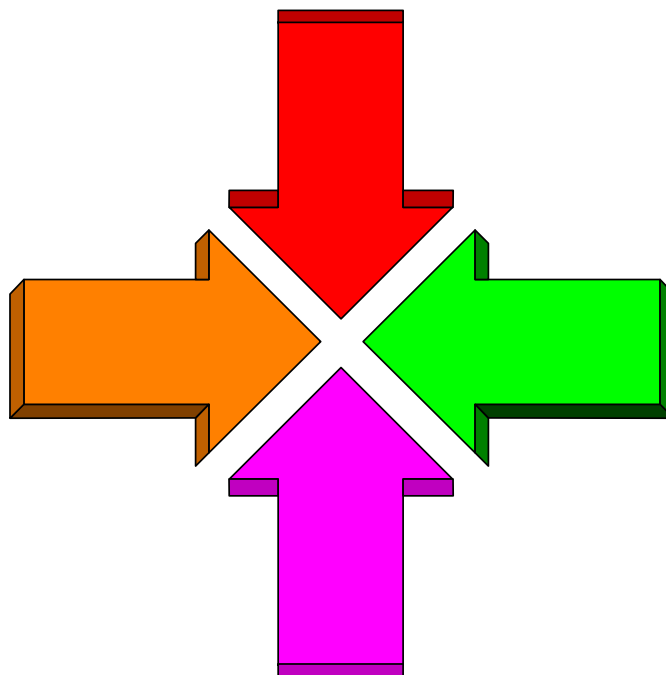


MERGING RACF DATABASES

Vanguard Enterprise Security Expo 2009 - Session RAA4 - June 2009



Robert L. Whittle

Senior RACF Specialist - RSH Consulting, Inc.

R.Whittle@rshconsulting.com - 617-969-9050 - www.rshconsulting.com

TOPICS

Planning & Preparation

Cleanup

Synchronization

Execution

RACF, OS/390, and z/OS are Trademarks of the International Business Machines Corporation

RSH-RDELTA and RSH-RCAPTURE are Trademarks of RSH Software, Inc.

PLANNING & PREPARATION

Obtain management commitment

Assemble project team

- **RACF Analysts**
- **Systems Programmers**
- **Application Developers**
- **End-user Representatives**

Obtain use of Systems Test environment (if available)

Stabilize RACF - minimize changes

- **Curtail non-essential RACF administration**
- **Suspend other RACF projects**
- **Complete merge expeditiously**

APPROACH

Consider scenario

- **Final configuration objective**
 - Existing system images will share merged RACF database
 - Existing system images will each have copy of merged database (RRSF)
 - System images being merged along with RACF database
- **Extent of class overlap & profile conflicts - requires remediation**
- **System & RACF exit & table differences - requires remediation**
- **Database size**
- **Project timeframe & resources**
- **RACF release(s)**
- **RACF database structure - pre/post-AIM reorganization**

Alternatives

- **Use RACF commands to migrate profiles**
- **Use IRRUT400 to merge databases**

SOFTWARE TOOLS

IRRDBU00	database unload - profile info & diagnostics
DBSYNC	generate commands to synchronize profiles
PWDCOPY	copy passwords from one database to another
IRRUT400	merge databases & flag conflicts
IRRUT200	database copy & diagnostics
RSH-RDELTA	compare unloads & identify all differences
RSH-RCAPTURE	compare z/OS & RACF options & tables

CLEANUP

Verify RACF database integrity & fix errors

- **IRRUT200 - Database copy & verification**
- **IRRDBU00 - Profile anomalies**
- **IRRUT400 - Database split/merge/extend utility**

Review and clean up RACF databases

- **Focus on areas of overlap**
- **Remove obsolete profile references - run IRRRID00**
- **Deactivate unused classes**
- **Eliminate obsolete users and groups**
- **Clean up Global Access Table entries**
- **Eliminate obsolete Started Tasks**
- **Delete obsolete dataset and general resource profiles**

SYNCHRONIZATION

Exits & Interfaces

Tables

SETROPTS Options

Started Tasks

Global Access Table

Users & Groups

Dataset Profiles

General Resource Profiles

Administration

Maintaining Synchronization

SYNCHRONIZATION

Analysis Questions:

- Will a table entry, exit, option, or class that is active on one system adversely affect the operation of the other?
- Will a profile and its access list on one system adversely affect access on the other by either granting or denying access?
- Will an attribute held by a user on one system adversely affect access on the other by either granting or denying access?

When practical, implement configuration and profile changes before the merge to ensure they work properly

Careful, thorough analysis is essential -- option-by-option, class-by-class, profile-by-profile -- down to individual fields as necessary

SYNCHRONIZE EXITS & INTERFACES

Due to IPL requirements, make this an early objective

RACF Exits

- **RACF Pre/Post-Processing Exits**
- **Password & Encryption Exits**
- **Command Exits**

Exits & interfaces using RACF

- **System product exits & external security interface options**
- **Application software interfaces**
- **Local system modifications (e.g., JES, DADSM exits, PPT)**

Considerations

- **Each system can have its own exits and remain independent**
- **Prefer having common set of exits used by all systems**

SYNCHRONIZE TABLES

Due to IPL requirements, make this an early objective

**Class Descriptor & Router Tables - ICHRRCDE & ICHRFR01
and/or CDT class**

- **Characteristics of installation & software vendor classes**
- **POSIT values**
- **Profile length & composition**
- **Default Return Codes**
- **OPERATIONS authority honored**
- **GENLIST / RACLIST allowed**

Dataset Name Table - ICHRDSNT

Naming Conventions Table - ICHNCV00

Database Range Table - ICHRRNG

Authorized Caller Table - ICHAUTAB

SYNCHRONIZE SETROPTS OPTIONS

Class status - need to equalize

- **ACTIVE / WHEN(PROGRAM)**
- **GENERIC**
- **GENCMD**
- **RACLIST**
- **GENLIST**
- **AUDIT**
- **LOGOPTIONS**
- **STATISTICS**

Beware DFTRETC=8 classes - no profile = no access

If a class is active on only one system, may have to implement on the other before merge (e.g., OPERCMDS)

SYNCHRONIZE SETROPTS OPTIONS

Control & audit options (partial list)

- **PROTECTALL**
- **EGN**
- **JES(BATCHALLRACF)**
- **TAPEDSN**
- **PREFIX**
- **ERASE**
- **GRPLIST**
- **PASSWORD(INTERVAL REVOKE WARNING HISTORY RULEn)**
- **SAUDIT, OPERAUDIT, APPLAUDIT**

Alternatives for addressing inconsistencies

- **Revert to least-restrictive (less secure) setting**
- **Implement option on system where inactive**

SYNCHRONIZE STARTED TASKS

Due to IPL and software change requirements, make this an early objective

STARTED class profile & ICHRIN03 table entry analysis

- **Identify all Started Tasks & corresponding profile / entry**
- **Compare all elements of STDATA segment & ICHRIN03 entries**
 - **Address ID assignment differences - implement matching ID**
 - **Address Trusted / Privileged differences**
- **Address Started Task ID differences**
 - **Attributes - SPECIAL, OPERATIONS, PROTECTED**
 - **OMVS uid / gid**
 - **Default Group**
 - **Access permissions**
- **Address default entry differences - ID, group, attributes**

IPL with changes prior to merge to verify they work correctly

SYNCHRONIZE GLOBAL ACCESS TABLE

Analyze & compare

- Entries -to- Profiles on each system
- Entries -to- Entries on merging systems

Avoid undermining access restrictions

Build & implement single, composite table prior to merge

GLOBAL -- DATASET

System 'A'

SYS1.HELP / READ

SYS1.MACLIB / READ

System 'B'

SYS1.RACF / NONE

SYS1.** / READ

Different schemes - which one to adopt?

SYNCHRONIZE USERS & GROUPS

User Profiles

- **Attributes (e.g., PROTECTED, OPERATIONS, RESTRICTED)**
- **Installation Data - may be using for control (e.g., INFOPAC)**
- **Password Interval**
- **OMVS Segment - uid assignment**
- **TSO Segment - TSOPROC, ACCTNUM differ by system**
- **CICS Segment - OPIDENT assignment**
- **Other Segments**

Group Profiles

- **Superior Group**
- **Installation Data**
- **TERMUACC**
- **OMVS Segment**

SYNCHRONIZE USERS & GROUPS

Beware USERID-Group collisions - will have to convert one

- **Conversion process (assumes DATASET profiles exist)**
 - **Deactivate PROTECTALL (if applicable)**
 - **Delete DATASET profiles**
 - **Remove users from group (if applicable)**
 - **Delete existing user / group profile & all references (use IRRRID00)**
 - **Create replacement group / user profile**
 - **Recreate DATASET profiles**
 - **Issue SETR GENERIC(DATASET) REFRESH**
 - **Reactivate PROTECTALL (if applicable)**
- **May need to add or delete general resource profiles containing embedded IDs (e.g., JESSPOOL)**
- **May have to alter access permissions if group permissions are eliminated**

SYNCHRONIZE DATASET PROFILES

Dataset Profile Options

- **UACC**
- **WARNING**
- **ERASE**
- **VOLUME (Discretes)**
- **AUDIT Levels**
- **Installation Data**
- **Access Lists**

Undercutting access - merged profiles interlace with existing ones

- **Affects HLQs common to both systems**
- **Assess similar but different coding scheme - HLQ.** vs HLQ.*.****
- **Equalize profiles on both systems before the merge**

SYNCHRONIZE GENERAL RESOURCES

General Resource Profile Options

- **UACC**
- **WARNING**
- **AUDIT Levels**
- **Installation Data**
- **APPLDATA**
- **Access Lists**

Undercutting access - merged profiles interlace with existing ones

- **Affects classes common to both systems, even if no profiles on one**
- **Assess similar but different coding scheme - catch-all ** vs ***
- **Equalize profiles on both systems before the merge**

SYNCHRONIZE GENERAL RESOURCES

Member / Grouping classes

- **Identify every profile where all resources are defined**
 - **Defined to both Member and Grouping profiles**
 - **Defined to more than one Grouping profile**
- **Determine & assess RACLIST results**
 - **Combined profile list**
 - **Composite profiles**
 - ◆ **UACC – lowest level of access**
 - ◆ **Permit – combines access list with highest level of access**
 - ◆ **WARNING – first profile encountered**
 - ◆ **Auditing – most inclusive**

RACFVARS - RACF Variables

- **Identify matching variables and assess combination of values**
- **&RACLNDE - RACF Local Nodes - used by NJE**

SYNCHRONIZE GENERAL RESOURCES

Node-specific profiles - JESSPOOL & JESJOBS

- JESJOBS profiles - *SUBMIT.nodename.jobname.userid*
- JESSPOOL profiles - *nodename.userid.jobname...*
 - JESSPOOL - differing implementation schemes create conflicts
 - NODEA user based **.userid*. ***
 - NODEB jobname based **.*.jobname*. ***
 - Possible solution - make profiles node-specific
 - NODEA.*userid*. ***
 - NODEB.**.jobname*. ***

JES-specific profiles - WRITER, SDSF, OPERCMDS

- WRITER profiles - *jesname.resource* (e.g., JES2.NJE.NODEX)
- SDSF profiles - *resource.jesname* (e.g., ISFCMD.ODSP.JESX)
- OPERCMDS profiles - *jesname.command* (e.g., JESA.DISPLAY.JOBS)
- If JES names match (e.g., JES2), rename one & change profiles

SYNCHRONIZE GENERAL RESOURCES

SDSF - server-name specific profiles

- Profile - **GROUP.groupname.servername** (e.g., GROUP.G1.SDSF)
- If SDSF server names match (e.g., SDSF), rename one & change profiles

CICS, DB2, IMS, etc. - optional installation-defined class names

- **Conflicts may arise when regions share classes (e.g., TCICSTRN)**
 - Resources collide (e.g., same transcode used by different applications)
 - Generics undercut one another
 - Different implementation schemes - ** with UACC READ / NONE
- **Possible solution - shift regions to different classes**
 - Define new set(s) of resource classes
 - Migrate profiles from old to new classes
 - Modify region(s) to begin using new classes
 - Delete profiles from old classes
 - End result - no conflict

SYNCHRONIZE GENERAL RESOURCES

Other problem classes

- **UNIXMAP - Deactivate, delete profiles, and rebuild after merge**
- **RACGLIST - Deactivate, delete profiles, and rebuild after merge**

Unix System Services issues

- **HFS permissions**
- **uid / gid assignments**
- **BPX.DEFAULT.USER & associated ID-uid, group-gid**

SYNCHRONIZE ADMINISTRATION

Merge may cause changes in ownership / group hierarchy

- **Ownership change affects decentralized administration**
 - **Group-Special**
 - **Group-Auditor**
 - **Group-Operations**
- **Ownership / group hierarchy may need redesign**

Areas of concern - may broaden authority

- **FIELD class profiles**
- **FACILITY class IRR.PASSWORD.RESET**
- **User Class Authorization (CLAUTH)**

MAINTAINING SYNCHRONIZATION

Track profile changes due to normal administration during project

Focus on areas of overlap

Make adjustments as needed to remain synchronized

For critical classes and profiles, review synchronization just prior to merge execution

EXECUTION

General

- Execute during dedicated system maintenance period
- Perform in isolation - suspend all other system activity
- Be on-call and standing by for rapid problem resolution
- Verify logon and RACF command execution at the system console is working before performing the merge
- Have RVARV passwords readily available
- Ensure IRRUT200 backups are prepared and accessible

If user profiles were simply added via RACF commands, migrate passwords

- Test password migration prior to final event
- Perform final password copy just prior to activation

If using IRRUT400 - see next slides

EXECUTION - IRRUT400 MERGE

Merging different databases not an "officially" sanctioned use

Sample merge job

```
//RSHMERGE JOB
//          EXEC PGM=IRRUT400,
//          PARM='NOLOCKINPUT,NODUPDATASET,FREESPACE(50),ALIGN'
//SYSPRINT DD SYSOUT=*
//INDD1    DD DSN=COPY.SYSA.RACF.DATABASE,DISP=OLD
//INDD2    DD DSN=COPY.SYSB.RACF.DATABASE,DISP=OLD
//OUTDD1   DD DSN=NEW.MERGED.RACF.DATABASE,DISP=(NEW,CATLG,DELETE),
//          UNIT=SYSDA,VOL=SER=RPVOL,
//          DCB=DSORG=PSU,
//          SPACE=(CYL,20,,CONTIG)
```

Merges whole profiles only - does not combine them

INDD1 - "base" database - SETROPTS options, templates, and profiles take precedence

INDD2 - "merge" database - profiles are incorporated only if they do not already exist in base; otherwise, they are dropped

EXECUTION - IRRUT400 MERGE

Breaks inter-record references when duplicate user and group profiles are dropped

- **User / Group membership & Group authority**
- **Superior / Subordinate group**

Can result in lost permissions and grouping class profile members when duplicate resource profiles are dropped

Requires execution of commands to fix references and restore permissions immediately after the merge

RACF configuration issues

- **Merge to the higher RACF release**
- **Always use the RACF utilities associated with the higher release**
- **Only merge databases with same AIM level**

EXECUTION - IRRUT400 MERGE

Make IRRUT200 backup copies of all RACF databases

On test system (make sure up level on RACF and has updated tables & exits previously implemented):

- **Transfer copies of databases to be merged to test system**
- **Generate IRRDBU00 unload for each database copy**
- **Execute Merge Utility IRRUT400 with database copies**
 - **Inspect results to confirm expectations (e.g., duplicates dropped)**
- **Activate merged database (RVARY "dance") and IPL**
 - **For recovery, backup / rename prior database rather than delete**
- **Make backup copy using IRRUT200 and unload using IRRDBU00**
 - **Inspect IRRUT200 & IRRDBU00 for integrity errors**
- **Generate Post-Merge Commands**
 - **Generate commands using database unload comparison software**
 - **Prepare additional hand-coded commands as needed (in advance)**

EXECUTION - IRRUT400 MERGE

Still on test system:

- **Run Post-Merge Commands to fix:**
 - **Broken User Default Groups**
 - **Broken User Connects**
 - **Broken Group Hierarchy**
 - **Lost Permits**
 - **Lost Grouping Class Profile Members**
 - **Inspect all command execution results & spot check profiles**
- **Make backup copy using IRRUT200 and unload using IRRDBU00**
 - **Inspect IRRUT200 & IRRDBU00 for integrity errors**
- **Test**
 - **System testing**
 - **End user testing**
- **If dry run, restore prior RACF database (RVARY "dance")**
- **If final implementation, make backup copy using IRRUT200 (if adjustments were made) and transfer to Production systems**

EXECUTION - IRRUT400 MERGE

Production implementation

- Build merged database on test system and transfer to production
- If database name unchanged for certain systems:
 - Activate merged database (RVARY "Dance")
 - SETROPTS GENERIC & RACLIST REFRESH everything
 - IPL recommended for at least one system in each Sysplex
 - IPL one system at a time
 - IPL for other systems optional but preferred
 - If not IPLing, perform system software refreshes (e.g., MQSeries)
- If database name changed for certain systems:
 - Implement new ICHRDSNT
 - IPL all related systems

May want to change database names pre-merge to avoid last minute implementation of new ICHRDSNT

EXECUTION - IRRUT400 MERGE

RVARY "Dance"

- **Requires RVARY passwords**
- **Make IRRUT200 backups of current databases beforehand**
- **Process overview**
 - **Switch to current Backup database (deactivates current Primary; current Backup becomes Primary)**
 - **Replace old Primary with copy of merged database - retain prior dsname**
 - **Activate merged database (becomes new Backup)**
 - **Switch to new Backup database (deactivates old Primary (was Backup))**
 - **Replace old Backup with copy of merged database - retain prior dsname**
 - **Activate new Backup**
- **Ensure new databases are cataloged and properly allocated (contiguous, no extents, non-SMS, non-movable)**
- **Ensure DASD volume has enough space for merged database**

Ensure you can access RACF database DASD volumes from other systems to recover prior databases if necessary