Generic Profile Load Performance

New York RACF User's Group

Speaker: Russ Hardgrove
RACF lvl2
hardgrov@us.ibm.com

Trademarks

For a list of trademarks see URL :

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

Session Objectives

- Explain what is the purpose of the Generic Profile Load Performance support
- Describe what function has changed and how it works
- Describe the external interfaces introduced or modified for this new function
- Impart some late breaking info (aka APARs of interest)

Overview

- **Problem Statement / Need Addressed:**
  - Customers could have RACF or system performance impacted when processing generic profiles for resource classes that are neither RACLISTed nor SETROPTS GENLISTed when either:
    - Customers define a large number of generic profiles within some data set high level qualifier (HLQ) or within some non-RACLISTed general resource class; or
    - Customer applications reference many non-RACLISTed general resource classes or data sets with many different HLQs.
- **Solution:**
  - This support modifies the way that RACF loads and searches generic profiles for resource classes that are neither RACLISTed nor SETROPTS GENLISTed.
- **Benefit:**
  - Improved performance.

Overview

- Prior to this change
  - RACF could cache up to 4 sets of generic profile names per address space, known/referred to as **GATE**s (**G**eneric **A**nchor **T**able **E**ntries). These are used to speed generic profile coverage checking.
  - If a particular address space frequently uses more than 4 sets of profiles, thrashing could occur.
  - The only workarounds for these problems have involved customers either
    - Splitting the RACF database to reduce contention, or
    - Physically renaming data sets to reduce the number of generic profiles under a single HLQ, or
    - Doing an analysis of their existing generic profiles to try to reduce their numbers, or
    - Implementing a RACF Naming Convention Table so RACF will pretend that the data sets have different HLQs, and thus can end up with more logical HLQs, each with fewer profiles, for the same number of physical HLQs. This is a complex process, often requiring assistance from IBM.

Overview

- Generic Profile Load Performance Improvement
  - The number of sets of profiles cached is now configurable via the RACF ***SET*** command, system wide or by job name, in the event that 4 is not sufficient.
    - The maximum number of GATEs per address space can be up to 99.
    - The minimum/default is 4.
  - An additional ***TRACE*** option is added to the RACF ***SET*** command to capture data about the caching of generic profiles to assist IBM support in diagnosing problems.

Overview

- Generic Profile Load Performance Improvement
  - Significant internal improvements and changes to RACF's caching and organization of generic profiles was also part of this support.
    - The profiles are no longer cached in ELSQA, but in 64-bit storage.
    - These lists of profiles (GATEs) are no longer searched sequentially, but in a hybrid way consisting of sequential and binary search.

Overview

- Generic Profile Load Performance Improvement …
  - Significant internal improvements …
    - The **ICHEINTY** macro (RACF low level database interface) has been changed to return more than one generic profile name, from the same level 1 index block, that also have the same class name or **HLQ** (**H**igh **L**evel **Q**ualifier).
    - The data is returned in the workarea specified by the caller which must be at least 4K in size.

Usage & Invocation

- The configuration of caching sets is implemented via the RACF **SET** command.

```
SET ...
   [ GENERICANCHOR(
     {SYSTEM | JOBNAME(jobname ...) }
     {COUNT( number ) | RESET })
   ]
```

- **SYSTEM** can be used to increase the number of generic profile caches system wide, for all jobs which do not have an overriding value.
- The additional tracing can be activated using the SET command.

```
SET ... [ TRACE ( ...{ GENERICANCHOR...} )
```


Usage & Invocation

- The **SET LIST** command output has been updated to include information about the caching and tracing configuration.

```
RACFR12 IRRH005I (@) RACF SUBSYSTEM INFORMATION:
  TRACE OPTIONS - IMAGE
  - NOAPPC
  - SYSTEMSSL
  - RACROUTE
    2 5 9
  - NOCALLABLE
  - NOPDCALLABLE
  - NODATABASE
  - GENERICANCHOR (or NOGENERICANCHOR)
  ...
  PASSWORD SYNCHRONIZATION IS *NOT* ALLOWED
  AUTOMATIC DIRECTION OF APPLICATION UPDATES IS *NOT* ALLOWED
  GENERICANCHOR:
   SYSTEM: COUNT(nn)
   JOBNAME: job1 COUNT(nn)
   job2* COUNT(nn)
```

Usage & Invocation

- **ICHEINTY** macro changes
  - The **ICHEINTY** macro (RACF low level database interface) has been changed to add a new option, ***INDEX=MULTIPLE***
  - The new ***INDEX=MULTIPLE*** option tells RACF to return more than one generic profile name from the same level 1 index block that also have the same class name or **HLQ** (**H**igh **L**evel **Q**ualifier).
  - The data is returned in the workarea specified by the caller which must be at least 4K in size.

Usage & Invocation

Data returned by INDEX=MULTIPLE option on ICHEINTY

Offset	Offset (hex)	Length	Description
0	0	4	Length of entire work area
4	4	6	Reserved
10	A	1	Reserved
11	B	1	Reserved
12	C	4	Reserved
16	10	8	Reserved
24	18	4	Length of data returned into work area
28	1C	2	Number of profile names returned
30	1E	1	Flags:
			1... .... (x'80') No more profiles after this set
31	1F	1	Reserved
32	20	variable	Start of list of returned profile names. Each returned name has the format:
		1	Length of the profile name
		variable	Profile name

Usage & Invocation

- Other changes:
  - **ICHPRCVT** - New data area to contain **GENERICANCHOR** settings
  - **IRRSAFT** - New indicator containing TRACE setting for **GENERICANCHOR**
  - **ICHGAPL**, **ICHGRPF** both have changes to support the new function.

A peek at a 'gate' and a 'gprf (pr64)'

```
ASID(X'0263') ADDRESS(7FFF8920.) STORAGE ---------------------------------
Command ===> L 009BDAC8+70?
7FFF8920   C7C1E3C5   02000000   E1000038   C4E2D5D9   | GATE........DSNR |
7FFF8930   40404040   00010000   00000000   00000000   |     ............ |
7FFF8940   00000000   7FFF8B08 ! 00000008   00800000   | ....".......... |
7FFF8950   C6B86FD4   8AE2E98D                         | F.?M.SZ.         |

ASID(X'0263') ADDRESS(08_00800000.) STORAGE ------------------------------
Command ===>
_0800000   D7D9F6F4   00000001   00000000   00000000   | PR64............ |
_0800010   00000000   00000000   00000008   00600000   | .............-.. |
_0800020   00000008   008FFFD8   00000008   008FFFD8   | .......Q.......Q |
_0800030.:08_008FFFCF. LENGTH(X'0FFFA0')--All bytes contain X'00'
_08FFFD0   00000000   00000000   00058000   00000000   | ................ |
_08FFFE0   00000008   00600010   00000000   00000000   | .....-.......... |
_08FFFF0   0202C4C2   FE000000   00000000   00000000   | ..DB............ |
```

Usage & Invocation

- New/Changed Messages
  - **IRRH083I** THE GENERICANCHOR KEYWORD REQUIRES ADDITIONAL SPECIFICATION.
  - **IRRH084I** THE SET COMMAND HAD NO EFFECT ON THE GENERICANCHOR SETTINGS.

Late Breaking Info

- APAR OA34311 RACHECK ABEND0C4 PIC3B SUBSPACE MODE
- APAR OA34332 NO details –are- available ☹
- APAR OA34694 ABEND0C4 in IRRGLS22 when sharing ACEEs

Session Summary

- Customers that may have had RACF or system performance impacted when processing generic profiles for resource classes that are neither **RACLIST**ed nor **SETROPTS GENLIST**ed, should not be adversely impacted any longer.
- Not necessary any more to resort to any of the workarounds that have been used by affected customers to solve these problems:
  - Splitting the RACF database to reduce contention, or
  - Physically renaming data sets to reduce the number of generic profiles under a single **HLQ**, or
  - Doing an analysis of their existing generic profiles to try to reduce their numbers, or
  - Implementing a RACF Naming Convention Table so RACF will pretend that the data sets have different HLQs, and thus can end up with more logical HLQs, each with fewer profiles, for the same number of physical HLQs. This is a complex process, often requiring assistance from IBM.
  - Some late breaking INFO..

Appendix

- **Publications affected**
  - **GA22-7680** - z/OS Security Server RACF Data Areas
  - **GA22-7689** - z/OS Security Server RACF Diagnosis Guide
  - **SA22-7681** - z/OS Security Server RACF System Programmer's Guide
  - **SA22-7682** - z/OS Security Server RACF Macros and Interfaces
  - **SA22-7683** - z/OS Security Server RACF Security Administrator's Guide
  - **SA22-7686** - z/OS Security Server RACF Messages and Codes
  - **SA22-7687** - z/OS Security Server RACF Command Language Reference

Questions,
comments?