RSH CONSULTING

RACFVARS

RUGONE - October 2013

Robert S. Hansel Lead RACF Consultant R.Hansel@rshconsulting.com 617-969-9050

Robert S. Hansel

Robert S. Hansel is Lead RACF Specialist and founder of RSH Consulting, Inc., an IT security professional services firm he established in 1992 and dedicated to helping clients strengthen their IBM z/OS mainframe access controls by fully exploiting all the capabilities and latest innovations in RACF. He has worked with IBM mainframes since 1976 and in information systems security since 1981. Mr. Hansel began working with RACF in 1986 and has been a RACF administrator, manager, auditor, instructor, developer, and consultant. He has reviewed, implemented, and enhanced RACF controls for major insurance firms, financial institutions, utilities, payment card processors, universities, hospitals, and international retailers. Mr. Hansel is especially skilled at redesigning and refining large-scale implementations of RACF using role-based access control concepts. He has also created elaborate automated tools to assist clients with RACF administration, database merging, identity management, and quality assurance.

Contact and background information:

- 617-969-8211
- R.Hansel@rshconsulting.com
- www.linkedin.com/in/roberthansel
- www.rshconsulting.com

RACFVARS

- RACFVARS - RACF Variables

- Used to define variables containing lists of character strings intended to match characters in a resource name

```
RACFVARS      &FN             ADDMEM( NYC  CLE )
NODES         &FN.USERJ.*     NYC.USERJ.MIN003
                              CLE.USERJ.CA7USR
```

- Used to build a single profile that can protect multiple resources having dissimilar names
  - Most useful with classes lacking a grouping class
  - Also valuable when qualifiers exist across multiple classes (e.g., JES nodes)

- Can only be used with General Resource profiles and not Dataset profiles

RACFVARS - CDT Attributes

- RACFVARS (RVARSMBR)
  - CASE = UPPER
  - DFTRETC = 4
  - DFTUACC = NONE
  - DYNAMIC = NO
  - FIRST = ANY
  - GENLIST = DISALLOWED
  - ID = 74 (73)
  - KEYQUAL = 0
  - MAXLENX = 8 (39)

  MAXLNTH = 8 (39)

  OPER = NO

  OTHER = ALPHANUM (ANY)

  POSIT = 102

  PROFDEF = YES

  RACLIST = ALLOWED (DISALLOWED)

  RACLREQ = YES (NO)

  RVRSMAC = NO

  SLBLREQ = NO

RACFVARS

- RACF variables are defined as discrete profiles in RACFVARS Class

```
RDEFINE  RACFVARS  &ABCLST
```

- RACFVARS profile names
  - Begin with an ampersand '&'
  - Up to 8 characters in length, including the &
  - May not contain generic masking characters or periods (.)
  - Prefix &RAC should be reserved for IBM use
  - &RACUID and &RACGPID are not RACFVARS variables

- Variable character strings
  - Added as members to the RACFVARS profile

    ```
    RALTER  RACFVARS  &ABCLST  ADDMEM( OPN1 DTL3 )
    ```

  - May be up to 39 characters in length
  - May not contain generic masking characters (can be added, but will be ignored)
  - Can match one, several, or all qualifiers of a resource name as well as partial qualifiers (e.g., LOCAL.PRT4)

RACFVARS - Sample

```
RLIST RACFVARS &PAYPRTR
CLASS      NAME
-----      ----
RACFVARS   &PAYPRTR

RESOURCES IN GROUP
--------- -- -----
LOCAL.PRT75
LOCAL.PRT82
LOCAL.PRT94
```

```
RLIST WRITER JES2.&PAYPRTR AUTH
CLASS      NAME
-----      ----
WRITER     JES2.&PAYPRTR

LEVEL  OWNER      UNIVERSAL ACCESS   YOUR ACCESS  WARNING
-----  --------   ----------------   ---- ------  -------
 00    SYSTEMS         NONE                NONE    NO

INSTALLATION DATA
---------------------------------------------------
PAYROLL SPECIAL PRINTERS

USER      ACCESS    ACCESS COUNT
----      ------    ------ -----
PAYSPVR    READ
COMPOPER  ALTER
```

Protects resource named …
JES2.LOCAL.PRT75
JES2.LOCAL.PRT82
JES2.LOCAL.PRT94

Caution: Undercut by profile JES2.LOCAL.** - be careful of using multiple qualifiers in RACFVARS

RACFVARS

- Within a profile, variable names are terminated by:

Period .	X.&USERVAR.YZ
Masking character % *	X.&USERVAR*
Another variable &	X.&USERVAR&V1.*
End of the profile	X.&USERVAR
8th character following the &	X.&USERVARABC (&USERVAR + ABC)

- May be used in combination and with other masking

  &RACLNDE.&ABCLST.*.*

- In order of precedence, '&' is considered more specific than other generic characters

RACFVARS

- RACFVARS profile UACC and access list determine who can view and administer the profile
  - NONE - No ability to list the profile
  - READ - List the profile using RLIST
  - ALTER - Add/Delete variable members and permissions
  - Access to a variable does not grant access to profiles using the variable

- RACFVARS must be RACLISTed, and if changed, must be refreshed

```
SETROPTS  RACLIST( RACFVARS )  [ REFRESH ]
```

- Any class with profiles containing a RACFVARS variable that has been changed must be also be refreshed, either ...

```
SETROPTS  RACLIST( class )  [ REFRESH ]
SETROPTS  GENERIC( class )  [ REFRESH ]
```

RACFVARS - &RACLNDE

- &RACLNDE - Local JES nodes

```
RACFVARS  &RACLNDE  ADDMEM( MYNODEA  MYNODEB )
```

- JES nodes defined to &RACLNDE are regarded as "TRUSTED" nodes
  - Equivalent to defining NODES *nodename*.*.* UACC(CONTROL)
  - NODES class is ignored for nodes listed in &RACLNDE - submitter USERID is automatically propagated as is
  - Intended to expedite the routing of jobs between internal nodes
  - Recommendation: &RACLNDE should only include names of JES nodes sharing the same RACF database

- Often used with JESJOBS and JESSPOOL profiles

```
JESJOBS     CANCEL.&RACLNDE.*
JESSPOOL  &RACLNDE.IBMUSER.*
```

- Always define &RACLNDE and add the local nodename, even on a stand-alone system, to support spool reload functions

RACFVARS - Examples

- Define set of JES printers to be managed by a particular group

RACFVARS	&PAYP	ADDMEM(PRT5 PRT23 PRT33)
WRITER	JES2.LOCAL.&PAYP	PERMIT ID(PAYROLL) ACC(ALTER)

- Allow a group of TSO users access to one another's output

GROUP	PGMRGRP1	CONNECT (HRW IBS TU1)
RACFVARS	&PGMRG1	ADDMEM(HRW IBS TU1)
JESSPOOL	&RACLNDE.&PGMRG1.*	PERMIT ID(PGMRGRP1) ACC(READ)

- Allow Network Systems Programmers to manage their Started Tasks

RACFVARS	&NETSTC	ADDMEM(NETVIEW NET TCPIP FTPSERVE)
OPERCMDS	MVS.*.STC.&NETSTC.**	PERMIT ID(NETSP) ACC(CONTROL)

RACFVARS

- Variables are checked in the order they appear in the profile
  - RLIST lists the variables in alphanumeric sequence, which may not be their order in the profile
  - IRRDBU00 lists the variables in the order they will be processed

- RDEFINE adds members in the order they are listed in the command (FIFO)
  - RDEFINE &X ADDMEM(A B) results in a member list of: A B

- RALTER adds members in the reverse order they are listed in the command and adds them to the front of the list (LIFO)
  - RALTER &X ADDMEM(X Y) results in a member list of : Y X A B

- Administrative tip - when adding new members to a RACFVARS profile, delete (RDELETE) and then recreate (RDEFINE) the profile with the desired member sequence as opposed to trying to modify (RALTER) the list

RACFVARS

- RACF attempts to find the sequence of characters matching each variable string and stops when the first match is found
  - This can result in unintended matches when one name is a subset of another (e.g., NET and NETVIEW), so when possible put the subset (shorter) name last

- Problem #1
  - Intend for resource PAYU.SUBMIT to match profile &PID.SUBMIT
  - RDEFINE &PID ADDMEM(PAY PAYU)
  - RACF will match string PAY to PAYU - equivalent to PAY.SUBMIT (does not match)
  - Correction - reorder members - RDEFINE &PID ADDMEM(PAYU PAY)

- Problem #2
  - Intend for resource A1.ABC to match profile &X%.*
  - RDEFINE &X ADDMEM(A1 A)
  - RACF will match A1 to A1 in its entirety, leaving no character to match the %
  - Correction - reorder members - RDEFINE &X ADDMEM(A A1)

RACFVARS

- If the same list of character strings is needed by two different profiles but in different sequences (e.g., XA X for one and X XA for another), you will need to create two RACFVARS variables each with the list in the required sequence

- If abutting variables - &A&B - consider all possible string combinations and whether an &A member could match an intended &A&B combination (abutting variables is not recommended due to complexity)

- Variables can be used with member/grouping class pairs, either in a member class profile or in a member of a grouping class profile (use in grouping members is not recommended - cannot be found with RLIST RESGROUP)

- RACFVARS may not work with mixed-case classes
  - Variable names must be entered in upper-case - Firm.Acct.&ACCT#
  - RACF only supports upper-case variable strings - &ACCT# ADDMEM(AC2 BD3)