A View From RACF Level 2
RAA10

Vanguard EXPO 2007

June 11-14, 2007

St. Louis, MO

John Reale, III

RACF Level 2

IBM – z/OS Software Support

Poughkeepsie, NY

jreale@us.ibm.com

Agenda

- ICH408I - get the WHOLE message
- The caller versus RACF
- Locating the caller of RACF
- SAFTRACE Facility - brief overview
- Some NEW stuff
- Current APARS of some interest, with a few last minute adds
- Level2 Personnel
- Any Questions?

ICH408I - get the WHOLE message

ICH408I USER(SMITH) GROUP(DEPT60) NAME(R.L.SMITH)
ICH408I DEPT58.CLIST.CNTL CL(DATASET) VOL(TSO035)
ICH408I INSUFFICIENT ACCESS AUTHORITY
ICH408I FROM DEPT58.CLIST.* (G)
ICH408I ACCESS INTENT(READ) ACCESS ALLOWED(NONE)

- It tells a story. **WHO, WHAT, and WHY**

- Perhaps, most important is the WHO. The "who" should be an identifiable userid and group.
- ---
- It is important to note if it's **JOB( ) STEP( )**.
- Indicates that task is either unidentified (to RACF) or call was made with token / userid not defined to RACF.
- In the case of user / token undefined, it may be another address space's "fault".

…more ICH408I

ICH408I USER(HARDGR3) GROUP( ) NAME(???)
 LOGON/JOB INITIATION - USER AT TERMINAL NOT RACF-DEFINED
IRR012I VERIFICATION FAILED. USER PROFILE NOT FOUND.

- This next one is NOT a RACF violation per se - it is a USS auditing message

ICH408I USER(HARDGRO) GROUP(TSOGRP) NAME(HARDGRO ID#1)
 /.sh_history CL(DIRACC) FID(01C8D9E9F1F7F00001040000000000003)
 INSUFFICIENT AUTHORITY TO OPEN
 ACCESS INTENT(-W-) ACCESS ALLOWED(OTHER R-X)
 EFFECTIVE UID(0000000001) EFFECTIVE GID(0000000099)

See Info APAR II12593 for more details, and the AUDITID tool to decipher the FID at:
www-1.ibm.com/servers/eserver/zseries/zos/unix/bpxa1ty2.html

The CALLER vs RACF

- Who is responsible for what?
  - Caller's
    - parameter list setup
    - handling return code
  - RACF's
    - accurately handling call
    - passing info back to caller

- Who actually denies access?? **The caller !!** >> (not RACF) <<

- The biggest factor is what call was made when (and why). For example, CATALOG calls RACF several times and suppresses many / most messages.
- Best to take up with our caller :-D

Locating the caller of RACF service

- Easiest when an SVC was issued by caller; ie, the old macros.
  - pickup PSW from SYSTRACE or the RB in SUMM FORMAT.
  - R1 has Parameter list of SVC call.

- Most common call is RACROUTE
  - uses BALR 14,15
  - need to use R0 of the SVC call (points to 2 words).
    - word1→ RACF Service Parms; word2 → SAF Parms
  - Need to go back thru SAF Parms/saveareas.
  - get R0 from either SYSTRACE entry or the next RB in SUMM FORMAT.

- We won't be covering OMVS callable services calls in this presentation.

Locating the caller – an example

05 00A2 009DD928 SVC 82 070C2000 **80DA266E** 00000000 **00005910** 00000000
05 00A2 009DD928 SVC 77 070C2000 88203442 88201FE4 FFF00000 000E0001
05 00A2 009DD928 SVCR 77 070C2000 88203442 00000000 FFF00000 80004000

IPCS OUTPUT STREAM ----------
Command ===> ip where **00DA266E**
****
ASID(X'00A2') 00DA266E. ICHRFR00+266E IN PLPA
****

ASID(X'00A2') STORAGE ----------
Command ===> **L 00005910**
00005910 **00005A90 00005A28** 00000000 884C68B2 | ..!...!.....h<.. |
00005920 TO 000059FF (X'000000E0' bytes)--All bytes contain X'00'
00005A00 00000000 00005A90 00000000 00000000 | .....!..... |
00005A10 TO 00005A2F (X'00000020' bytes)--All bytes contain X'00'
00005A30 00AC0000 00010000 00000000 00000000 | ..... |

…more example

```
SVRB: 009FF928
  -0020  XSB..... 7FFFC918  FLAGS2... 00       RTPSW1...
  -0008  FLAGS1... 02000000  WLIC..... 00020082
  +0000  RSV..... 00000000  00000000           SZSTAB...
  +0018  Q....... 00000000  LINK..... 009DD8A0
  +0020  GPR0-3... 00000000  009B8A5C  009ACED9  009CA480
  +0030  GPR4-7... 009CAE30  009AA688  009C9340  009D6F08
  +0040  GPR8-11.. 009B88B8  07F3C2F5  07F3B2F6  07F3A2F7
  +0050  GPR12-15. 87F392F8  009B88B8  87F3B2C6  00000000

SVRB: 009FF738
  -0020  XSB..... 7FFFC7B8  FLAGS2... 80       RTPSW1...
  -0008  FLAGS1... 02000004  WLIC..... 00040010
  +0000  RSV..... 00000000  00000000           SZSTAB...
  +0018  Q....... 00000000  LINK..... 009FF928
  +0020  GPR0-3... 00005910 00000000  00DA1C50  00F4B530
  +0030  GPR4-7... 00000002  00000001  00000001  00005A90
  +0040  GPR8-11.. 00005A28  00005888  00000000  00005890
  +0050  GPR12-15. 00DA1C50  00005890  00000002  00000000
```

Example - Step 2

ASID(X'00A2') STORAGE ----------
Command ===> **1st word** is address of **RACF service** parameter list
00005910 **? 00005A90 00005A28** 00000000 884C68B2 | ..!...!.....h<.. |
00005920 TO 000059FF (X'000000E0' bytes)--All bytes contain X'00'
00005A00 00000000 00005A90 00000000 00000000 | .....!..... |
00005A10 TO 00005A2F (X'00000020' bytes)--All bytes contain X'00'
00005A30 00AC0000 00010000 00000000 00000000 | ..... |

ASID(X'00A2') STORAGE ----------
Command ===>
00005A90 44000000 98000000 04000000 00000000 | ....q..... |
00005AA0 TO 00005AAF (X'00000010' bytes)--All bytes contain X'00'
00005AB0 00000000 00005AF8 00005B24 00005B2C | .....!8..$...$. |
00005AC0 TO 00005AEF (X'00000030' bytes)--All bytes contain X'00'

...more of Step 2

ASID(X'00A2') STORAGE ----------
Command ===> **2nd word** is address of **SAF** parameter list
00005910 **00005A90 ? 00005A28** 00000000 884C68B2 |..!...!.....h<..|
00005920 TO 000059FF (X'000000E0' bytes)--All bytes contain X'00'
00005A00 00000000 00005A90 00000000 00000000 |.....!.....|
00005A10 TO 00005A2F (X'00000020' bytes)--All bytes contain X'00'
00005A30 00AC0000 00010000 00000000 00000000 |.....|

ASID(X'00A2') STORAGE ----------
Command ===> from here issue **L x+18?+14?+C** …because…
00005A28 TO 00005A2F (X'00000008' bytes)--All bytes contain X'00'
00005A30 00AC0000 00010000 00000000 00000000 |.....|
00005A40 00005828 00000000 00000000 00000068 |.....|
00005A50 TO 00005A8F (X'00000040' bytes)--All bytes contain X'00'
00005A90 44000000 98000000 04000000 00000000 |....q.....|
00005AA0 TO 00005AAF (X'00000010' bytes)--All bytes contain X'00'

The Caller at
L R0+4?+18?+14?+C?-2

This way of finding the caller is to go to the SAF workarea....

SVC 8X R0 → two-word ParmList
+0 → RACF Service ParmList
+4 → SAF ParmList
+18 → SAF WorkArea (SAFPWA)
+14 → SaveArea

SAVEAREA

Words:	1st	2nd	3rd	4th	5th
	+0	+4	+8	+C	+10
	Garbage	HSA	LSA	R14 (caller)	R15 (ICHSFR00+0)

Go to the saved R14 and backup 2 to confirm the BALR, farther to find the module/level.

You can issue IP Where on the saved R14 and R15, also. (good to verify R15)

An Example – Step 3

```
ASID(X'00A2') STORAGE  ----------------------------------------------
Command ===>
00005398                    884C68B2  00D9A000   |.      h<...R..|
000053A0   08150008   10745064   00280000   08150008   |.....&.....|
000053B0   00000001   10745000   00005818   00005828   |.....&.....|
000053C0   10745000   00FBE700   084C770B   000052F0   |..&...X..<.....0|
000053D0   884C670C   00000000   00000000   00000000   |h<.............|
000053E0 TO 000053FF (X'00000020' bytes)--All bytes contain X'00'
```

…more of Step 3

CALLED (where you're headed)
IPCS OUTPUT STREAM ----------
Command ===> ip where **00D9A000** reg 15 (to addr)
****
ASID(X'00A2') 00D9A000. **ICHSFR00+00** IN PLPA
****

CALLER (where you came from)
IPCS OUTPUT STREAM ----------
Command ===> ip where **084C68B2** reg 14 (from addr)
****
ASID(X'00A2') 084C68B2. **IGG0CLHA+78B2** IN EXTENDED PLPA
****

You should see the 05EF in the code (if storage is IN dump of course).

Caller – Lastly

Remember, RACROUTE is called via a BALR R14,R15. This invokes SAF router (ICHSFR00) which in turn calls the RACF router (ICHRFR00) to do the actual SVC (x'82', x'83', x'84', x'85') invocation. (if necessary - some calls don't issue SVCs)

SAFTRACE Facility

- all supported releases of z/OS
- Traces:
  - RACROUTEs (Auths, Verifies, etc.)
  - Manager calls (ICHEINTYs)
  - OMVS Callable Services
  - Doesn't trace branch entered services, or older SVC interfaces
- Use SET command to initiate, filter, stop
  - so RACF Address space required
- GTF required to be running
- Two trace records are generated: pre- & post-event
- You will need a good knowledge of:
  - RACF parameter lists
  - RACF in general

More SAFTRACE

- Output (GTF trace records) can be formatted by IPCS.
- Each record shows the Caller, both ParmLists, and many of the Parms.
- Practice makes perfect - (become familiar before that first fire - needed use)

SAFTRACE - Instructions

To activate the RACF GTF Trace:

- start GTF with a dataset and the parms TRACE=USRP,USR=(F44).
- issue console command:

```
   @SET TRACE( JOBNAME(jobname*)
               RACROUTE(TYPE(1,4))       /* Auth, Define
              CALLABLE(TYPE(38) ) )      /* initACEE
              )  LIST
```

- Do your thing...
- take a console dump of the ASID/job if one is not automatic.
  (used to find the calling code from the caller's address)
- issue console command:

```
   @SET TRACE( NOJOBNAME NORACROUTE NOCALLABLE )  LIST
```

- stop GTF, collect the trace dataset.
- terse/FTP the dump and trace to IBM.

NOTE: the '@' represents the subsystem cmd prefix you assigned to RACF.

Some NEW Stuff

- z/OS V1R8 – not much experience
  - Password Phrase
  - More Health Checks
  - UT200/UT400 safety features
  - More ChangeLogging

Some NEW Stuff

- z/OS V1R7 - Health Checker reports
  - APARs for ‘corner cases’ - OA15290, OA17127, OA17994, OA18868
- z/OS V1R6 - Dynamic CDT
  - Eliminates ICHRFR01
  - Eliminates IPLs to load or change classes
  - Instead, uses new class CDT.
  - APAR OA11874 - a MUST to have on at 1.6+

Some NEW Stuff

- z/OS V1R5 – Dynamic Templates
  - Templates no longer shipped as MODGEN(IRRTEMP1).
  - Now IRRTEMP2 imbedded in IRRMIN00 & ICHSEC00.
  - Will load in storage even if NOT on database. See:

ICH579E ICH579E RACF TEMPLATES ON DATABASE ARE DOWNLEVEL: *FMID or APAR* rrrrrrrr.aaaaaaaa; USING TEMPLATES AT LEVEL *FMID or APAR* rrrrrrrr.aaaaaaaa FROM IRRTEMP2. RUN IRRMIN00 PARM=UPDATE

  - Re: APAR OA07917 & INFO APARs II07031, II13997

More on Templates & Dynamic Parse

- Interaction of these can (and does) cause much confusion.
  ICH577E WARNING: BASE SEGMENT OF USER TEMPLATE AT LEVEL template-level DOES NOT CONTAIN FIELD ffffffff.
- With dynamic templates, this ought to be a thing of the past.

- TEMPLATES - Should match highest level of software using this RACF database. Execute Pgm IRRMIN00 with PARM=UPDATE. Pre-V1R5, you had to specify the IRRTEMP1 file. If an IPL is not required, you can also execute IRRMIN00 with PARM=ACTIVATE.

- DYNAMIC PARSE TABLE SEGMENTS – MUST match software level of running system. Execute Pgm IRRDPI00 (usually via task IRRDPTAB) to process the IRRDPSDS file.

- REF: section 1.5.1.1 of the RACF System Programmer's Guide

IRRIRA00 – AIM Stages

- AIM – Alternate Index Mapping.
- Replacement for mainly UNIXMAP.
- Also helps other segments; eg, NOTELINK.
- Improves performance of a ‘reverse’ search.
- On 1.7-, must have OA11805, and OA12443 when sharing with 1.8.
- Also see Info APAR II12972.
- REF: section 7.1 of the RACF System Programmer’s Guide

Other Stuff

RACF SUBSYSTEM ASID:

- One of RRSF's major components.
- Used to service IRRSEQ00 calls (ie, R_Admin callable service)
  - Used much more now by LDAP.
- Needed for SAFTRACE for the SET cmd used to start, filter, stop.
- Enables ability to issue RACF commands via an MVS console.
  - SET and RVARY don't require logon (plus those dealing with RRSF; TARGET, etc).
  - Most other commands do require logon (AU, AD, CO, RDEF, SETR, etc).
- Used for APPC PV (persistent verification - RACROUTE REQ=SIGNON).
- Highly recommended to have up and running.

- It is NOT part of garden variety RACF functionality.
- For RACF address space, we _recommend_ giving it the trusted attribute.

More Other Stuff

Emergency ICHRDSNT

- To specify a different DB (or an *)
- Turn off the sysplex comm bit

D/R practice - a few "what if" scenarios:

- Primary and/or Backup DB no good at IPL
- out of space conditions
- database corruption
- do you know the RVARY passwords?

When migrating to a new OS, remember:

- ICHRIN03
- ICHRDSNT and ICHRRNG
- ICHRRCDE and ICHRFR01 (ICHRFR01 not used at 1.6 and up)
- ICHDEX01 (or not)
- ICHR?X01/02 (where ? is I, C and D) and other miscellaneous exits
- compare ICH508I message from two IPLs
- look for ICH524I / ICH525I as appropriate

APARS of Interest:
V1R8 7/06 – 3/07

OA19617	3	RACF PANEL ICHP41 GENERATES MSGICHM412A FOR OWNER	FIN
OA18327	2	AT Z/OS V1R8 FIN APAR OA14273 CHANGED ADDUSER AND	PER
OA18087	3	USING A RACROUTE EXTRACT,TYPE=REPLACE TO DELETE TH	PER
OA17429	3	RACF_SENSITIVE_RESOURCES CHECKS AN INCORRECT GENER	PER
OA17400	3	RACF SUPPORT FOR DB2 V9 TO PROVIDE ROLE BASED SECU	UR1
OA17388	3	ABEND878 REASON CODE 10 IN R_DATALIB CALLABLE SERV	PER
OA17259	2	ABSTRACT: RACROUTE REQUEST=VERIFY PERFORMANCE DEG	PER
OA17115	3	A TSO LOGON OR RACROUTE REQUEST=VERIFY WITH A PASS	PER
OA16923	3	THE PASS PHRASE HISTORY FIELD IS NOT CALCULATED CO	PER
OA16782	3	IRRREQ00 (THE R_ADMIN CALLABLE SERVICE) TAKES AN A	PER
OA16757	3	RACF ENVIRONMENT SERVICE IRRENS00 MUST REPORT ONLY	PER
OA16752	3	RACF ENVIRONMENT SERVICE IRRENS00 QUERY FUNCTION	PER
OA16257	3	LISTDSD DSNS FAILS WITH A 0C4-11 ABEND IN LOAD MOD	PER

APARS of interest:
V1R7 7/06 – 3/07

OA20513	2	UNABLE TO USE BLKUPD (AKA IRRUT300) TO LOCATE / DI	OPEN
OA20502	2	AUTO APPL DIRECTION DOESN'T FLOW IF A DEFAULT	OPEN
OA20304	3	NEW FUNCTION	OPEN
OA20162	3	RACF SUPPORT FOR CICS TRANSACTION SERVER	OPEN
OA20144	3	NEW FUNCTION	OPEN
OA20002	3	IRRRID00 OUTPUT CONTAINS RACDCERT CMDS FOR USERS T	OPEN
OA19769	2	RACF DOESN'T DISPLAY RSN07 IN MSGICH505A ABENDAC5	OPEN
OA18958	3	APF LIBRARY NAME'S ALIAS WILL BE FLAGGED AS AN ERR	OPEN (5752SC1CJ)
OA18243	2	MSGICH427I MSGICH428I WHEN(PROGRAM -- ENHANCED WAR	DOC

APARS of interest:
V1R6 7/06 – 3/07

OA20124	3	HAVING FAILURES USING THE RACF ISPF ADMIN PANELS.	FIN
OA19761	2	ABEND0C4-11 ICHRAU00+1C18 UA17182 MOVING TOO MUCH	OPEN
OA19640	2	ICH408I INSUFFICIENT AUTHORITY TO <BLANK>	PER
OA19353	3	SSL CAN'T FIND CERT ON RACF KEYRING; RC8 RC8 RSN2C	DOC
OA18868	3	MISSING INFO FROM RACF_SENSITIVE_RESOURCES HEALTH	PER
OA18728	2	PREVIOUS GID USED AS FILE'S GROUP WHEN OMVS BATCH	PER
OA18698	2	R_ADMIN ADMN_XTR_SETR OUTPUT GLOBAL LIST DOES NOT	PER
OA18575	2	A TIMING WINDOW EXISTS IN THE RACF RRSF IRRPOST MA	PER
OA18540	3	SECTION 12.6.5.4 OF RACF SECURITY ADMIN GUIDE IS U	DOC
OA18340	2	RRSF (RACF SUBSYSTEM) ADDRESS SPACE ABEND878 DUE T	CAN
OA18242	3	COMMTASK ABEND378-10 IRRRSM00+X06E2 CALLED BY IRRA	CAN
OA18073	3	ABEND0C4 IRRCOP21+334 DURING IPL WHILE A SETROPTS	PER
OA17994	4	RACF HEALTH CHECKER SENSITIVE DATASET REPORT CAN F	PER
OA17727	3	SMFTYPE80 SMF80EVT=27X (TERMOEDP) CONTAINS INVALID	PER
OA17675	2	APPLICATIONS USING R_ADMIN RACF INTERFACE FROM MUL	SUG
OA17588	3	RACF OPTION 7.2 (CREATE CSR) RESULTS IN ISPP100 ME	PER

Personnel
RACF Level 2 - Poughkeepsie NY

- John Reale III - team leader
- Russ Hardgrove
- Bill Garvin – also covers RACF on VM

Any Questions ?

- The presentation?
- Service?
- IBM?
- Other?

Session Summary

- ICH408I - get the whole message
- The caller versus RACF
- Locating the caller of RACF
- SAFTRACE Facility
- Some NEW Stuff
- APARS of interest
- RACF Level2 Personnel
- Some questions and answers