

SAFTRACE

**Vanguard's Enterprise Security Expo
2003**

July 13-17, 2003

Orlando Fla

at z/OS Version 1 Release 2 >



**Russ Hardgrove
RACF Level 2 Team Leader
IBM - OS/390 & z/OS Software Service
Poughkeepsie, NY 12601
hardgrov@us.ibm.com**

Agenda

- Overview
 - What is SAFTRACE?
 - Who should use it and how do I use it?
 - examples - RACROUTE - AUTH
- Objectives
 - Understand how to use SAFTRACE
- Example of it IN ACTION !!
- Session Summary
- Sources and Additional Information
- Appendices (A & B)



What is SAFTRACE?

A security product trace

- SAFTRACE provides the ability to trace all Racroutes, RACF Callable services, and RACF Database Manager requests that go through the RACF routers. When tracing these services, the trace routine will copy the parameter list into a GTF record **before and after** the service runs.
(creating a pre and post trace record)
- Available at z/OS VIR2 and above.....

IPCS formatting

- Once collected records are formatted with IPCS exit IRRUSR57 (alias AMDUSR57) making them readable.



Who should use it?

SAFTRACE is a powerful DIAGNOSTIC TOOL targeted for use by IBM Support or customers with a STRONG working knowledge of RACF interfaces.

- **Primarily RACF L2**
- Other L2 teams
- On site support (with direction if required)
- **Customers with "some" RACF internals knowledge (i.e. especially parameter lists)**

How can SAFTRACE be used?

Diagnostic situations where SAFTRACE can be helpful:

- To understand WHAT calls are being made for any given situation.
- If RACF database contention has been observed:
Trace DATABASE(ALTER) requests on the specific ASID indicated via GRS contention displays. Alter requests generally prevent readers (majority) from getting service.
- Excessive database i/o (for a given address space)
Trace reads to see what CLASS / ENTITIES are related.
- Excessive Verify's:
If your systems has an excessive amount of Verify's, set a trace on RACROUTE(TYPE(2,5,9)) and determine who is issuing all of the RACROUTE calls.
- Timings - use a statistical package to calculate time difference between pre to post call for each 'pair' of GTF records. Insure pairings are correct.

Where are the trace points?

Where do we trace?



Trace Points...

IBM SAF routers ICHSFR00 and IRRSFR11

- Internal calls to the security product may not be traced.
- All calls made via RACROUTE or Callable Service interface will be traced.
- Calls that issue SVC (pre-RACROUTE, such as RACINIT, RACDEF, RACHECK, RACLIST, etc.) or directly enter the security product **will not be** traced

RACF Database manager ICHEINTY interface

- All ICHEINTYs and internal security product calls to the database manager.



How Do I use it?

Activating the trace?



Activating a trace...

```
[subsystem-prefix] SET [TRACE(  
    [ RACROUTE(ALL * | NONE | TYPE(t1, t2,...)) |  
        NORACROUTE ] |  
  
    [ DATABASE ([ALL * | NONE] |  
        [ALTER | NOALTER] |  
        [ALTERI | NOALTERI] |  
        [READ | NOREAD]) |  
        NODATABASE ] |  
  
    [ CALLABLE(ALL | NONE | TYPE(t1, t2, ...)) |  
        NOCALLABLE]  
  
    [ ASID(aside1, aside2, .. |*) | NOASID | ALLASIDS ]  
    [ JOBNAME(jobname1, jobname2, ... | *) |  
        NOJOBNAME | ALLJOBNAMES ]  
    )]
```



* ALL is not recommended for RACROUTE or DATABASE as trace output could be **very** large.

Activating a trace

Sample GTF PROC

```
//GTFRACF PROC MEMBER=GTFPRM#O
//BR14 EXEC PGM=IEFBR14,REGION=512K
//SYSPRINT DD SYSOUT=*
//D DD DISP=(OLD,DELETE),UNIT=3380,VOL=SER=TEMP01,
// DSN=SYS1.TRACE
//IEFPROC EXEC PGM=AHLGTF,PARM='MODE=EXT,DEBUG=NO,SA=100K,AB=100K',
// REGION=2880K,TIME=NOLIMIT
//IEFRDER DD DSN=SYS1.TRACE,UNIT=3380,VOL=SER=TEMP01,
// DISP=(NEW,CATLG),SPACE=(TRK,(100))
//SYSLIB DD DSN=RACFDRVR.PARMLIB.R6(&MEMBER),DISP=SHR
```

Sample Parmlib Member: GTFPRM#O

```
TRACE=USRP
USR=(F44),END
```



Activating a trace...

1. Start the GTF using the GTFRACF (see Sample PROC 1.) or other procedure:

START GTFRACF.GTF,,NOPROMPT

Noprompt implies that the PROC has what it needs.

2. Use the SET command to enable your trace:

**@SET TRACE(RACROUTE(TYPE(1))
JOBNAME(HARDGR2)) LIST**

3. Reproduce the scenario that trace is required for, e.g.; start batch job, login, start application, use CICS application or access resource.



Activating a trace...

4. Next stop GTF to prevent excessive traces


STOP GTF

5. shut off the TRACE

@SET TRACE(NORACROUTE NOJOBNAME)

6. Use IPCS to view the trace data.

The input trace data is contained in the dataset specified on the IEFRDER DD card in the GTFRACF (or other) procedure. The sample GTFRACF procedure specifies 'SYSI.TRACE' . Once the TSO IPCS session is active use the IPCS subcommand "IP GTF USR" to display the formatted trace.



Usage notes

Things to know:

- The RACF subsystem must be up and running.
- GTF must be active.
- For OMVS calls, you need an '*' in the jobname filter to trace spawned processes. Otherwise, you will not get a complete set of records. Example:
- @SET TRACE(CALLABLE(TYPE(xx))
JOBNAME(HARDGR*)) LIST will trace
jobnames HARDGR1, HARDGR2 etc



SET LIST - Sample output

```
- RACFR12  IRRH005I (@) RACF SUBSYSTEM INFORMATION:
-   TRACE OPTIONS                - NOIMAGE
-                                - NOAPPC
-                                - RACROUTE
-                                1
-                                - NOCALLABLE
-                                - NODATABASE
-                                - NOASID
-                                - JOBNAME
-                                HARDGR2
- SUBSYSTEM USERID              - IBMUSER
- JESNODE (FOR TRANSMITS)       - POKVMMCL
- AUTOMATIC COMMAND DIRECTION IS *NOT* ALLOWED
- AUTOMATIC PASSWORD DIRECTION IS *NOT* ALLOWED
- PASSWORD SYNCHRONIZATION IS *NOT* ALLOWED
- AUTOMATIC DIRECTION OF APPLICATION UPDATES IS *NOT* ALLOWED
- RACF STATUS INFORMATION:
-   TEMPLATE VERSION            - HRF7705
00-   DYNAMIC PARSE VERSION      - HRF7705
```

Output trace format

```

Trace Identifier:      00000036
Record Eyecatcher:   RTRACE
Trace Type:          RACFPRE
Ending Sequence:     .....
Calling address:     00000000 8B04A24E
Requestor/Subsystem: RSSC06  RACF
Task address:        00000000 006EC1A0
Task ACEEP:          00000000 00000000
Time:                B5773AAD 0E780C4B
Error class:         .....
Service number:      00000005
RACF Return code:    00000000
RACF Reason code:    00000000
Return area address: 00000000 00000001
Parameter count:     0000000A
    
```

Header portion
(fixed length)

```

Area length:         00000068

Area value:
00000000 00000000 00680200 00055800 | .....
0B089158 0B089160 0B08916C 00000000 | ..j...j-..j%...
00000000 00000068 00000000 00000000 | .....
00400000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
    
```

Unloaded parameters
from RACF parameter
list

```

Area length:         0000006C

Area value:
6C0000A0 00000000 00000000 00000000 | %.....
00000000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
00000000 0B089154 00000000 00000000 | .....j.....
00000000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
00000000 00000000 00000000 00000000 | .....
    
```

```

Hexadecimal dump of record follows:
+0000 00000036 D9E3D9C1 C3C54040 D9C1C3C6 | ...RTRACE RACF
+0010 D7D9C540 00000000 00000000 00000000 | PRE.....
+0020 00000000 00000000 00000000 8B04A24E | .....S+
+0030 D9E2E2C3 F0F640F9 00000000 00000000 | RSSC06 9.....
+0040 D9C1C3C6 40404040 006EC1A0 00FA9B00 | RACF .>A.....
+0050 00FA9B00 0000001D 0000001D D9C1C3C6 | .....RACF
+0060 40404040 D9C1C3C6 40404040 006FFDC0 | RACF .?.{
+0070 006FFDC0 00000000 B5773AAD 0E780C4B | .?.{.....
+0080 00000000 00000001 0000000A 00000005 | .....
+0090 00000068 00000000 00000000 00680200 | .....
+00A0 00055800 0B089158 0B089160 0B08916C | .....j...j-..j%
+00B0 00000000 00000000 00000068 00000000 | .....
    
```

Raw hex dump
of entire GTF
record including
header



Header portion of trace output

Following is a formatted R_TRACE record.

This trace record was generated by IRRTRC00 with IDENT(R_TRACE).

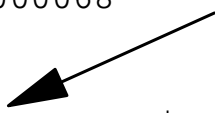
Trace Identifier:	00000036	
Record Eyecatcher:	RTRACE	
Trace Type:	RACFPRE / RACFPOST	
(will be one of these:)	OMVSPRE / OMVSPOST	
	MNGRPRE / MNGRPOST	
Ending Sequence:	
Calling address:	00000000	85D4872E
Requestor/Subsystem:
Primary jobname:	HARDGR2	
Primary asid:	00000018	
Primary ACEEP:	00000000	005FF340
Home jobname:	HARDGR2	
Home asid:	00000018	
Home ACEEP:	00000000	005FF340
Task address:	00000000	005C8D90
Task ACEEP:	00000000	00000000
Time:	B97004AF	74D51E48
Error class:	
Service number:	00000001	(in HEX)
RACF Return code:	00000008	
RACF Reason code:	00000000	
Return area address:	00000000	0005AB90
Parameter count:	0000000B	

Parameter portion of trace output - I

Area length: 00000068

Area value:

00000000	00000000	00D00000	00010000	}	
00000000	00000000	0005A990	00000000	z	
00000000	00000068	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000				



Area length: 00000068


Area value:

68000000	9C000000	80000000	00000000		
00000000	00000000	00000000	00000000		
00000000	0005AC60	0005AC8C	0005AC94	-.....m	
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000				

Area length: 00000008

Area value:

D6C6C6E2	C5E30024				OFFSET..	
----------	-----------------	--	--	--	----------	--



Area length: 0000002C

Area value:

D7C1C7C5	F0F84BC3	C1E3C1D3	D6C74040		PAGE08.CATALOG	
40404040	40404040	40404040	40404040			
40404040	40404040	40404040				

Doc'd in book:
z/OS V1R4.0 Security Server RACF Diagnosis Guide
Chapter 6. Diagnosis reference for RACF
6.1 Parameter list descriptions

Parameter portion of trace output -2

```

Area length:                00000008

Area value:
D6C6C6E2  C5E30028      | OFFSET..      |
                                     ^
Area length:                00000008

Area value:
07C4C1E3  C1E2C5E3      | .DATASET     |
Area length:                00000008

Area value:
D6C6C6E2  C5E3002C      | OFFSET..      |
                                     ^
Area length:                00000006

Area value:
D7C1C7C5  F0F8          | PAGE08       |
Area length:                000000A8

Area value:
C1C3C5C5  FF0000A8  02000000  00000000 | ACEE...y..... |
00000000  07C8C1D9  C4C7D9F2  4006E3E2 | .....HARDGR2 .TS |
D6C7D9D7  40408101  8003138F  40404040 | OGRP  a..... |
40404040  00A85B00  20000000  00000000 |      .y$...... |
D3D6C3C1  D3C3F1F1  00000000  00800000 | LOCALC11..... |
00000000  00000000  40404040  40404040 | ..... |
00000000  005FF3E8  00000000  005C8A08 | .....¬3Y.....*.. |
7FFFB9B0  005FF438  00000000  0103138F | ".....¬4..... |
00000000  00200000  00000000  00000000 | ..... |
00000000  00000000  005FF470  7F6C0000 | .....¬4."%. |
00000000  005FF500  | .....¬5. |

```

Parameter portion of trace output -3

Area length: 00000050

Area value:

50012206	0001C000	00000000	00000000		&.....{.....	
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
D3D6C3C1	D3C3F1F1	00000000	00000000		LOCALC11.....	
C8C1D9C4	C7D9F240	E3E2D6C7	D9D74040		HARDGR2 TSOGRP	

Area length: 00000090

Area value:

C1C3C5E7	03000000	00FAA5F8	00000000		ACEX.....v8....	
00000000	00000000	00000000	00000000		
00000000	00000024	005FF550	00000000	-5&....	
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		
00000000	00000000	00000000	00000000		

How do I read a trace? (continued)

Special Control Blocks and other handling:

- ACEE: not only is the ACEE unloaded, but so are the USP, TOKEN, and ACEX if available.
- CRED: (IRRPCRED) After the CRED structure is unloaded, the first path name, the second path name, the first filename and second filename are unloaded.
- Work Areas: Work Areas in general are not unloaded.
- Passwords: Are not unloaded.
- Installation parameters
- ENVIRIN and ENVIROUT parameters
- Certificates

Raw data portion of trace output

Hexadecimal dump of record follows:

+0000	00000036	D9E3D9C1	C3C54040	D9C1C3C6RTRACE RACF
+0010	D7D9C540	00000000	00000000	00000000	PRE
+0020	00000000	00000000	00000000	00000000
+0030	85D4872E	00000000	00000000	40400000	eMg..... ..
+0040	00000000	00000000	00000000	00000000
+0050	00000000	00000000	005C8D90	00000000*.....
+0060	00F9CA00	00000000	00F9CA00	00000018	.9.....9.....
+0070	00000018	C8C1D9C4	C7D9F240	C8C1D9C4HARDGR2 HARD
+0080	C7D9F240	00000000	005FF340	00000000	GR2-3
+0090	005FF340	00000000	00000000	B97004AF	.-3
+00A0	74D51E48	00000000	00000000	0005AB90	.N.....
+00B0	0000000B	00000001	00000068	00000000
+00C0	00000000	00D00000	00010000	00000000}
+00D0	00000000	0005A990	00000000	00000000z.....
+00E0	00000068	00000000	00000000	00000000
+00F0	00000000	00000000	00000000	00000000
+0100	00000000	00000000	00000000	00000000
+0110	00000000	00000000	00000000	00000000
+0120	00000000	00000068	68000000	9C000000
+0130	80000000	00000000	00000000	00000000
+0140	00000000	00000000	00000000	0005AC60-
+0150	0005AC8C	0005AC94	00000000	00000000m.....
+0160	00000000	00000000	00000000	00000000
+0170	00000000	00000000	00000000	00000000
+0180	00000000	00000000	00000000	00000000
+0190	00000008	D6C6C6E2	C5E30024	0000002COFFSET.....
+01A0	D7C1C7C5	F0F84BC3	C1E3C1D3	D6C74040	PAGE08.CATALOG
+01B0	40404040	40404040	40404040	40404040	
+01C0	40404040	40404040	40404040	00000008
+01D0	D6C6C6E2	C5E30028	00000008	07C4C1E3	OFFSET.....DAT
+01E0	C1E2C5E3	00000008	D6C6C6E2	C5E3002C	ASET....OFFSET..
+01F0	00000006	D7C1C7C5	F0F80000	00A8C1C3PAGE08...yAC
+0200	C5C5FF00	00A80200	00000000	00000000	EE...y.....
+0210	000007C8	C1D9C4C7	D9F24006	E3E2D6C7	...HARDGR2 .TSOG
+0220	D9D74040	81018003	138F4040	40404040	RP a.....
+0230	404000A8	5B002000	00000000	0000D3D6	.y\$......LO
+0240	C3C1D3C3	F1F10000	00000080	00000000	CALC11.....
+0250	00000000	00004040	40404040	40400000
+0260	0000005F	F3E80000	0000005C	8A087FFF	...-3Y.....*..".
+0270	B9B0005F	F4380000	00000103	138F0000	...-4.....
+0280	00000020	00000000	00000000	00000000
+0290	00000000	0000005F	F4707F6C	00000000-4."%.....
+02A0	0000005F	F5000000	00505001	22060001	...-5.....&&.....
+02B0	C0000000	00000000	00000000	00000000	{.....

Reading a trace...

**Sample RACROUTE
trace for a specific
situation....**



Situation to be traced

USERID HARDGR2 has OPERATIONS authority at userid level but is permitted to dataset profile (by virtue of a group connect) with only READ access.

Why can this userid DELETE (uncatalog and scratch) a dataset covered by this profile?

userid HARDGR2 issues:

```
DELETE 'IBMUSER.DELETE.DATASET2'
```

and sees

```
IDC0550I ENTRY (A) IBMUSER.DELETE.DATASET2  
DELETED
```

why did the delete occur? why wasn't authority denied?

SAFTRACE can and will demonstrate exactly why..

DELETE 'IBMUSER.DELETE.DATASET2'



Following is a formatted R_TRACE record.

This trace record was generated by IRRTRC00 with
IDENT(R_TRACE).

Trace Identifier:	00000036	
Record Eyecatcher:	RTRACE	
Trace Type:	RACFPOST	
Ending Sequence:	
Calling address:	00000000	85D1349C
Requestor/Subsystem:
Primary jobname:	HARDGR2	
Primary asid:	000001F7	
Primary ACEEP:	00000000	005C83D8
Home jobname:	HARDGR2	
Home asid:	000001F7	
Home ACEEP:	00000000	005C83D8
Task address:	00000000	005C8D90
Task ACEEP:	00000000	00000000
Time:	B986A9AA	984ECB07
Error class:	
Service number:	00000001	
RACF Return code:	00000008	
RACF Reason code:	00000000	
Return area address:	00000000	00000000
Parameter count:	0000000B	

IGG0290A

DADSM SCRATCH

igg0290a 10/12/01 HDZ11E0 EP AT 05D12AE8 balr 14,15 AT

05D1349A 05EF X'09b2'

RACROUTE REQUEST=AUTH,CLASS=DATASET

Area length: 00000028

Area value:

00000008 00000000 00280000 00010000 |\$.y.... |
00000000 00000000 005B69A8 00000000 | |
00000000 00000028

Area length: 0000003C

Area value: << 1100 1100 >>

3C000000 CC000000 80000000 00000000 | |
00000000 00000000 00000000 00000000 | |
00000000 005B6638 005B6910 005B5CC4 |\$.\$.\$.D..... |
00000000 00000000 00000080

CC 11001100
^

LOG=NOFAIL

If the authorization check fails, the attempt is not recorded. If the
authorization check succeeds, the attempt is recorded as in ASIS

Area length: 00000008

Area value:

D6C6C6E2 C5E30024 | OFFSET.. |

Area length: 0000002C

Area value:

C9C2D4E4 E2C5D94B C4C5D3C5 E3C54BC4 | IBMUSER.DELETE.DATASET2 |
C1E3C1E2 C5E3F240 40404040 40404040 | |
40404040 40404040 40404040

Area length: 00000008

Area value:

D6C6C6E2 C5E30028 | OFFSET.. |

Area length: 00000008

Area value:

07C4C1E3 C1E2C5E3 | .DATASET |

Area length: 00000008

Area value:

D6C6C6E2 C5E3002C | OFFSET.. |

Area length: 00000006

Area value:

E3C5D4D7 F0F1 | TEMP01 |

DELETE 'IBMUSER.DELETE.DATASET2'

no, next check...



Following is a formatted R_TRACE record.
 This trace record was generated by IRRTRC00 with
 IDENT(R_TRACE).

Trace Identifier:	00000036	
Record Eyecatcher:	RTRACE	
Trace Type:	RACFPOST	
Ending Sequence:	
Calling address:	00000000	85D13B4C
Requestor/Subsystem:
Primary jobname:	HARDGR2	
Primary asid:	000001F7	
Primary ACEEP:	00000000	005C83D8
Home jobname:	HARDGR2	
Home asid:	000001F7	
Home ACEEP:	00000000	005C83D8
Task address:	00000000	005C8D90
Task ACEEP:	00000000	00000000
Time:	B986A9AA	98DD5007
Error class:	
Service number:	00000001	
RACF Return code:	00000000	
RACF Reason code:	00000000	
Return area address:	00000000	00000000
Parameter count:	00000009	

IGG0290A

DADSM SCRATCH

igg0290a 10/12/01 HDZ11E0 EP AT 05D12AE8

balr 14,15 AT 05D13B4A 05EF x'1062'

RACROUTE REQUEST=AUTH,CLASS=DASDVOL

Area length: 00000028

Area value:

00000000	00000000	00280000	00010000	\$.y.....	
00000000	00000000	005B69A8	00000000		
00000000	00000028					

Area length: 0000003C

Area value:

3C000000	0C000000	80000000	00000000		
00000000	00000000	00000000	00000000			
00000000	005B5CC4	005B6910	00000000	\$.*D.\$.....	
00000000	00000000	00000000				

Area length: 00000008

Area value:

D6C6C6E2	C5E30024			OFFSET..		
----------	----------	--	--	----------	--	--

Area length: 00000006

Area value:

E3C5D4D7	F0F1			TEMP01		
----------	------	--	--	--------	--	--

Area length: 00000008

Area value:

D6C6C6E2	C5E30028			OFFSET..		
----------	----------	--	--	----------	--	--

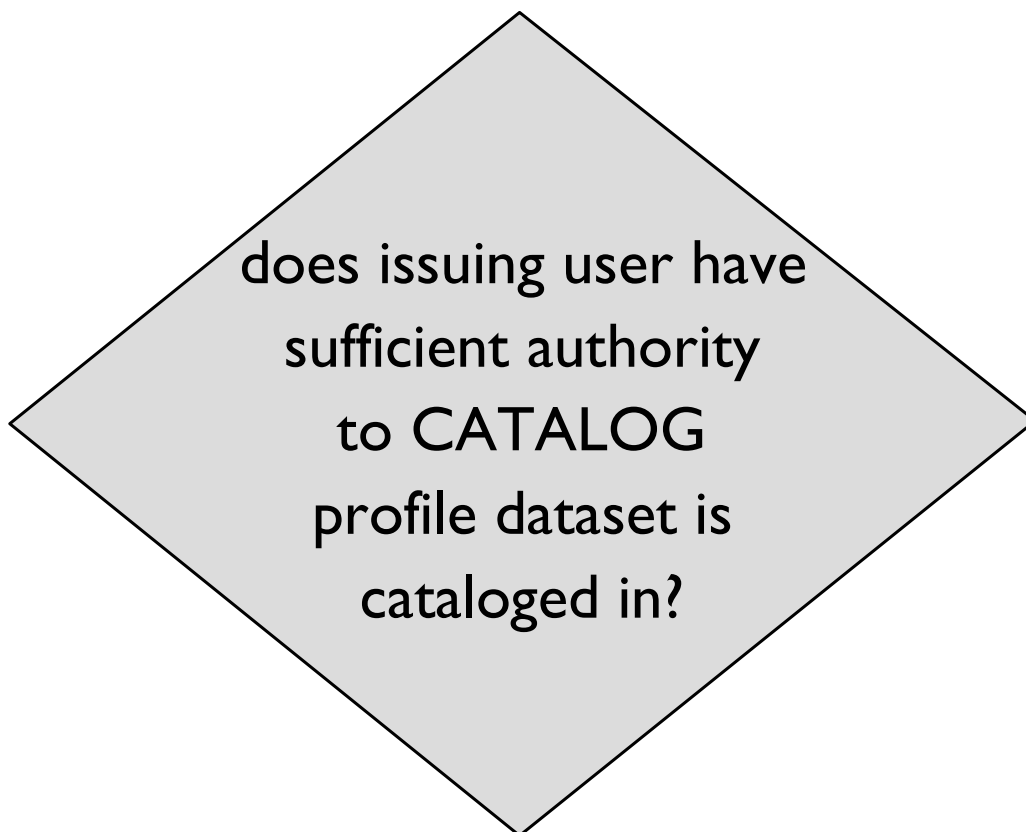
Area length: 00000008

Area value:

07C4C1E2	C4E5D6D3			.DASDVOL		
----------	----------	--	--	----------	--	--

DELETE 'IBMUSER.DELETE.DATASET2'

yes, dataset can be SCRATCHed.
may it be UNCATALOGed also?



Following is a formatted R_TRACE record.
This trace record was generated by IRRTRC00 with
IDENT(R_TRACE).

Trace Identifier:	00000036	
Record Eyecatcher:	RTRACE	
Trace Type:	RACFPOST	
Ending Sequence:	
Calling address:	00000000	85D4872E
Requestor/Subsystem:
Primary jobname:	HARDGR2	
Primary asid:	000001F7	
Primary ACEEP:	00000000	005C83D8
Home jobname:	HARDGR2	
Home asid:	000001F7	
Home ACEEP:	00000000	005C83D8
Task address:	00000000	005C8D90
Task ACEEP:	00000000	00000000
Time:	B986A9A9	6F2E5E00
Error class:	
Service number:	00000001	
RACF Return code:	00000000	
RACF Reason code:	00000000	
Return area address:	00000000	0005CB90
Parameter count:	0000000B	

IGG0CLH0

FUNCTION = PERFORMING SAF, RACF, TGET,
DADSM SCRATCH, USVR, WTOR IN THE
USER'S ADDRESS SPACE.

igg0CLH0 10/13/01 HDZ11G0 ep at 05D483B8
balr 14,15 at 05D4872C 05EF X'0374'

Area length: 00000068

Area value:

00000000	00000000	00D00000	00010000	}I.....	
00000000	00000000	0005C990	00000000				
00000000	00000068	00000000	00000000			
00000000	00000000	00000000	00000000			
00000000	00000000	00000000	00000000			
00000000	00000000	00000000	00000000			
00000000	00000000					

Area length: 00000068

Area value:

68000000	9C 000000	80000000	00000000			
00000000	00000000	00000000	00000000				
00000000	0005CC60	0005CC8C	0005CC94	-m.....	
00000000	00000000	00000000	00000000			
00000000	00000000	00000000	00000000			
00000000	00000000	00000000	00000000			
00000000	00000000					

Area length: 00000008

Area value:

D6C6C6E2	C5E30024				OFFSET..		
----------	----------	--	--	--	----------	--	--

Area length: 0000002C

Area value:

D7C1C7C5	F0F84BC3	C1E3C1D3	D6C74040		PAGE08.CATALOG		
40404040	40404040	40404040	40404040				
40404040	40404040	40404040					

Area length: 00000006

Area value:

D7C1C7C5	F0F8				PAGE08		
----------	------	--	--	--	--------	--	--

Area length: 00000008

Area value:

D6C6C6E2 C5E30028 | OFFSET.. |

Area length: 00000008

Area value:

07C4C1E3 C1E2C5E3 | .DATASET |

Area length: 00000008

Area value:

D6C6C6E2 C5E3002C | OFFSET.. |

Conclusion:

Even though this user has

OPERATIONS but is specifically permitted to dataset profile with **READ** authority, it gives implicit **ALTER** to both **DASDVOL** profile and **CATALOG** dataset profile allowing **SCRATCH** and **UNCATALOG** to **STILL** occur.

:-b

Session Summary

- What SAFTRACE is.
- Uses of the trace.
- Activating a trace.
- Read a trace.
- Real world example



Additional information

- ▶ **z/OS VIR4.0 Security Server RACF Callable Services**
- ▶ **z/OS VIR4.0 Security Server RACF Diagnosis Guide**
- ▶ **z/OS VIR4.0 Security Server RACF Command Language Reference**
- ▶ **z/OS VIR4.0 Security Server RACF Data Areas**



Appendix A

RACROUTE Types

RACROUTE REQUEST	Service Number or TYPE (HEX)	Service Number or Type (Decimal)
AUTH	1	1
FASTAUTH	2	2
LIST	3	3
DEFINE	4	4
VERIFY	5	5
EXTRACT	6	6
DIRAUTH	7	7
TOKENMAP	8	8
VERIFYX	9	9
TOKENXTR	A	10
TOEKNBLD	B	11
EXTRACT, BR=YES	C	12
AUDIT	D	13
STAT	E	14
SIGNON	F	15
TOKENMAP, XMEM	10	16
TOKENXTR, XMEM	11	17

Appendix B

Callable Service Types

CALLABLE SERVICE	Service Number or TYPE (HEX)	Service Number or TYPE (DECIMAL)
IRRRIU00 - initUSP	1	1
IRRREDU00 - deleteUSP	2	2
IRRRFMF00 - makeFSP	3	3
reserved	4	4
IRRMM00 - R ₀ umask	5	5
IRRKA00 - ck ₀ access	6	6
IRRRKP00 - ck ₀ priv	7	7
IRRUM00 - getUMAP	8	8
IRRRGM00 - getGMAP	9	9
IRRRGG00 - R ₀ getgroups	A	10
IRRRSU00 - R ₀ setuid	B	11
IRRRREU00 - R ₀ seteuid	C	12
IRRRSG00 - R ₀ setgid	D	13
IRRRREG00 - R ₀ setegid	E	14
IRRRCO00 - R ₀ chown	F	15

Appendix B

Callable Service Types

CALLABLE SERVICE	Service Number or TYPE (HEX)	Service Number or TYPE (DECIMAL)
IRRRCF00 - Rchmod	10	16
IRRRC A00 - Rchaudit	11	17
IRRREX00 - Rexec	12	18
IRRRAU00 - Raudit	13	19
IRR RK000 - ckprocessowner	14	20
IRR S00 - querysystemsecurityoptions	15	21
IRR F00 - queryfilesecurityoptions	16	22
IRR RCS00 - clearsetid	17	23
IRR RKF00 - chfileowner	18	24
IRR RMR00 - makerootPSP	19	25
IRR RPT00 - Rptrace	1A	26
IRR RUG00 - Rgetgroupsbyname	1B	27
IRR RFK00 - Rfork	1C	28
IRR RMI00 - makelSP	1D	29
IRR RKI00 - ckIPCaccess	1E	30

Appendix B

Callable Service Types

CALLABLE SERVICE	Service Number or TYPE (HEX)	Service Number or TYPE (DECIMAL)
IRRRCI00 - RRPCctl	1F	31
IRRRC200 - ckownertwofiles	20	32
IRRERGE00 - getuidgidsupgrps	21	33
IRRREDI00 - Rdiceinfo	22	34
IRRREDK00 - Rdicekey	23	35
IRRREDU00 - Rdiceruid	24	36
IRRREDA00 - Rdiceauth	25	37
IRRRIA00 - Initacee	26	38
*IRRSE00 - Radmin	27	39
*IRRSIM00 - Rusermap	28	40
*IRRSDL00 - Rdatalib	29	41
*IRRSMK00 -	2A	42
*IRRSPK00 - Rticketserve	2B	43
IRRSPX00 - RPKIServ	2C	44
IRRSCH00 - Rcacheserv	2D	45
IRRSPY00 - Rproxyserv	2E	46