	0	2
RLX1GGL	24	80	2	RLX1OWNX	24	22	2
RLX1GGM	24	40	2	RLX1OWNY	4	20	2
RLX1GGW	24	10	2	RLX1POST	8	80	2

## RLX1P

Name	Hex Offset	Hex Value	Level
RLX1PRE	8	0	2
RLX1RESL	24	44	2
RLX1RULE	24		2
RLX1SLBL	24	11	2
RLX1SUBP	1C		2
RLX1TERM	24	11	2
RLX1TREE	34		2
RLX1UACC	24	44	2
RLX1WA	30		2
RLX1XAMT	28		2

**RLX2P: RACROUTE REQUEST=LIST Selection Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	ADDRESS	4	RLX2LEN	Number address: points to a fullword containing the number of parameters in this list, including itself.
4	(4)	ADDRESS	4	RLX2PRPA	Preparameters address: address of the parameter list passed to the RACROUTE REQUEST=LIST preprocessing exit. This parameter is passed to allow communication between the RACROUTE REQUEST=LIST pre- and postprocessing exits and the RACROUTE REQUEST=LIST selection exit, 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
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.
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.

## RLX2P

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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: X'80' - ALTER X'40' - CONTROL X'20' - UPDATE X'10' - READ 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.
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 you want, 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.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.
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 a 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 a 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.
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: B'1000000' 1... .... The resource has a session from being established. .111 1111 Reserved
		1... ....		RLX2SLCK	LAST CHANGE DATE address: points to a 4-byte area containing the last change date for the session key.
92	(5C)	ADDRESS	4	RLX2LKYP	SESSION KEY INTERVAL address: points to a 2-byte area containing the maximum number of days for which the key is valid.
96	(60)	ADDRESS	4	RLX2KYIN	MAXIMUM FAILURES address: points to a 2-byte area containing the maximum number of failed attempts permitted before the resource is locked.
100	(64)	ADDRESS	4	RLX2MXFL	SECLABEL address: points to an 8-byte area containing the SECLABEL of the resource profile.
104	(68)	ADDRESS	4	RLX2SLBL	

## RLX2P

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
108	(6C)	ADDRESS	4	RLX2ACL2	<p>SECOND ACCESS 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:</p> <p>8 byte program name of which the first byte is considered a flag byte.            8 byte user ID or group name.            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 length data.</p>
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

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
RLX2ACCL	30		2	RLX2UPDA	1C	20	2
RLX2ACL2	6C		2	RLX2WARN	50		2
RLX2ALTE	1C	80	2				
RLX2AL2C	70		2				
RLX2ANCH	34		2				
RLX2APPL	3C		2				
RLX2AUD	20		2				
RLX2CONT	1C	40	2				
RLX2CTGY	48		2				
RLX2DATA	2C		2				
RLX2DGRP	8	0	2				
RLX2DRES	8	40	2				
RLX2FLAG	8		2				
RLX2GLAU	24		2				
RLX2KEYP	54		2				
RLX2KYIN	60		2				
RLX2LEN	0		2				
RLX2LKYD	5C		2				
RLX2MXFL	64		2				
RLX2NONE	1C	1	2				
RLX2NPRV	8	80	2				
RLX2NTFY	40		2				
RLX2OWNN	38		2				
RLX2PRPA	4		2				
RLX2PRVE	8	0	2				
RLX2READ	1C	10	2				
RLX2RGCA	18		2				
RLX2RGRP	10		2				
RLX2RLVL	28		2				
RLX2RNAM	C		2				
RLX2RNCL	14		2				
RLX2SCON	74		2				
RLX2SLBL	68		2				
RLX2SLCK	58	80	2				
RLX2SLSF	58		2				
RLX2SLVL	44		2				
RLX2TERM	4C		2				
RLX2UACC	1C		2				

**RNG: RACF Database Range Table**

NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

End of NOT Programming Interface Information

---

**RRPF: Resident Profile Map**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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-VSAM
		.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 FLAG

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
69	(45)	UNSIGNED	1	DSPVRSN	VERSION = 1
70	(46)	BITSTRING	1	DSPWARN	WARNING FLAG BIT 7 = 1 - RESOURCE HAS WARNING ATTRIBUTE
71	(47)	BITSTRING	1	DSPEOS	ERASE-ON-SCRATCH FLAG BIT 0 = 1 - DATASET IS ERASED WHEN SCRATCHED
72	(48)	SIGNED	4	DSPINST	OFFSET TO INSTALLATION DATA
76	(4C)	ADDRESS	4	DSPNEXTP	ADDR NEXT MODEL
80	(50)	BITSTRING	1	DSPFNF	MODEL FOUND INDICATOR 0,FD -1,NFD
81	(51)	UNSIGNED	1	DSPSLVL	RESOURCE SECURITY LEVEL
82	(52)	SIGNED	2	DSPRTPD	RETENTION PERIOD
84	(54)	CHARACTER	8	DSPOWNER	RESOURCE 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 THE TERMINAL MAY NOT BE USED (BIT 0 - SUNDAY, BIT 1 - MONDAY,...
110	(6E)	CHARACTER	3	DSPLOGNT	EARLIEST TIME THAT THE TERMINAL BE USED.(HHMM)
113	(71)	CHARACTER	3	DSPLOGFT	LATEST TIME THAT THE TERMINAL BE USED.(HHMM)
116	(74)	CHARACTER	3	DSPTZONE	TIME OFFSET OF TERMINAL FROM THE CPU. (+ = EAST, - = WEST)
119	(77)	CHARACTER	1	*	RESERVED
120	(78)	CHARACTER	8	DSPSLABL	SECLABEL
128	(80)	CHARACTER	4	DSPDSNBF	Character form of offset to resource
128	(80)	SIGNED	4	DSPDSNOF	Offset to resource name in extended format
132	(84)	CHARACTER	4	DSPAPOFF	Offset to the application data.
132	(84)	SIGNED	4	DSPAPOFF	Offset to the application data.
136	(88)	SIGNED	4	DSPIPOFF	Offset to IP lookup value

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	DSPVOLS	VOLSER LIST
0	(0)	UNSIGNED	2	DSPVOLCT	NUMBER OF ENTRIES
2	(2)	CHARACTER	6	DSPVOLSR (*)	VOLSERS

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	DSPACCES	ACCESS LIST
0	(0)	UNSIGNED	2	DSPACT	NUMBER OF ENTRIES
2	(2)	CHARACTER	9	DSPACCLE (*)	ACCESS LIST ENTRIES
2	(2)	CHARACTER	8	DSPAUSER	USERID/GROUPNAME
10	(A)	BITSTRING	1	DSPACS	ACCESS AUTHORITY

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	DSPINSTD	INSTALLATION DATA
0	(0)	SIGNED	2	DSPLINST	LENGTH OF INSTALLATION DATA
2	(2)	CHARACTER	*	DSPIDATA	INSTALLATION DATA

## RRPF

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	DSPDPTD	CATEGORY LIST
0	(0)	SIGNED	2	DSPDPTCT	NUMBER OF CATEGORIES
2	(2)	SIGNED	2	DSPDEPT (*)	CATEGORY LIST

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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	Entry structure
4	(4)	CHARACTER	8	DSP2ENT	Program Name / Flags
4	(4)	CHARACTER	1	DSPPGFLG	Flag byte
5	(5)	CHARACTER	7	DSPA2RST	The rest of name or flags
12	(C)	CHARACTER	8	DSP2USR	User ID / Group Name
20	(14)	BITSTRING	1	DSP2ACS	Access authority
21	(15)	UNSIGNED	2	DSP2GACS	Access Count
23	(17)	UNSIGNED	1	DSP2GVRL	Variable entity length
24	(18)	CHARACTER	*	DSP2GVAR	Variable entity information
24	(18)	CHARACTER	8	DSP2CLID	Class ID
32	(20)	CHARACTER	2	DSP2RSVD	Reserved
34	(22)	UNSIGNED	1	DSP2VENL	Variable Length
35	(23)	CHARACTER	*	DSP2VENT	Variable Entity

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	DSPBUF	Resource name in extended format
0	(0)	CHARACTER	2	DSPDLEN	Character form of resource name length
0	(0)	SIGNED	2	DSPDSNML	Resource name length
2	(2)	CHARACTER	*	DSPDSNME	Resource name

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
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.

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	CHARACTER	16	DSPIPLOK	IP lookup value for SERVAUTH class profiles

## Constants

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.

## Cross Reference

Name	Hex Offset	Hex Value	Level	Name	Hex Offset	Hex Value	Level
DSPACCES	0		1	DSP2ACCS	0		1
DSPACCLE	2		2	DSP2ACS	14		3
DSPACCOF	38		4	DSP2CLID	18		3
DSPACS	A		3	DSP2ENT	4		3
DSPACT	0		2	DSP2GACS	15		3
DSPAPLDT	2		2	DSP2GCT	0		2
DSPAPOFF	84		4	DSP2GLN	2		2
DSPAPPL	0		1	DSP2GVAR	18		2
DSPAPPLN	0		2	DSP2GVRL	17		3
DSPAPPOF	84		5	DSP2RSVD	20		3
DSPIPOFF	88		4	DSP2USR	C		3
DSPAUDIT	31		4	DSP2VENL	22		3
DSPAUSER	2		3	DSP2VENT	23		3
DSPA2RST	5		4	RRPF	0		1
DSPBUF	0		1	RRPLEN	1		3
DSPCLASS	3C		4	RRPSP	0		3
DSPCORE	0		2	RRPVDATA	4		2
DSPDEPT	2		2				
DSPDLEN	0		2				
DSPDPTCT	0		2				
DSPDPTD	0		1				
DSPDPTOF	64		4				
DSPDSNBF	80		4				
DSPDSNM	4		4				
DSPDSNME	2		2				
DSPDSNML	0		3				
DSPDSNOF	80		5				
DSPEOS	47		4				
DSPFNF	50		4				
DSPGAUD	44		4				
DSPIDATA	2		2				
DSPINST	48		4				
DSPINSTD	0		1				
DSPIPLOK	0		2				
DSPLEVEL	33		4				
DSPLINST	0		2				
DSPLOGFT	71		4				
DSPLOGNT	6E		4				
DSPMDL	32	40	5				
DSPNEXTP	4C		4				
DSPNOTFY	5C		4				
DSPOWNER	54		4				
DSPPGFLG	4		4				
DSPPGMOF	68		4				
DSPRESF	6C		4				
DSPRTPD	52		4				
DSPSLABL	78		4				
DSPSLVL	51		4				
DSPSUB	4		3				
DSPTAPE	32	20	5				
DSPTDAYS	6D		4				
DSPTP	32	80	5				
DSPTYPE	32		4				
DSPTZONE	74		4				
DSPUACC	30		4				
DSPVOLCT	0		2				
DSPVOLOF	34		4				
DSPVOLS	0		1				
DSPVOLSR	2		2				
DSPVRSN	45		4				
DSPWARN	46		4				
DSP2ACCL	4		2				

---

**RSMXP: RACF Report Writer Selection Exit Parameter List**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	ADDRESS	4	RSMCALLR	Reason code address: points to a fullword containing the reason for this call: X'00000000' - Prerecord selection X'00000004' - Postrecord selection X'00000008' - End-of-file
		.... .... .... .1.. .... 1...		RSMPRES RSMPOSTS RSMEOF	
4	(4)	ADDRESS	4	RSMMDSTRA	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 selects or rejects 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 <i>z/OS Security Server RACF System Programmer's Guide</i> .
		.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.



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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.

## Cross Reference

Name	Hex Offset	Hex Value	Level
RSMCALLR	0		2
RSMCOMM	14		2
RSMSTRA	4		2
RSMEOF	0	8	2
RSMNONRS	8	40	2
RSMPOSTS	0	4	2
RSMPRDCB	10		2
RSMPRES	0	0	2
RSMRCD	C		2
RSMREFRS	8	0	2
RSMRJECT	8	80	2
RSMSLECT	8	0	2
RSMSTCH	8		2

**RUTKN: Resource/User Security Token**

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	80	TOKEN	
0	(0)	UNSIGNED	1	TOKLEN	UTOKEN / RTOKEN LENGTH
1	(1)	UNSIGNED	1	TOKVERS	UTOKEN / RTOKEN VERSION #
2	(2)	BITSTRING	1	TOKFLG1	MISCELLANEOUS FLAGS
		1... ....		TOKENCR	TOKEN IS ENCRYPTED
		.1.. ....		*	RESERVED
		..1. ....		TOKLT19	TOKEN CREATED BY PRE 1.9 RACF CALL
		...1 ....		TOKVXPRP	VERIFYX PROPAGATION FLAG
		.... 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 COMP BASE
		.... .1..		TOKSUS	SURROGATE USERID
		.... ..1.		TOKREMOT	REMOTE JOB INDICATOR
		.... ...1		TOKPRIV	PRIVILEGED 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 FOR EXPANSION
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 GROUPNAME

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
48	(30)	CHARACTER	8	TOKPOE	Port of entry(CONS ID,TERM ID-first half of IP value for SERVAUTH)
56	(38)	CHARACTER	8	TOKNETW	Remote network name, if TOKNETF is on. Second half of IP value if TOKIPV is on.
64	(40)	CHARACTER	8	TOKUSER	USERID
72	(48)	CHARACTER	8	TOKGRUP	GROUPNAME

## Constants

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
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

Name	Hex Offset	Hex Value	Level
TOKDFLT	4	80	3
TOKDGRP	6	80	3
TOKDSEC	6	40	3
TOKEN	0		1
TOKENCR	2	80	3
TOKERR	4	10	3
TOKFLG1	2		2
TOKFLG2	4		2
TOKFLG3	6		2
TOKGRUP	48		2
TOKIPV	6	10	3
TOKLEN	0		2
TOKLOGU	2	04	3
TOKLT19	2	20	3
TOKNETF	6	20	3
TOKNETW	38		2
TOKPOE	30		2
TOKPOEX	5		2
TOKPRIV	4	01	3
TOKREMOT	4	02	3
TOKRSPEC	2	02	3
TOKSCL	8		2
TOKSGRP	28		2
TOKSNOD	20		2
TOKSTYP	3		2
TOKSUS	4	04	3
TOKSUSR	18		2
TOKTRST	4	08	3
TOKUDUS	4	40	3
TOKUNUSR	2	08	3
TOKUSER	40		2
TOKVERS	1		2
TOKVXPRP	2	10	3
TOKWDWN	6	8	3
TOKXNOD	10		2

## RXTL: RACROUTE REQUEST=EXTRACT Parameter List (Request Section)

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	12	EXTLIST	EXTRACT parameter 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
5	(5)	BITSTRING	1	EXTFLAGS	Flag byte
		1... ..		EXTBRNCH	Branch entry requested
		.1.. ..		EXTENX	0 => ENTITY is specified and 1 => ENTITYX is specified
		..1. ....		EXTPROP	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 parameter

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
12	(C)	STRUCTURE	12	EXTEXT	TYPE=EXTRACT parameter 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		EXTEND1	End of fixed part of parameter

# RXTL

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
24	(18)	STRUCTURE	20	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)	CHARACTER		EXTENDX	End of fixed part of parameter

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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
16	(10)	SIGNED	4	EXTMETH	Encryption method: 1 = RACF DES method 2 = RACF hashing method 3 = Installation defined method 4 = NBS DES method

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
12	(C)	STRUCTURE	8	EXTENVX	TYPE=ENVRXTR parameter list
12	(C)	ADDRESS	4	EXTENVO	Address of data structure to retrieve the security environment
16	(10)	ADDRESS	4	EXTACE2	Address of ACEE

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	SEGDATS	Map SEGDATA parameter
0	(0)	SIGNED	4	SEGFLN	Size of data
4	(4)	CHARACTER	*	SEGFDTA	Segment data

## Constants

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
EXTACEE	20		2
EXTACE2	10		2
EXTBRNCH	5	80	3
EXTCLAS	C		2
EXTDATA	C		2
EXTDRV	28	02	3
EXTENB	18		1
EXTENC	C		1
EXTEND	C		2
EXTENDX	2C		2
EXTEND1	18		2
EXTENT	8		2
EXTENTX	8		3
EXTENVO	C		2
EXTENVX	C		1
EXTENX	5	40	3
EXTEXT	C		1
EXTFLAC	28	01	3
EXTFLAGS	5		2
EXTFLDS	14		2
EXTFUNCT	2		2
EXTGEN	28	04	3
EXTLEN	0		2
EXTLIST	0		1
EXTMATCH	28	08	3
EXTMETH	10		2
EXTOFF	6		2
EXTPROP	5	20	3
EXTRES1	28		3
EXTSEGDT	1C		2
EXTSEGM	18		2
EXTSP	10		2
EXTSPR	28		2
EXTTYPE	3		2
EXTVER	4		2
EXTVOL	24		2
FIELDS	0		1
FLDCNT	0		2
FLDNAME	4		2
SEGDATS	0		1
SEGFDTA	4		2
SEGFLEN	0		2

## RXTW: RACROUTE REQUEST=EXTRACT Result Area Mapping

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	72	EXTWKEA	
0	(0)	SIGNED	4	EXTWPLEN	
0	(0)	UNSIGNED	1	EXTWSP	Area subpool
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 bits
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 current connect group
40	(28)	CHARACTER	32	*	Reserved
72	(48)	CHARACTER		EXTWEND	End of fixed part



Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	EXTWANM	WORKATTR - User name
0	(0)	SIGNED	4	EXTWNMLN	Length of user name
4	(4)	CHARACTER	*	EXTWNAME	User name for SYSOUT

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	EXTWABG	WORKATTR - Building name
0	(0)	SIGNED	4	EXTWBDLN	Length of building name
4	(4)	CHARACTER	*	EXTWBLDG	Building name for delivery

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	EXTWADP	WORKATTR - Department name
0	(0)	SIGNED	4	EXTWDTLN	Length of department name
4	(4)	CHARACTER	*	EXTWDEPT	Department name for delivery

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	EXTWARM	WORKATTR - Room name
0	(0)	SIGNED	4	EXTWRMLN	Length of room name
4	(4)	CHARACTER	*	EXTWROOM	Room for delivery

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	*	EXTWAS4	WORKATTR - SYSOUT line 4

## RXTW

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	SIGNED	4	EXTWS4LN	Length of SYSOUT line 4
4	(4)	CHARACTER	*	EXTWSYS4	SYSOUT delivery line 4

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	*	EXTWAAC	WORKATTR - Account number
0	(0)	SIGNED	4	EXTWATLN	Length of account number
4	(4)	CHARACTER	*	EXTWACCT	Account number

## Cross Reference

Name	Hex Offset	Hex Value	Level
EXTFLAG	6		2
EXTGENRC	6	80	3
EXTWAAC	0		1
EXTWABG	0		1
EXTWACCT	4		2
EXTWADP	0		1
EXTWANM	0		1
EXTWAOFF	8		2
EXTWARM	0		1
EXTWAS1	0		1
EXTWAS2	0		1
EXTWAS3	0		1
EXTWAS4	0		1
EXTWATLN	0		2
EXTWBDLN	0		2
EXTWBLDG	4		2
EXTWDEPT	4		2
EXTWDTLN	0		2
EXTWEND	48		2
EXTWGRP	20		2
EXTWKEA	0		1
EXTWLN	1		3
EXTWNAME	4		2
EXTWNMLN	0		2
EXTWOFF	4		2
EXTWOPT	48		1
EXTWPLEN	0		2
EXTWPRLN	10		2
EXTWPRUS	16		2
EXTWPSWD	48		2
EXTWRMLN	0		2
EXTWROOM	4		2
EXTWRTAD	C		2
EXTWRTAS	A		2
EXTWSELN	13		2
EXTWSEUS	17		2
EXTWSP	0		3
EXTWSYS1	4		2
EXTWSYS2	4		2
EXTWSYS3	4		2
EXTWSYS4	4		2
EXTWS1LN	0		2
EXTWS2LN	0		2
EXTWS3LN	0		2
EXTWS4LN	0		2
EXTWUID	18		2
EXTWVERS	7		2

## SAFP: SAF Router Parameter List

**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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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)
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 is 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, contains the address of the message(s) returned from the invoked function (if SAFPMSGR is on), or zero if no message is returned
32	(20)	ADDRESS	4	*	Reserved
36	(24)	SIGNED	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
60	(3C)	ADDRESS	4	SAFPECB1	Address of ECB1
64	(40)	ADDRESS	4	SAFPECB2	Address of ECB1
68	(44)	ADDRESS	4	SAFPREV	Address of previous flat list

## SAFP

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
72	(48)	ADDRESS	4	SAFPNEXT	Address of next flat list
76	(4C)	ADDRESS	4	SAFPORIG	Address of original list
80	(50)	SIGNED	4	SAFPFLEN	Flat parameter list length
84	(54)	SIGNED	4	SAFPUSRW	User Word - Identifier
88	(58)	ADDRESS	4	SAFPFREE	Address of pre-processing exit
92	(5C)	ADDRESS	4	SAFPPOST	Address of post-processing exit
96	(60)	ADDRESS	4	SAFPSYNC	Address of synchronous ECB
100	(64)	UNSIGNED	1	SAFPSKEY	Requestor's storage key
101	(65)	UNSIGNED	1	SAFPMODE	Requestor's 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		*	

## Constants

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	SAFPVKMP	RACKSRV - security token services
1	DECIMAL	9	SAFPVERX	RACROUTE REQUEST=VERIFYX
1	DECIMAL	10	SAFPKXT	RACKSRV - 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	SAFPXMP	Token map request for XMREQ=YES is specified.
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

Len	Type	Value	Name	Description
1	DECIMAL	12	SAFPVXEF	ESTAE failed
1	DECIMAL	16	SAFPVXUT	TOKNOUT keyword missing - 9C7 SAF abend
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
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	21	SAFPCURR	Current level of RACF
1	DECIMAL	0	SAFPRL19	Indicates RACF 1.9.0
1	DECIMAL	2	SAFPRL92	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

# SAFP

## Cross Reference

Name	Hex Offset	Hex Value	Level
SAFP	0		1
SAFPDCPL	E	10	3
SAFPECB1	3C		2
SAFPECB2	40		2
SAFPFLAT	38		2
SAFPFLEN	50		2
SAFPFLGS	E		2
SAFPGCS	66	80	3
SAFPMODE	65		2
SAFPMSAD	1C		2
SAFPMSGR	E	80	3
SAFPMSPL	F		2
SAFPNEXT	48		2
SAFPOLEN	32		2
SAFPORIG	4C		2
SAFPPLN	8		2
SAFPPLNX	30		2
SAFPPOST	5C		2
SAFPPREE	58		2
SAFPPREV	44		2
SAFPRACP	24		2
SAFPREQR	10		2
SAFPREQT	C		2
SAFPRETD	34		2
SAFPRREA	4		2
SAFPRRET	0		2
SAFPR18	E	40	3
SAFPSBYT	66		2
SAFPSFRC	28		2
SAFPSFRS	2C		2
SAFPSFSU	66	40	3
SAFPSKEY	64		2
SAFPSUBS	14		2
SAFPSUPP	E	20	3
SAFPSYNC	60		2
SAFPSYST	E	08	3
SAFPUSRW	54		2
SAFPVRRL	A		2
SAFPWA	18		2

---

## SAFR: Number of z/OS UNIX System Services Callable Services

**Common Name:** Number of z/OS UNIX 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

Offsets			Len		
Dec	Hex	Type		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.

### Constants

Len	Type	Value	Name	Description
2	DECIMAL	4	SAFR#LEN	Length of SAFR

## SAFV: SAF Router Vector Table

NOT Programming Interface Information

The following fields are not programming interface information:

- SAFVIDEN
- SAFVVRSN
- SAFVEXIT
- SAFVSAFR
- SAFVSECT

End of NOT Programming Interface 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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
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 - '00'X
5	(5)	CHARACTER	3	*	Reserved
8	(8)	ADDRESS	4	SAFVEXIT	Address of the SAF router exit routine (ICHRTX00)
12	(C)	ADDRESS	4	SAFVSAFR	Address of the SAF router routine (ICHFR00)
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 UNIX System Services (IRRRFR10)
32	(20)	ADDRESS	4	SAFVXIT2	Address of the SAF router exit routine for UNIX System Services (IRRSXT00)
36	(24)	ADDRESS	4		Reserved (alignment).
40	(28)	ADDRESS	4	SAFVXITX	Address of the 64-bit SAF router exit routine (IRRSXTOX).
44	(2C)	ADDRESS	4	SAFVZXIT	Address of the SAF router exit routine for PD (IRRSZT00)



Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
48	(30)	ADDRESS	4	SAFVZCVT	Address of the PD communication vector table
52	(34)	SIGNED	4	* (3)	Reserved
64	(3C)	CHARACTER		*	Ensure doubleword length

## Constants

Len	Type	Value	Name	Description
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 SAFVVRN
1	DECIMAL	0	SAFVVN0	Original value of SAF version number stored in SAFVVRN
1	DECIMAL	1	SAFVVN1	Value of SAF version number for UNIX System Services stored in SAFVVRN
1	DECIMAL	2	SAFVVN2	Value of SAF version number for PD stored in SAFVVRN

## Cross Reference

Name	Hex Offset	Hex Value	Level
SAFV	0		1
SAFVEXIT	8		2
SAFVEXUS	18		2
SAFVIDEN	0		2
SAFVRACR	10		2
SAFVRACT	14		2
SAFVRAC2	1C		2
SAFVSAFR	C		2
SAFVVRN	4		2
SAFVXIT2	20		2
SAFVXITX	28		2
SAFVSECT	24		2
SAFVZCVT	30		2
SAFVZPDR	28		2
SAFVZXIT	2C		2

---

## SECUR: OS/390 Security Context

**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

Offsets			Len		
Dec	Hex	Type		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

Len	Type	Value	Name	Description
32	CHARACTER	See note.	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

**Note:** The value for SECUR\_Context\_Key is IRR\_SECURITY\_CONTEXT.

## SGNPL: RACROUTE REQUEST=SIGNON Parameter List (Request Section)

**Common Name:** Request-specific portion of the RACROUTE REQUEST=SIGNON parameter list

**Macro ID:** IRRSGNPL

**DSECT Name:** SIGENV D, 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 SAFR plus offset in SAFPRACP

**Serialization:** None

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

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	48	SIGPLIST	
0	(0)	SIGNED	2	SIGPLEN	SIGNON parameter list length
2	(2)	UNSIGNED	1	SIGTYPE	Subfunction Type
3	(3)	UNSIGNED	1	SIGLTYPE	List Type
4	(4)	ADDRESS	4	SIGAPPLP	Address of APPL name buffer
8	(8)	ADDRESS	4	SIGPOEP	Address of POE name buffer
12	(C)	ADDRESS	4	SIGACEPP	Address of ACEE
16	(10)	ADDRESS	4	SIGGRUPP	Address of GROUP name buffer
20	(14)	ADDRESS	4	SIGSECLP	Address of SECLABEL buffer
24	(18)	ADDRESS	4	SIGENVRO	Address of output ENVR data
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
40	(28)	ADDRESS	4	SIGENVRI	Address of input ENVR data
44	(2C)	ADDRESS	4	SIGPOENP	Address of POENET name buffer, 1 byte length plus 1-8 byte network name, or zero if not specified.

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	14	SIGENV D	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	SIGENSPP	ENVR OBJECT STORAGE AREA SUBPOOL
13	(D)	UNSIGNED	1	SIGENSKY	ENVR OBJECT STORAGE AREA KEY

## SGNPL

### Cross Reference

Name	Hex Offset	Hex Value	Level
SIGACEPP	C		2
SIGAPPLP	4		2
SIGENLEN	0		2
SIGENSAD	8		2
SIGENSKY	D		2
SIGENSLN	4		2
SIGENSSP	C		2
SIGENV D	0		1
SIGENVRI	28		2
SIGENVRO	18		2
SIGGRUPP	10		2
SIGLTYPE	3		2
SIGOTKP	1C		2
SIGPLIST	0		1
SIGPLEN	0		2
SIGPOENP	2C		2
SIGPOEP	8		2
SIGSECLP	14		2
SIGTYPE	2		2
SIGUSRDP	20		2
SIGVEXIT	24		2

**SGX1P: RACROUTE REQUEST=SIGNON Parameter List Mapping**

**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

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
0	(0)	STRUCTURE	20	SNGFP	
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
16	(10)	ADDRESS	4	SNGFGDVA_ADDR	ADDRESS OF GDS VARIABLE
		1... ....		SNGF_LAST_PARM	END OF PARM LIST INDICATOR

**Constants**

Len	Type	Value	Name	Description
4	DECIMAL	20	SNGFPLEN	

---

**STAT: RACROUTE REQUEST=STAT Parameter List (Request Section)**

**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

Offsets			Len		
Dec	Hex	Type		Name (Dim)	Description
0	(0)	STRUCTURE	24	STATPARM	STAT parameters.
0	(0)	ADDRESS	4	STATCLAS	Address of class entry
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
12	(C)	STRUCTURE	0	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	0	STATEXT3	RELEASE=7709 extension
20	(14)	ADDRESS	4	STATNEXT	Address of the class name for the NEXT=keyword

**Constants**

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

## Cross Reference

Name	Hex Offset	Hex Value	Level
STATCDTP	4		2
STATCLAS	0		2
STATCPYA	C		3
STATCPYL	10		3
STATEXT1	8		2
STATEXT2	C		2
STATEXT3	14		2
STATLEN	8		3
STATNEXT	14		3
STATPARM	0		1

**TSRV: RACROUTE REQUEST=TOKENMAP/TOKENXTR Parameter List (Request Section)**

**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 SAFR plus offset in SAFPRACP

**Serialization:** None

**Function:** Maps the request-specific portion of the parameter list passed to the RACROUTE REQUEST=TOKENMAP/TOKENXTR routine

Offsets		Type	Len	Name (Dim)	Description
Dec	Hex				
0	(0)	STRUCTURE	24	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
		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

**Cross Reference**

Name	Hex Offset	Hex Value	Level
TSRVACEE	4		2
TSRVFLGS	C		2
TSRVFMT	C	80	3
TSRVLEN	E		2
TSRVPARM	0		1
TSRVTKIN	0		2
TSRVTKOT	8		2



## WORK: z/OS UNIX System Services Work Area for SAF and RACF

**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

Offsets			Len	Name (Dim)	Description
Dec	Hex	Type			
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 nonzero (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.
The following 944 bytes are used by both RACF and the User Exit					
16	(10)	CHARACTER	944	WORK_RACF_WORKA	RACF work area.
16	(10)	CHARACTER	136	WORK_USER_WORKA	User work area.
152	(98)	CHARACTER	808	*	
The following 64 bytes are reserved for the SAF Router for future use.					
960	(3C0)	CHARACTER	64	*	Reserved for future use.

### Constants

Len	Type	Value	Name	Description
4	DECIMAL	1024	WORK_LEN	Constant WORK length

## Cross Reference

Name	Hex Offset	Hex Value	Level
WORK	0		1
WORK_RACF_WORKA	10		2
WORK_USER_PERM	0		2
WORK_USER_SIGNAL	0		3
WORK_USER_WORKA	10		3

---

## Appendix A. Notices

This information was developed for products and services offered in the USA.

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 give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
USA

For license inquiries regarding double-byte (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.  
1623-14, Shimotsuruma, Yamato-shi  
Kanagawa 242-8502 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 DOCUMENTATION "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 documentation. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this documentation at any time without notice.

Any references in this information to non-IBM web sites are provided for convenience only and do not in any manner serve as an endorsement of those web sites. The materials at those web sites are not part of the materials for this IBM product and use of those web sites is at your own risk.

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 want 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:

Site Counsel  
IBM Corporation  
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 information 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.

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.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

This product contains code licensed from RSA Data Security Incorporated.

---

## Programming interface information

This primarily documents intended Programming Interfaces that allow the customer to write programs to obtain the services of Security Server for z/OS RACF.

This also documents information that is NOT intended to be used as Programming Interfaces of Security Server for z/OS RACF.

This information is identified where it occurs, either by an introductory statement to a chapter or section or by the following marking:

NOT Programming Interface information

Data Area Name

End of NOT Programming Interface information

---

## Policy for unsupported hardware

Various z/OS elements, such as DFSMS, HCD, 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.

---

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.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

<http://www.ibm.com/legal/copytrade.shtml>

Lotus and Lotus Notes are trademarks of Lotus Development Corporation in the United States, or other countries, or both.

Tivoli is a trademark of Tivoli Systems, Inc. in the United States, or other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

---

# Index

## Numerics

64-bit Storage profile list mapping 129

## A

accessor  
  authorities 3  
  environment element 3  
ACEE  
  data area 3  
  DSECT 3  
  expansion/compression exit 12  
  work attributes data  
    extracted from 230  
ACEEUSRI user ID 21  
ACHKL data area 9  
ACHKLIST DSECT 9  
ACXP data area 12  
address space  
  authorities of accessor in 3  
ADDSD command 27  
administration  
  classroom courses vii  
AFC data area 14  
alphabetic range of profiles 217  
ALTDSD command 27  
ALTUSER command 156  
area  
  result  
    RACROUTE REQUEST=EXTRACT 230  
attachment interface  
  CICS/ESA 19  
audit data 94  
audit function codes  
  z/OS UNIX System Services 14  
AUDLIST DSECT 17  
AUL data area 17  
authorities  
  of accessor in address space 3  
authorization checking

## C

callable services  
  initUSP (IRRSIU00) 152  
  makeISP (IRRSMI00) 145  
  RACF 92, 136, 140  
  routers  
    RACF 30, 75  
    SAF 30, 75  
  SAF 92, 136, 140  
  z/OS UNIX System Services  
    number of 237

CCXP data area 18  
CCXPL DSECT 18  
CDXP data area 19  
CDXPLIST DSECT 19  
CGRP  
  data area 21  
  DSECT 21  
CGRPENTD DSECT 21  
checking  
  authorization 218  
CICS/ESA 19  
class name and syntax table 23, 26  
classes  
  general resource 23, 26  
  resource 23, 26  
classroom courses, RACF vii  
CNST  
  data area 23, 26  
  DSECT 23, 26  
CNSX  
  data area 23, 26  
  DSECT 23, 26  
CNXP data area 27  
CNXPL DSECT 27  
command  
  exit 112  
  preprocessing  
    ICHCCX00 exit 18  
  processor envelope 112  
  parameter list 112  
command preprocessing exit  
  ICHCCX00 18  
  ICHCNX00 27  
commands  
  ADDSD 27  
  ALTDSD 27  
  ALTUSER 156  
  DELDSD 27  
  DELGROUP 18  
  DELUSER 18  
  LISTDSD 27  
  PASSWORD 156  
  PERMIT 27  
  REMOVE 18  
  SEARCH 27  
common SAF callable services 92, 136, 140  
common SAF/RACF parameter list 30, 75  
communication  
  area 161  
  vector table 161  
COMP data area 30

compression/expansion exit (ACEE) 12  
 COMX data area 75  
 COMY data area 92  
 connect group name table 21  
 constants  
     for z/OS UNIX System Services audit function codes 14  
 context, security 240  
 control blocks  
     ICB  
         See z/OS Security Server RACF Diagnosis Guide  
 inventory  
     See z/OS Security Server RACF Diagnosis Guide  
 courses on RACF vii  
 CRED  
     data area 94  
     DSECT 94  
 credential structure 94, 97  
 CREI  
     data area 97  
     DSECT 97

## D

data  
     audit 94  
     passed to SAF router  
         descriptor for 233  
     work attributes  
         extracted from ACEE 230  
 data area map vi  
 data areas  
     ACEE 3  
     ACHKL 9  
     ACXP 12  
     AFC 14  
     AUL 17  
     CCXP 18  
     CDXP 19  
     CGRP 21  
     CNST 23, 26  
     CNSX 23, 26  
     CNXP 27  
     COMP 30, 75  
     CRED 94  
     CREI 97  
     DAUT 99  
     DEXP 101  
     ENF2 111  
     ENFP 110  
     EVXP 112  
     FACL 115  
     FAST 118  
     FC 120  
     FXAP 122  
     GAPL 126  
     GRPF 132

data areas (*continued*)  
     IFSP 143  
     IISP 145  
     IRRGPRFL 129  
     IRRPGANC 124  
     ISP 146  
     Ousp 152  
     PWXP 156  
     RCVT 161  
     RCXP 171  
     RDDFL 178  
     RDXP 181  
     RFXP 189  
     RIPL 192  
     RIXP 198  
     RLST 206  
     RLX1P 208  
     RLX2P 213  
     RNG 217  
     RRPF 218  
     RSMXP 222  
     RUTKN 224  
     RXTL 227  
     RXTW 230  
     SAFP 233  
     SAFR 237  
     SAFV 238  
     SECUR 240  
     SGNPL 241  
     SGX1P 243  
     STAT 244  
     TSRV 246  
     WORK 247  
 data encryption exits 101  
 data segments  
     in RACF database 19  
 data set  
     backup 103  
     descriptor table 103  
     high-level qualifier 126, 129  
     primary 103  
 database  
     inventory control block (ICB)  
         See z/OS Security Server RACF Diagnosis Guide  
     RACF  
         data segments 19  
         range table  
             RACF 217  
     database range table 217  
     DAUT data area 99  
 defaults  
     RACF 217  
 definitions  
     connect group name table 21  
 DELDSD command 27

DELGROUP command 18  
 DELUSER command 18  
 descriptor  
     for data passed to SAF router 233  
 DEXP data area 101  
 DEXPL DSECT  
 distributed identity support 138  
 documents  
     softcopy vii  
 DSDT  
     data area 103  
     DSECT 103  
 DSECTs  
     ACEE 3  
     ACHKLIST 9  
     ADMN 30  
     AUDLIST 17  
     CACH 30  
     CAUD 30  
     CCXPL 18  
     CDDL 30  
     CDDLX 75  
     CDXPLIST 19  
     CGRP 21  
     CGRPENTD 21  
     CHKP 30  
     CKO2 30  
     CKPO 30  
     CLID 30  
     CMOD 30  
     CNST 23, 26  
     CNSX 23, 26  
     CNXPL 27  
     COMP 30  
     COMX 75  
     COMY 92  
     COWN 30  
     CRED 94  
     CREI 97  
     CSID 30  
     DAUT 30  
     DEXPL 101  
     DINF 30  
     DKEY 30  
     DRUR 30  
     DSDT 103  
     DSP2ACCS 218  
     DSPACCES 218  
     DSPDPTD 218  
     DSPINSTD 218  
     DSPVOLS 218  
     EVXPL 112  
     EXID 30  
     EXTWAAC 230  
     EXTWABG 230  
     EXTWADP 230

DSECTs (*continued*)

EXTWANM 230  
 EXTWARM 230  
 EXTWAS1 230  
 EXTWAS2 230  
 EXTWAS3 230  
 EXTWAS4 230  
 EXTWKEA 230  
 FACL 115  
 FORK 30  
 GENANC\_JOBNAME\_DATA 124  
 GENANC\_SETTINGS 124  
 GENATE 126  
 GENPLE2 129  
 GENPLEL 126  
 GENPRFL 126  
 GETE 30  
 GETG 30  
 GINF 30  
 GMAP 30  
 GPRFL64 129  
 GRPF 132  
 GSEC 75  
 GUGP 30  
 IACC 30  
 ICRX 136  
 ICTX 138  
 IDID 140  
 IFSP 143  
 IISP 145  
 INITPARM 192  
 INTA 30  
 INVICB  
     See z/OS Security Server RACF Diagnosis Guide  
 IOWN 30  
 IRR\_SECURITY\_CONTEXT 240  
 IUSP 30  
 KACC 30  
 KERB 30  
 KFOW 30  
 MFSP 30  
 MISP 30  
 MKRT 30  
 OUSP 152  
 PGSN 30  
 PKIS 30  
 PRXY 30  
 PTRC 30  
 PWX2L 154  
 PWXPL 156  
 QRYF 30  
 QRYS 30  
 RACL 30  
 RACRNE 146  
 RACRPE 146  
 RACRSE 146

## DSECTs (continued)

RACRTE 146  
RAUD 30  
RAUX 30, 92  
RCVI 159  
RCVT 161  
RCXPL 171  
RDDFLIST 178  
RDXPL 181  
RFXAPL 122  
RFXPL 189  
RIXPL 198  
RLSTPARM 206  
RLX1PL 208  
RLX2PL 213  
RPEACCLE 146  
RPEACL2 146  
RPEAPPL 146  
RPEINST 146  
RPEMEM 146  
RPEPTD 146  
RPESESS2 146  
RPESESSN 146  
RRPF 218  
RSMXPL 222  
SAFP 233  
SAFR 237  
SAFV 238  
SECL 30  
SIGENVD 241  
SIGPLIST 241  
TKTS 30  
TOKEN 224  
UMAP 30  
UMSK 30  
WORK 247  
WPRV 30  
DSP2ACCS DSECT 218  
DSPACCES DSECT 218  
DSPDPTD DSECT 218  
DSPINSTD DSECT 218  
DSPVOLS DSECT 218  
dynamic parsing 19

## E

encryption, data 101  
ENF2 data area 111  
ENFP data area 110  
EVXP data area 112  
EVXPL DSECT 112

### exits

ICHCCX00 (command preprocessing) 18  
ICHCNX00 (command preprocessing) 27  
ICHDEX01, ICHDEX11 (data encryption) 101  
IHPWX01 (password) 156

## exits (continued)

ICHRCX01 (RACROUTE REQUEST=AUTH) 171  
ICHRCX02 (RACROUTE REQUEST=AUTH) 171  
ICHRDX01 (RACROUTE REQUEST=DEFINE) 181  
ICHRDX02 (RACROUTE REQUEST=DEFINE) 181  
ICHRFX01 (RACROUTE REQUEST=FASTAUTH) 189  
ICHRFX02 (RACROUTE REQUEST=FASTAUTH) 189  
ICHRFX03 (RACROUTE REQUEST=FASTAUTH  
extended function) 122  
ICHRFX04 (RACROUTE REQUEST=FASTAUTH  
extended function) 122  
ICHRIX01 (RACROUTE  
REQUEST=VERIFY/VERIFYX) 198  
ICHRIX02 (RACROUTE  
REQUEST=VERIFY/VERIFYX) 198  
ICHRLX01 (RACROUTE REQUEST=LIST) 208  
ICHRLX02 (RACROUTE REQUEST=LIST selection) 213  
ICHRSME (report writer selection) 222  
IRRACX01 (ACEE expansion/compression) 12  
IRRACX02 (ACEE expansion/compression) 12  
IRREX01 (RACF command exit) 112  
notify/verify/list exit routines 19  
expansion/compression exit (ACEE) 12  
EXTWAAC DSECT 230  
EXTWABG DSECT 230  
EXTWADP DSECT 230  
EXTWANM DSECT 230  
EXTWARM DSECT 230  
EXTWAS1 DSECT 230  
EXTWAS2 DSECT 230  
EXTWAS3 DSECT 230  
EXTWAS4 DSECT 230  
EXTWKEA DSECT 230

## F

FACL  
data area 115  
DSECT 115  
FAST data area 118  
FC data area 120  
file security packet  
mapping 143  
FXAP data area 122

## G

GAPL data area 126  
GENANC\_JOBNAME\_DATA DSECT 124  
GENANC\_SETTINGS DSECT 124  
GENATE DSECT 126  
general resource  
profile  
mapping 218  
profiles 146



- general resource class 126, 129
  - table, RACF 23, 26
- generic anchor table entry 126
- generic profile map 132
- generic profile names 126, 129
- GENERICANCHOR Settings Mapping 124
- GENPLE2 DSECT 129
- GENPLEL DSECT 126
- GENPRFL DSECT 126
- GPRFL64 DSECT 129
- group names
  - ACEEUSRI user ID 21
- groups
  - ACEEUSRI user ID 21
- GRPF
  - data area 132
  - DSECT 132

## H

- hardw 250
- header v
- high-level qualifier
  - data set 126, 129

## I

- ICB (inventory control block)
  - See z/OS Security Server RACF Diagnosis Guide
- ICHACHKL macro 9, 99
- ICHCCX00 exit 18
- ICHCCXP macro 18
- ICHCDXP macro 19
- ICHCNX00 exit 27
- ICHCNXP macro 27
- ICHDEX01, ICHDEX11 exits 101
- ICHDEXP macro 101
- ICHGAPL macro 126
- ICHGLS00 126, 132
- ICHGRP macro 132
- ICHPAUL macro 17
- ICHPCGRP macro 21
- ICHPCNST macro 23
- ICHPDSDT macro 103
- ICHPISP macro 146
- ICHPRCVT macro 161
- ICHPWX01 exit 156
- ICHPWXP macro 154, 156
- ICHRCX01, ICHRCX02 exits 171
- ICHRCXP macro 171
- ICHRDDL macro 178
- ICHRDX01, ICHRDY02 exits 181
- ICHRDXP macro 181
- ICHRFX01, ICHRFY02 exits 189
- ICHRFX03, ICHRFY04 exits 122

- ICHRFXAP macro 122
- ICHRFXP macro 189
- ICHRX01, ICHRX02 exits 198
- ICHRXP macro 198
- ICHLX01 exit 208
- ICHLX02 exit 213
- ICHLX1P macro 208
- ICHLX2P macro 213
- ICHRNG module 217
- ICHRPF macro 218
- ICHRSMFE exit 222
- ICHRSMXP macro 222
- ICHRUTKN macro 175, 203, 224
- ICHSAFP macro 233
- ICHSAFV macro 238
- ICHSEC00 24, 103
- ICHSGX1P macro 243
- ICRX data area 136
- ICTX data area 138
- identity cache communication vendor 159
- IDID data area 140
- IFSP
  - data area 143
  - DSECT 143
- IHAACEE macro 3
- IISP
  - data area 145
  - DSECT 145
- in-storage generic profile 132
- in-storage profile 146
- initialization 161
  - RACF 23, 217
- INITPARM DSECT 192
- initUSP callable service 152
- input
  - SAF callable services 92, 136, 140
  - SAF/RACF parameter list 30, 75
- installation exits
  - postprocessing
    - RACROUTE REQUEST=AUTH 171
    - RACROUTE REQUEST=DEFINE 181
    - RACROUTE REQUEST=FASTAUTH 189
    - RACROUTE REQUEST=FASTAUTH extended function 122
    - RACROUTE REQUEST=LIST 208
    - RACROUTE REQUEST=VERIFY/VERIFYX 198
  - preprocessing
    - command exit (ICHCCX00) 18
    - command exit (ICHCNX00) 27
    - RACROUTE REQUEST=AUTH 171
    - RACROUTE REQUEST=DEFINE 181
    - RACROUTE REQUEST=FASTAUTH 189
    - RACROUTE REQUEST=FASTAUTH extended function 122
    - RACROUTE REQUEST=LIST 208
    - RACROUTE REQUEST=VERIFY/VERIFYX 198

interface  
 attachment  
   CICS/ESA 19  
 inventory control block (ICB)  
   See z/OS Security Server RACF Diagnosis Guide  
 INVICB DSECT  
   See z/OS Security Server RACF Diagnosis Guide  
 IPC  
   security credentials 97  
   security packet 145  
 IRRACX01 exit 12  
 IRRACX02 exit 12  
 IRRACXP macro 12  
 IRREVX01 112  
 IRREVXP macro 112  
 IRRGLS21 129  
 IRRGPRFL data area 129  
 IRRGPRFL macro 129  
 IRRPAFC macro 14  
 IRRPCNST macro 26  
 IRRPCOMP macro 30  
   pointed to by 94, 97, 143, 145, 247  
 IRRPCOMX macro 75  
 IRRPCOMY macro 92  
 IRRPCRED macro 94  
   pointed to by 115  
 IRRPCREI macro 97  
 IRRPEN2 macro 111  
 IRRPENFP macro 110  
 IRRPFACL macro 115  
 IRRPFC macro 120  
 IRRPGANC data area 124  
 IRRPGANC macro 124  
 IRRPICRX macro 136  
 IRRPICTX macro 138  
 IRRPIDID macro 140  
 IRRPIFSP macro 143  
 IRRPIISP macro 145  
 IRRPOUSP macro 152  
 IRRPRCVI macro 159  
 IRRPRIPL macro 192  
 IRRPRLST macro 206  
 IRRPRXTL macro 227  
 IRRPRXTW macro 230  
 IRRPSAFR macro 237  
 IRRPWORK macro 247  
 IRRSECUR macro 240  
 IRRSET03 124  
 IRRSGNPL macro 241  
 IRRSIU00 callable service 152  
 IRRSMI00 callable service 145  
 IRRUT100 utility 27  
 ISP data area 146

## L

list  
 of addresses 101  
   passed to data encryption exits 101  
   passed to ICHCCX00 exit 18  
   passed to password exit 156  
   passed to RACF Command Processor Envelope 112  
   passed to RACROUTE REQUEST=AUTH exits 171  
   passed to RACROUTE REQUEST=DEFINE exits 181  
   passed to RACROUTE REQUEST=FASTAUTH  
     exits 189  
   passed to RACROUTE REQUEST=FASTAUTH  
     extended function exits 122  
   passed to RACROUTE REQUEST=LIST exits 208  
   passed to RACROUTE  
     REQUEST=VERIFY/VERIFYX 198  
   RACROUTE REQUEST=LIST selection exit 213  
   report writer selection exit 222  
 parameter 222  
   command preprocessing exit (ICHCNX00) 27  
   data encryption exits 101  
   for initUSP 152  
   ICHCCX00 command preprocessing exit 18  
   mapping of ACEE expansion/compression exit 12  
   notify/verify/list exit routines 19  
   passed to RACROUTE REQUEST=EXTRACT  
     routine 227  
   passed to RACROUTE REQUEST=STAT routine 244  
   passed to RACROUTE REQUEST=TOKENMAP  
     routine 246  
   passed to RACROUTE REQUEST=TOKENXTR  
     routine 246  
   password exit 156  
   RACF command exit 112  
   RACF/SAF 30, 75  
   RACROUTE REQUEST=AUTH exits 171  
   RACROUTE REQUEST=DEFINE 178  
   RACROUTE REQUEST=DEFINE exits 181  
   RACROUTE REQUEST=FASTAUTH exits 189  
   RACROUTE REQUEST=FASTAUTH extended function  
     exits 122  
   RACROUTE REQUEST=LIST 206  
   RACROUTE REQUEST=LIST exit 208  
   RACROUTE REQUEST=LIST selection exit 213  
   RACROUTE REQUEST=SIGNON 241, 243  
   RACROUTE REQUEST=TOKENBLD 192  
   RACROUTE REQUEST=VERIFY/VERIFYX 192  
   RACROUTE REQUEST=VERIFY/VERIFYX exits 198  
   report writer selection exit 222  
   SAF router 233  
   SAF/RACF 30, 75  
   RACROUTE REQUEST=AUDIT 17  
   RACROUTE REQUEST=AUTH 9  
   RACROUTE REQUEST=DIRAUTH 99  
   RACROUTE REQUEST=FASTAUTH 118

LISTDSD command 27

## M

### macros

ICHACHKL 9  
ICHCCXP 18  
ICHCDXP 19  
ICHCNXP 27  
ICHDEXP 101  
ICHGAPL 126  
ICHGRP 132  
ICHPAUL 17  
ICHPCGRP 21  
ICHPCNST 23  
ICHPDSDT 103  
ICHPICB  
    See z/OS Security Server RACF Diagnosis Guide  
ICHPISP 146  
ICHPRCVT 161  
ICHPWXP 154, 156  
ICHRCXP 171  
ICHRDDFL 178  
ICHRDXP 181  
ICHRFXAP 122  
ICHRFXP 189  
ICHRIXP 198  
ICHLX1P 208  
ICHLX2P 213  
ICHRPF 218  
ICHRSMXP 222  
ICHRUTKN 175, 203, 224  
ICHSAFP 233  
ICHSAFV 238  
ICHSGX1P 243  
IHAACEE 3  
IRRACXP 12  
IRREVXP 112  
IRRGPRFL 129  
IRRP AFC 14  
IRRPCNST 26  
IRRPCOMP 30  
IRRPCOMX 75  
IRRPCOMY 92  
IRRPCRED 94  
IRRPCREI 97  
IRRPFACL 115  
IRRPFC 120  
IRRP GANC 124  
IRRPICRX 136  
IRRP ICTX 138  
IRRP IDID 140  
IRRP IFSP 143  
IRRP IISP 145  
IRRP OUSP 152  
IRRP RCVI 159

### macros (continued)

IRRPRIPL 192  
IRRPRLST 206  
IRRPRTL 227  
IRRPRTW 230  
IRRP SAFR 237  
IRRPWORK 247  
IRRSECUR 240  
IRRS GNPL 241  
RACROUTE  
    data passed to SAF router by 233  
mainframe ix  
    education ix  
makeISP callable service 145  
map  
    generic profile 132  
map, data area vi  
mapping  
    of parameter list  
        for ACEE expansion/compression exit 12  
    of SAF router vector table 238  
    parameter list  
        data encryption exits 101  
        passed to RACROUTE REQUEST=EXTRACT  
            routine 227  
        RACF command exit 112  
        RACROUTE REQUEST=AUTH exits 171  
        RACROUTE REQUEST=DEFINE 178  
        RACROUTE REQUEST=DEFINE exits 181  
        RACROUTE REQUEST=FASTAUTH exits 189  
        RACROUTE REQUEST=LIST 206  
        RACROUTE REQUEST=TOKENBLD 192  
        RACROUTE REQUEST=VERIFY/VERIFYX 192  
    profile  
        general resource 218  
    result area  
        RACROUTE REQUEST=EXTRACT 230  
modules  
    ICHR RNG 217

## N

notify/verify/list exit routines 19

## O

OS/390 security context 240  
OUSH  
    data area 152  
    DSECT 152  
output parameter list for initUSH 152

## P

parameter list  
    exit routines  
        notify/verify/list 19

- parameter list (*continued*)
  - for initUSP 152
  - ICHCCX00 exit 18
  - mapping
    - data encryption exits 101
    - for ACEE expansion/compression exit 12
    - of RACF command exit 112
    - of RACROUTE REQUEST=DEFINE exits 181
    - of RACROUTE REQUEST=FASTAUTH extended function exits 122
    - RACROUTE REQUEST=AUTH exits 171
    - RACROUTE REQUEST=FASTAUTH exits 189
  - of addresses 27
    - passed to ICHCNX00 27
  - passed to ICHCNX00 27
  - passed to RACROUTE REQUEST=
    - TOKENMAP routine 246
    - TOKENXTR routine 246
  - passed to RACROUTE REQUEST=EXTRACT routine 227
  - passed to RACROUTE REQUEST=STAT routine 244
  - RACF/SAF 30, 75
  - RACROUTE
    - REQUEST=LIST exit 208
  - RACROUTE REQUEST=
    - LIST 206
    - TOKENBLD 192
    - VERIFY/VERIFYX 192
  - RACROUTE REQUEST=AUDIT 17
  - RACROUTE REQUEST=AUTH 9
  - RACROUTE REQUEST=DEFINE 178
  - RACROUTE REQUEST=DIRAUTH 99
  - RACROUTE REQUEST=FASTAUTH 118
  - RACROUTE REQUEST=LIST
    - selection exit 213
  - RACROUTE REQUEST=SIGNON 241, 243
  - report writer selection exit
    - installation 222
    - selection 222
  - SAF router 233
  - SAF/RACF 30, 75
- parsing, dynamic 19
- password
  - new
    - exit 156
- PASSWORD command 156
- password exit 156
- password phrase 154
- PERMIT command 27
- processor
  - RACF Command Processor Envelope 112
  - RACROUTE
    - REQUEST=LIST 208, 213
  - SVC
    - RACROUTE REQUEST=VERIFY 156
    - RACXTRT 101

- profile
  - resident
    - mapping of 218
- profiles
  - general resource 146
  - in courage
    - in-storage 146
  - individual 146
  - range of alphabetic 217
- programming interface information 250
- PWX2 data area 154
- PWXP data area 156
- PWXPL DSECT 156

## R

- RACF
  - administration courses vii
  - and SAF
    - z/OS UNIX System Services work area 247
  - and SAF parameter list 30, 75
  - classroom courses vii
  - command exit 112
  - command processor envelope 112
  - communication vector table 161
  - data set
    - backup 103
    - descriptor table 103
    - primary 103
  - database
    - data segments 19
    - range table 217
  - database inventory control block (ICB)
    - See z/OS Security Server RACF Diagnosis Guide
  - defaults 217
  - documents
    - softcopy vii
  - dynamic parsing 19
  - ENF EVENT CODE 62
  - ENF EVENT CODE 71
  - in-storage profile 146
  - initialization 23, 161, 217
  - report writer
    - selection exit 222
  - security tokens
    - resource 224
    - user 224
  - security topics vii
- RACRNE DSECT 146
- RACROUTE
  - macro
    - data passed to SAF router by 233
  - REQUEST=AUDIT
    - request parameter list 17
  - REQUEST=AUTH
    - exit parameter list 171
    - request parameter list 9

RACROUTE (*continued*)

- REQUEST=AUTH (*continued*)
  - resident profile map 218
- REQUEST=DEFINE
  - exit parameter list 181
  - request parameter list 178
- REQUEST=DIRAUTH
  - request parameter list 99
- REQUEST=EXTRACT
  - request parameter list 227
  - result area mapping 230
- REQUEST=FASTAUTH 189
  - exit parameter list 189
  - extended function exit parameter list 122
  - request parameter list 118
- REQUEST=LIST 146
  - exit parameter list 208
  - request parameter list 206
  - selection exit parameter list 213
- REQUEST=SIGNON
  - request parameter list 241
  - verbexit parameter list 243
- REQUEST=STAT 23, 26
  - request parameter list 244
- REQUEST=TOKENBLD
  - request parameter list 192
  - security token maps 224
- REQUEST=TOKENMAP/TOKENXTR
  - request parameter list 246
- REQUEST=VERIFY 3
  - SVC processor 156
- REQUEST=VERIFY/VERIFYX
  - exit parameter list 198
  - request parameter list 192
  - security token maps 224
- RACRPE DSECT 146
- RACRSE DSECT 146
- RACRTE DSECT 146
- RACXTRT SVC processor 101
- range of profiles
  - alphabetic 217
- RCVI data area 159
- RCVT
  - data area 161
  - DSECT 161
- RCXP data area 171
- RCXPL DSECT 171
- RDDFL data area 178
- RDDFLIST DSECT 178
- RDXP data area 181
- RDXPL DSECT 181
- REMOVE command 18
- report writer selection exit 222
- requests
  - passed to RACROUTE REQUEST=
    - DEFINE 178
    - LIST 206
- requests (*continued*)
  - passed to RACROUTE REQUEST= (*continued*)
    - TOKENBLD 192
    - TOKENMAP routine 246
    - TOKENXTR routine 246
    - VERIFY/VERIFYX 192
  - passed to RACROUTE REQUEST=AUDIT 17
  - passed to RACROUTE REQUEST=AUTH 9
  - passed to RACROUTE REQUEST=DIRAUTH 99
  - passed to RACROUTE REQUEST=EXTRACT
    - routine 227
  - passed to RACROUTE REQUEST=FASTAUTH 118
  - passed to RACROUTE REQUEST=SIGNON 241
  - passed to RACROUTE REQUEST=STAT routine 244
  - RACROUTE REQUEST=AUTH 218
- resident profile map 218
- resource
  - general
    - profile 146, 218
- resource class
  - general 126, 129
  - table, RACF 23, 26
- result area
  - mapping
    - RACROUTE REQUEST=EXTRACT 230
- RFXAPL DSECT 122
- RFXP data area 189
- RFXPL DSECT 189
- RIPL data area 192
- RIXP data area 198
- RIXPL DSECT 198
- RLST data area 206
- RLSTPARM DSECT 206
- RLX1P data area 208
- RLX1PL DSECT 208
- RLX2P data area 213
- RLX2PL DSECT 213
- RNG data area 217
- router
  - SAF
    - parameter list 233
    - vector table 238
- routers
  - callable services
    - RACF 30, 75
    - SAF 30, 75
- routines
  - RACROUTE REQUEST=
    - LIST 206
    - TOKENBLD 192
    - TOKENMAP 246
    - TOKENXTR 246
    - VERIFY/VERIFYX 192
  - RACROUTE REQUEST=AUDIT 17
  - RACROUTE REQUEST=AUTH 9
  - RACROUTE REQUEST=DEFINE 178

routines (*continued*)

- RACROUTE REQUEST=DIRAUTH 99
- RACROUTE REQUEST=EXTRACT 227
- RACROUTE REQUEST=FASTAUTH 118
- RACROUTE REQUEST=STAT 244
- RPEACCLE DSECT 146
- RPEACL2 DSECT 146
- RPEAPPL DSECT 146
- RPEINST DSECT 146
- RPEMEM DSECT 146
- RPEPTD DSECT 146
- RPESESS2 DSECT 146
- RPESESSN DSECT 146
- RRPF 218
  - data area 218
  - DSECT 218
- RSMXP data area 222
- RSMXPL DSECT 222
- RUTKN data area 224
- RXTL data area 227
- RXTW data area 230

## S

- SAF
  - and RACF
    - z/OS UNIX System Services work area 247
  - (system authorization facility)
    - router
      - parameter list 233
      - vector table 238
  - callable services 92, 136, 140
  - RACF parameter list 30, 75
- SAFP
  - data area 233
  - DSECT 233
- SAFR
  - data area 237
  - DSECT 237
- SAFV
  - data area 238
  - DSECT 238
- SEARCH command 27
- SECUR
  - data area 240
  - DSECT 240
- security context 240
- security credentials 94
  - for IPC 97
- security function code table 120
- security functions
  - z/OS UNIX System Services 30, 75, 247
- security packet
  - IPC 145
- security tokens
  - resource
    - mapping 224

security tokens (*continued*)

- user
  - mapping 224
- security topics for RACF
  - classroom courses vii
- selection exit
  - parameter list 213
- SGNPL data area 241
- SGX1P data area 243
- SIGENVD DSECT 241
- SIGPLIST DSECT 241
- STAT data area 244
- SVC processor
  - RACROUTE
    - REQUEST=VERIFY 156
  - RACXTRT 101
- system authorization facility (SAF)
  - router
    - parameter list 233
    - vector table 238

## T

- tables
  - ACEEUSRI user ID
    - of group names 21
  - class name and syntax 23, 26
  - communication vector 161
  - connect group name 21
  - data set descriptor 103
  - database range 217
  - in-storage profile 146
  - security function codes 120
  - vector
    - SAF router 238
- TOKEN DSECT 224
- tokens
  - security
    - resource 224
    - user 224
- trademarks 250
- TSRV data area 246

## U

- user IDs
  - ACEEUSRI 21
- utilities
  - IRRUT100 27

## V

- vector table
  - SAF router 238

## W

### WORK

- data area 247
- DSECT 247

### work area

- z/OS UNIX System Services  
for SAF and RACF 247

### work attributes data

- extracted from ACEE 230

## Z

- z/OS Basic Skills informationcenter ix

### z/OS UNIX System Services

- access control list 115

- audit function codes 14

- callable services

  - number of 237

- credential structure 94, 97

- file security packet 143

- security function code table 120

- security functions 30, 75

- work area

  - for SAF and RACF 247



Program Number: 5694-A01



Printed in the United States of America  
on recycled paper containing 10%  
recovered post-consumer fiber.

GA22-7680-13

