

## USS / RACF Mapping Methods

New York RACF User's Group

Speaker: Russ Hardgrove  
RACF IV2  
hardgrov@us.ibm.com

### Trademarks

For a list of trademarks see URL :

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

## Agenda

- . A bit of history (a quick look back)
- . The 'three' methods. Plus VLF...
- . What we recommend, and why..
- . Demise of BPX.DEFAULT.USER

## A look back

- . Why? USS (OMVS on z/OS) uses the binary integer 0 to 2,147,483,647.
- . RACF uses their traditional 1-8 char userids and groups (A-Z, 0-9 and @, \$, and #) .
- . HOW to tie them together..
- . Commands in USS that show UID/ GID want to show a corresponding RACF userid versus a UID and a RACF group versus a GID (uid 0 is usually the interesting case here).
- . When USS wants to do something "RACF wise", who's userid to actually USE?
- . Why mapping is key (and the last one critical..)

## The 'three' methods

1. **Initially a SERIAL search of the RACF DB.** Reading each userid (in collating sequence) to see if they are are UID n. Or each group looking for GID n. Painfull, to say the least. And I've recently SEEN it occur too.
 

VLF (not a method per se) was then added, but that ONLY helped SECOND and subsequent mapping and then ONLY if a VLF purge had not occurred.
2. **Use of UNIXMAP class profiles.** Used standard ACCESS list (first user or group ON the access list). Unnn where nnn was the UID. Or Gnnn where nnn was the GID. These were 'klunky'. By that, we mean a) easy to reset access list b) delete c) were 'large'. Having many of them took up space in the RACF DB. If class is populated, the class MUST be active.
3. **Use of alias indices.** Requires a one time DB conversion. Running IRRIRA00 utility on the RACF DB. Will cover this conversion in a later slide. Use of an alternate index structure for fast lookup. Uses FAR less DB space. And can NOT be easily "reset" (as UNIXMAP profiles sometimes are). Initially VLF support was dropped, but re-introduced via APAR OA02721.

## IRRIRA00 conversion

- Do NOT need to first intro UNIXMAP. But can't imagine a shop doing any mapping with method 1 (and surviving to tell about it)
  - No longer really need to pause after stage 1 or 2. Best to go fully to stage 3.
  - STAGE1 means to keep indices up-to-date as well as UNIXMAP, but use only UNIXMAP for actual mapping.
  - STAGE2 means keep both up-to-date, use indices for mapping, but if a failure (not found) occurs fall back to UNIXMAP. Cut a LOGREC record in that case as well.
  - STAGE3 use only indices, UNIXMAP profiles gone and never used or kept up.
  - This method is REQUIRED if considering using BPX.UNIQUE.USER. Also is (and always was) required if other things like AUTOUID and AUTOUID are to be employed.
- See INFO APAR II12972

UPDATED / CONSOLIDATED DOCUMENTATION FOR IRRIRA00: CONVERSION UTILITY TO BUILD AIM INDICES AND REMOVE UNIXMAP PROFILES

## Demise of BPX.DEFAULT.USER

- Mentioned in z/OS V1R13 preview (i.e. an early) announcement under 'Statements of Direction'.

"z/OS V1.13 is planned to be the last release to support BPX.DEFAULT.USER. IBM recommends that you either use the BPX.UNIQUE.USER support that was introduced in z/OS V1.11, or assign unique UIDs to users who need them and assign GIDs for their groups."

- Can't go into great detail here and now. Recommend planning now for either assignment of uids / gids as necessary or use of BPX.UNIQUE.USER. And implicit in that is the need to employ method 3 → Use of Alias Indices..
- It's intended replacement (if you wish to not give each user an overt OMVS segment) is BPX.UNIQUE.USER

## Session Summary

- **A bit of history (a quick look back).** Why the need for mapping
- **The 'three' methods. Plus VLF...** bad to best
- **What we recommend, and why..** Implicitly covered as use of indices and use of BPX.UNIQUE.USER
- **Demise of BPX.DEFAULT.USER** - briefly. Couldn't embellish too much.

## 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

## Extra Slides:

### 18.4.3.1 Special RRSF considerations for automatic unique IDs

When you implement unique ID assignment using the BPX.UNIQUE.USER profile, RACF updates the OMVS segments of user and group profiles. If your installation synchronizes profiles in the USER and GROUP classes, you should enable automatic direction of application updates for these classes.

Example:

```
SET AUTOAPPL
RDEFINE RRSFDATA AUTODIRECT.*.USER.APPL UACC(READ)
RDEFINE RRSFDATA AUTODIRECT.*.GROUP.APPL UACC(READ)
```

Questions,  
comments?

