13 RACF

August 1998

update

In this issue

RACF Update

Published by
Xephon
27-35 London Road
Newbury
Berkshire RG14 1JL
England
Telephone: 01635 38030
From USA: 01144 1635 38030
E-mail: xephon@compuserve.com

North American office
Xephon/QNA
1301 West Highway 407, Suite 201-405
Lewisville, TX 75077-2150
USA
Telephone: 940 455 7050

Contributions
Articles published in *RACF Update* are paid for at the rate of £170 ($250) per 1000 words and £90 ($140) per 100 lines of code for original material. To find out more about contributing an article, without any obligation, please contact us at any of the addresses above and we will send you a copy of our *Notes for Contributors*.

RACF Update on-line
Code from *RACF Update* can be downloaded from our Web site at http://www.xephon.com; you will need the user-id shown on your address label.

Editor
Robert Burgess

Disclaimer
Readers are cautioned that, although the information in this journal is presented in good faith, neither Xephon nor the organizations or individuals that supplied information in this journal give any warranty or make any representations as to the accuracy of the material it contains. Neither Xephon nor the contributing organizations or individuals accept any liability of any kind howsoever arising out of the use of such material. Readers should satisfy themselves as to the correctness and relevance to their circumstances of all advice, information, code, JCL, and other contents of this journal before making any use of it.

Subscriptions and back-issues
A year's subscription to *RACF Update*, comprising twelve monthly issues, costs £185.00 in the UK; $280.00 in the USA and Canada; £191.00 in Europe; £197.00 in Australasia and Japan; and £195.50 elsewhere. In all cases the price includes postage. Individual issues, starting with the August 1995 issue, are available separately to subscribers for £46.50 ($70.00) each including postage.

An extension to SETROPTS

INTRODUCTION

The distinction between the function of a security officer and that of a systems programmer, however logical nowadays, has not always been so clear. In the old days, it was often a member of the systems group that maintained the security profiles. Looking at RACF, one wonders what knowledge the ideal person should possess to administer the security items of a large mainframe site. The person in question should not only be knowledgeable about the working of RACF, but also of Assembler language (RACF exits, privileged instructions), MVS internals (the APF mechanism, SVC processing, supervisor mode), JES2 processing (SDSF protection, operator commands), SYS1.PARMLIB specifics (the program properties table, subsystems), IPL happenings (one pack bootable disks, SVC table), networking (IP spoofing, GTF traces), the organization itself (functional groups), printing (CA-View, VPS), and so on.

These requirements make it hard to fill in the profile, especially since, with these qualifications, it often pays better to do system work than to become a security officer. A lot of mainframe shops compromise on these requirements by delegating part of the responsibility for security to the systems group. Sometimes this strategy works, often it doesn't – for the simple reason that systems people in general tend to look upon security as something that offers them nothing but trouble. This attitude is mostly misunderstood by the security people because they are sometimes unaware of the freedom a systems programmer needs to accomplish his tasks. Even worse, a cold war sometimes occurs between the two departments – a war that the security people can only lose (see *Software and hardware protection mechanisms, RACF Update,* Issue 8, May 1997).

In this article we concentrate on the SETROPTS command, a RACF processor that allows one to set the global options for the entire system. The way RACF is designed, I often wonder whether IBM is aware of the different needs that exist for systems programmers and

security officers. From a practical point of view, a systems programmer sometimes needs to know things about RACF, the global options settings being a nice example. However, the way RACF is devised leaves no space for the security officer to grant this authority, other than by giving away all his privileges – namely by giving the systems programmers the attribute of SPECIAL. As a systems programmer, I don't want either this attribute or OPERATIONS. On the other hand, it annoys me if I cannot perform my work the way I should do when RACF stops me doing the things I want to do.

One way around this is fairly simple, I can stop RACF (this takes about 20 lines of Assembler), change my user attributes to SPECIAL (see the above mentioned article), do whatever I feel like, and finally reset everything. This can be accomplished by any systems programmer, without leaving a trace. However, what annoys me about this scheme is that I really believe in the necessity for a security policy so, for ethical reasons, this is a last resort way of dealing with RACF.

Unfortunately, only a few weeks ago, I was forced to handle a database problem late at night this way – when security could not be reached and all production chains came screaming to a halt.

The alternative is to extend RACF in such a way that systems people can perform their duties without being granting the same privileges as the security team. The code below is an example of a way to achieve this. The Assembler will list all the settings of the SETROPTS command in a way controlled by RACF, without offering a means to change them. In this way we can give 'READ' access to the SETROPTS command to a limited group of people. However, one must be aware that this is purely optional, every single TSO/E user on your system can display the control blocks by the TEST command, a little Assembler program with a SNAP, or even by a few lines of REXX. The proposed solution is only viable if the system people sign a contract of honour only to do it this way, a way they can perform their tasks without bothering about RACF.

TECHNICAL NOTES

The program refers to R#PUT. This is a macro that writes to the spool

in batch and uses the ISPF browse interface in the foreground. Register 10 points to the DCB or the BRIF control block at call time. For our purposes it is sufficient to say that PUT will do the same thing in batch after pointing register 10 to the DCB (replace the lines ‘LR RA,R1’ and ‘L RA,0(RA)’ at the start of the program with ‘LA RA,A_DCB’ where ‘A_DCB’ is the label of the DCB macro. The C#REGS, EYECATCH, and AMODE24 macros do what their names indicate – the first includes a register equates, the second an eyecatcher, and the third changes the amode to 24-bit processing.

The S#TODECX routine takes a pointer to a full word containing a binary value and prints it in human-readable form in the field the second parameter is pointing to.

The program runs smoothly on a system (MVS 4.3) with RACF 1.9 installed, but should run on higher levels without any modification. The RACF profile is called JEDSP.SETROPTS and is kept is the class FACILITY, a class intended for this purpose.

RACF DEFINITIONS CLIST

```
PROC Ø
PROFILE NOPREFIX
/*                                                              */
/* Define jedsp.setropts as a profile in the class facility,    */
/* you can add notify, warning, audit, ...                      */
/* see the RACF Security Administrator's Guide for a complete   */
/* description                                                  */
/*                                                              */
RDEFINE FACILITY                                                    +
        (JEDSP.SETROPTS)                                            +
        DATA('JED:SP SYSTEM PROGRAMMER UTILITIES: SETROPTS')        +
        UACC(NONE)                                                  +
        NOTIFY(TZØØ)
/*                                                              */
/* Activate the class facility (if not already done)            */
/*                                                              */
SETROPTS CLASSACT(FACILITY)
/*                                                              */
/* This is a generic class                                      */
/*                                                              */
SETROPTS GENERIC(FACILITY)
/*                                                              */
/* Give the necessary permits – limit this for instance to    */
/*                                the systems group             */
```

```
 /*                                                                   */
 PERMIT JEDSP.SETROPTS CLASS(FACILITY) GENERIC ACCESS(READ) +
                       ID(GSYSTEMS)
 /*                                                                   */
 /* Refresh the in storage profiles                                   */
 /*                                                                   */
 SETROPTS REFRESH GENERIC(FACILITY)

         TITLE '*** JED:SP REPORTS: RACF MISCELLANEOUS OPTIONS      JANX
                DE DECKER ***'
*———————————————————————————————————
*—JAN-DE-DECKER-JED:SP-WASHUISSTRAAT-24-1ØØØ-BRUSSEL-+32(Ø)2 513Ø398—
*———————————————————————————————————
*
* NAME:
*
* PURPOSE:      RACF MISCELLANEOUS OPTIONS (SETROPTS)
*
* PARAMETERS:   AT ENTRY RA -> OUTPUT DCB
*
* LINK:
*
* SYSTEM:       MVS/ESA 4.3
*
* DATASETS:
*
* LIMITATIONS:  RACF 1.9
*
* MODIFICATION:
*
*———————————————————————————————————
*-FACSIMILE +32 (Ø)2 5138499 E-MAIL:-JAN@JEDSP.COM HTTP://WWW.JEDSP.COM
*———————————————————————————————————
         EJECT
RBØ38Ø   CSECT
RBØ38Ø   AMODE 24
RBØ38Ø   RMODE 24
         COPY  C#REGS
         PRINT NOGEN
         EJECT
         STM   RE,RC,12(RD)           STORE REGS IN SAVEA CALLER
         LR    RB,RF                  LOAD
         LA    RC,4Ø95(RB)                 BASE
         LA    RC,1(RC)                         REGISTERS
         USING RBØ38Ø,RB,RC           RB, RC ARE BASE REGISTERS
         EYECATCH
         ST    RD,SAVEAREA+4          KEEP A(SAVEA CALLER)
         LR    R9,RD                  COPY A(SAVE CALLER) TO R9
         LA    RD,SAVEAREA            RD POINTS TO OWN SAVEAREA
         ST    RD,8(R9)               STORE A(OWN SAVEAREA) IN CALLERS
```

```
         EJECT
*
* START OF PROCESSING
*
         LR    RA,R1                   RA -> A(DCB OR BRIF)
         L     RA,Ø(RA)                RA -> DCB OR BRIF
         AMODE24
*
* CHECK RACF AUTHORITY CALLER
*
         LA    R6,RACF_D1              R6 -> RACF DYNAMIC AREA
         LA    R7,RACROUL1             R7 = L(RACF MACRO)
         LA    R8,RACF_S1              R8 -> RACF STATIC AREA
         LR    R9,R7                   R9 = L(RACF MACRO)
         MVCL  R6,R8                   COPY STATIC TO DYNAMIC AREA
*
         LA    R5,RENTDE               R5 -> RESOURCE DESCRIPTION
         LA    R7,WORKA                R7 -> SAF ROUTER AREA
         RACROUTE REQUEST=AUTH,        REQUEST AUTHORITY CHECK         X
               ENTITYX=((R5),NONE),    -> GENERAL RESOURCE             X
               RELEASE=(1.9.2,CHECK)   VERIFY RELEASE                  X
               WORKA=(R7),             R7 -> SAF ROUTER AREA           X
               MF=(E,RACF_D1)          EXECUTE MACRO FORMAT
*
* CHECK RACF RETURN AND REASON CODES
*
         LTR   RF,RF                   SAF OK?
         BNZ   THE_END                 NO —> GO RETURN TO CALLER
         EJECT
         XR    R2,R2                   R2 = Ø
         USING PSA,R2                  R2 ADDRESSES PSA
         L     R2,FLCCVT               R2 = ADDRESS(CVT)
         DROP  R2
         USING CVTMAP,R2               R2 ADDRESSES CVT PREFIX
         L     R2,CVTRAC               R2 -> RCVT
         DROP  R2
         USING RCVT,R2                 R2 ADDRESSES RCVT
         CLC   RCVTID,=C'RCVT'         CHECK ID
         BE    LØØØØ                   OK —> CONTINUE
         ABEND 1ØØ,DUMP                NOK -> DUMP
         EJECT
*
* SET VARIABLES TO DEFAULTS
*
LØØØØ    DS    ØH
         MVC   ADSP,=C'NOT IN EFFECT'
         MVC   CATDSNS,=C'NOT IN EFFECT'
         MVC   COMPAT,=C'NOT IN EFFECT'
         MVC   DASD,=C'NOT IN EFFECT'
         MVC   EGN,=C'NOT IN EFFECT'
```

```
         MVC   EOS,=C'NOT IN EFFECT'
         MVC   GO,=C'NOT IN EFFECT'
         MVC   IDSN,=C'NOT IN EFFECT'
         MVC   LGC,=C'NOT IN EFFECT'
         MVC   CMDV,=C'NOT IN EFFECT'
         MVC   MLA,=C'NOT IN EFFECT'
         MVC   MLQ,=C'NOT IN EFFECT'
         MVC   MLS,=C'NOT IN EFFECT'
         MVC   MLT,=C'NOT IN EFFECT'
         MVC   MGDG,=C'NOT IN EFFECT'
         MVC   MGRP,=C'NOT IN EFFECT'
         MVC   MUSR,=C'NOT IN EFFECT'
         MVC   DUPD,=C'NOT IN EFFECT'
         MVC   PACC,=C'NOT IN EFFECT'
         MVC   PALL,=C'NOT IN EFFECT'
         MVC   SELC,=C'NOT IN EFFECT'
         MVC   TAPV,=C'NOT IN EFFECT'
         MVC   TAPD,=C'NOT IN EFFECT'
         MVC   TERM,=C'NOT IN EFFECT'
         MVC   TUAC,=C'         READ'
         MVC   SSIGNON,=C'NOT IN EFFECT'
*
* OPTION FIELDS TO BLANKS
*
         MVC   LCATDSNS,BLANKS
         MVC   LEOS,BLANKS
         MVC   LMLA,BLANKS
         MVC   LMLS,BLANKS
         MVC   LPALL,BLANKS
         EJECT
*
* LOOK-UP VALUES
*
         TM    RCVTSTAT,RCVTNADS       NO AUTOMATIC DATASET PROTECTION?
         BO    L00010                  INDEED, CONTINUE
         MVC   ADSP,=C'    IN EFFECT'  YES -> INDICATE
L00010   DS    0H
         TM    RCVTFLG2,RCVTCATD       CATDSNS ON?
         BZ    L00020                  NO -> CONTINUE
         MVC   CATDSNS,=C'    IN EFFECT'  YES -> INDICATE
         MVC   LCATDSNS,=C'FAILURES'   SET DEFAULT LEVEL
         TM    RCVTFL2X,RCVTCATF       CHECK LEVEL (== FAILURES)
         BO    L00020                  IF ON (DEFAULT)
         MVC   LCATDSNS,=C'WARNING '   SET LEVEL
L00020   DS    0H
         TM    RCVTFL2X,RCVTCMPM       COMPAT MODE ON?
         BZ    L00030                  NO -> CONTINUE
         MVC   COMPAT,=C'    IN EFFECT' YES -> INDICATE
L00030   DS    0H
         TM    RCVTSTA1,RCVTDASD       DASDVOL PROTECTION
```

```
         BZ    L0040                   NO -> CONTINUE
         MVC   DASD,=C'    IN EFFECT' YES -> INDICATE
L0040    DS    0H
         TM    RCVTSTAT,RCVTEGN        ENHANCED GENERIC NAMING?
         BZ    L0050                   NO -> CONTINUE
         MVC   EGN,=C'    IN EFFECT'   YES -> INDICATE
         EJECT
L0050    DS    0H
         TM    RCVTFLG1+1,RCVTEOS      ERASE ON SCRATCH?
         BZ    L0080                   NO -> CONTINUE
         MVC   EOS,=C'    IN EFFECT'   YES -> INDICATE
         TM    RCVTFLG1+1,RCVTEOSL     ERASE ON SCRATCH BY SECLEVEL
         BZ    L0060                   NO -> CONTINUE
         MVC   LEOS,=C'SECLEVEL'       YES -> INDICATE
         B     L0080                       -> NEXT OPTION
L0060    DS    0H
         TM    RCVTFLG1+1,RCVTEOSA     ERASE ON SCRATCH ALL DSN
         BZ    L0070                   NO -> MUST BE NOSECLEVEL
         MVC   LEOS,=C'ALL DSN '       YES -> INDICATE
         B     L0080                       -> NEXT OPTION
L0070    DS    0H
         MVC   LEOS,=C'PROFILE '       YES -> INDICATE
L0080    DS    0H
         TM    RCVTFLG2,RCVTGNOW       GENERIC OWNER?
         BZ    L0090                   NO -> CONTINUE
         MVC   GO,=C'    IN EFFECT'    YES -> INDICATE
L0090    DS    0H
         TM    RCVTSTA1,RCVTRDSN       INPUT DSN FOR LOGGING AND MESS.
         BZ    L0100                   NO -> CONTINUE
         MVC   IDSN,=C'    IN EFFECT'  YES -> INDICATE
L0100    DS    0H
         MVC   SDSN,RCVTQUAL           GET SINGLE LEVEL DSN PREFIX
         MVC   WORKWORD,=F'0'          CLEAN UP WORKWORD
         MVC   WORKWORD+3(1),RCVTQLLN  WORKWORD = L(SINGLE LEVEL DSN)
         MVC   A_REC,=AL4(LSDSN)
         MVC   LEN,=H'2'
         LA    R1,P_TODECX
         CALL  S#TODECX
         EJECT
         TM    RCVTOPTX,RCVTLGRP       LIST OF GROUPS CHECKING?
         BZ    L0110                   NO -> CONTINUE
         MVC   LGC,=C'    IN EFFECT'   YES -> INDICATE
L0110    DS    0H
         TM    RCVTEROP,RCVTAVIO       NO COMMAND VIOLATIONS LOGGING?
         BO    L0120                   YES -> CONTINUE
         MVC   CMDV,=C'    IN EFFECT'  NO -> INDICATE
L0120    DS    0H
         TM    RCVTFLG2,RCVTMLAC       MLACTIVE ACTIVE?
         BZ    L0130                   NO -> CONTINUE
         MVC   MLA,=C'    IN EFFECT'   YES -> INDICATE
```

```
         MVC   LMLA,=C'FAILURES'              DEFAULT TO FAILURES
         TM    RCVTFL2X,RCVTMLAF              FAILURES ?
         BO    L013Ø                          YES -> CONTINUE
         MVC   LMLA,=C' WARNING'              NO —> WRITE WARNING
L013Ø    DS    ØH
         TM    RCVTFLG2,RCVTMLQT       MLQUIET ACTIVE?
         BZ    L014Ø                   NO —> CONTINUE
         MVC   MLQ,=C'    IN EFFECT'   YES -> INDICATE
L014Ø    DS    ØH
         TM    RCVTFLG2,RCVTMLS        MLSECURE ACTIVE?
         BZ    L015Ø                   NO —> CONTINUE
         MVC   MLS,=C'    IN EFFECT'   YES -> INDICATE
         MVC   LMLS,=C'FAILURES'              DEFAULT TO FAILURES
         TM    RCVTFL2X,RCVTMLSF              FAILURES ?
         BO    L015Ø                          YES -> CONTINUE
         MVC   LMLS,=C' WARNING'              NO —> WRITE WARNING
L015Ø    DS    ØH
         TM    RCVTFLG2,RCVTMLST       MLSTABLE ACTIVE?
         BZ    L016Ø                   NO —> CONTINUE
         MVC   MLT,=C'    IN EFFECT'   YES -> INDICATE
         EJECT
L016Ø    DS    ØH
         TM    RCVTMDEL,RCVTMGDG       MODEL GDG ACTIVE?
         BZ    L017Ø                   NO —> CONTINUE
         MVC   MGDG,=C'    IN EFFECT'  YES -> INDICATE
L017Ø    DS    ØH
         TM    RCVTMDEL,RCVTMGRP       MODEL GROUP ACTIVE?
         BZ    L018Ø                   NO —> CONTINUE
         MVC   MGRP,=C'    IN EFFECT'  YES -> INDICATE
L018Ø    DS    ØH
         TM    RCVTMDEL,RCVTMUSR       MODEL USER ACTIVE?
         BZ    L019Ø                   NO —> CONTINUE
         MVC   MUSR,=C'    IN EFFECT'  YES -> INDICATE
L019Ø    DS    ØH
         TM    RCVTFLGS,RCVTNDUP       NO DUPLICATE DSNS
         BZ    L020Ø                   NO —> CONTINUE
         MVC   DUPD,=C'    IN EFFECT'  YES -> INDICATE
L020Ø    DS    ØH
         TM    RCVTFLG1+2,RCVTPROG     ACCESS CONTROL BY PROGRAM?
         BZ    L021Ø                   NO —> CONTINUE
         MVC   PACC,=C'    IN EFFECT'  YES -> INDICATE
L021Ø    DS    ØH
         TM    RCVTFLG1+1,RCVTPRO      PROTECT ALL?
         BZ    L0215                   NO —> CONTINUE
         MVC   PALL,=C'    IN EFFECT'  YES -> INDICATE
         MVC   LPALL,=C'WARNING '      SET DEFAULT LEVEL
         TM    RCVTFLG2,RCVTPROF       CHECK LEVEL (== WARNING)
         BO    L0215                   IF ON (DEFAULT)
         MVC   LCATDSNS,=C'WARNING '   SET LEVEL
L0215    DS    ØH
```

```
         TM    RCVTFLG2,RCVTSLCL       SECLABELCONTROL?
         BZ    LØ22Ø                   NO —> CONTINUE
         MVC   SELC,=C'    IN EFFECT'  YES -> INDICATE
LØ22Ø    DS    ØH
         MVC   WORKWORD,=F'Ø'          CLEAN UP WORKWORD
         MVC   WORKWORD+3(1),RCVTSLAU  WORKWORD = SECLEVEL TO AUDIT
         MVC   A_REC,=AL4(SELA)
         MVC   LEN,=H'3'
         LA    R1,P_TODECX
         CALL  S#TODECX
*
         MVC   WORKWORD,=F'Ø'          CLEAN UP WORKWORD
         MVC   WORKWORD+2(2),RCVTSINT  WORKWORD = SESSION INTERVAL
         MVC   A_REC,=AL4(SESI)
         MVC   LEN,=H'5'
         LA    R1,P_TODECX
         CALL  S#TODECX
*
         MVC   WORKWORD,=F'Ø'          CLEAN UP WORKWORD
         MVC   WORKWORD+2(2),RCVTRTPD  WORKWORD = SECURITY RETENTION
         MVC   A_REC,=AL4(SREP)
         MVC   LEN,=H'5'
         LA    R1,P_TODECX
         CALL  S#TODECX
         EJECT
         TM    RCVTFLG1,RCVTTDSN       TAPE DSN PROTECTION?
         BZ    LØ23Ø                   NO —> CONTINUE
         MVC   TAPD,=C'    IN EFFECT'  YES -> INDICATE
LØ23Ø    DS    ØH
         TM    RCVTSTA1,RCVTTAPE       TAPE VOLUME PROTECTION?
         BZ    LØ24Ø                   NO —> CONTINUE
         MVC   TAPV,=C'    IN EFFECT'  YES -> INDICATE
LØ24Ø    DS    ØH
         TM    RCVTEROP,RCVTTERP       TERMINAL AUTHORIZATION CHECK?
         BZ    LØ25Ø                   NO —> CONTINUE
         MVC   TERM,=C'    IN EFFECT'  YES -> INDICATE
LØ25Ø    DS    ØH
         TM    RCVTEROP,RCVTTUAC       TERMINAL UACC NONE?
         BZ    LØ26Ø                   NO —> CONTINUE
         MVC   TERM,=C'         NONE'  YES -> INDICATE
         EJECT
LØ26Ø    DS    ØH
         CLC   RCVTPTGN,=F'Ø'          SECURED SIGNON AVAILABLE?
         BE    LØ27Ø
         MVC   SSIGNON,=C'    IN EFFECT'  YES -> INDICATE
*
* PRINT INFORMATION TO THE PASSED DCB
*
LØ27Ø    DS    ØH
         R#PUT LINEØØ
```

```
         R#PUT LINEØ1
         R#PUT BLANKS
         R#PUT LINEØ2
         R#PUT LINEØ3
         R#PUT LINEØ4
         R#PUT LINEØ5
         R#PUT LINEØ6
         R#PUT LINEØ7
         R#PUT LINEØ8
         R#PUT LINEØ9
         R#PUT LINE1Ø
         R#PUT LINE11
         R#PUT LINE12
         R#PUT LINE13
         R#PUT LINE14
         R#PUT LINE15
         R#PUT LINE16
         R#PUT LINE17
         R#PUT LINE18
         R#PUT LINE19
         R#PUT LINE2Ø
         R#PUT LINE21
         R#PUT LINE22
         R#PUT LINE23
         R#PUT LINE24
         R#PUT LINE25
         R#PUT LINE26
         R#PUT LINE27
         R#PUT LINE28
         R#PUT LINE29
         R#PUT LINE3Ø
         R#PUT BLANKS
         R#PUT LINE31
         EJECT
*
* END OF PROCESSING
*
THE_END  DS    ØH                    MY ONLY FRIEND, THE END
         LA    RD,SAVEAREA           RD -> SAVEAREA
         L     RD,4(RD)              RD -> SAVEAREA CALLER
         LM    RE,RC,12(RD)          RESTORE REGISTERS
         XR    RF,RF                 RETURN CODE Ø
         BSM   RØ,RE                 RETURN TO CALLER
         EJECT
*
* VARIABLES AND CONSTANTS
*
SAVEAREA DS    18F
*
* RACF CALL: STATIC INFORMATION
```

```
*
RACF_S1  RACROUTE REQUEST=AUTH,        REQUEST INFORMATION             X
               CLASS='FACILITY',       -> RACF CLASS DESCRIPTION       X
               ENTITYX=(RENTDE,NONE),  -> GENERAL RESOURCE             X
               RELEASE=1.9.2,          RACF RELEASE 1.9.2              X
               LOG=ASIS,               LOG ACCORDING TO PROFILE        X
               MF=L                    LIST MACRO FORMAT
RACROUL1 EQU   *-RACF_S1               L(STATIC RACF LIST MACRO)
*
RCLASSDE DS    ØF                      KEEP ON FULLWORD BOUNDARY
         DC    AL1(L'RCLASS)           L(CLASS FIELD)
RCLASS   DC    C'FACILITY'             GENERAL RESOURCE LASS NAME
RENTDE   DS    ØF                      KEEP ON FULLWORD BOUNDARY
         DC    H'14'                   L(BUFFER FOR RESOURCE NAME)
         DC    H'14'                   L(RESOURCE NAME)
         DC    CL14'JEDSP.SETROPTS'    RESOURCE NAME
*
* RACF CALL: DYNAMIC STORAGE RESERVATION
*
RACF_D1  RACROUTE REQUEST=AUTH,        REQUEST INFORMATION             X
               CLASS='FACILITY',       -> RACF CLASS DESCRIPTION       X
               RELEASE=1.9.2,          RACF RELEASE 1.9.2              X
               MF=L
WORKA    DS    ØF                      KEEP ON FULLWORD BOUNDARY
         DS    XL1Ø24                  SAF ROUTER WORK AREA
         EJECT
*
* PARAMETER LISTS
*
P_TODECX DS    ØF
A_HEX    DC    AL4(WORKWORD)
A_REC    DS    AL4
A_LEN    DC    AL4(LEN)
WORKWORD DS    F
LEN      DS    H
*
         EJECT
*
* OUTPUT LINES
*
LINEØØ   DC    C'  OPTION                                '
         DC    C' SETROPTS          '
         DC    C'VALUE  '
BLANKS   DC    131C' '
*
LINEØ1   DC    C'  ———————————————————'
         DC    C'———————————'
         DC    C'———————————————'
         DC    131C' '
         ORG   LINEØ1+131
```

```
*
LINEØ2   DC    C'  AUTOMATIC DATASET PROTECTION          '
         DC    C' ADSP                '
ADSP     DS    CL13
         DC    131C' '
         ORG   LINEØ2+131
*
LINEØ3   DC    C'  CATALOGUED DATASETS ONLY               '
         DC    C' CATDSNS             '
CATDSNS  DS    CL13
         DC    C'   '
LCATDSNS DS    CL8
         DC    131C' '
         ORG   LINEØ3+131
*
LINEØ4   DC    C'  COMPATIBILITY MODE                    '
         DC    C' COMPATMODE          '
COMPAT   DS    CL13
         DC    131C' '
         ORG   LINEØ4+131
*
LINEØ5   DC    C'  DASD VOLUME PROTECTION                '
         DC    C' N/A                 '
DASD     DS    CL13
         DC    131C' '
         ORG   LINEØ5+131
*
LINEØ6   DC    C'  ENHANCED GENERIC NAMING               '
         DC    C' EGN                 '
EGN      DS    CL13
         DC    131C' '
         ORG   LINEØ6+131
*
LINEØ7   DC    C'  ERASE-ON-SCRATCH                      '
         DC    C' ERASE               '
EOS      DS    CL13
         DC    C'   '
LEOS     DS    CL8
         DC    131C' '
         ORG   LINEØ7+131
         EJECT
LINEØ8   DC    C'  GENERIC OWNER ONLY                    '
         DC    C' GENERICOWNER        '
GO       DS    CL13
         DC    131C' '
         ORG   LINEØ8+131
*
LINEØ9   DC    C'  INPUT DSN FOR LOGGING AND MESSAGES    '
         DC    C' REALDSN             '
IDSN     DS    CL13
```

```
         DC    131C' '
         ORG   LINEØ9+131
*
LINE1Ø   DC    C'  LIST OF GROUPS ACCESS CHECKING       '
         DC    C' GRPLIST            '
LGC      DS    CL13
         DC    131C' '
         ORG   LINE1Ø+131
*
LINE11   DC    C'  LOG RECORD FOR COMMAND VIOLATIONS    '
         DC    C' CMDVIOL            '
CMDV     DS    CL13
         DC    131C' '
         ORG   LINE11+131
*
LINE12   DC    C'  MULTI-LEVEL ACTIVE                   '
         DC    C' MLACTIVE           '
MLA      DS    CL13
         DC    C'  '
LMLA     DS    CL1Ø
         DC    131C' '
         ORG   LINE12+131
*
LINE13   DC    C'  MULTI-LEVEL QUIET                    '
         DC    C' MLQUIET            '
MLQ      DS    CL13
         DC    131C' '
         ORG   LINE13+131
*
LINE14   DC    C'  MULTI-LEVEL SECURE                   '
         DC    C' MLS                '
MLS      DS    CL13
         DC    C'  '
LMLS     DS    CL1Ø
         DC    131C' '
         ORG   LINE14+131
*
LINE15   DC    C'  MULTI-LEVEL STABLE                   '
         DC    C' MLSTABLE           '
MLT      DS    CL13
         DC    131C' '
         ORG   LINE15+131
         EJECT
LINE16   DC    C'  MODEL-GDG                            '
         DC    C' MODEL(GDG)         '
MGDG     DS    CL13
         DC    131C' '
         ORG   LINE16+131
*
LINE17   DC    C'  MODEL-GROUP                          '
```

```
         DC    C' MODEL(GROUP)       '
MGRP     DS    CL13
         DC    131C' '
         ORG   LINE17+131
*
LINE18   DC    C'  MODEL-USER                          '
         DC    C' MODEL(USER)        '
MUSR     DS    CL13
         DC    131C' '
         ORG   LINE18+131
*
LINE19   DC    C'  NO DUPLICATE DATASET NAMES          '
         DC    C' N/A                '
DUPD     DS    CL13
         DC    131C' '
         ORG   LINE19+131
*
LINE2Ø   DC    C'  PROGRAM ACCESS CONTROL              '
         DC    C' WHEN(PROGRAM)      '
PACC     DS    CL13
         DC    131C' '
         ORG   LINE2Ø+131
*
LINE21   DC    C'  PROTECT-ALL                         '
         DC    C' PROTECTALL         '
PALL     DS    CL13
         DC    C'   '
LPALL    DS    CL8
         DC    131C' '
         ORG   LINE21+131
*
LINE22   DC    C'  SECLABEL USE ON RACF COMMANDS LIMITED '
         DC    C' SECLABELCONTROL    '
SELC     DS    CL13
         DC    131C' '
         ORG   LINE22+131
*
LINE23   DC    C'  SECURITY LEVEL AUDITING             '
         DC    C' SECLEVELAUDIT      '
         DC    1ØC' '
SELA     DS    3C
         DC    131C' '
         ORG   LINE23+131
*
LINE24   DC    C'  SESSION KEY INTERVALS MAXIMUM       '
         DC    C' SESSIONINTERVAL    '
         DC    8C' '
SESI     DS    5C
         DC    131C' '
         ORG   LINE24+131
```

```
         EJECT
*
LINE25   DC    C'  SINGLE LEVEL DSN PREFIX AND LENGTH    '
         DC    C' PREFIX              '
         DC    5C' '
SDSN     DS    CL8
         DC    C'         '
LSDSN    DS    CL2
         DC    131C' '
         ORG   LINE25+131
*
LINE26   DC    C'  SYSTEM SECURITY RETENTION PERIOD      '
         DC    C' RETPD               '
         DC    8C' '
SREP     DS    CL5
         DC    131C' '
         ORG   LINE26+131
*
LINE27   DC    C'  TAPE DSN PROTECTION                   '
         DC    C' TAPEDSN             '
TAPD     DS    CL13
         DC    131C' '
         ORG   LINE27+131
*
LINE28   DC    C'  TAPE VOL PROTECTION                   '
         DC    C' N/A                 '
TAPV     DS    CL13
         DC    131C' '
         ORG   LINE28+131
*
LINE29   DC    C'  TERMINAL AUTHORIZATION CHECKING       '
         DC    C' TERMINAL            '
TERM     DS    CL13
         DC    131C' '
         ORG   LINE29+131
*
LINE3Ø   DC    C'  TERMINAL UACC IF NOT DEFINED          '
         DC    C' TERMINAL(UACC)      '
TUAC     DS    CL13
         DC    131C' '
         ORG   LINE3Ø+131
*
LINE31   DC    C'  SECURED SIGNON SPE (PASSTICKET)        '
         DC    C' N/A                 '
SSIGNON  DS    CL13
         DC    131C' '
         ORG   LINE31+131
         EJECT
*
* LITERAL POOL
```

```
*
         LTORG
         EJECT
*
* DSECT'S
*
         PRINT NOGEN
*
         IHAPSA DSECT=YES
         CVT    DSECT=YES,PREFIX=YES
         ICHPRCVT
*
         END
```

CONCLUSION

In my experience, a number of misunderstandings exist between the two distinct groups of people that, in general, maintain the security of a enterprise wide system (the network group is yet another story with the growing importance of cheap protocols and connections like the Internet and intranets). Some of these grievances can be solved by accomplishing extensions to RACF that are more realistic in their approach than the original IBM design. In the program above, 'READ' access to the SETROPTS command can be given to certain users without compromising the overall RACF authority. It must be emphasized however that this consists of a gentlemen's agreement, a real systems programmer with bad intentions cannot be stopped.

In the past I have implemented similar solutions to list the RACF active exits, the password rules, the started procedures table, the general resources, the RACF/JES2 options, the statistics and audit settings, the authorized caller table, the global access table (GAC) and the databases in use at a particular point in time.

Editor's note: If there is sufficient interest, a number of follow-up articles are possible, based on the subjects mentioned above. Please address any comments or feedback to the author at jan@jedsp.com.

jan de decker
JED:SP (Belgium)

Using TMON to find a RACF user-id for a Netname

RACF detects and logs an access violation when it denies a user access to a resource – which occurs because that user is not authorized to access the resource. Therefore, an access violation is a sign that someone either does not understand his or her role as a RACF user, or is trying to bypass RACF protection. You can use a report of access violations to identify these users as well as to audit your security.

When CICS is secured with RACF each user must sign on with a user-id and a password. To retrieve the association between a user-id and a netname, I have written a job TERMSERD with The Monitor for CICS Version 1.3 and CICS Version 3. The output is directed to a file (see usernetna). TERMSERD calls a procedure TMON1RPT and my CICS name is CIC5. Figure 1 shows an example of the report obtained.

TERMSERD

```
//TERMSERD JOB SYS,SYSTEME,CLASS=Y,MSGCLASS=9,NOTIFY=&SYSUID
//*****************************************************************
//*   CREATe a file for netname and user-id association  *
//*   *
//*****************************************************************
//*
//STATERM  EXEC TMON1RPT,TIME=20
//*
//*
//REPORTS  DD SYSOUT=*
//*
//SYSIN    DD  *
00330000
UFLD
  NAME TRANRATE
  HEAD1 = 'TERMINAL'
  HEAD2 = 'USERID'
  TYPE COUNT

REPORT COVER NO
  LINE TATERID TAUSRID TARSPTM TACPURTM TAIOCT
  PROC
    WHEN TASYSID NE 'CIC5'
         THEN REJECT
```

```
1DATE: Ø1/2Ø/98      T H E   M O N I T O R   F O R   C I C S / E S A   PAGE: 2
 TIME: Ø6:17:28      A C T I V I T E   D E S   T E R M I N A U X
                              DATA IS FROM Ø1/Ø5/98
Ø     START        TERM         USER           TRANS         TRANS
      TIME         ID           ID              ID           COUNT
                                                             TOT
Ø  7:19:33.4ØØØ                 CIC5            CRSQ          1
   7:19:33.4Ø1                  CIC5            CSFU          1
   7:19:33.91Ø1                 CIC5            CXRE          1
   7:19:34.2596    WSVB         CICSUSER        SRCØ          1
   7:19:34.2631    WRVG         CICSUSER        SRCØ          1
   7:19:34.2696                 CIC5            CATR          1
   7:19:34.3123                 CIC5            CLS1          1
   7:19:34.3812    WDVR         CICSUSER        SRCØ          1
   7:19:36.Ø716                 CIC5            ISER          1
   7:19:36.Ø733                 CIC5            ISER          2
   7:19:4Ø.5347                 CIC5            SRXE          1
   7:19:41.3837                 CIC5            SRXØ          1
   7:19:46.Ø819                 CIC5            SRXE          1
   7:19:46.2618                 CIC5            CNTL          1
   7:19:46.2619                 CIC5            AAON          1
   7:19:46.7756                 CIC5            CRSQ          1
   7:19:46.7757                 CIC5            CSFU          1
   7:19:46.868Ø                 CIC5            CNTL          1
   7:19:46.8682                 CIC5            AAON          1
   7:19:46.9119                 CIC5            SRXØ          1
   7:19:47.2816                 CIC5            CXRE          1
   7:19:47.5326                 CIC5            CRSQ          1
   7:19:47.5327                 CIC5            CSFU          1
   7:19:47.7958     WDVA        CICSUSER        SRCØ          2
   7:19:47.8175                 CIC5            CATR          1
   7:19:47.8495                 CIC5            CLS1          1
   7:19:48.Ø455                 CIC5            CXRE          1
   7:19:48.3357     WKVC        CICSUSER        SRCØ          1
   7:19:48.4925                 CIC5            CATR          1
   7:19:48.5251                 CIC5            CLS1          1
   7:19:48.618Ø                 YSVØØDØØ        CSSY          17
   7:19:49.433Ø                 CIC5            ISER          1
   7:19:49.4348                 CIC5            ISER          2
```

Figure 1: Example of report

```
    ENDWHEN
  SUMF TAUSRID
  TITLE 'TERMINAL USERID       '

/*
//*****************************************************************
//*   CREATE A STATISTICAL FILE FOR TERMINAL ACTIVITY FOR A      *
//*   TERMINAL AND A JOURNAL                                     *
//*****************************************************************
//DELSTAT  EXEC PGM=IDCAMS
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
 DEL  EXPL69.CIC5.USERID.TERMID NONVSAM PURGE
/*
//STATERM  EXEC TMON1RPT,TIME=20
//*
//*
//REPORTS  DD DSN=EXPL69.CIC5.USERID.TERMID,UNIT=SYSDA,DISP=(,CATLG),
//         SPACE=(TRK,(150,75),RLSE),LABEL=RETPD=30,
//         DCB=(RECFM=FBA,LRECL=133)
//*
//SYSIN    DD  *
00330000
UFLD
  NAME TRANRATE
  HEAD1 = 'TERMINAL'
  HEAD2 = 'USERID'
  TYPE COUNT

REPORT COVER NO
  LINE TATSKSTS TATERID TAUSRID TATRNID TATRANCT
  PROC
    WHEN TASYSID NE 'CIC5'
         THEN REJECT
    ENDWHEN
  SORTF TATSKSTS
  TITLE 'ACTIVITE DES TERMINAUX'

/*
/*
//*****************************************************************
//*   CREATE A STATISTICAL FILE FOR TERMINAL ACTIVITY FOR A      *
//*   TERMINAL AND FOR A JOURNAL                                 *
//*****************************************************************
//DELSTAT  EXEC PGM=IDCAMS
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
 DEL  EXPL69.CIC5.USERID.TERMID.B NONVSAM PURGE
/*
//STATERM  EXEC TMON1RPT,TIME=20
```

```
//*
//*
//REPORTS  DD DSN=EXPL69.CIC5.USERID.TERMID.B,UNIT=SYSDA,DISP=(,CATLG),
//         SPACE=(TRK,(150,75),RLSE),LABEL=RETPD=30,
//         DCB=(RECFM=FBA,LRECL=133)
//*
//SYSIN    DD  *

REPORT COVER NO
  LINE TATSKSTS TATERID TAUSRID TATRNID TATRANCT
  PROC
    WHEN TASYSID NE 'CIC5'
         THEN REJECT
    ENDWHEN
  SORTF TATSKSTS
  TITLE 'TERMINAL ACTIVITY'

  SORTF TATERID TATSKSTS
  LINE TATSKSTS TATRNID TAUSRID TATERID TATRANCT TARSPTM TACPURTM TAIOCT
  SUMF TATRNID TAUSRID TATERID
  SFORM TATRANCT TARSPTM TACPURTM TAIOCT
/*
//*****************************************************************
//*   CREATE A STATISTICAL FILE FOR TERMINAL ACTIVITY FOR A       *
//*   TERMINAL AND A JOURNAL                                      *
//*****************************************************************
//DELSTAT  EXEC PGM=IDCAMS
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD *
 DEL  EXPL69.CIC5.TIMEOUT NONVSAM PURGE
/*
/*
//STATERM  EXEC TMON1RPT,TIME=20
//*
//*
//REPORTS  DD DSN=EXPL69.CIC5.TIMEOUT,UNIT=SYSDA,DISP=(,CATLG),
//         SPACE=(TRK,(150,75),RLSE),MGMTCLAS=INTERIM,
//         DCB=(RECFM=FBA,LRECL=133)
//*
//SYSIN    DD  *
00330000
UFLD
  NAME TRANRATE
  HEAD1 = 'TERMINAL'
  HEAD2 = 'USERID'
  TYPE COUNT

REPORT COVER NO
  LINE TATSKSTS TATERID TAUSRID TATRNID TATRANCT
  PROC
```

```
    WHEN TASYSID NE 'CIC5'
         THEN REJECT
    ENDWHEN
  SORTF TATSKSTS
  TITLE 'TERMINAL ACTIVITY'

/*
//
```

TMON1RPT

```
//TMON1RPT  PROC REPORT=
//REPORT   EXEC PGM=LMRK7ØØ,REGION=4ØØØK,PARM='TMONØ8Ø'
//*
//STEPLIB  DD DSN=LMRK.PSV2R4.TCELOAD,DISP=SHR
//         DD DSN=LMRK.PSV2R4.LMKLOAD,DISP=SHR
//CTLTMON  DD DSN=TMON.V1R3.CONTROL,DISP=SHR
//*
//* D U M P I N  -  I N P U T   F I L E S
//*
//HISTIN DD DSN=TMON.TMONV8.HIST(Ø),DISP=SHR  PHILG
//* DUMPIN    DD DSN=TMON.V1R3.TMONØ1.ARCHIVE9(Ø),DISP=SHR
//*
//TEMPFIL  DD DSN=&&TEMP,DISP=(,DELETE,DELETE),
//      UNIT=SYSDA,SPACE=(CYL,(1ØØ,5),RLSE),
//      DCB=(RECFM=VB,LRECL=32596,BLKSIZE=32600)
//*
//REPORTS  DD SYSOUT=*,
//         DCB=(BLKSIZE=133Ø,RECFM=FBA,LRECL=133)
//*
//SYSOUT   DD SYSOUT=*,DCB=BUFNO=1
//SORTWKØ1 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//SORTWKØ2 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//SORTWKØ3 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//SORTWKØ4 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//SORTWKØ5 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//SORTWKØ6 DD SPACE=(CYL,(5,1Ø)),UNIT=SYSDA
//*
//PRTCTL   DD SYSOUT=*,DCB=BLKSIZE=133
//LMRKSNAP DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*,DCB=BLKSIZE=133
//SYSIN    DD DSN=LMRK.PSV2R4.TCESAMP(&REPORT),DISP=SHR
//*        PEND  REMOVE ASTERISK IF USING IN LINE PROCS
```

Claude Dunand (France)

Replacement for the RACF Report Writer – part 2

This month we continue the code for the reporting tool in SAS, which can act as a replacement for the RACF Report Writer. This enables you to have SQL-like capabilities without having to keep your SMF data in databases.

The article is an extensive piece of work, which will be published over several issues of RACF Update. *To enable the article to be published in a manageable form, considerable editing of the original code has been necessary. The original, unedited code is available on our Web site (www.xephon.com) and can be downloaded in the usual way. This service is free to subscribers.*

```
./        ADD   LIST=ALL,NAME=RALTER
%MACRO RALTER(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from RALTER extension;
      RACF.RALTER (KEEP=%SMFHDR
                        %SMF8ØHDR(REQ=DEFINE)
                        RA_CLASS
                        RA_OWNID
                        RA_USERN
                        RA_OLSCL
                        RA_UTKNE
                        RA_UPRE
                        RA_UVFYX
                        RA_UNJEU
                        RA_UUAUD
                        RA_USPEC
                        RA_UDFLT
                        RA_UUNDF
                        RA_UERR
                        RA_UTRST
                        RA_USEST
                        RA_USURO
                        RA_URMT
                        RA_UPRVL
                        RA_USECL
                        RA_UEXND
                        RA_USUSR
                        RA_USNOD
                        RA_USGRP
```

```
                        RA_USPOE
                        RA_USPCL
                        RA_UTUSR
                        RA_UTGRP
                        RA_UTDFT
                        RA_UTSEC
                        RA_APPC
                        RA_RESNM
                        RA_SPEFD
                        RA_FAILD
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for RALTER extension;
    WHEN('RALTER') DO;
      INPUT %SMF80HDR(REQ=EXTRACT)
            RA_CLASS $       282-289
            RA_OWNID $       291-298
            RA_USERN $       300-319
            RA_OLSCL $       321-328
            RA_UTKNE $       330-333
            RA_UPRE  $       335-338
            RA_UVFYX $       340-343
            RA_UNJEU $       345-348
            RA_UUAUD $       350-353
            RA_USPEC $       355-358
            RA_UDFLT $       360-363
            RA_UUNDF $       365-368
            RA_UERR  $       370-373
            RA_UTRST $       375-378
            RA_USEST $       380-387
            RA_USURO $       389-392
            RA_URMT  $       394-397
            RA_UPRVL $       399-402
            RA_USECL $       404-411
            RA_UEXND $       413-420
            RA_USUSR $       422-429
            RA_USNOD $       431-438
            RA_USGRP $       440-447
            RA_USPOE $       449-456
            RA_USPCL $       458-465
            RA_UTUSR $       467-474
            RA_UTGRP $       476-483
            RA_UTDFT $       485-488
            RA_UTSEC $       490-493
            RA_APPC  $       495-510
            RA_RESNM $       512-711
            RA_SPEFD $       768-967
            RA_FAILD $      1793-1992
```

```
                ;
LABEL RA_CLASS = 'Class name'
      RA_OWNID = 'Profile owner'
      RA_USERN = 'User name'
      RA_OLSCL = 'SECLABEL deleted'
      RA_UTKNE = 'Utoken encr.?'
      RA_UPRE  = 'Pre-1.9?'
      RA_UVFYX = 'VERIFYX propagation?'
      RA_UNJEU = 'Undefined NJE user?'
      RA_UUAUD = 'UAUDIT?'
      RA_USPEC = 'RACF special?'
      RA_UDFLT = 'Default token?'
      RA_UUNDF = 'Undefined user?'
      RA_UERR  = 'Token in error?'
      RA_UTRST = 'User trusted?'
      RA_USEST = 'Session type'
      RA_USURO = 'Surrogate user?'
      RA_URMT  = 'Remote job?'
      RA_UPRVL = 'Privileged user?'
      RA_USECL = 'User SECLABEL'
      RA_UEXND = 'Execution node'
      RA_USUSR = 'Submitting user'
      RA_USNOD = 'Submitting node'
      RA_USGRP = 'Submitting group'
      RA_USPOE = 'Port of entry'
      RA_USPCL = 'Class of POE'
      RA_UTUSR = 'Userid'
      RA_UTGRP = 'Groupid'
      RA_UTDFT = 'Default group?'
      RA_UTSEC = 'Default SECLABEL?'
      RA_APPC  = 'APPC key link'
      RA_RESNM = 'Resource name'
      RA_SPEFD = 'Keywords specified'
      RA_FAILD = 'Keywords failed'
      ;
        OUTPUT RACF.RALTER;
      END;
    %END;
%MEND RALTER;
./        ADD   LIST=ALL,NAME=RDEFINE
%MACRO RDEFINE(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from RDEFINE extension;
      RACF.RDEFINE (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         RD_CLASS
                         RD_OWNID
                         RD_USERN
```

```
                        RD_SECL
                        RD_UTKNE
                        RD_UPRE
                        RD_UVFYX
                        RD_UNJEU
                        RD_UUAUD
                        RD_USPEC
                        RD_UDFLT
                        RD_UUNDF
                        RD_UERR
                        RD_UTRST
                        RD_USEST
                        RD_USURO
                        RD_URMT
                        RD_UPRVL
                        RD_USECL
                        RD_UEXND
                        RD_USUSR
                        RD_USNOD
                        RD_USGRP
                        RD_USPOE
                        RD_USPCL
                        RD_UTUSR
                        RD_UTGRP
                        RD_UTDFT
                        RD_UTSEC
                        RD_APPC
                        RD_RESNM
                        RD_SPEFD
                        RD_FAILD
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for RDEFINE extension;
    WHEN('RDEFINE') DO;
      INPUT %SMF80HDR(REQ=EXTRACT)
            RD_CLASS $       282-289
            RD_OWNID $       291-298
            RD_USERN $       300-319
            RD_SECL  $       321-328
            RD_UTKNE $       330-333
            RD_UPRE  $       335-338
            RD_UVFYX $       340-343
            RD_UNJEU $       345-348
            RD_UUAUD $       350-353
            RD_USPEC $       355-358
            RD_UDFLT $       360-363
            RD_UUNDF $       365-368
            RD_UERR  $       370-373
```

```
                RD_UTRST $       375-378
                RD_USEST $       380-387
                RD_USURO $       389-392
                RD_URMT  $       394-397
                RD_UPRVL $       399-402
                RD_USECL $       404-411
                RD_UEXND $       413-420
                RD_USUSR $       422-429
                RD_USNOD $       431-438
                RD_USGRP $       440-447
                RD_USPOE $       449-456
                RD_USPCL $       458-465
                RD_UTUSR $       467-474
                RD_UTGRP $       476-483
                RD_UTDFT $       485-488
                RD_UTSEC $       490-493
                RD_APPC  $       495-510
                RD_RESNM $       512-711
                RD_SPEFD $       768-967
                RD_FAILD $      1793-1992
                 ;
LABEL RD_CLASS = 'Class name'
      RD_OWNID = 'Profile owner'
      RD_USERN = 'User name'
      RD_SECL  = 'SECLABEL'
      RD_UTKNE = 'Utoken encr.?'
      RD_UPRE  = 'Pre-1.9?'
      RD_UVFYX = 'VERIFYX propagation?'
      RD_UNJEU = 'Undefined NJE user?'
      RD_UUAUD = 'UAUDIT?'
      RD_USPEC = 'RACF special?'
      RD_UDFLT = 'Default token?'
      RD_UUNDF = 'Undefined user?'
      RD_UERR  = 'Token in error?'
      RD_UTRST = 'User trusted?'
      RD_USEST = 'Session type'
      RD_USURO = 'Surrogate user?'
      RD_URMT  = 'Remote job?'
      RD_UPRVL = 'Privileged user?'
      RD_USECL = 'User SECLABEL'
      RD_UEXND = 'Execution node'
      RD_USUSR = 'Submitting user'
      RD_USNOD = 'Submitting node'
      RD_USGRP = 'Submitting group'
      RD_USPOE = 'Port of entry'
      RD_USPCL = 'Class of POE'
      RD_UTUSR = 'Userid'
      RD_UTGRP = 'Groupid'
      RD_UTDFT = 'Default group?'
      RD_UTSEC = 'Default SECLABEL?'
```

```
      RD_APPC  = 'APPC key link'
      RD_RESNM = 'Resource name'
      RD_SPEFD = 'Keywords specified'
      RD_FAILD = 'Keywords failed'
      ;
        OUTPUT RACF.RDEFINE;
      END;
    %END;
%MEND RDEFINE;
./        ADD   LIST=ALL,NAME=RDELETE
%MACRO RDELETE(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from RDELETE extension;
      RACF.RDELETE (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         RL_CLASS
                         RL_OWNID
                         RL_USERN
                         RL_SECL
                         RL_UTKNE
                         RL_UPRE
                         RL_UVFYX
                         RL_UNJEU
                         RL_UUAUD
                         RL_USPEC
                         RL_UDFLT
                         RL_UUNDF
                         RL_UERR
                         RL_UTRST
                         RL_USEST
                         RL_USURO
                         RL_URMT
                         RL_UPRVL
                         RL_USECL
                         RL_UEXND
                         RL_USUSR
                         RL_USNOD
                         RL_USGRP
                         RL_USPOE
                         RL_USPCL
                         RL_UTUSR
                         RL_UTGRP
                         RL_UTDFT
                         RL_UTSEC
                         RL_APPC
                         RL_RESNM
                         RL_SPEFD
                    )
```

```
    %END;
  %IF &REQ = EXTRACT %THEN
    %DO;
      %PUT Including datadefinition for RDELETE extension;
      WHEN('RDELETE') DO;
        INPUT %SMF80HDR(REQ=EXTRACT)
              RL_CLASS $       282-289
              RL_OWNID $       291-298
              RL_USERN $       300-319
              RL_SECL  $       321-328
              RL_UTKNE $       330-333
              RL_UPRE  $       335-338
              RL_UVFYX $       340-343
              RL_UNJEU $       345-348
              RL_UUAUD $       350-353
              RL_USPEC $       355-358
              RL_UDFLT $       360-363
              RL_UUNDF $       365-368
              RL_UERR  $       370-373
              RL_UTRST $       375-378
              RL_USEST $       380-387
              RL_USURO $       389-392
              RL_URMT  $       394-397
              RL_UPRVL $       399-402
              RL_USECL $       404-411
              RL_UEXND $       413-420
              RL_USUSR $       422-429
              RL_USNOD $       431-438
              RL_USGRP $       440-447
              RL_USPOE $       449-456
              RL_USPCL $       458-465
              RL_UTUSR $       467-474
              RL_UTGRP $       476-483
              RL_UTDFT $       485-488
              RL_UTSEC $       490-493
              RL_APPC  $       495-510
              RL_RESNM $       512-711
              RL_SPEFD $       768-967
               ;
LABEL RL_CLASS = 'Class name'
      RL_OWNID = 'Profile owner'
      RL_USERN = 'User name'
      RL_SECL  = 'SECLABEL'
      RL_UTKNE = 'Utoken encr.?'
      RL_UPRE  = 'Pre-1.9?'
      RL_UVFYX = 'VERIFYX propagation?'
      RL_UNJEU = 'Undefined NJE user?'
      RL_UUAUD = 'UAUDIT?'
      RL_USPEC = 'RACF special?'
      RL_UDFLT = 'Default token?'
```

```
      RL_UUNDF = 'Undefined user?'
      RL_UERR  = 'Token in error?'
      RL_UTRST = 'User trusted?'
      RL_USEST = 'Session type'
      RL_USURO = 'Surrogate user?'
      RL_URMT  = 'Remote job?'
      RL_UPRVL = 'Privileged user?'
      RL_USECL = 'User SECLABEL'
      RL_UEXND = 'Execution node'
      RL_USUSR = 'Submitting user'
      RL_USNOD = 'Submitting node'
      RL_USGRP = 'Submitting group'
      RL_USPOE = 'Port of entry'
      RL_USPCL = 'Class of POE'
      RL_UTUSR = 'Userid'
      RL_UTGRP = 'Groupid'
      RL_UTDFT = 'Default group?'
      RL_UTSEC = 'Default SECLABEL?'
      RL_APPC  = 'APPC key link'
      RL_RESNM = 'Resource name'
      RL_SPEFD = 'Keywords specified'
      ;
        OUTPUT RACF.RDELETE;
      END;
    %END;
%MEND RDELETE;
./        ADD   LIST=ALL,NAME=REMOVE
%MACRO REMOVE(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from REMOVE extension;
      RACF.REMOVE (KEEP=%SMFHDR
                        %SMF8ØHDR(REQ=DEFINE)
                        REMOWNID
                        REMUSERN
                        REMUTKNE
                        REMUPRE
                        REMUVFYX
                        REMUNJEU
                        REMUUAUD
                        REMUSPEC
                        REMUDFLT
                        REMUUNDF
                        REMUERR
                        REMUTRST
                        REMUSEST
                        REMUSURO
                        REMURMT
                        REMUPRVL
```

```
                        REMUSECL
                        REMUEXND
                        REMUSUSR
                        REMUSNOD
                        REMUSGRP
                        REMUSPOE
                        REMUSPCL
                        REMUTUSR
                        REMUTGRP
                        REMUTDFT
                        REMUTSEC
                        REMAPPC
                        REMUSRID
                        REMSPEFD
                        REMFAILD
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for REMOVE extension;
    WHEN('REMOVE') DO;
      INPUT %SMF80HDR(REQ=EXTRACT)
            REMOWNID $       282-289
            REMUSERN $       291-310
            REMUTKNE $       312-315
            REMUPRE  $       317-320
            REMUVFYX $       322-325
            REMUNJEU $       327-330
            REMUUAUD $       332-335
            REMUSPEC $       337-340
            REMUDFLT $       342-345
            REMUUNDF $       347-350
            REMUERR  $       352-355
            REMUTRST $       357-360
            REMUSEST $       362-369
            REMUSURO $       371-374
            REMURMT  $       376-379
            REMUPRVL $       381-384
            REMUSECL $       386-393
            REMUEXND $       395-402
            REMUSUSR $       404-411
            REMUSNOD $       413-420
            REMUSGRP $       422-429
            REMUSPOE $       431-438
            REMUSPCL $       440-447
            REMUTUSR $       449-456
            REMUTGRP $       458-465
            REMUTDFT $       467-470
            REMUTSEC $       472-475
            REMAPPC  $       477-492
```

```
              REMUSRID $       494-501
              REMSPEFD $       503-702
              REMFAILD $      1528-1727
              ;
LABEL REMOWNID = 'Profile owner'
      REMUSERN = 'User name'
      REMUTKNE = 'Utoken encr.?'
      REMUPRE  = 'Pre-1.9?'
      REMUVFYX = 'VERIFYX propagation?'
      REMUNJEU = 'Undefined NJE user?'
      REMUUAUD = 'UAUDIT?'
      REMUSPEC = 'RACF special?'
      REMUDFLT = 'Default token?'
      REMUUNDF = 'Undefined user?'
      REMUERR  = 'Token in error?'
      REMUTRST = 'User trusted?'
      REMUSEST = 'Session type'
      REMUSURO = 'Surrogate user?'
      REMURMT  = 'Remote job?'
      REMUPRVL = 'Privileged user?'
      REMUSECL = 'User SECLABEL'
      REMUEXND = 'Execution node'
      REMUSUSR = 'Submitting user'
      REMUSNOD = 'Submitting node'
      REMUSGRP = 'Submitting group'
      REMUSPOE = 'Port of entry'
      REMUSPCL = 'Class of POE'
      REMUTUSR = 'Userid'
      REMUTGRP = 'Groupid'
      REMUTDFT = 'Default group?'
      REMUTSEC = 'Default SECLABEL?'
      REMAPPC  = 'APPC key link'
      REMUSRID = 'User id'
      REMSPEFD = 'Keywords specified'
      REMFAILD = 'Keywords failed'
      ;
        OUTPUT RACF.REMOVE;
      END;
    %END;
%MEND REMOVE;
./        ADD   LIST=ALL,NAME=SETROPTS
%MACRO SETROPTS(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from SETROPTS extension;
      RACF.SETROPTS (KEEP=%SMFHDR
                          %SMF80HDR(REQ=DEFINE)
                          SETOWNID
                          SETUSERN
```

```
                        SETUTKNE
                        SETUPRE
                        SETUVFYX
                        SETUNJEU
                        SETUUAUD
                        SETUSPEC
                        SETUDFLT
                        SETUUNDF
                        SETUERR
                        SETUTRST
                        SETUSEST
                        SETUSURO
                        SETURMT
                        SETUPRVL
                        SETUSECL
                        SETUEXND
                        SETUSUSR
                        SETUSNOD
                        SETUSGRP
                        SETUSPOE
                        SETUSPCL
                        SETUTUSR
                        SETUTGRP
                        SETUTDFT
                        SETUTSEC
                        SETAPPC
                        SETSPEFD
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for SETROPTS extension;
    WHEN('SETROPTS') DO;
      INPUT %SMF80HDR(REQ=EXTRACT)
            SETOWNID $       282-289
            SETUSERN $       291-310
            SETUTKNE $       312-315
            SETUPRE  $       317-320
            SETUVFYX $       322-325
            SETUNJEU $       327-330
            SETUUAUD $       332-335
            SETUSPEC $       337-340
            SETUDFLT $       342-345
            SETUUNDF $       347-350
            SETUERR  $       352-355
            SETUTRST $       357-360
            SETUSEST $       362-369
            SETUSURO $       371-374
            SETURMT  $       376-379
            SETUPRVL $       381-384
```

```
                SETUSECL $       386-393
                SETUEXND $       395-402
                SETUSUSR $       404-411
                SETUSNOD $       413-420
                SETUSGRP $       422-429
                SETUSPOE $       431-438
                SETUSPCL $       440-447
                SETUTUSR $       449-456
                SETUTGRP $       458-465
                SETUTDFT $       467-470
                SETUTSEC $       472-475
                SETAPPC  $       477-492
                SETSPEFD $       494-693
                ;
LABEL SETOWNID = 'Profile owner'
      SETUSERN = 'User name'
      SETUTKNE = 'Utoken encr.?'
      SETUPRE  = 'Pre-1.9?'
      SETUVFYX = 'VERIFYX propagation?'
      SETUNJEU = 'Undefined NJE user?'
      SETUUAUD = 'UAUDIT?'
      SETUSPEC = 'RACF special?'
      SETUDFLT = 'Default token?'
      SETUUNDF = 'Undefined user?'
      SETUERR  = 'Token in error?'
      SETUTRST = 'User trusted?'
      SETUSEST = 'Session type'
      SETUSURO = 'Surrogate user?'
      SETURMT  = 'Remote job?'
      SETUPRVL = 'Privileged user?'
      SETUSECL = 'User SECLABEL'
      SETUEXND = 'Execution node'
      SETUSUSR = 'Submitting user'
      SETUSNOD = 'Submitting node'
      SETUSGRP = 'Submitting group'
      SETUSPOE = 'Port of entry'
      SETUSPCL = 'Class of POE'
      SETUTUSR = 'Userid'
      SETUTGRP = 'Groupid'
      SETUTDFT = 'Default group?'
      SETUTSEC = 'Default SECLABEL?'
      SETAPPC  = 'APPC key link'
      SETSPEFD = 'Keywords specified'
      ;
        OUTPUT RACF.SETROPTS;
      END;
    %END;
%MEND SETROPTS;
./        ADD   LIST=ALL,NAME=RVARY
%MACRO RVARY(REQ=);
```

```
%LET REQ = %UPCASE(&REQ);
%IF &REQ = DEFINE %THEN
  %DO;
    %PUT Including variables from RVARY extension;
    RACF.RVARY (KEEP=%SMFHDR
                     %SMF8ØHDR(REQ=DEFINE)
                     RV_USERN
                     RV_UTKNE
                     RV_UPRE
                     RV_UVFYX
                     RV_UNJEU
                     RV_UUAUD
                     RV_USPEC
                     RV_UDFLT
                     RV_UUNDF
                     RV_UERR
                     RV_UTRST
                     RV_USEST
                     RV_USURO
                     RV_URMT
                     RV_UPRVL
                     RV_USECL
                     RV_UEXND
                     RV_USUSR
                     RV_USNOD
                     RV_USGRP
                     RV_USPOE
                     RV_USPCL
                     RV_UTUSR
                     RV_UTGRP
                     RV_UTDFT
                     RV_UTSEC
                     RV_APPC
                )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for RVARY extension;
    WHEN('RVARY') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            RV_USERN $       282-3Ø1
            RV_UTKNE $       3Ø3-3Ø6
            RV_UPRE  $       3Ø8-311
            RV_UVFYX $       313-316
            RV_UNJEU $       318-321
            RV_UUAUD $       323-326
            RV_USPEC $       328-331
            RV_UDFLT $       333-336
            RV_UUNDF $       338-341
            RV_UERR  $       343-346
```

```
                RV_UTRST $       348-351
                RV_USEST $       353-360
                RV_USURO $       362-365
                RV_URMT  $       367-370
                RV_UPRVL $       372-375
                RV_USECL $       377-384
                RV_UEXND $       386-493
                RV_USUSR $       395-402
                RV_USNOD $       404-411
                RV_USGRP $       413-420
                RV_USPOE $       422-429
                RV_USPCL $       431-438
                RV_UTUSR $       440-447
                RV_UTGRP $       449-456
                RV_UTDFT $       458-461
                RV_UTSEC $       463-466
                RV_APPC  $       468-483
                ;
LABEL RV_USERN = 'User name'
      RV_UTKNE = 'Utoken encr.?'
      RV_UPRE  = 'Pre-1.9?'
      RV_UVFYX = 'VERIFYX propagation?'
      RV_UNJEU = 'Undefined NJE user?'
      RV_UUAUD = 'UAUDIT?'
      RV_USPEC = 'RACF special?'
      RV_UDFLT = 'Default token?'
      RV_UUNDF = 'Undefined user?'
      RV_UERR  = 'Token in error?'
      RV_UTRST = 'User trusted?'
      RV_USEST = 'Session type'
      RV_USURO = 'Surrogate user?'
      RV_URMT  = 'Remote job?'
      RV_UPRVL = 'Privileged user?'
      RV_USECL = 'User SECLABEL'
      RV_UEXND = 'Execution node'
      RV_USUSR = 'Submitting user'
      RV_USNOD = 'Submitting node'
      RV_USGRP = 'Submitting group'
      RV_USPOE = 'Port of entry'
      RV_USPCL = 'Class of POE'
      RV_UTUSR = 'Userid'
      RV_UTGRP = 'Groupid'
      RV_UTDFT = 'Default group?'
      RV_UTSEC = 'Default SECLABEL?'
      RV_APPC  = 'APPC key link'
      ;
        OUTPUT RACF.RVARY;
      END;
    %END;
%MEND RVARY;
```

```
./        ADD   LIST=ALL,NAME=APPCLU
%MACRO APPCLU(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from APPCLU extension;
      RACF.APPCLU (KEEP=%SMFHDR
                        %SMF8ØHDR(REQ=DEFINE)
                        APCRESNM
                        APCCLASS
                        APCTYPE
                        APCNAME
                        APCOWNID
                        APCUSERN
                        APCUTKNE
                        APCUPRE
                        APCUVFYX
                        APCUNJEU
                        APCUUAUD
                        APCUSPEC
                        APCUDFLT
                        APCUUNDF
                        APCUERR
                        APCUTRST
                        APCUSEST
                        APCUSURO
                        APCURMT
                        APCUPRVL
                        APCUSECL
                        APCUEXND
                        APCUSUSR
                        APCUSNOD
                        APCUSGRP
                        APCUSPOE
                        APCUSPCL
                        APCUTUSR
                        APCUTGRP
                        APCUTDFT
                        APCUTSEC
                        APCAPPC
                        APCRNAME
                  )
    %END;
  %IF &REQ = EXTRACT %THEN
    %DO;
      %PUT Including datadefinition for APPCLU extension;
      WHEN('APPCLU') DO;
        INPUT %SMF8ØHDR(REQ=EXTRACT)
              APCRESNM $       282-481
              APCCLASS $       538-545
```

```
                APCTYPE  $       547-554
                APCNAME  $       556-755
                APCOWNID $       804-811
                APCUSERN $       813-832
                APCUTKNE $       834-837
                APCUPRE  $       839-842
                APCUVFYX $       844-847
                APCUNJEU $       849-852
                APCUUAUD $       854-857
                APCUSPEC $       859-862
                APCUDFLT $       864-867
                APCUUNDF $       869-872
                APCUERR  $       874-877
                APCUTRST $       879-882
                APCUSEST $       884-891
                APCUSURO $       893-896
                APCURMT  $       898-901
                APCUPRVL $       903-906
                APCUSECL $       908-915
                APCUEXND $       917-924
                APCUSUSR $       926-933
                APCUSNOD $       935-942
                APCUSGRP $       944-951
                APCUSPOE $       953-960
                APCUSPCL $       962-969
                APCUTUSR $       971-978
                APCUTGRP $       980-987
                APCUTDFT $       989-992
                APCUTSEC $       994-997
                APCAPPC  $       999-1014
                APCRNAME $      1016-1215
                 ;
LABEL APCRESNM = 'Resource name'
      APCCLASS = 'Class name'
      APCTYPE  = 'Type of resource'
      APCNAME  = 'Profile name'
      APCOWNID = 'Profile owner'
      APCUSERN = 'User name'
      APCUTKNE = 'Utoken encr.?'
      APCUPRE  = 'Pre-1.9?'
      APCUVFYX = 'VERIFYX propagation?'
      APCUNJEU = 'Undefined NJE user?'
      APCUUAUD = 'UAUDIT?'
      APCUSPEC = 'RACF special?'
      APCUDFLT = 'Default token?'
      APCUUNDF = 'Undefined user?'
      APCUERR  = 'Token in error?'
      APCUTRST = 'User trusted?'
      APCUSEST = 'Session type'
      APCUSURO = 'Surrogate user?'
```

```
      APCURMT  = 'Remote job?'
      APCUPRVL = 'Privileged user?'
      APCUSECL = 'User SECLABEL'
      APCUEXND = 'Execution node'
      APCUSUSR = 'Submitting user'
      APCUSNOD = 'Submitting node'
      APCUSGRP = 'Submitting group'
      APCUSPOE = 'Port of entry'
      APCUSPCL = 'Class of POE'
      APCUTUSR = 'Userid'
      APCUTGRP = 'Groupid'
      APCUTDFT = 'Default group?'
      APCUTSEC = 'Default SECLABEL?'
      APCAPPC  = 'APPC key link'
      APCRNAME = 'Resource name'
      ;
        OUTPUT RACF.APPCLU;
      END;
    %END;
%MEND APPCLU;
./        ADD   LIST=ALL,NAME=GENERAL
%MACRO GENERAL(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from GENERAL extension;
      RACF.GENERAL (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         GENCLASS
                         GENLOGST
                         GENUSERN
                         GENUTKNE
                         GENUPRE
                         GENUVFYX
                         GENUNJEU
                         GENUUAUD
                         GENUSPEC
                         GENUDFLT
                         GENUUNDF
                         GENUERR
                         GENUTRST
                         GENUSEST
                         GENUSURO
                         GENURMT
                         GENUPRVL
                         GENUSECL
                         GENUEXND
                         GENUSUSR
                         GENUSNOD
                         GENUSGRP
```

```
                          GENUSPOE
                          GENUSPCL
                          GENUTUSR
                          GENUTGRP
                          GENUTDFT
                          GENUTSEC
                          GENAPPC
                    )
    %END;
  %IF &REQ = EXTRACT %THEN
    %DO;
      %PUT Including datadefinition for GENERAL extension;
      WHEN('GENERAL') DO;
        INPUT %SMF8ØHDR(REQ=EXTRACT)
              GENCLASS $       282-289
              GENLOGST $       291-49Ø
              GENUSERN $       547-566
              GENUTKNE $       568-571
              GENUPRE  $       573-576
              GENUVFYX $       578-581
              GENUNJEU $       583-586
              GENUUAUD $       588-591
              GENUSPEC $       593-596
              GENUDFLT $       598-6Ø1
              GENUUNDF $       6Ø3-6Ø6
              GENUERR  $       6Ø8-611
              GENUTRST $       613-616
              GENUSEST $       618-625
              GENUSURO $       627-63Ø
              GENURMT  $       632-635
              GENUPRVL $       637-64Ø
              GENUSECL $       642-649
              GENUEXND $       651-658
              GENUSUSR $       66Ø-667
              GENUSNOD $       669-676
              GENUSGRP $       678-685
              GENUSPOE $       687-694
              GENUSPCL $       696-7Ø3
              GENUTUSR $       7Ø5-712
              GENUTGRP $       714-721
              GENUTDFT $       723-726
              GENUTSEC $       728-731
              GENAPPC  $       733-748
               ;
LABEL GENCLASS = 'Class name'
      GENLOGST = 'Logstr'
      GENUSERN = 'User name'
      GENUTKNE = 'Utoken encr.?'
      GENUPRE  = 'Pre-1.9?'
      GENUVFYX = 'VERIFYX propagation?'
```

```
      GENUNJEU = 'Undefined NJE user?'
      GENUUAUD = 'UAUDIT?'
      GENUSPEC = 'RACF special?'
      GENUDFLT = 'Default token?'
      GENUUNDF = 'Undefined user?'
      GENUERR  = 'Token in error?'
      GENUTRST = 'User trusted?'
      GENUSEST = 'Session type'
      GENUSURO = 'Surrogate user?'
      GENURMT  = 'Remote job?'
      GENUPRVL = 'Privileged user?'
      GENUSECL = 'User SECLABEL'
      GENUEXND = 'Execution node'
      GENUSUSR = 'Submitting user'
      GENUSNOD = 'Submitting node'
      GENUSGRP = 'Submitting group'
      GENUSPOE = 'Port of entry'
      GENUSPCL = 'Class of POE'
      GENUTUSR = 'Userid'
      GENUTGRP = 'Groupid'
      GENUTDFT = 'Default group?'
      GENUTSEC = 'Default SECLABEL?'
      GENAPPC  = 'APPC key link'
      ;
        OUTPUT RACF.GENERAL;
      END;
    %END;
%MEND GENERAL;
./        ADD   LIST=ALL,NAME=DIRSRCH
%MACRO DIRSRCH(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from DIRSRCH extension;
      RACF.DIRSRCH (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         DSCCLASS
                         DSCUSERN
                         DSCUTKNE
                         DSCUPRE
                         DSCUVFYX
                         DSCUNJEU
                         DSCUUAUD
                         DSCUSPEC
                         DSCUDFLT
                         DSCUUNDF
                         DSCUERR
                         DSCUTRST
                         DSCUSEST
                         DSCUSURO
```

```
                        DSCURMT
                        DSCUPRVL
                        DSCUSECL
                        DSCUEXND
                        DSCUSUSR
                        DSCUSNOD
                        DSCUSGRP
                        DSCUSPOE
                        DSCUSPCL
                        DSCUTUSR
                        DSCUTGRP
                        DSCUTDFT
                        DSCUTSEC
                        DSCAPPC
                        DSCAUDIT
                        DSCORUID
                        DSCOEUID
                        DSCOSUID
                        DSCORGID
                        DSCOEGID
                        DSCOSGID
                        DSCPATHN
                        DSCFILID
                        DSCFOUID
                        DSCFOGID
                        DSCREQRD
                        DSCREQWR
                        DSCREQEX
                        DSCREQSC
                        DSCACTYP
                        DSCALWRD
                        DSCALWWR
                        DSCALWEX
                        DSCREQP2
                        DSCSRVCD
                        DSCHFSDS
                        DSCSYMLK
                        DSCFILNM
                        DSCPTHTP
                        DSCFILPL
                        DSCFILSP
                        DSCINODE
                        DSCSCID
                        DSCDCELK
                        DSCAUTYP
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for DIRSRCH extension;
```

```
WHEN('DIRSRCH') DO;
  INPUT %SMF80HDR(REQ=EXTRACT)
        DSCCLASS $       282-289
        DSCUSERN $       291-310
        DSCUTKNE $       312-315
        DSCUPRE  $       317-320
        DSCUVFYX $       322-325
        DSCUNJEU $       327-330
        DSCUUAUD $       332-335
        DSCUSPEC $       337-340
        DSCUDFLT $       342-345
        DSCUUNDF $       347-350
        DSCUERR  $       352-355
        DSCUTRST $       357-360
        DSCUSEST $       362-369
        DSCUSURO $       371-374
        DSCURMT  $       376-379
        DSCUPRVL $       381-384
        DSCUSECL $       386-393
        DSCUEXND $       395-402
        DSCUSUSR $       404-411
        DSCUSNOD $       413-420
        DSCUSGRP $       422-429
        DSCUSPOE $       431-438
        DSCUSPCL $       440-447
        DSCUTUSR $       449-456
        DSCUTGRP $       458-465
        DSCUTDFT $       467-470
        DSCUTSEC $       472-475
        DSCAPPC  $       477-492
        DSCAUDIT $       494-504
        DSCORUID         506-515
        DSCOEUID         517-526
        DSCOSUID         528-537
        DSCORGID         539-548
        DSCOEGID         550-559
        DSCOSGID         561-570
        DSCPATHN $       572-771
        DSCFILID $      1596-1627
        DSCFOUID        1629-1638
        DSCFOGID        1640-1649
        DSCREQRD $      1651-1654
        DSCREQWR $      1656-1659
        DSCREQEX $      1661-1664
        DSCREQSC $      1666-1669
        DSCACTYP $      1671-1678
        DSCALWRD $      1680-1683
        DSCALWWR $      1685-1688
        DSCALWEX $      1690-1693
        DSCREQP2 $      1695-1894
```

```
                DSCSRVCD $      2719-2729
                DSCHFSDS $      2731-2774
                DSCSYMLK $      2776-2975
                DSCFILNM $      3800-3999
                DSCPTHTP $      4057-4060
                DSCFILPL $      4062-4069
                DSCFILSP $      4071-4078
                DSCINODE        4080-4089
                DSCSCID         4091-4100
                DSCDCELK $      4102-4117
                DSCAUTYP $      4119-4131
                 ;
LABEL DSCCLASS = 'Class name'
      DSCUSERN = 'User name'
      DSCUTKNE = 'Utoken encr.?'
      DSCUPRE  = 'Pre-1.9?'
      DSCUVFYX = 'VERIFYX propagation?'
      DSCUNJEU = 'Undefined NJE user?'
      DSCUUAUD = 'UAUDIT?'
      DSCUSPEC = 'RACF special?'
      DSCUDFLT = 'Default token?'
      DSCUUNDF = 'Undefined user?'
      DSCUERR  = 'Token in error?'
      DSCUTRST = 'User trusted?'
      DSCUSEST = 'Session type'
      DSCUSURO = 'Surrogate user?'
      DSCURMT  = 'Remote job?'
      DSCUPRVL = 'Privileged user?'
      DSCUSECL = 'User SECLABEL'
      DSCUEXND = 'Execution node'
      DSCUSUSR = 'Submitting user'
      DSCUSNOD = 'Submitting node'
      DSCUSGRP = 'Submitting group'
      DSCUSPOE = 'Port of entry'
      DSCUSPCL = 'Class of POE'
      DSCUTUSR = 'Userid'
      DSCUTGRP = 'Groupid'
      DSCUTDFT = 'Default group?'
      DSCUTSEC = 'Default SECLABEL?'
      DSCAPPC  = 'APPC key link'
      DSCAUDIT = 'Audit code'
      DSCORUID = 'Old real UID'
      DSCOEUID = 'Old effective UID'
      DSCOSUID = 'Old saved UID'
      DSCORGID = 'Old real GID'
      DSCOEGID = 'Old effective GID'
      DSCOSGID = 'Old saved GID'
      DSCPATHN = 'Path name'
      DSCFILID = 'File id'
      DSCFOUID = 'Owner UID'
```

```
      DSCFOGID = 'Owner GID'
      DSCREQRD = 'Read acc req?'
      DSCREQWR = 'Write acc req?'
      DSCREQEX = 'Exec acc req?'
      DSCREQSC = 'Dir search req?'
      DSCACTYP = 'Access type'
      DSCALWRD = 'Read allowed?'
      DSCALWWR = 'Write allowed?'
      DSCALWEX = 'Exec allowed?'
      DSCREQP2 = '2nd path name'
      DSCSRVCD = 'Service code'
      DSCHFSDS = 'HFS datasetname'
      DSCSYMLK = 'SYMLINK'
      DSCFILNM = 'Filename checked'
      DSCPTHTP = 'Path type'
      DSCFILPL = 'File pool'
      DSCFILSP = 'File space'
      DSCINODE = 'Inode'
      DSCSCID  = 'File SCID'
      DSCDCELK = 'DCE link'
      DSCAUTYP = 'Request type'
      ;
        OUTPUT RACF.DIRSRCH;
      END;
    %END;
%MEND DIRSRCH;
./        ADD   LIST=ALL,NAME=DACCESS
%MACRO DACCESS(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from DACCESS extension;
      RACF.DACCESS (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         DACCLASS
                         DACUSERN
                         DACUTKNE
                         DACUPRE
                         DACUVFYX
                         DACUNJEU
                         DACUUAUD
                         DACUSPEC
                         DACUDFLT
                         DACUUNDF
                         DACUERR
                         DACUTRST
                         DACUSEST
                         DACUSURO
                         DACURMT
                         DACUPRVL
```

```
                        DACUSECL
                        DACUEXND
                        DACUSUSR
                        DACUSNOD
                        DACUSGRP
                        DACUSPOE
                        DACUSPCL
                        DACUTUSR
                        DACUTGRP
                        DACUTDFT
                        DACUTSEC
                        DACAPPC
                        DACAUDIT
                        DACORUID
                        DACOEUID
                        DACOSUID
                        DACORGID
                        DACOEGID
                        DACOSGID
                        DACPATHN
                        DACFILID
                        DACFOUID
                        DACFOGID
                        DACREQRD
                        DACREQWR
                        DACREQEX
                        DACREQSC
                        DACACTYP
                        DACALWRD
                        DACALWWR
                        DACALWEX
                        DACREQP2
                        DACSYMLK
                        DACFILNM
                        DACPTHTP
                        DACFILPL
                        DACFILSP
                        DACINODE
                        DACSCID
                        DACDCELK
                        DACAUTYP
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for DACCESS extension;
    WHEN('DACCESS') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            DACCLASS $       282-289
            DACUSERN $       291-31Ø
```

```
DACUTKNE $       312-315
DACUPRE  $       317-320
DACUVFYX $       322-325
DACUNJEU $       327-330
DACUUAUD $       332-335
DACUSPEC $       337-340
DACUDFLT $       342-345
DACUUNDF $       347-350
DACUERR  $       352-355
DACUTRST $       357-360
DACUSEST $       362-369
DACUSURO $       371-374
DACURMT  $       376-379
DACUPRVL $       381-384
DACUSECL $       386-393
DACUEXND $       395-402
DACUSUSR $       404-411
DACUSNOD $       413-420
DACUSGRP $       422-429
DACUSPOE $       431-438
DACUSPCL $       440-447
DACUTUSR $       449-456
DACUTGRP $       458-465
DACUTDFT $       467-470
DACUTSEC $       472-475
DACAPPC  $       477-492
DACAUDIT $       494-504
DACORUID         506-515
DACOEUID         517-526
DACOSUID         528-537
DACORGID         539-548
DACOEGID         550-559
DACOSGID         561-570
DACPATHN $       572-771
DACFILID $      1596-1627
DACFOUID        1629-1638
DACFOGID        1640-1649
DACREQRD $      1651-1654
DACREQWR $      1656-1659
DACREQEX $      1661-1664
DACREQSC $      1666-1669
DACACTYP $      1671-1678
DACALWRD $      1680-1683
DACALWWR $      1685-1688
DACALWEX $      1690-1693
DACREQP2 $      1695-1894
DACSYMLK $      2719-2918
DACFILNM $      3743-3942
DACPTHTP $      4000-4003
DACFILPL $      4005-4012
```

```
                DACFILSP $      4014-4021
                DACINODE        4023-4032
                DACSCID         4034-4043
                DACDCELK $      4045-4060
                DACAUTYP $      4062-4074
                 ;
LABEL DACCLASS = 'Class name'
      DACUSERN = 'User name'
      DACUTKNE = 'Utoken encr.?'
      DACUPRE  = 'Pre-1.9?'
      DACUVFYX = 'VERIFYX propagation?'
      DACUNJEU = 'Undefined NJE user?'
      DACUUAUD = 'UAUDIT?'
      DACUSPEC = 'RACF special?'
      DACUDFLT = 'Default token?'
      DACUUNDF = 'Undefined user?'
      DACUERR  = 'Token in error?'
      DACUTRST = 'User trusted?'
      DACUSEST = 'Session type'
      DACUSURO = 'Surrogate user?'
      DACURMT  = 'Remote job?'
      DACUPRVL = 'Privileged user?'
      DACUSECL = 'User SECLABEL'
      DACUEXND = 'Execution node'
      DACUSUSR = 'Submitting user'
      DACUSNOD = 'Submitting node'
      DACUSGRP = 'Submitting group'
      DACUSPOE = 'Port of entry'
      DACUSPCL = 'Class of POE'
      DACUTUSR = 'Userid'
      DACUTGRP = 'Groupid'
      DACUTDFT = 'Default group?'
      DACUTSEC = 'Default SECLABEL?'
      DACAPPC  = 'APPC key link'
      DACAUDIT = 'Audit code'
      DACORUID = 'Old real UID'
      DACOEUID = 'Old effective UID'
      DACOSUID = 'Old saved UID'
      DACORGID = 'Old real GID'
      DACOEGID = 'Old effective GID'
      DACOSGID = 'Old saved GID'
      DACPATHN = 'Path name'
      DACFILID = 'File id'
      DACFOUID = 'Owner UID'
      DACFOGID = 'Owner GID'
      DACREQRD = 'Read acc req?'
      DACREQWR = 'Write acc req?'
      DACREQEX = 'Exec acc req?'
      DACREQSC = 'Dir search req?'
      DACACTYP = 'Access type'
```

```
      DACALWRD = 'Read allowed?'
      DACALWWR = 'Write allowed?'
      DACALWEX = 'Exec allowed?'
      DACREQP2 = '2nd path name'
      DACSYMLK = 'SYMLINK'
      DACFILNM = 'Filename checked'
      DACPTHTP = 'Path type'
      DACFILPL = 'File pool'
      DACFILSP = 'File space'
      DACINODE = 'Inode'
      DACSCID  = 'File SCID'
      DACDCELK = 'DCE link'
      DACAUTYP = 'Request type'
      ;
        OUTPUT RACF.DACCESS;
      END;
    %END;
%MEND DACCESS;
./        ADD   LIST=ALL,NAME=FACCESS
%MACRO FACCESS(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from FACCESS extension;
      RACF.FACCESS (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         FACCLASS
                         FACUSERN
                         FACUTKNE
                         FACUPRE
                         FACUVFYX
                         FACUNJEU
                         FACUUAUD
                         FACUSPEC
                         FACUDFLT
                         FACUUNDF
                         FACUERR
                         FACUTRST
                         FACUSEST
                         FACUSURO
                         FACURMT
                         FACUPRVL
                         FACUSECL
                         FACUEXND
                         FACUSUSR
                         FACUSNOD
                         FACUSGRP
                         FACUSPOE
                         FACUSPCL
                         FACUTUSR
```

```
                        FACUTGRP
                        FACUTDFT
                        FACUTSEC
                        FACAPPC
                        FACAUDIT
                        FACORUID
                        FACOEUID
                        FACOSUID
                        FACORGID
                        FACOEGID
                        FACOSGID
                        FACPATHN
                        FACFILID
                        FACFOUID
                        FACFOGID
                        FACREQRD
                        FACREQWR
                        FACREQEX
                        FACREQSC
                        FACACTYP
                        FACALWRD
                        FACALWWR
                        FACALWEX
                        FACREQP2
                        FACFILNM
                        FACPTHTP
                        FACFILPL
                        FACFILSP
                        FACINODE
                        FACSCID
                        FACDCELK
                        FACAUTYP
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for FACCESS extension;
    WHEN('FACCESS') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            FACCLASS $       282-289
            FACUSERN $       291-31Ø
            FACUTKNE $       312-315
            FACUPRE  $       317-32Ø
            FACUVFYX $       322-325
            FACUNJEU $       327-33Ø
            FACUUAUD $       332-335
            FACUSPEC $       337-34Ø
            FACUDFLT $       342-345
            FACUUNDF $       347-35Ø
            FACUERR  $       352-355
```

```
                FACUTRST $       357-360
                FACUSEST $       362-369
                FACUSURO $       371-374
                FACURMT  $       376-379
                FACUPRVL $       381-384
                FACUSECL $       386-393
                FACUEXND $       395-402
                FACUSUSR $       404-411
                FACUSNOD $       413-420
                FACUSGRP $       422-429
                FACUSPOE $       431-438
                FACUSPCL $       440-447
                FACUTUSR $       449-456
                FACUTGRP $       458-465
                FACUTDFT $       467-470
                FACUTSEC $       472-475
                FACAPPC  $       477-492
                FACAUDIT $       494-504
                FACORUID         506-515
                FACOEUID         517-526
                FACOSUID         528-537
                FACORGID         539-548
                FACOEGID         550-559
                FACOSGID         561-570
                FACPATHN $       572-771
                FACFILID $      1596-1627
                FACFOUID        1629-1638
                FACFOGID        1640-1649
                FACREQRD $      1651-1654
                FACREQWR $      1656-1659
                FACREQEX $      1661-1664
                FACREQSC $      1666-1669
                FACACTYP $      1671-1678
                FACALWRD $      1680-1683
                FACALWWR $      1685-1688
                FACALWEX $      1690-1693
                FACREQP2 $      1695-1894
                FACFILNM $      2719-2918
                FACPTHTP $      2976-2979
                FACFILPL $      2981-2988
                FACFILSP $      2990-2997
                FACINODE        2999-3008
                FACSCID         3010-3019
                FACDCELK $      3021-3036
                FACAUTYP $      3038-3050
                ;
LABEL FACCLASS = 'Class name'
      FACUSERN = 'User name'
      FACUTKNE = 'Utoken encr.?'
      FACUPRE  = 'Pre-1.9?'
```

```
FACUVFYX = 'VERIFYX propagation?'
FACUNJEU = 'Undefined NJE user?'
FACUUAUD = 'UAUDIT?'
FACUSPEC = 'RACF special?'
FACUDFLT = 'Default token?'
FACUUNDF = 'Undefined user?'
FACUERR  = 'Token in error?'
FACUTRST = 'User trusted?'
FACUSEST = 'Session type'
FACUSURO = 'Surrogate user?'
FACURMT  = 'Remote job?'
FACUPRVL = 'Privileged user?'
FACUSECL = 'User SECLABEL'
FACUEXND = 'Execution node'
FACUSUSR = 'Submitting user'
FACUSNOD = 'Submitting node'
FACUSGRP = 'Submitting group'
FACUSPOE = 'Port of entry'
FACUSPCL = 'Class of POE'
FACUTUSR = 'Userid'
FACUTGRP = 'Groupid'
FACUTDFT = 'Default group?'
FACUTSEC = 'Default SECLABEL?'
FACAPPC  = 'APPC key link'
FACAUDIT = 'Audit code'
FACORUID = 'Old real UID'
FACOEUID = 'Old effective UID'
FACOSUID = 'Old saved UID'
FACORGID = 'Old real GID'
FACOEGID = 'Old effective GID'
FACOSGID = 'Old saved GID'
FACPATHN = 'Path name'
FACFILID = 'File id'
FACFOUID = 'Owner UID'
FACFOGID = 'Owner GID'
FACREQRD = 'Read acc req?'
FACREQWR = 'Write acc req?'
FACREQEX = 'Exec acc req?'
FACREQSC = 'Dir search req?'
FACACTYP = 'Access type'
FACALWRD = 'Read allowed?'
FACALWWR = 'Write allowed?'
FACALWEX = 'Exec allowed?'
FACREQP2 = '2nd path name'
FACFILNM = 'Filename checked'
FACPTHTP = 'Path type'
FACFILPL = 'File pool'
FACFILSP = 'File space'
FACINODE = 'Inode'
FACSCID  = 'File SCID'
```

```
      FACDCELK = 'DCE link'
      FACAUTYP = 'Request type'
      ;
        OUTPUT RACF.FACCESS;
      END;
    %END;
%MEND FACCESS;
./        ADD   LIST=ALL,NAME=CHAUDIT
%MACRO CHAUDIT(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from CHAUDIT extension;
      RACF.CHAUDIT (KEEP=%SMFHDR
                         %SMF8ØHDR(REQ=DEFINE)
                         CAUCLASS
                         CAUUSERN
                         CAUUTKNE
                         CAUUPRE
                         CAUUVFYX
                         CAUUNJEU
                         CAUUUAUD
                         CAUUSPEC
                         CAUUDFLT
                         CAUUUNDF
                         CAUUERR
                         CAUUTRST
                         CAUUSEST
                         CAUUSURO
                         CAUURMT
                         CAUUPRVL
                         CAUUSECL
                         CAUUEXND
                         CAUUSUSR
                         CAUUSNOD
                         CAUUSGRP
                         CAUUSPOE
                         CAUUSPCL
                         CAUUTUSR
                         CAUUTGRP
                         CAUUTDFT
                         CAUUTSEC
                         CAUAPPC
                         CAUAUDIT
                         CAUORUID
                         CAUOEUID
                         CAUOSUID
                         CAUORGID
                         CAUOEGID
                         CAUOSGID
```

```
                        CAUPATHN
                        CAUFILID
                        CAUFOUID
                        CAUFOGID
                        CAUREQRD
                        CAUREQWR
                        CAUREQEX
                        CAUUOLRD
                        CAUUOLWR
                        CAUUOLEX
                        CAUAOLRD
                        CAUAOLWR
                        CAUAOLEX
                        CAUUNWRD
                        CAUUNWWR
                        CAUUNWEX
                        CAUANWRD
                        CAUANWWR
                        CAUANWEX
                        CAUFILPL
                        CAUFILSP
                        CAUINODE
                        CAUSCID
                        CAUDCELK
                        CAUAUTYP
                    )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for CHAUDIT extension;
    WHEN('CHAUDIT') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            CAUCLASS $       282-289
            CAUUSERN $       291-31Ø
            CAUUTKNE $       312-315
            CAUUPRE  $       317-32Ø
            CAUUVFYX $       322-325
            CAUUNJEU $       327-33Ø
            CAUUUAUD $       332-335
            CAUUSPEC $       337-34Ø
            CAUUDFLT $       342-345
            CAUUUNDF $       347-35Ø
            CAUUERR  $       352-355
            CAUUTRST $       357-36Ø
            CAUUSEST $       362-369
            CAUUSURO $       371-374
            CAUURMT  $       376-379
            CAUUPRVL $       381-384
            CAUUSECL $       386-393
            CAUUEXND $       395-4Ø2
```

```
                CAUUSUSR $       4Ø4-411
                CAUUSNOD $       413-42Ø
                CAUUSGRP $       422-429
                CAUUSPOE $       431-438
                CAUUSPCL $       44Ø-447
                CAUUTUSR $       449-456
                CAUUTGRP $       458-465
                CAUUTDFT $       467-47Ø
                CAUUTSEC $       472-475
                CAUAPPC  $       477-492
                CAUAUDIT $       494-5Ø4
                CAUORUID         5Ø6-515
                CAUOEUID         517-526
                CAUOSUID         528-537
                CAUORGID         539-548
                CAUOEGID         55Ø-559
                CAUOSGID         561-57Ø
                CAUPATHN $       572-771
                CAUFILID $      1596-1627
                CAUFOUID        1629-1638
                CAUFOGID        164Ø-1649
                CAUREQRD $      1651-1658
                CAUREQWR $      166Ø-1667
                CAUREQEX $      1669-1676
                CAUUOLRD $      1678-1685
                CAUUOLWR $      1687-1694
                CAUUOLEX $      1696-17Ø3
                CAUAOLRD $      17Ø5-1712
                CAUAOLWR $      1714-1721
                CAUAOLEX $      1723-173Ø
                CAUUNWRD $      1732-1739
                CAUUNWWR $      1741-1748
                CAUUNWEX $      175Ø-1757
                CAUANWRD $      1759-1766
                CAUANWWR $      1768-1775
                CAUANWEX $      1777-1784
                CAUFILPL $      1786-1793
                CAUFILSP $      1795-18Ø2
                CAUINODE        18Ø4-1813
                CAUSCID         1815-1824
                CAUDCELK $      1824-1841
                CAUAUTYP $      1843-1855
                 ;
LABEL CAUCLASS = 'Class name'
      CAUUSERN = 'User name'
      CAUUTKNE = 'Utoken encr.?'
      CAUUPRE  = 'Pre-1.9?'
      CAUUVFYX = 'VERIFYX propagation?'
      CAUUNJEU = 'Undefined NJE user?'
      CAUUUAUD = 'UAUDIT?'
```

```
CAUUSPEC = 'RACF special?'
CAUUDFLT = 'Default token?'
CAUUUNDF = 'Undefined user?'
CAUUERR  = 'Token in error?'
CAUUTRST = 'User trusted?'
CAUUSEST = 'Session type'
CAUUSURO = 'Surrogate user?'
CAUURMT  = 'Remote job?'
CAUUPRVL = 'Privileged user?'
CAUUSECL = 'User SECLABEL'
CAUUEXND = 'Execution node'
CAUUSUSR = 'Submitting user'
CAUUSNOD = 'Submitting node'
CAUUSGRP = 'Submitting group'
CAUUSPOE = 'Port of entry'
CAUUSPCL = 'Class of POE'
CAUUTUSR = 'Userid'
CAUUTGRP = 'Groupid'
CAUUTDFT = 'Default group?'
CAUUTSEC = 'Default SECLABEL?'
CAUAPPC  = 'APPC key link'
CAUAUDIT = 'Audit code'
CAUORUID = 'Old real UID'
CAUOEUID = 'Old effective UID'
CAUOSUID = 'Old saved UID'
CAUORGID = 'Old real GID'
CAUOEGID = 'Old effective GID'
CAUOSGID = 'Old saved GID'
CAUPATHN = 'Path name'
CAUFILID = 'File id'
CAUFOUID = 'Owner UID'
CAUFOGID = 'Owner GID'
CAUREQRD = 'Req audit opt read'
CAUREQWR = 'Req audit opt write'
CAUREQEX = 'Req audit opt exec'
CAUUOLRD = 'Old user audit opt read'
CAUUOLWR = 'Old user audit opt write'
CAUUOLEX = 'Old user audit opt exec'
CAUAOLRD = 'Old aud. audit opt read'
CAUAOLWR = 'Old aud. audit opt write'
CAUAOLEX = 'Old aud. audit opt exec'
CAUUNWRD = 'New user audit opt read'
CAUUNWWR = 'New user audit opt write'
CAUUNWEX = 'New user audit opt exec'
CAUANWRD = 'New aud. audit opt read'
CAUANWWR = 'New aud. audit opt write'
CAUANWEX = 'New aud. audit opt exec'
CAUFILPL = 'File pool'
CAUFILSP = 'File space'
CAUINODE = 'Inode'
```

```
      CAUSCID  = 'File SCID'
      CAUDCELK = 'DCE link'
      CAUAUTYP = 'Request type'
      ;
        OUTPUT RACF.CHAUDIT;
      END;
    %END;
%MEND CHAUDIT;
./        ADD   LIST=ALL,NAME=CHDIR
%MACRO CHDIR(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from CHDIR extension;
      RACF.CHDIR (KEEP=%SMFHDR
                       %SMF8ØHDR(REQ=DEFINE)
                       CHDCLASS
                       CHDUSERN
                       CHDUTKNE
                       CHDUPRE
                       CHDUVFYX
                       CHDUNJEU
                       CHDUUAUD
                       CHDUSPEC
                       CHDUDFLT
                       CHDUUNDF
                       CHDUERR
                       CHDUTRST
                       CHDUSEST
                       CHDUSURO
                       CHDURMT
                       CHDUPRVL
                       CHDUSECL
                       CHDUEXND
                       CHDUSUSR
                       CHDUSNOD
                       CHDUSGRP
                       CHDUSPOE
                       CHDUSPCL
                       CHDUTUSR
                       CHDUTGRP
                       CHDUTDFT
                       CHDUTSEC
                       CHDAPPC
                       CHDAUDIT
                       CHDORUID
                       CHDOEUID
                       CHDOSUID
                       CHDORGID
                       CHDOEGID
```

```
                    CHDOSGID
                    CHDPATHN
                    CHDFILID
                    CHDFOUID
                    CHDFOGID
                    CHDDCELK
                    CHDAUTYP
                )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for CHDIR extension;
    WHEN('CHDIR') DO;
      INPUT %SMF80HDR(REQ=EXTRACT)
            CHDCLASS $       282-289
            CHDUSERN $       291-310
            CHDUTKNE $       312-315
            CHDUPRE  $       317-320
            CHDUVFYX $       322-325
            CHDUNJEU $       327-330
            CHDUUAUD $       332-335
            CHDUSPEC $       337-340
            CHDUDFLT $       342-345
            CHDUUNDF $       347-350
            CHDUERR  $       352-355
            CHDUTRST $       357-360
            CHDUSEST $       362-369
            CHDUSURO $       371-374
            CHDURMT  $       376-379
            CHDUPRVL $       381-384
            CHDUSECL $       386-393
            CHDUEXND $       395-402
            CHDUSUSR $       404-411
            CHDUSNOD $       413-420
            CHDUSGRP $       422-429
            CHDUSPOE $       431-438
            CHDUSPCL $       440-447
            CHDUTUSR $       449-456
            CHDUTGRP $       458-465
            CHDUTDFT $       467-470
            CHDUTSEC $       472-475
            CHDAPPC  $       477-492
            CHDAUDIT $       494-504
            CHDORUID         506-515
            CHDOEUID         517-526
            CHDOSUID         528-537
            CHDORGID         539-548
            CHDOEGID         550-559
            CHDOSGID         561-570
            CHDPATHN $       572-771
```

```
               CHDFILID $      1596-1627
               CHDFOUID        1629-1638
               CHDFOGID        164Ø-1649
               CHDDCELK $      1651-1666
               CHDAUTYP $      1668-168Ø
                ;
LABEL CHDCLASS = 'Class name'
      CHDUSERN = 'User name'
      CHDUTKNE = 'Utoken encr.?'
      CHDUPRE  = 'Pre-1.9?'
      CHDUVFYX = 'VERIFYX propagation?'
      CHDUNJEU = 'Undefined NJE user?'
      CHDUUAUD = 'UAUDIT?'
      CHDUSPEC = 'RACF special?'
      CHDUDFLT = 'Default token?'
      CHDUUNDF = 'Undefined user?'
      CHDUERR  = 'Token in error?'
      CHDUTRST = 'User trusted?'
      CHDUSEST = 'Session type'
      CHDUSURO = 'Surrogate user?'
      CHDURMT  = 'Remote job?'
      CHDUPRVL = 'Privileged user?'
      CHDUSECL = 'User SECLABEL'
      CHDUEXND = 'Execution node'
      CHDUSUSR = 'Submitting user'
      CHDUSNOD = 'Submitting node'
      CHDUSGRP = 'Submitting group'
      CHDUSPOE = 'Port of entry'
      CHDUSPCL = 'Class of POE'
      CHDUTUSR = 'Userid'
      CHDUTGRP = 'Groupid'
      CHDUTDFT = 'Default group?'
      CHDUTSEC = 'Default SECLABEL?'
      CHDAPPC  = 'APPC key link'
      CHDAUDIT = 'Audit code'
      CHDORUID = 'Old real UID'
      CHDOEUID = 'Old effective UID'
      CHDOSUID = 'Old saved UID'
      CHDORGID = 'Old real GID'
      CHDOEGID = 'Old effective GID'
      CHDOSGID = 'Old saved GID'
      CHDPATHN = 'Path name'
      CHDFILID = 'File id'
      CHDFOUID = 'Owner UID'
      CHDFOGID = 'Owner GID'
      CHDDCELK = 'DCE link'
      CHDAUTYP = 'Request type'
      ;
        OUTPUT RACF.CHDIR;
      END;
```

```
    %END;
%MEND CHDIR;
./        ADD   LIST=ALL,NAME=CHMOD
%MACRO CHMOD(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from CHMOD extension;
      RACF.CHMOD (KEEP=%SMFHDR
                       %SMF8ØHDR(REQ=DEFINE)
                       CHMCLASS
                       CHMUSERN
                       CHMUTKNE
                       CHMUPRE
                       CHMUVFYX
                       CHMUNJEU
                       CHMUUAUD
                       CHMUSPEC
                       CHMUDFLT
                       CHMUUNDF
                       CHMUERR
                       CHMUTRST
                       CHMUSEST
                       CHMUSURO
                       CHMURMT
                       CHMUPRVL
                       CHMUSECL
                       CHMUEXND
                       CHMUSUSR
                       CHMUSNOD
                       CHMUSGRP
                       CHMUSPOE
                       CHMUSPCL
                       CHMUTUSR
                       CHMUTGRP
                       CHMUTDFT
                       CHMUTSEC
                       CHMAPPC
                       CHMAUDIT
                       CHMORUID
                       CHMOEUID
                       CHMOSUID
                       CHMORGID
                       CHMOEGID
                       CHMOSGID
                       CHMPATHN
                       CHMFILID
                       CHMFOUID
                       CHMFOGID
                       CHMOLSGI
```

```
                    CHMOLSUI
                    CHMOLSVT
                    CHMOLORD
                    CHMOLOWR
                    CHMOLOEX
                    CHMOLGRD
                    CHMOLGWR
                    CHMOLGEX
                    CHMOLWRD
                    CHMOLWWR
                    CHMOLWEX
                    CHMNWSGI
                    CHMNWSUI
                    CHMNWSVT
                    CHMNWORD
                    CHMNWOWR
                    CHMNWOEX
                    CHMNWGRD
                    CHMNWGWR
                    CHMNWGEX
                    CHMNWWRD
                    CHMNWWWR
                    CHMNWWEX
                    CHMRQSGI
                    CHMRQSUI
                    CHMRQSVT
                    CHMRQORD
                    CHMRQOWR
                    CHMRQOEX
                    CHMRQGRD
                    CHMRQGWR
                    CHMRQGEX
                    CHMRQWRD
                    CHMRQWWR
                    CHMRQWEX
                    CHMFILPL
                    CHMFILSP
                    CHMINODE
                    CHMSCID
                    CHMDCELK
                    CHMAUTYP
                )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for CHMOD extension;
    WHEN('CHMOD') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            CHMCLASS $       282-289
            CHMUSERN $       291-31Ø
```

```
CHMUTKNE $       312-315
CHMUPRE  $       317-320
CHMUVFYX $       322-325
CHMUNJEU $       327-330
CHMUUAUD $       332-335
CHMUSPEC $       337-340
CHMUDFLT $       342-345
CHMUUNDF $       347-350
CHMUERR  $       352-355
CHMUTRST $       357-360
CHMUSEST $       362-369
CHMUSURO $       371-374
CHMURMT  $       376-379
CHMUPRVL $       381-384
CHMUSECL $       386-393
CHMUEXND $       395-402
CHMUSUSR $       404-411
CHMUSNOD $       413-420
CHMUSGRP $       422-429
CHMUSPOE $       431-438
CHMUSPCL $       440-447
CHMUTUSR $       449-456
CHMUTGRP $       458-465
CHMUTDFT $       467-470
CHMUTSEC $       472-475
CHMAPPC  $       477-492
CHMAUDIT $       494-504
CHMORUID         506-515
CHMOEUID         517-526
CHMOSUID         528-537
CHMORGID         539-548
CHMOEGID         550-559
CHMOSGID         561-570
CHMPATHN $       572-771
CHMFILID $      1596-1627
CHMFOUID        1629-1638
CHMFOGID        1640-1649
CHMOLSGI $      1651-1654
CHMOLSUI $      1656-1659
CHMOLSVT $      1661-1664
CHMOLORD $      1666-1669
CHMOLOWR $      1671-1674
CHMOLOEX $      1676-1679
CHMOLGRD $      1681-1684
CHMOLGWR $      1686-1689
CHMOLGEX $      1691-1694
CHMOLWRD $      1696-1699
CHMOLWWR $      1701-1704
CHMOLWEX $      1706-1709
CHMNWSGI $      1711-1714
```

```
                CHMNWSUI $      1716-1719
                CHMNWSVT $      1721-1724
                CHMNWORD $      1726-1729
                CHMNWOWR $      1731-1734
                CHMNWOEX $      1736-1739
                CHMNWGRD $      1741-1744
                CHMNWGWR $      1746-1749
                CHMNWGEX $      1751-1754
                CHMNWWRD $      1756-1759
                CHMNWWWR $      1761-1764
                CHMNWWEX $      1766-1769
                CHMRQSGI $      1771-1774
                CHMRQSUI $      1776-1779
                CHMRQSVT $      1781-1784
                CHMRQORD $      1786-1789
                CHMRQOWR $      1791-1794
                CHMRQOEX $      1796-1799
                CHMRQGRD $      18Ø1-18Ø4
                CHMRQGWR $      18Ø6-18Ø9
                CHMRQGEX $      1811-1814
                CHMRQWRD $      1816-1819
                CHMRQWWR $      1821-1824
                CHMRQWEX $      1826-1829
                CHMFILPL $      1831-1838
                CHMFILSP $      184Ø-1847
                CHMINODE        1849-1858
                CHMSCID         186Ø-1869
                CHMDCELK $      1871-1886
                CHMAUTYP $      1888-19ØØ
                 ;
LABEL CHMCLASS = 'Class name'
      CHMUSERN = 'User name'
      CHMUTKNE = 'Utoken encr.?'
      CHMUPRE  = 'Pre-1.9?'
      CHMUVFYX = 'VERIFYX propagation?'
      CHMUNJEU = 'Undefined NJE user?'
      CHMUUAUD = 'UAUDIT?'
      CHMUSPEC = 'RACF special?'
      CHMUDFLT = 'Default token?'
      CHMUUNDF = 'Undefined user?'
      CHMUERR  = 'Token in error?'
      CHMUTRST = 'User trusted?'
      CHMUSEST = 'Session type'
      CHMUSURO = 'Surrogate user?'
      CHMURMT  = 'Remote job?'
      CHMUPRVL = 'Privileged user?'
      CHMUSECL = 'User SECLABEL'
      CHMUEXND = 'Execution node'
      CHMUSUSR = 'Submitting user'
      CHMUSNOD = 'Submitting node'
```

```
CHMUSGRP = 'Submitting group'
CHMUSPOE = 'Port of entry'
CHMUSPCL = 'Class of POE'
CHMUTUSR = 'Userid'
CHMUTGRP = 'Groupid'
CHMUTDFT = 'Default group?'
CHMUTSEC = 'Default SECLABEL?'
CHMAPPC  = 'APPC key link'
CHMAUDIT = 'Audit code'
CHMORUID = 'Old real UID'
CHMOEUID = 'Old effective UID'
CHMOSUID = 'Old saved UID'
CHMORGID = 'Old real GID'
CHMOEGID = 'Old effective GID'
CHMOSGID = 'Old saved GID'
CHMPATHN = 'Path name'
CHMFILID = 'File id'
CHMFOUID = 'Owner UID'
CHMFOGID = 'Owner GID'
CHMOLSGI = 'Old S_ISGID requested?'
CHMOLSUI = 'Old S_ISUID requested?'
CHMOLSVT = 'Old S_ISVTX requested?'
CHMOLORD = 'Old Owner read?'
CHMOLOWR = 'Old Owner write?'
CHMOLOEX = 'Old Owner exec?'
CHMOLGRD = 'Old Group read?'
CHMOLGWR = 'Old Group write?'
CHMOLGEX = 'Old Group exec?'
CHMOLWRD = 'Old Other read?'
CHMOLWWR = 'Old Other write?'
CHMOLWEX = 'Old Other exec?'
CHMNWSGI = 'New S_ISGID requested?'
CHMNWSUI = 'New S_ISUID requested?'
CHMNWSVT = 'New S_ISVTX requested?'
CHMNWORD = 'New Owner read?'
CHMNWOWR = 'New Owner write?'
CHMNWOEX = 'New Owner exec?'
CHMNWGRD = 'New Group read?'
CHMNWGWR = 'New Group write?'
CHMNWGEX = 'New Group exec?'
CHMNWWRD = 'New Other read?'
CHMNWWWR = 'New Other write?'
CHMNWWEX = 'New Other exec?'
CHMRQSGI = 'Req S_ISGID?'
CHMRQSUI = 'Req S_ISUID?'
CHMRQSVT = 'Req S_ISVTX?'
CHMRQORD = 'Req Owner read?'
CHMRQOWR = 'Req Owner write?'
CHMRQOEX = 'Req Owner exec?'
CHMRQGRD = 'Req Group read?'
```

```
      CHMRQGWR = 'Req Group write?'
      CHMRQGEX = 'Req Group exec?'
      CHMRQWRD = 'Req Other read?'
      CHMRQWWR = 'Req Other write?'
      CHMRQWEX = 'Req Other exec?'
      CHMFILPL = 'File pool'
      CHMFILSP = 'File space'
      CHMINODE = 'Inode'
      CHMSCID  = 'File SCID'
      CHMDCELK = 'DCE link'
      CHMAUTYP = 'Request type'
      ;
        OUTPUT RACF.CHMOD;
      END;
    %END;
%MEND CHMOD;
./        ADD   LIST=ALL,NAME=CHOWN
%MACRO CHOWN(REQ=);
  %LET REQ = %UPCASE(&REQ);
  %IF &REQ = DEFINE %THEN
    %DO;
      %PUT Including variables from CHOWN extension;
      RACF.CHOWN (KEEP=%SMFHDR
                       %SMF8ØHDR(REQ=DEFINE)
                       CHOCLASS
                       CHOUSERN
                       CHOUTKNE
                       CHOUPRE
                       CHOUVFYX
                       CHOUNJEU
                       CHOUUAUD
                       CHOUSPEC
                       CHOUDFLT
                       CHOUUNDF
                       CHOUERR
                       CHOUTRST
                       CHOUSEST
                       CHOUSURO
                       CHOURMT
                       CHOUPRVL
                       CHOUSECL
                       CHOUEXND
                       CHOUSUSR
                       CHOUSNOD
                       CHOUSGRP
                       CHOUSPOE
                       CHOUSPCL
                       CHOUTUSR
                       CHOUTGRP
                       CHOUTDFT
```

```
                    CHOUTSEC
                    CHOAPPC
                    CHOAUDIT
                    CHOORUID
                    CHOOEUID
                    CHOOSUID
                    CHOORGID
                    CHOOEGID
                    CHOOSGID
                    CHOPATHN
                    CHOFILID
                    CHOFOUID
                    CHOFOGID
                    CHOUID
                    CHOGID
                    CHOFILPL
                    CHOFILSP
                    CHOINODE
                    CHOSCID
                    CHODCELK
                    CHOAUTYP
                )
  %END;
%IF &REQ = EXTRACT %THEN
  %DO;
    %PUT Including datadefinition for CHOWN extension;
    WHEN('CHOWN') DO;
      INPUT %SMF8ØHDR(REQ=EXTRACT)
            CHOCLASS $       282-289
            CHOUSERN $       291-31Ø
            CHOUTKNE $       312-315
            CHOUPRE  $       317-32Ø
            CHOUVFYX $       322-325
            CHOUNJEU $       327-33Ø
            CHOUUAUD $       332-335
            CHOUSPEC $       337-34Ø
            CHOUDFLT $       342-345
            CHOUUNDF $       347-35Ø
            CHOUERR  $       352-355
            CHOUTRST $       357-36Ø
            CHOUSEST $       362-369
            CHOUSURO $       371-374
            CHOURMT  $       376-379
            CHOUPRVL $       381-384
            CHOUSECL $       386-393
            CHOUEXND $       395-4Ø2
            CHOUSUSR $       4Ø4-411
            CHOUSNOD $       413-42Ø
            CHOUSGRP $       422-429
            CHOUSPOE $       431-438
```

```
                CHOUSPCL $       44Ø-447
                CHOUTUSR $       449-456
                CHOUTGRP $       458-465
                CHOUTDFT $       467-47Ø
                CHOUTSEC $       472-475
                CHOAPPC  $       477-492
                CHOAUDIT $       494-5Ø4
                CHOORUID         5Ø6-515
                CHOOEUID         517-526
                CHOOSUID         528-537
                CHOORGID         539-548
                CHOOEGID         55Ø-559
                CHOOSGID         561-57Ø
                CHOPATHN $       572-771
                CHOFILID $      1596-1627
                CHOFOUID        1629-1638
                CHOFOGID        164Ø-1649
                CHOUID          1651-166Ø
                CHOGID          1662-1671
                CHOFILPL $      1673-168Ø
                CHOFILSP $      1682-1689
                CHOINODE        1691-17ØØ
                CHOSCID         17Ø2-1711
                CHODCELK $      1713-1728
                CHOAUTYP $      173Ø-1742
                 ;
LABEL CHOCLASS = 'Class name'
      CHOUSERN = 'User name'
      CHOUTKNE = 'Utoken encr.?'
      CHOUPRE  = 'Pre-1.9?'
      CHOUVFYX = 'VERIFYX propagation?'
      CHOUNJEU = 'Undefined NJE user?'
      CHOUUAUD = 'UAUDIT?'
      CHOUSPEC = 'RACF special?'
      CHOUDFLT = 'Default token?'
```

Editor's note: this article will be continued in the next issue.

An enhanced LISTUSER command – revisited

THE PROBLEM

An enhanced LISTUSER command was published in *RACF Update*, Issue 10, November 1997, and Issue 11, February 1998. We have encountered a problem that, if the person issuing the LISTUSER does not have read access to *any one* of the segments (TSO, DFP, NETVIEW, CICS, OPERPARM), the routine goes into an infinite loop. This is because, instead of finding 'dfp information' or 'no dfp information', for example, as a segment delimiter, it neglects to check for 'IRR52021I You are not authorized to view DFP segments.', and keeps going.

THE FIX

The problem can be solved as follows:

- Everywhere you see an 'if' statement for 'npxxx information', add an 'if' for the 'not authorized' message.
- Everywhere you see a 'while (not) =' for 'no xx information' add an '& (not) =' for the IRR520211 message.

A couple more hints:

- When the routine does find a TSO segment, it looks for the USERDATA field as a delimiter (and therefore loop). There may not be such a field in a given TSO segment. The fix for this is to check for the 'The-End=Thats all folks' delimiter.
- The scrolling of the group connections seemed to get off by one extra line for each group. My fix for this was to eliminate the last 'Call Write_Table_Record' statement in the table build section. This can be found three statements before the 'ADDRESS "ISPEXEC" "BTOP RACFUSER"' statement.

Larry Huitt
State of Illinois (USA)

Quick RACF validate

A systems programmer can often be denied access to RACF commands, yet still need to check the RACF environment. This can be a nuisance if you are involved in setting up multi-LPAR environments and wish to validate the consistency of RACF across those LPARs. I developed the following REXX to provide a basic check for just this situation. It requires no RACF authority and picks up all the information it requires by using the REXX storage function to retrieve information from the RCVT control block (RACF's vector table). Place this REXX in an accessible library in your SYSPROC or SYSEXEC concatenation, with a suitable name, and issue it from your command line. The information returned is displayed in standard TSO line mode, being simply a quick check, and therefore has not had any ISPF panels wrapped around it. An example display is shown in Figure 1.

```
This RACF is version 2 release Ø2 modification Ø
The following information may be of interest
============================================
> The RACF dataset name is SYS1.RACF
> Number of password generations to maintain 18
> Revoke user after 3 attempts
> Minimum password length 6
> Maximum password length 8
> Password rule           LLLLLLLL
> Where L=Alphanumeric, A=Alphabetic, N=Numeric, V=Vowel,
>       C=Consonant, and W=No Vowels
> generic profile checking in effect for dataset class
> generic command processing in effect for dataset class
> Audit group class
> Audit user class
> Audit dataset class
> Audit dasdvol class
> Audit tapevol class
> Audit terminal class
> Fastpath for dataset class
***
```

Figure 1: Example of display

```
/* REXX */
/* */
/* locate RCVT */
/* */
CVTRAC=D2X(C2D(STORAGE(10,4))+992)
RCVT=D2X(C2D(STORAGE(CVTRAC,4)))
RACDSN=STORAGE(D2X(X2D(RCVT)+56),44)
RACSTAT=STORAGE(D2X(X2D(RCVT)+150),1)
RACAUD=STORAGE(D2X(X2D(RCVT)+151),1)
PASSGEN=STORAGE(D2X(X2D(RCVT)+240),1)
PASSMIN=STORAGE(D2X(X2D(RCVT)+244),1)
PASSMAX=STORAGE(D2X(X2D(RCVT)+245),1)
PASSTYP=STORAGE(D2X(X2D(RCVT)+246),8)
PASSRVK=STORAGE(D2X(X2D(RCVT)+241),1)
RCVTFLG1=STORAGE(D2X(X2D(RCVT)+392),1)
RCVTFLG2=STORAGE(D2X(X2D(RCVT)+393),1)
RCVTVRM=STORAGE(D2X(X2D(RCVT)+616),4)
/* */
/* State the RACF version release and modification levels */
/* */
say 'This RACF is version' left(rcvtvrm,1),
'release' substr(rcvtvrm,2,2) 'modification' right(rcvtvrm,1)
/* */
/* State the Number of password generations being kept */
/* */
say ''
say ' The following information may be of interest'
say ' ============================================='
/* */
/* State the RACF dataset name in use */
/* */
say '> The RACF dataset name is' racdsn
say ''
say '> Number of password generations to maintain' c2d(passgen)
/* */
/* State how many failures a user can have before being revoked */
/* */
say '> Revoke user after' c2d(passrvk) 'attempts'
/* */
/* State max and minimum password lengths */
/* */
say '> Minimum password length' c2d(passmin)
say '> Maximum password length' c2d(passmax)
say '> Password rule          ' passtyp
say '> Where L=Alphanumeric, A=Alphabetic, N=numeric, V=vowel,'
say '>       C=Consonant, and W=No Vowels'
say ''
/* */
/* Now analyse the status flag */
/* */
if BITAND('80'X,racstat) \= '00'X,
THEN say '> tape volume protection in effect'
```

```
if BITAND('40'X,racstat) \= '00'X,
THEN say '> dasd volume protection in effect'
if BITAND('20'X,racstat) \= '00'X,
THEN say '> generic profile checking in effect for dataset class'
if BITAND('10'X,racstat) \= '00'X,
THEN say '> generic command processing in effect for dataset class'
if BITAND('08'X,racstat) \= '00'X,
THEN say '> Input dataset name will be used for logging and messages'
if BITAND('04'X,racstat) \= '00'X,
THEN say '> JES-XBMALLRACF in effect'
if BITAND('02'X,racstat) \= '00'X,
THEN say '> JES-EARLYVERIFY in effect'
if BITAND('01'X,racstat) \= '00'X,
THEN say '> JES-BATCHALLRACF in effect'
/* */
/* Now analyse the audit status */
/* */
if BITAND('40'X,racaud) \= '00'X,
THEN say '> Audit group class'
if BITAND('20'X,racaud) \= '00'X,
THEN say '> Audit user class'
if BITAND('10'X,racaud) \= '00'X,
THEN say '> Audit dataset class'
if BITAND('08'X,racaud) \= '00'X,
THEN say '> Audit dasdvol class'
if BITAND('04'X,racaud) \= '00'X,
THEN say '> Audit tapevol class'
if BITAND('02'X,racaud) \= '00'X,
THEN say '> Audit terminal class'
if BITAND('01'X,racaud) \= '00'X,
THEN say '> Audit operation attribute'
/* */
/* Now analyse the flag1 */
/* */
if BITAND('80'X,rcvtflg1) \= '00'X,
THEN say '> Fastpath for dataset class'
if BITAND('40'X,rcvtflg1) \= '00'X,
THEN say '> Tape dataset protection in effect'
/* */
/* Now analyse the flag2 */
/* */
if BITAND('80'X,rcvtflg2) \= '00'X,
THEN say '> Protect all in effect'
if BITAND('40'X,rcvtflg2) \= '00'X,
THEN say '> Protect all warning in effect'
if BITAND('20'X,rcvtflg2) \= '00'X,
THEN say '> Protect all failure in effect'
if BITAND('10'X,rcvtflg2) \= '00'X,
THEN say '> Erase on scratch in effect'
```

C A Jacques
Systems Programmer (UK)

Open Software Technologies has announced Version 2.1 of REXX to RACF (RTOR), its development tool kit for building corporate RACF and OS/390 security server applications through a set of APIs. RTOR also has RACF command and display extensions for use under TSO ISPF.

RACF applications built with RTOR, and the command/display extensions, are fully protected through multiple authority scoping capabilities. This allows the security administrator to control distribution of the applications and/or commands.

RTOR can be used with REXXTOOLS/MVS from Open Software Technologies to store RACF data in VSAM, QSAM, and BPAM files, or DB2 tablespaces.

For further information contact:
Open Software Technologies, 1230 Douglas Avenue, 300 Longwood, FL 32779, USA.
Tel: (407) 788 7173.
URL: http://www.open-softech.com.

* * *

Schumann are to integrate their Security Administration Manager (SAM), which includes software support for RACF, with the AccessMaster component of Bull's OpenMaster integrated management framework.

SAM provides a single point of administration and control across different platforms and their respective security systems. It supports role-based access control procedures, automated and distributed security administration, and the integration of human resource, single sign-on, and system management solutions.

For further information contact:
Schumann Security Software, 312 Marshall Avenue, Suite 400, Laurel, MD 20707, USA.
Tel: (301) 483 8807.
URL: http://www.schumannsoftware.com.

* * *

BETA Systems has announced BETA 88 PCF, a PC tool for RACF administration. BETA 88 PCF provides PC graphic components and drag-and-drop utilities to make tasks such as copying user profiles more visible and straightforward. The user interface gives a clear picture of the RACF security structure to enable decentralized security personnel to understand the security relationship under their supervision. These administrators can make changes to the RACF database via a TCP/IP link with BETA 88 on the host. RACF commands are executed in the host environment.

For further information contact:
BETA Systems Software, One Securities Center, 3490 Piedmont Road, Suite 1100, Atlanta, GA 30305, USA.
Tel: (404) 812 1556.
BETA Systems Software, Highlands House, Basingstoke Road, Spencers Wood, Reading, RG7 1NT, UK.
Tel: (01734) 885175.
URL: http://www.betasystems.com.

* * *