RRE™ - RACF 'RULES' ENFORCER

Compliance and beyond:

"In many cases, organizations and their company officers found to be non-compliant may be subject to fines or legal action, in addition to facing exposure to risks associated with internal data breaches. "

In general, compliance means conforming to a specification or policy, standard or law that has been clearly defined.
Corporate scandals and breakdowns such as the Enron case in 2001 have highlighted the need for stronger compliance regulations for publicly listed companies. The most significant regulation in this context is the Sarbanes-Oxley Act developed by two U.S. congressmen, Senator Paul Sarbanes and Representative Michael Oxley in 2002 which defined significant tighter personal responsibility of corporate top management for the accuracy of reported financial statements.
Compliance in the USA generally means compliance with laws and regulations. These laws can have criminal or civil penalties or can be regulations. The definition of what constitutes an effective compliance plan has been elusive. Most authors, however, continue to cite the guidance provided by the United States Sentencing Commission in Chapter 8 of the Federal Sentencing Guidelines.
On October 12 2006, the U.S. Small Business Administration re-launched Business.gov which provides a single point of access to government services and information that help businesses comply with government regulations.
There are a number of other regulations such as GLBA, FISMA, and HIPAA. In some cases other compliance frameworks (such as COBIT) or standards (NIST) inform on how to comply with the regulations

The **Chief Compliance Officer** (CCO) of a company is the officer primarily responsible for overseeing and managing compliance issues within an organization. Generally, a CCO is in charge of overseeing and managing compliance issues within an organization, ensuring, for example, that a company is complying with regulatory requirements, and that the company and its employees are complying with internal policies and procedures. The CCO typically reports to the Chief Executive Officer. The role has long existed at companies that operate in heavily regulated industries such as financial services and healthcare.

Purpose:

- To verify all RACF profiles against a HR/CD/ID system and vice versa.
- To verify all RACF profiles against a set of user defined 'rules'.
- To enforce naming conventions in a RACF environment without having to have any exits.
- To simplify and automate future audits.
- To reduce the immense costs of any internal or external RACF audits.
- To keep HR/CD and RACF information in sync based on installation standards.
- To have a better control over all RACF profiles.
- To be able to manage multiple clients.
- To verify SETROPTS settings.
- To verify IKJTSOXX settings (AUTHCMD, AUTHPGM, AUTHTSF, PLATPGM, PLATCMD, NOTBKGND)
- To verify PPT settings (SCHED=)
- To verify subsystems (SSN)
- To verify LINKLIST settings and its RACF protection
- To verify APFLIST settings and its RACF protection
- To verify LPALIST settings and its RACF protection
- To verify CATALOG and its RACF protection
- To verify SMF datasets and its RACF protection
- To verify user datasets and its RACF protection
- To verify RACF profile owner ship
- To verify access lists, including conditional access lists
- To verify OMVS(HFS)
- To verify business-, application- and systems owner

Most RACF installations do no longer know why certain user-Ids are connected to various RACF Group-Ids. Even when installations utilize a corporate directory (ID or CD or HR) it never matches the RACF environment 100%. Ownership of profiles is not up-to-date either.

Especially large corporations with many decentralized RACF administrators face the immense problem to enforce standards. Manually controlling such RACF environments is almost impossible. Home-grown tools are in many cases no solution either to the well known problem.

This batch facility helps every RACF installation to verify corporate directories versus RACF. It lists all inconsistencies and generates the necessary RACF commands to alter/delete RACF profile information.

RRE consists of two parts:

- CD/ID/HR verification against RACF and vice versa
- Rules checking for RACF group-, user- (incl. connects), dataset- and general resource profiles

DEB$SW1H - CD/HR vs RACF verification

Purpose:

- Verify the HR/CD (corporate directory) against RACF and vice versa.

Note: It is the responsibility of each user to verify any generated RACF commands before executing them e.g. to alter or delete any user-Ids.

JCL required to run DEB$SW1H

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1H) to create the verification reports:

```
//RREVERIF   EXEC PGM=DEB$SW1H
//STEPLIB    DD   DISP=SHR,DSN=RA2002.LINKLIB
//*
//* INPUT FILES
//*
//IRRI0200   DD   DISP=SHR,DSN=YOUR.IRRDBU.IRRI0200.VB(0)
//HRSI0200   DD   DISP=SHR,DSN=YOUR.MYCORP.HRS
//*
//* HRS RELATED INFORMATION (ALL HR IDS AND THEIR MISSING IDS IN RACF)
//*
//VERHRLST   DD SYSOUT=*   * HR HRSI0200 USERIDS LISTED "ASIS"
//VERHRMIS   DD SYSOUT=*   * HR USERIDS NOT FOUND IN RACF
//*
//* RACF RELATED INFORMATION (ALL RACF UIDS AND THEIR MISSING HR UIDS)
//*
//VERPRINT   DD SYSOUT=*   * PRINT +VERIFY CONTROL STATEMENTS
//VERRALST   DD SYSOUT=*   * RACF IRRI0200 USERIDS LISTED "ASIS"
//VERRAMIS   DD SYSOUT=*   * RACF LIST ALL MISSING USERIDS IN "HR"
//VERRANEV   DD SYSOUT=*   * RACF LIST ALL USERIDS NEVER USER "ASIS"
//VERRAREV   DD SYSOUT=*   * RACF LIST ALL REVOKED USERIDS
//VERRAOPR   DD SYSOUT=*   * RACF LIST ALL OPER/SPECIAL USERIDS  "ASIS"
//VERRAPRO   DD SYSOUT=*   * RACF LIST ALL PROTECTED USERIDS "ASIS"
//VERRAUAU   DD SYSOUT=*   * RACF LIST ALL UAUDIT USERIDS "ASIS"
//VERRCDEL   DD SYSOUT=*   * RACF DELETE ALL USERIDS NOT FOUND IN "HR"
//VERRCREV   DD SYSOUT=*   * RACF REVOKE ALL USERIDS NOT FOUND IN "HR"
//VERRCALT   DD SYSOUT=*   * RACF ALU    ALL USERIDS NOT FOUND IN "HR"
//VERINPUT   DD *          * FILTER CRITERIA FOR "HR" AND "RACF"
*
*OPTIONS HEADING=YES,PROTECTED=YES,REVOKE=YES,
*SPECIAL=YES,OPERATIONS=YES
+OPTIONS HEADING=YES
+VERIFY_INCLUDE USERID=@*,DFLTGRP=*,OWNER=STD*
*VERIFY_EXCLUDE USERID=*,DFLTGRP=*,OWNER=RACF*
```

How to build your own //HRSI0200 file?

RRE does not know any of your HR/CD/ID systems as they may not reside on the IBM Host. You can build via REXX and LDAP searches the //HRSI0200 input file.

The //HRSI0200 file must have the same record format as the IRRDBU00 from IBM: RECFM=VB, LRECL=4096.
The record layout for the first two fields (record type and user-ID) has the same as the IBM IRRDBU00 user record type 0200.

Extract from your HR(human resources system)/CD(corporate directory)/ID(identity management) the user-Ids, which must have a RACF user-ID. Use REXX/LDAP or FTP the data to the host and modify the Host file to have the following format:

Pos. 1 – 4	record type	0200	Fix value
Pos. 6 – 13	User-ID	e.g. IBMUSER	
Pos. 15 – 22	Status	ENABLED or DISABLED	DISABLED=REVOKED inactive ENABLED=active

REXX/LDAP sample on how to build your own //HRSI0200 file?

You can build via REXX and LDAP searches the //HRSI0200 input file
e.g. use an LDAP search to obtain your data:

REXX LDAP sample:

```
/* REXX */

HOST = 'XXX.CH.SWISSCOM.COM'
PORTID = 389

LDAP_O = "CN=RACF,CN=TARGETSYSTEMS,CN=INTRANET",
         "CN=* OBJECTCLASS=* DXRSTATE DXRTSSTATE"
/*                                                    */
/* -D BINDDN     BIND DN                              */
/* -W PASSWD     BIND PASSWD                          */
/* -S SCOPE      ONE OF BASE, ONE, OR SUB (SEARCH SCOPE)   */
/*                                                    */
'GLDSRCH / -H 'HOST '-P 'PORTID' -L 120 -S SUB',
 '-D CN=XRZP001,CN=USERS,CN=INTRANET -W [PASSWORD] -B 'LDAP_O' >DD:HRSI0200'
IF RC /= 0 THEN DO
   SAY 'GLDSRCH ENDED WITH RETURN CODE = 'RC
END
EXIT RC
```

Filter Control Statements (//VERINPUT DD)

HR/CD verification against RACF and vice versa

Following control statements can be utilized to obtain the necessary HR versus RACF verification reports:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	HEADING=YES or NO	Print headings (title lines)	YES
		PROTECTED=YES or NO or blank. The keyword is not required.	Select only "RACF" user-Ids from //IRRI0200 DD DSN= marked as protected. This keyword is ignored by the selection process for //HRSI0200 records.	N/A
		REVOKE=YES or NO or blank. The keyword is not required.	Select only "RACF" user-Ids from //IRRI0200 DD DSN= marked as revoked. This keyword is ignored by the selection process for //HRSI0200 records.	N/A
		SPECIAL=YES or NO or blank. The keyword is not required.	Select only "RACF" user-Ids from //IRRI0200 DD DSN= marked as special This keyword is ignored by the selection process for //HRSI0200 records.	N/A
		OPERATIONS=YES or NO or blank. The keyword is not required.	Select only "RACF" user-Ids from //IRRI0200 DD DSN= marked as operations This keyword is ignored by the selection process for //HRSI0200 records.	N/A
	+VERIFY_INCLUDE Note: you can define as many +VERIFY_ statements as required. Make sure the region size is set to e.g. REGION=0M	USERID=	Select a user-ID. Generic Ids are supported incl. The '?' as substitution character. Only the user-ID will be compared against the //HRSI0200 input file.	Blanks=all
		DFLTGRP=	Select a default group-ID. Generic Ids are supported incl. The '?' as substitution character.	Blanks=all
		OWNER=	Select a default owner-ID. Generic Ids are supported incl. The '?' as substitution character.	Blanks=all
	+VERIFY_EXCLUDE	Note: the same rules apply like for +VERIFY_INCLUDE verb.		
	Note:			

	1. All records matching a "+VERIFY_" will be included or excluded. Input to the verification process are //HRSI0200 and //IRRI0200 (RACF offloaded file in IBM's IRRDBU00 format). Only record type 0200 will be processed. The include process will be performed first. An exclude of USERID=* will be ignored for the all //HRSI0200 records. 2. A compare will be done with all keywords except for the //HRSI0200 file. This file must have the same format as the IRRI0200 (IRRDBU00), whereby the tool checks only for record type 0200 at position 1-4 and at position 6-13 for the user-id. 3. The file //HRSI0200 must be build by the customer due to the fact that each customer has his own HR or CD system in place. Currently we are checking only the first 13 positions (record type and user-ID).

Sample:

```
//VERINPUT   DD *
*
+OPTIONS HEADING=YES
+VERIFY_INCLUDE USERID=@*,DFLTGRP=*,OWNER=MIX*
+VERIFY_INCLUDE USERID=$*,DFLTGRP=*,OWNER=MAX*
*
+VERIFY_EXCLUDE USERID=*,DFLTGRP=*,OWNER=RACF*
+VERIFY_EXCLUDE USERID=*,DFLTGRP=HKROC,OWNER=TEST*
```

DDNAMES related to the HR/CD and RACF verification process

DDNAME	Description
VERINPUT	Input file - Control statments
VERPRINT	Print file – lists all //VERINPUT control statements. If an error occurred please review this output.
VERHRLST	Print file – lists unfiltered all //HRSI0200 records "AS IS".
VERHRMIS	Print file – lists all user-Ids from the //HRSI0200 file, which could not be found in RACF. This means you have defined user-Ids in your HR or CD, which do simply not exist in RACF or your +VERIFY_ verbs have excluded these IDS.
VERRALST	Print file – lists unfiltered all //IRRI0200 records "AS IS".
VERRAMIS	Print file – lists all user-Ids from the //IRRI0200 file, which could not be found in HR/CD. This means you have defined user-Ids in your RACF, which do simply not exist in the HR/CD or your +VERIFY_ verbs have excluded these IDS.
VERRANEV	Print file – lists unfiltered all RACF user-Ids, which 'never' logged on (= never used).
VERRAREV	Print file – lists unfiltered all RACF user-Ids, which have the status 'revoked'.
VERRAOPR	Print file – lists unfiltered all RACF user-Ids, which have the attribute 'operations and/or special'.
VERRAPRO	Print file – lists unfiltered all RACF user-Ids, which have the attribute 'protected'.
VERRAUAU	Print file – lists unfiltered all RACF user-Ids, which have the attribute 'uaudit'.
VERRCDEL	RACF command file (DCB=(RECFM=FB,LRECL=80)) – contains RACF delete user-ID commands for user-Ids not found in //HRSI0200. It is up to each installation to decide on what they want to do with user-Ids not found in the HR/CD system.
VERRCREV	RACF command file (DCB=(RECFM=FB,LRECL=80)) – contains RACF ALTUSER REVOKE user-ID commands for user-Ids not found in //HRSI0200. It is up to each installation to decide on what they want to do with user-Ids not found in the HR/CD system.
VERRCALT	RACF command file (DCB=(RECFM=FB,LRECL=80)) – contains RACF ALTUSER OWNER(new_ID) DFLTGRP(new_ID) REVOKE user-ID commands for user-Ids not found in //HRSI0200. It is up to each installation to decide on what they want to do with user-Ids not found in the HR/CD system. The user must modify the generated control statements accordingly.

Output Samples:

//VERHRLST lists all HR/CD entries 'as is':

```
******************************** TOP OF DATA ********************************************************************************
DEB$SW15-10 HR USER-IDS ENTRIES AS IS (ALL)             ALS(C) V3R6M0 07/03/05 12.33  RACF VERS 2608               PAGE:         1
                                                                                                                     DATE:2005-07-06
            JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                            TIME:   8:10:01
USERID   INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------------------------------
VOGEL
TRXUMO

DEB$SW15-10 HR USER-IDS ENTRIES AS IS (ALL)             ALS(C) V3R6M0 07/03/05 12.33  RACF VERS 2608               PAGE:       213
                                                                                                                     DATE:2005-07-06
            JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                            TIME:   8:10:01
USERID   INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------------------------------


===> TOTAL NUMBER OF USER-IDS READ     :         10.553
```

//VERHRMIS lists all HR/CD entries missing in RACF based on filter criteria's:

```
******************************** TOP OF DATA ********************************************************************************
DEB$SW17-10 HR USER-IDS MISSING IN THE "RACF" SYSTEM(S)  ALS(C) V3R6M0 07/03/05 12.34  RACF VERS 2608               PAGE:         1
                                                                                                                     DATE:2005-07-06
            JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                            TIME:   8:14:17
USERID   INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------------------------------
ARM3SE
ART3SS

DEB$SW17-10 HR USER-IDS MISSING IN THE "RACF" SYSTEM(S)  ALS(C) V3R6M0 07/03/05 12.34  RACF VERS 2608               PAGE:         2
                                                                                                                     DATE:2005-07-06
            JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                            TIME:   8:14:17
USERID   INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------------------------------


===> TOTAL NUMBER OF USER-IDS VERIFIED :         10.552

===> TOTAL NUMBER OF USER-IDS MISSING  :             23
```

//VERPRINT lists all filter statements:

```
VERPRINT-10 CONTROL STATEMENTS (COMPARE HR:RACF AND RACF:HR)       ALS(C) V3R6M0 07/03/05 12.40   RACF VER:2608      PAGE:         1
                                                                                                                     DATE:2005-07-06
            JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                            TIME:   8:10:01

CONTROL CARD(S) READ VIA //VERINPUT                                    ERROR MESSAGE
-------------------------------------------------------------------------- -------------------------------------------------

*
*OPTIONS HEADING=YES,PROTECTED=YES,REVOKE=YES,
*SPECIAL=YES,OPERATIONS=YES
+OPTIONS HEADING=YES
+VERIFY_INCLUDE USERID=@*,DFLTGRP=*,OWNER=STD*
+VERIFY_EXCLUDE USERID=*,DFLTGRP=RACFTUID,OWNER=*
                                                                       >-- EXCLUDE OF "*" OR "**" FOR USERID=  WILL BE IGNORED.
                                                                        THIS RESTRICTION APPLIES ONLY TO "HR" DATA
```

//VERRALST lists all RACF user-IDS 'as is':

```
DEB$SW14-10 RACF IRRDBU00 TYPE 0200 USER RECORDS (ALL)   ALS(C) V3R6M0 07/03/05 12.33  RACF VERS 2608                PAGE:         1
                                                                                                                      DATE:2005-07-06
           JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                              TIME:   8:10:01
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
FIATFIAT  Intercept           2000-07-17 MERCURY  N N N Y N      RACFCICS 2000-07-17 14:17:46 JDBC-access
Etc.

DEB$SW14-10 RACF IRRDBU00 TYPE 0200 USER RECORDS (ALL)   ALS(C) V3R6M0 07/03/05 12.33  RACF VERS 2608                PAGE:       284
                                                                                                                      DATE:2005-07-06
           JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                              TIME:   8:10:01
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------


===> TOTAL NUMBER OF USER-IDS READ      :        14.138

===> TOTAL NUMBER OF USER-IDS PROTECTED :           820

===> TOTAL NUMBER OF USER-IDS SPECIAL   :            10

===> TOTAL NUMBER OF USER-IDS OPERATIONS:             3

===> TOTAL NUMBER OF USER-IDS REVOKED   :         1.377

===> TOTAL NUMBER OF USER-IDS NEVER USED:           931
```

//VERRAMIS lists all RACF user-IDS missing in HR/CD (HRSI0200) based on filter criteria's:

```
DEB$SW16-10 RACF USER-IDS MISSING IN THE "HR" SYSTEM(S)  ALS(C) V3R6M0 07/03/05 12.34  RACF VERS 2608                PAGE:         1
                                                                                                                      DATE:2005-07-06
           JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                              TIME:   8:14:16
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
AGR100   Mike Norton          2005-05-20 TETRAPAK N N N N N      TETRAPAK                    ? NO DESCRIPTION
                                                                                             USER NEVER LOGGED ON
MILE07   LEADS Bill          2005-05-24 TETRAPAK N N N N N      TETRAPAK                    ? NO DESCRIPTION

                                                                                             PROTECTED USER
DEB$SW16-10 RACF USER-IDS MISSING IN THE "HR" SYSTEM(S)  ALS(C) V3R6M0 07/03/05 12.34  RACF VERS 2608                PAGE:        43
                                                                                                                      DATE:2005-07-06
           JOBNAME :XRZP001S STEPNAME:RA2VERIF PROCNAME:                                                              TIME:   8:14:16
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INFORMATION (ERROR MESSAGES ETC.)
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------


===> TOTAL NUMBER OF USER-IDS VERIFIED  :        12.287

===> TOTAL NUMBER OF USER-IDS MISSING   :         1.754

===> TOTAL NUMBER OF USER-IDS PROTECTED :            20

===> TOTAL NUMBER OF USER-IDS SPECIAL   :            10

===> TOTAL NUMBER OF USER-IDS OPERATIONS:             0

===> TOTAL NUMBER OF USER-IDS REVOKED   :         1.166
```

Sample creating a PDF to Email it:

The XMITIP program is public domain to create a PDF and is not included on the product CD/TAPE:

```
//GETUSER   EXEC PGM=IKJEFT1B,DYNAMNBR=200
//SYSEXEC   DD   DISP=SHR,DSN=FERRARI.REXX.LIB
//SYSTSPRT  DD   SYSOUT=*
//HRSI0200  DD   DISP=(,PASS),DSN=&&TEMP,LRECL=80,RECFM=FB
//SYSTSIN   DD   *
%CDSEARCH
/*
```

```
//*
//* FILTER THE USERS (ONLY ENABLED ONES) AND OUTPUT '0200' RECORDS
//*
//FILTER    EXEC PGM=IKJEFT1B,COND=(0,LT),DYNAMNBR=200
//SYSEXEC   DD   DISP=SHR,DSN=FERRARI.REXX.LIB
//HRSI0200  DD   DISP=(,PASS),DSN=&&TEMP2,LRECL=4096,RECFM=VB
//INPUT     DD   DSN=&&TEMP,DISP=(OLD,DELETE,DELETE)
//SYSTSPRT  DD   SYSOUT=*
//SYSTSIN   DD   *
%CDFILTER
/*
```

```
//RA2VERIF  EXEC PGM=DEB$SW1H,COND=(0,LT)
//STEPLIB   DD   DISP=SHR,DSN=RA2002.LINKLIB
//*
//* INPUT FILES
//*
//IRRI0200  DD   DISP=SHR,DSN=RA2.IRRDBU.IRRI0200.VB(-0)
//HRSI0200  DD   DSN=&&TEMP2,DISP=(OLD,DELETE,DELETE)
//*
//* HRS RELATED INFORMATION (ALL HR IDS AND THEIR MISSING IDS IN RACF)
//*
//VERHRLST  DD   SYSOUT=*   * HR HRSI0200 USERIDS LISTED "ASIS"
//VERHRMIS  DD   DISP=(,PASS),DSN=&&VERHRMIS
//*
//* RACF RELATED INFORMATION (ALL RACF UIDS AND THEIR MISSING HR UIDS)
//*
//VERINPUT  DD DISP=SHR,DSN=RA2002.RULEDATA(R001VER)
//VERPRINT  DD   SYSOUT=*   * PRINT +VERIFY CONTROL STATEMENTS
//VERRALST  DD   SYSOUT=*   * RACF IRRI0200 USERIDS LISTED "ASIS"
//VERRAMIS  DD   DISP=(,PASS),DSN=&&VERRAMIS
//VERRANEV  DD   SYSOUT=*   * RACF LIST ALL USERIDS NEVER USER "ASIS"
//VERRAOPR  DD   SYSOUT=*   * RACF LIST ALL OPER/SPECIAL USERIDS "ASIS"
//VERRAPRO  DD   SYSOUT=*   * RACF LIST ALL PROTECTED USERIDS "ASIS"
//VERRAPRO  DD   SYSOUT=*   * RACF LIST ALL REVOKED USERIDS "ASIS"
//*
//* USER RACF COMMANDS BASED ON VERIFICATION PROCESSING
//* - AN INSTALLATION MUST DECIDE WHAT TO DO WITH USERIDS NOT FOUND
//*   IN THE "HR" (HRSI0200) FILE.
//* - EITHER YOU DELETE, REVOKE AND OR ALTER THE USERIDS
//* - YOU MIGHT AS WELL CHANGE THE +VERIFIY STATEMENTS TO EXCLUDE
//*   CERTAIN USERIDS
//*
//VERRCDEL  DD   SYSOUT=*   * RACF DELETE ALL USERIDS NOT FOUND IN "HR"
//VERRCREV  DD   SYSOUT=*   * RACF REVOKE ALL USERIDS NOT FOUND IN "HR"
//VERRCALT  DD   SYSOUT=*   * RACF ALU ALL USERIDS NOT FOUND IN "HR"
```

```
//* EMAIL ACCOUNTS THAT ARE IN CD (WITH RACF ROLE) BUT NOT IN RACF
//*
//EMAIL1    EXEC BATCHTMP
//TEMPDD    DD   DISP=(OLD,DELETE),DSN=&&VERHRMIS
//SYSIN     DD   *
 XMITIP MARCEL_SCHMIDT@SWISSCOM.COM +
   SUBJECT 'CD TO RACF VERIFICATION - MISSING ACCOUNTS IN RACF' +
   FROM MARCEL_SCHMIDT@SWISSCOM.COM +
   MSGDS 'FERRARI.XMITIP.LIB(IVPMSG)' +
   FILEDD TEMPDD +
   FORMAT PDF/DS:'FERRARI.XMITIP.LIB(CDCFG)'
/*
```

Config for PDF:

```
* TXT2PDF CONFIGURATION FILE CREATED ON 8 OCT 2002 06:51:28 BY %TXT2PDFI
CC YES
COMPRESS 9
ENCRYPT ST/FERRARI/CDTEAM/128/NE/NC
ORIENT  LANDSCAPE
PAPER A4/GREENBAR/HOLED
CONFIRM YES
OUTLINE RC/0/3/5



THIS E-MAIL, INCLUDING ATTACHMENTS, IS INTENDED FOR THE PERSON(S) OR
COMPANY NAMED AND MAY CONTAIN CONFIDENTIAL AND/OR LEGALLY PRIVILEGED
INFORMATION. UNAUTHORIZED DISCLOSURE, COPYING OR USE OF THIS INFORMATION
MAY BE UNLAWFUL AND IS PROHIBITED. IF YOU ARE NOT THE INTENDED RECIPIENT,
PLEASE DELETE THIS MESSAGE AND NOTIFY THE SENDER
```

DEB$SI10 - RACF password verification

Purpose:

- Verify/test the RACF password rules.
- Some installations have to prove to corporate Audit, that the implemented RACF password rules really do work. To simplify this process a special program has been developed allowing an installation to test the password rules.

Password validation (NEWPASSW=) can be performed without having to know the current password. However only authorized users can perform such a task. The following RACF profile(s) must be present:

RACF class	Resource Profile (UACC=NONE)	Comment
FACILITY	RA2SOX.DEB$SI13.userid	If the keyword PASSWORD= is missing on the +VERIFY statement the RACF User-Id selected will be resumed and the password set to a specific value. This has to be done to avoid that during the testing the User-ID does not get revoked. If you test password rules there may be dozens of combinations possible you might have to test. Please note that the User-ID you have used becomes unusable concerning the PASSWORD. You must assign a new PASSWORD via the RACF ALU command. We highly recommend to utilize a special test User-ID to perform the validation process. Note: When using PASSWORD=x,NEWPASSW=y fails after n-1 attempts, the User-ID gets automatically revoked by RACF.

Note:

Each failed attempt concerning the password validation will be listed.

```
09.22.12 JOB06972  ICH408I USER(E       ) GROUP(SYS     ) NAME(####################)
                     LOGON/JOB INITIATION - INVALID PASSWORD ENTERED AT TERMINAL
09.22.12 JOB06972  IRR013I  VERIFICATION FAILED. INVALID PASSWORD GIVEN.
09.22.12 JOB06972  ICH408I USER(C       ) GROUP(SYS1    ) NAME(####################)
                     LOGON/JOB INITIATION - REVOKED USER ACCESS ATTEMPT
09.22.12 JOB06972  ICH408I USER(XRZP001 ) GROUP(SYS     ) NAME(RAND                )
                     RA2002.DEB$SI13.C CL(FACILITY)
                     INSUFFICIENT ACCESS AUTHORITY
                     FROM RA2002.* (G)
                     ACCESS INTENT(READ   )  ACCESS ALLOWED(NONE   )
09.22.12 JOB06972  ICH408I USER(F       ) GROUP(        ) NAME(???                 )
                     LOGON/JOB INITIATION - USER AT TERMINAL          NOT RACF-DEFINED
```

JCL required to run DEB$SI10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SI10) to create the verification reports:

```
//PASSWORD   EXEC PGM=DEB$SI10
//STEPLIB    DD   DISP=SHR,DSN=RA2002.LINKLIB
//VERPRINT   DD SYSOUT=*   * PRINT CONTROL STATEMENTS
//VERINPUT   DD *
+OPTIONS HEADING=YES
*
* VERIFY FOLLOWING USERIDS WITH A GIVEN PASSWORD
*
+VERIFY USERID=E,NEWPASSW=SYS1
+VERIFY USERID=E,NEWPASSW=SYS2
+VERIFY USERID=E,NEWPASSW=EUGENE
+VERIFY USERID=B,PASSWORD=SYS1,NEWPASSW=SYS2,
 SUSERID=GAGA,SGROUPID=BANANA,
        JOBNAME=MYJOB,
        APPL=MYAPPL,
        LOGSTR=LOGSTRIN,
        PGMNAME=PGMNAME,
        POE=JES2,
        SGROUPID=SGRUPPE,
        SESSION=APPC,
        SNODE=NODESUB
+VERIFY USERID=B,PASSWORD=SYS2,NEWPASSW=SYS3
+VERIFY USERID=B,PASSWORD=SYS3,NEWPASSW=SYS4
+VERIFY USERID=B,PASSWORD=SYS4,NEWPASSW=SYS5
+VERIFY USERID=B,PASSWORD=SYS5,NEWPASSW=SYS6
+VERIFY USERID=B,PASSWORD=SYS6,NEWPASSW=SYS7
+VERIFY USERID=B,PASSWORD=SYS7,NEWPASSW=SYS8
+VERIFY USERID=B,PASSWORD=SYS8,NEWPASSW=SYS9
+VERIFY USERID=B,PASSWORD=SYS9,NEWPASSW=SYSA
+VERIFY USERID=B,PASSWORD=SYSA,NEWPASSW=SYSB
+VERIFY USERID=B,PASSWORD=SYSB,NEWPASSW=SYSC
+VERIFY USERID=B,PASSWORD=SYSC,NEWPASSW=SYSD
+VERIFY USERID=B,PASSWORD=SYSD,NEWPASSW=SYSE
+VERIFY USERID=B,PASSWORD=SYSE,NEWPASSW=SYS1
```

Filter Control Statements (//VERINPUT DD)

RACF password validation

Following control statements can be utilized to obtain the RACF verification reports:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	HEADING=YES or NO	Print headings (title lines)	YES
	+VERIFY Note: you can define as many +VERIFY statements as required. Make sure the region size is set to e.g. REGION=0M	USERID=	Specifies the user identification of the user who has entered the system. Userids which have the attribute SPECIAL, OPERATIONS, AUDIT or privileged will be ignored for security reasons. Userid=IBMUSER will be ignored as well.	N/A
		PASSWORD=	Specifies the currently defined password of the user who has entered the system.	Note 1
		NEWPASSW=	Specifies the password that is to replace the user's currently defined password. With the assignment of a new password all the relevant RACF password rules checking will take	optional

			place.	
		APPL=	Specifies the name of the application issuing the RACROUTE REQUEST=VERIFY to verify the user's authority to access the application.	optional
		GROUPID=	Specifies the group specified by the user who has entered the system.	optional
		JOBNAME=	Specifies the job name of a background job.	optional
		LOGSTR=	Specifies character data to be written to the system-management-facilities (SMF) data set together with any RACF audit information, if logged.	optional
		PGMNAME=	Specifies the address of the name of the user who has entered the system. This 20-byte area is passed to the RACINIT installation exit routine; it is not used by the RACROUTE REQUEST=VERIFY routine.	optional
		POE=	Specifies the address of the port of entry into the system. The address points to the name of the input device through which the user or job entered the system. For example, this could be the name of the input device through which the job was submitted or the terminal logged onto. The port of entry is an 8-character field that is left-justified and padded with blanks.	optional
		SGROUPID=	Specifies the address of an area that contains the group name of the user who submitted the unit of work. The group ID cannot exceed eight bytes.	optional
		SUSERID=	Specifies the address of an area that contains the user ID of the user who submitted the unit of work. The user ID cannot exceed eight bytes.	optional
		TERMID=	Specifies the address of the identifier for the terminal through which the user is accessing the system.	optional

	Note: 1. If no password is supplied, the RACF profile FACILITY RA2002.DEB$SI13.userid will be checked. A internal password will be assigned and the user-ID will be resumed as well. 2. If a password is supplied, then RACF will verify the supplied password. 3. The implemented RACF password validation process has been set-up in such a way, that no unauthorized user can manipulate the RACF DB. Make sure there is no RA2002 profile called RA2002.** defined e.g. with UACC(READ).

RACF Return codes

To perform the password validation, standard IBM functions will be invoked. In case of RACF errors the following return and reason codes should be observed to resolve any problems. The RACF errors will be listed in //VERUSERS as follows: xxyyzz. All return and reason codes are shown in hexadecimal. Also, please note that SAF return code is presented as SAF RC and RACF return code is presented as RACF RC in the following section.

SAF R15 xx	Description	RACF RC yy	Description	RACF RS zz	Description
04	Requested function could not be completed.	00	No security decision could be made.	00	RACF was not called to process the request.
		04	The user profile is not defined to RACF.		
		20	RACF is not active.		
		58	RJE or NJE operator FACILITY class profile not found.		
08	Requested function has failed.	04	The user profile is not defined to RACF.		
		08	The password is not authorized		
		0C	The password has expired		
		10	The new password is not valid.		* failing password rule
		14	The user is not defined to the group.		
		18	RACROUTE REQUEST=VERIFY was failed by the installation exit routine.		
		1C	The user's access has been revoked.		
		24	The user's access to the specified group has been revoked.		
		28	OIDCARD parameter is required but not supplied.		
		2C	OIDCARD parameter is not valid for specified user.		
		30	The user is not authorized to the port of entry in the TERMINAL, JESINPUT, or CONSOLE class.	00	Indicates the user is not authorized to the port of entry.
				04	Indicates the user is not authorized to access the system on this day, or at this time of day.
				08	Indicates the port of entry cannot be used on this day, or at this time of day.
		34	The user is not authorized to use the application.		

DDNAMES related to the RACF password validation process

DDNAME	Description
VERPRINT	Print file – lists all //VERINPUT control statements. If an error occurred please review this output.
VERUSERS	Print file – lists all processed RACF User-ids found via //VERINPUT
VERINPUT	Input file - Control statements

Output Samples:

//VERPRINT lists all +VERIFY entries 'as is':

```
VERPRINT-10 CONTROL STATEMENTS USED TO VERIFY PASSWORDS            ALS(C) V3R6M0 08/15/05 19.44   RACF VER:2608      PAGE:         1
                                                                                                                     DATE:2005-08-17
            JOBNAME :XRZP001C STEPNAME:PASSWORD PROCNAME:                                                            TIME:   9:22:09

CONTROL CARD(S) READ VIA //VERINPUT                                      ERROR MESSAGE
------------------------------------------------------------------------ --------------------------------------------------

+OPTIONS HEADING=YES
*
+VERIFY USERID=E,NEWPASSW=SYS1
+VERIFY USERID=E,NEWPASSW=SYS2
+VERIFY USERID=E,NEWPASSW=SYS3
+VERIFY USERID=E,NEWPASSW=SYS4
```

//VERUSERS lists verified entries:

```
DEB$SI13-10 PASSWORD VALIDATION INTERFACE               ALS(C) V3R6M0 08/17/05 09.22  RACF VERS 2608               PAGE:         1
                                                                                                                     DATE:2005-08-17
            JOBNAME :XRZP001C STEPNAME:PASSWORD PROCNAME:                                                            TIME:   9:22:09
USERID   GROUPID  JOBNAME  PASSWORD NEWPASSW APPL     POE      SESSION  S-USER   S-GROUP  S-NODE   TERMID   INFORMATION
-------- -------- -------- -------- -------- -------- -------- -------- -------- -------- -------- -------------------------
E
E                                  SYS1
E                                  SYS2
E                                  SYS3
E                                  SYS4
E                                  SYS5
E                                  SYS6
E                                  SYS7
E                                  SYS8
E                                  SYS9
E                                  SYS0
E                                  SYSa                                                          CREATE FAILED     081000
E                                  SYSb                                                          CREATE FAILED     081000
```

RACF rules Verification (RRE)

Purpose:

- Verify the RACF based on installation defined 'rules' without having to utilize any exits at all.
- An installation can specify rules for all RACF base segments (groups, users, connects, datasets and general resource profiles.
- The intention of this utility is to simplify audits and profile verification.
  - Verify attributes e.g. special, operations etc.
  - Verify class authorizations
  - Verify owners
  - Verify members e.g. global access list
  - Verify access lists

Especially if you have one or more RACF environments, which have been maintained by a number of people over the last 10-20 years, it is most difficult to find out "what is what" and if all the rules (if any) are properly used. RACF Exits are for most companies not a feasible option due to the ever-changing security environment.

To execute this batch program an 'off-loaded' RACF DB is required. To setup all the rules will take a considerable amount of time, especially in case no proper naming standards have been implemented.
RA/2 (search and tag facility under option 3.100, 200, 205, 400 and 500) can be utilized to generate the majority of the rules you may required, saving an installation a lot of time and money.

DEB$SW1G – RACF Group Verification (RRE)

Purpose:

- Verify RACF group profiles.

JCL required to run DEB$SW1G

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1G) to create the reports:

```
//RA2VERIF   EXEC PGM=DEB$SW1G
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* COMMANDS
//*
//COMMANDS   DD   DISP=SHR,DSN=RA2002.V?R?M?.COMMANDS
//*
//* INPUT FILES
//*
//IRRI0100   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0100.VB(0)
//IRRI0102   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0102.VB(0)
//*
//* OUTPUT FILES
//*
//GRPPRINT   DD SYSOUT=*   * PRINT RESNAME_RULE CONTROL STATEMENTS
//GRPC0100   DD SYSOUT=*   * RACF GROUPS - GENERATED COMMANDS
//GRPG0100   DD SYSOUT=*   * RACF GROUPS - MATCHING RULES
//GRPF0100   DD SYSOUT=*   * RACF GROUPS - FAILED RULES
//GRPX0100   DD SYSOUT=*   * RACF GROUPS - NO RULES APPLY
//GRPT0100   DD SYSOUT=*   * RACF GROUPS - SUMMARY
//*
//GRPC0102   DD SYSOUT=*   * RACF GROUPS - GENERATED COMMANDS
//GRPG0102   DD SYSOUT=*   * RACF GROUPS - MATCHING RULES
//GRPF0102   DD SYSOUT=*   * RACF GROUPS - FAILED RULES
//GRPX0102   DD SYSOUT=*   * RACF GROUPS - NO RULES APPLY
//GRPT0102   DD SYSOUT=*   * RACF GROUPS - SUMMARY
//GRPRULES   DD *          * RACF BASE USERID RULES
*
* +OPTIONS SPECIFIES THE DEFAULT VALUES TO BE ASSIGNED IF
*  NO RULE DOES MATCH
*
*
+OPTIONS     SET_OWNER=NEWOWNER,SET_SUPGROUP=TEST
+GROUP_RULE  NAME='SYS1   ',GROUPID=T*
+GRP_RULE    NAME='SYS1   ',GROUPID=T*
*
+GROUPID_RULE NAME='SYS1   ',GROUPID=T*,
             OWNER=TEST*,SUPGROUP=SYS1,
             SET_OWNER=T*USR,SET_SUPGROUP=SYS1SUP
*
+GROUPID_RULE NAME='Z GROUP',GROUPID=Z*,
             OWNER=TEST*,SUPGROUP=SYS1,
             SET_OWNER=Z*USR,SET_SUPGROUP=SYS1SUP
*
+GROUPID_RULE NAME='SYS1DATA',GROUPID=@*,
             OWNER=TEST*,SUPGROUP=SYS1,DATA=YES,
             SET_OWNER=SYS1DATA,SET_SUPGROUP=SYS1SUP
*
+GROUPID_RULE NAME='$$ DATA',GROUPID=$$*,
             OWNER=$$*,SUPGROUP=$$*,
             SET_OWNER=$$OWNER,SET_SUPGROUP=$$SUPG
*
+GROUPID_RULE NAME='$  DATA',GROUPID=$*,
             OWNER=$*,SUPGROUP=$*,
             SET_OWNER=$OWNER,SET_SUPGROUP=$SUPG
*
```

DDnames:

- //IRRIxxxx must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. These files are used as input to the program. In case the files are not split by record type then define on all //IRRIxxxx the offloaded RACF database flat file.
  - xxxx = IRRDBU00 record type.
- //COMMANDS must be RECFM=FB, LRECL=80, DSORG=PO. This file contains the product-supplied commands and the user defined commands. The first letter '$' is reserved for product supplied commands.
- //SORTxxxxx DDNAMES are required when the option VERIFY=YES is set.
- //SYSOUT is required by the SORT program
- //???Cxxxx must be RECFM=FB, LRECL=80, DSORG=PS.
  - ??? = USR, GRP, DSN, CON or RES
  - xxxx = IRRDBU00 record type

Note:

Control cards generated by this program must reside in separate flat files and not e.g. in one common PDS, otherwise you will encounter the following ABEND:
IEC143I 213-30,IFG0194D,MYJOBID,RA2VERIF,GRPC0100. This is due to the fact that multiple files are open at the same time to generate control cards.

Group-ID Rules (Filter) Control Statements (//GRPRULES DD *)

Following control statements can be utilized to perform the RACF group-ID verification:

DDname	Verbs	Keywords	Comment	Default
//GRPRULES	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	SET_OWNER=	Assign new default owner if all rules fail. The global variable name &SGOWNER can be used in the command member for the failing rule.	N/A
		SET_SUPGROUP=	Assign new superior group if all rules fail. The global variable name &SGSUPGRP can be used in the command member for the failing rule.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0100 base record) will be bypassed for further processing.	N/A
		Note: all generated RACF control statements must be reviewed prior executing them. This utility does not automatically update the RACF DB.		

+GR - Group basic data record (0100)

The Group Basic Data record defines the basic information that defines a group. There is one record per group.

	+GROUPID_RULE or +GROUP_RULE or +GRP_RULE or +GR	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a	N/A

	Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M		meaningful name.	
		GROUPID=	Specifies a RACF group-ID to be verified.	N/A
		OWNER=	Specifies a RACF Owner-ID to be verified.	N/A
		SUPGROUP=	Specifies a RACF superior-group-ID to be verified.	N/A
		DATA=YES or NO	Specifies that installation data must be present.	N/A
		UACC=	Specifies a RACF group-ID UACC to be verified.	N/A
		AUTHDATE=(yyyy-mm-dd,??) or AUTH_DATE=(yyyy-mm-dd,??)	AUTHDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. AUTHDATE=(2000,GE) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. AUTHDATE is the date when a profile was 'defined' to RACF.	N/A
		SET_OWNER=	Assign new default owner if this rule fails. The global variable name &SOWNER can be used in the command member for the failing rule.	N/A
		SET_SUPGROUP=	Assign new default superior group if this rule fails. The global variable name &SSUPGRP can be used in the command member for the failing rule.	N/A
		BYPASS_OWNER=(…, …)	Specifies RACF Owner-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_SUPGROUPID=(…, …)	Specifies RACF superior Group-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_UACC=(…, …)	Specifies RACF UACC(s) to be excluded from the validation process. Max. 8 ID's can be specified.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails. The output will be written to //GRPC0100.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0100 base record) will be bypassed for further processing.	N/A

	Note: • The key fields for rule checking support generic filtering. A key field can also contain the substitution character '?' e.g. IBM?A* • If all supplied rules fail, the program will create the necessary reports and RACF control cards to alter the profiles.			

+CR - Group members record (0102)

The Group Members record defines the relationship between a group and the members of the group. There is one record per group/member combination.

DDname	Verbs	Keywords	Comment	Default
continued	**+CONNECT_RULE or** +CR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M This rule applies to record type 0102 only.	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		GROUPID=	Specifies a RACF group-ID to be verified. //IRRI0102 DD file must be defined on the JCL. The global variable name &R12NAME can be used in the command member for the failing rule.	N/A
		USERID=(… ,…)	Specifies a RACF User-ID to be verified. The global variable name &R12USERN can be used in the command member for the failing rule. Up 128 User-IDs can be defined	N/A
		BYPASS_USERID=(…, …)	Specifies RACF user-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		AUTH=	Connect authorization e.g. JOIN, USE, CREATE The global variable name &R12UACS can be used in the command member for the failing rule.	N/A
		SET_AUTH=	Assign new connect authorization if this rule fails. The global variable name &SGAUTH can be used in the command member for the failing rule.	N/A
		BYPASS_AUTH=(…, …)	Specifies RACF authorization codes to be excluded from the validation process. Max. codes can be specified.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails. The output will be written to //GRPC0102.	N/A

Sample: Failing Group-IDs

```
DEB$SW50-10 RACF GROUP-IDS WHICH FAILED RULES CHECKING   ALS(C) V3R6M0 10/26/05 14.05  RACF VERS 2608              PAGE:        1
                                                                                                                 DATE:2005-10-27
           JOBNAME :XRZP001A STEPNAME:RA2RULES PROCNAME:                                                         TIME:   9:37:07
GROUPID  SUPGROUP AUTHDATE   OWNER    UACC     DATA (INSTALLATION DATA)                                  FAILING RULE NAME(S)
-------- -------- ---------- -------- -------- --------------------------------------------------------- ----------------------------

$$$$TEST SYS1     2005-11-09 IBMUSER  NONE     AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA '$  DATA'
$$DB2    SYS1     2005-04-10 SYS1     NONE     DB2      STC FUNC                                         '$  DATA'
$$FUNC   SYS1     2005-04-10 SYS1     NONE     GROUP FOR SYSTEM FUNCTIONS                                '$  DATA'
$$STC    SYS1     2005-04-10 SYS1     NONE     GROUP FOR STARTED TASKS                                   '$  DATA'
```

Command generation

For each failing rule it is possible to generate any kind of commands. Below are the variable names listed which can be used in a command member (DDname //COMMANDS). Each time a rule fails and the keyword COMMAND=member name has been specified, the member will be read from //COMMANDS PDS file and all variables replaced. The output will be written to //GRPC0100 and //GRPC0102. This facility works similar to the ISPF FTINCL function, however it does not support e.g.)SEL,)IM logic.

Variable names filled in by the IRRI0100 record:

The following variables can be used to generate commands related to group-Ids:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT	SAMPLE DATA
NAME	&R10NAME		$$WEBPUB
SUPGRP_ID	&R10SUPG		$$WEB
CREATE_DATE	&R10AUTHD	YYYY-MM-DD	
OWNER_ID	&R10OWNER		
UACC	&R10UACC		NONE
NOTERMUACC	&R10TUACC	Y OR N (?)	
INSTALL_DATA	&R10DATA		
MODEL	&R10MODEL		

Variable names filled in by the failing rule:

OPTIONS KEYWORD	OPTIONS VARIABLE	FORMAT
SET_SUPGROUP=NAME	&SGSUPGRP	MAX. 8 CHAR
SET_OWNER=NAME	&SGOWNER	MAX. 8 CHAR

RULE KEYWORD	RULE VARIABLE	FORMAT
SET_SUPGROUP=NAME	&SSUPGRP	MAX. 8 CHAR
SET_OWNER=NAME	&SOWNER	MAX. 8 CHAR

Variable names filled in by the IRRI0102 record:

The following variables can be used to generate commands related to group-Ids:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT	SAMPLE DATA
NAME	&R12NAME		SYS1
MEMBER_ID	&R12USERN		IBMUSER
AUTH	&R12UACS		JOIN

Variable names filled in by the failing rule:

RULE KEYWORD	RULE VARIABLE	FORMAT
SET_AUTH=NAME	&SGAUTH	MAX. 8 CHAR

Rules and command sample:

```
+CR           NAME='CONNECT 3',G=*,AUTH=CREATE,SET_AUTH=USE,
 USERID=(DFS,DFSCM),COMMAND=GRPR0002

IF ABOVE RULE FAILS THE FOLLOWING COMMAND WILL BE GENERATED
WITH THE CONNECT AUTHORIZATION 'USE' AS DEFINED IN THE RULE DEFINED ABOVE.


COMMAND MEMBER 'GRPR0002' IN //COMMANDS:
/* OLD CONNECT AUTH VALUE:            AUTH(&R12UACS)
 CONNECT  (&R12USERN) GROUP(&R12NAME) AUTH(&SGAUTH)
/*                                                    */
```

Sample: List all Group-connects as failed where the authority is 'NOT USE':

```
+CONNECT_RULE NAME='GROUP-CONNECT>USE',
              USERID=*,
              GROUPID=*,
              AUTH=USE
```

DEB$SW1U – RACF User Verification (RRE)

Purpose:

- Verify RACF user profiles:
  - Base records - (record type 0200)
  - CLAUTH (class authorization) - (record type 0200)
  - NETVIEW - (record type 0280-282)
  - Create a delta between a set of userids. This allows to check if they all have the same attributes. Record type 0200 – 02F0 are supported.

JCL required to run DEB$SW1U

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1U) to create the reports:

```
//RA2VERIF   EXEC PGM=DEB$SW1U
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* COMMANDS
//*
//COMMANDS   DD   DISP=SHR,DSN=RA2002.V?R?M?.COMMANDS
//*
//* INPUT FILES
//*
//IRRI0200   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0200.VB(0)
//IRRI0202   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0202.VB(0)
//IRRI0280   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0280.VB(0)
//IRRI0281   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0281.VB(0)
//IRRI0282   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0282.VB(0)
//*
//* OUTPUT FILES
//*
//USRPRINT   DD SYSOUT=*   * PRINT RESNAME_RULE CONTROL STATEMENTS
//USRC0200   DD SYSOUT=*   * RACF USERS - GENERATED COMMANDS
//USRG0200   DD SYSOUT=*   * RACF USERS - MATCHING RULES
//USRF0200   DD SYSOUT=*   * RACF USERS - FAILED RULES
//USRX0200   DD SYSOUT=*   * RACF USERS - NO RULES APPLY
//USRT0200   DD SYSOUT=*   * RACF USERS - SUMMARY
//*
//USRC0202   DD SYSOUT=*   * RACF USERS - GENERATED COMMANDS
//USRG0202   DD SYSOUT=*   * RACF USERS - MATCHING RULES
//USRF0202   DD SYSOUT=*   * RACF USERS - FAILED RULES
//USRX0202   DD SYSOUT=*   * RACF USERS - NO RULES APPLY
//USRT0202   DD SYSOUT=*   * RACF USERS - SUMMARY
//*
//USRC0280   DD SYSOUT=*   * RACF USERS - GENERATED COMMANDS
//USRG0280   DD SYSOUT=*   * RACF USERS - MATCHING RULES
//USRF0280   DD SYSOUT=*   * RACF USERS - FAILED RULES
//USRX0280   DD SYSOUT=*   * RACF USERS - NO RULES APPLY
//USRT0280   DD SYSOUT=*   * RACF USERS - SUMMARY
//*
//USRRULES   DD *          * RACF BASE USERID RULES
+OPTIONS     SET_OWNER=USRREV01,SET_DFLTGRP=USRREV01,
             SET_REVOKE=YES,SET_PROTECTED=YES
*
* RULES AT OUR CORPORATION
*
+UR NAME='P390        ',U=P390,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL),
 BYPASS_USERID=(IBM*)
+UR NAME='P390        ',U=P390,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL),
 BYPASS_USERID=(A,1,2,3,4,5,6,7)
+UR NAME='P390        ',U=P390,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL),
 BYPASS_USERID=(A,1,2,3,4,5,6,12345689)
+UR NAME='P390AUSERIDS',U=P390A,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL)
+UR NAME='P390BUSERIDS',U=P390B,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL)
+UR NAME='P390CUSERIDS',U=P390C,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL)
+UR NAME='P390DUSERIDS',U=P390D,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL)
+UR NAME='P390ZUSERIDS',U=P390Z,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL)
+UR NAME='PROT        ',U=*,PROTECTED=YES
+CLR NAME='CLAUTH1    ',U=*,CL=(DATASET,1,2,3,4)
+CLR NAME='CLAUTH2    ',U=*,CL=(TAPEVOL,A,B,C,D,E)
```

continued . . . (+DELTA U= and/or G=)

```
OPTIONAL JCL AND RULES FOR DELTA PROCESSING . . .


//IRRIDELT   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0200.VB(0)
//                . . . UP TO IRRI02F0

//DLTCXXXX   DD SYSOUT=*    * CONTROL STATEMENTS +DELTA
//DLTFXXXX   DD SYSOUT=*    * FAILED


//USRRULES   DD *
+DELTA NAME=OWNA,G=(FCTXXXXX)
+DELTA NAME=BASE,U=(MOKKEG,MOKSM1,MOKDME,MOKESH,MOKXCI,VOGT)
+DELTA NAME=STOR,U=(MOKMHO,MOKHEJ,MOKECM)
+DELTA NAME=ASYS,U=(MOKXUO,MOKSP1,JAMES)
```

DDnames:

- //IRRIxxxx must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. These files are used as input to the program. In case the files are not split by record type then define on all //IRRIxxxx the offloaded RACF database flat file.
  - xxxx = IRRDBU00 record type.
- //IRRIDELT must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. This file is used as input to the program utilizing the +DELTA function. In case the files are split by record type then define/concatenate all IRRIxxxx offloaded RACF database flat files. Only one single DDNAME is required.
- //COMMANDS must be RECFM=FB, LRECL=80, DSORG=PO. This file contains the product-supplied commands and the user defined commands. The first letter '$' is reserved for product supplied commands.
- //SORTxxxxx DDNAMES are required when the option VERIFY=YES is set.
- //SYSOUT is required by the SORT program
- //???Cxxxx must be RECFM=FB, LRECL=80, DSORG=PS.
  - ??? = USR, GRP, DSN, CON, RES or DLT(+DELTA)
  - xxxx = IRRDBU00 record type

Note:

Control cards generated by this program must reside in separate flat files and not e.g. in one common PDS, otherwise you will encounter the following ABEND:
IEC143I 213-30,IFG0194D,MYJOBID,RA2VERIF,DSNC0400. This is due to the fact that multiple files are open at the same time to generate control cards.

User-ID Rules (Filter) Control Statements (//USRRULES DD *)

Following control statements can be utilized to perform the RACF user-ID verification:

DDname	Verbs	Keywords	Comment	Default
//USRRULES	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	SET_OWNER=	Assign new default owner if all rules fail. Variable name &SUOWNER will be set.	N/A
		SET_DFLTGRP=	Assign new default group if all rules fail. Variable name &SUDFLTGRP will be set.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0200 base record) will be bypassed for further processing.	N/A

		Note: all generated RACF control statements must be reviewed prior executing them. This utility does not automatically update the RACF DB.		

+UR - User basic data record (0200)

The User Basic Data record defines the basic information about a user. There is one record per user.

	+USERID_RULE or +USER_RULE +USR_RULE +UR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type=0200	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		USERID=	Specifies a RACF user-ID to be verified.	N/A
		OWNER=	Specifies a RACF Owner-ID to be verified.	N/A
		DFLTGRP=	Specifies a RACF default-group-ID to be verified.	N/A
		DATA=YES or NO	Specifies that installation data must be present.	N/A
		PROTECTED=YES or NO	Specifies that the user-ID must be protected ('Y') or not.	N/A
		REVOKE=YES or NO	Specifies that the user-ID must be revoked ('Y') or not.	N/A
		SPECIAL=YES or NO	Specifies that the user-ID must have the special attribute ('Y') or not.	N/A
		OPERATIONS=YES or NO	Specifies that the user-ID must have the operations attribute ('Y') or not.	N/A
		AUDITOR=YES or NO	Specifies that the new user has full responsibility for auditing the use of system resources, and is able to control the logging of detected accesses to any RACF-protected resources during RACF authorization checking and accesses to the RACF database.	N/A
		UAUDIT==YES or NO	Specifies that RACF is to log all RACROUTE REQUEST=AUTH and RACROUTE REQUEST=FASTAUTH services that are eligible for logging, and all RACROUTE REQUEST=DEFINE services issued for the user, and all RACF commands.	N/A
		ATTRIBUTE=	Other user attributes (RSTD for users with RESTRICTED attribute).	N/A
		LJDATE=YES or NO	Specifies that a logon date must be present ('Y') or not.	N/A
		AUTHDATE=(yyyy-mm-dd,??) or AUTH_DATE=(yyyy-mm-dd,??)	AUTHDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. AUTHDATE=(2000,GE) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. AUTHDATE is the date when a profile was 'defined' to RACF.	N/A

		INITDATE=(yyyy-mm-dd,??) or INIT_DATE=(yyyy-mm-dd,??)	INITDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. INITDATE=(2000,LT) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. INITDATE is the date when a profile was last used e.g. LOGON (=JOBINIT). To find userids, which have NO logon date (never used) you can specify e.g. INITDATE=(1,LT).	N/A
		SET_OWNER=	Assign new default owner if this rule fails. Variable name &SOWNER will be set.	N/A
		SET_DFLTGRP=	Assign new default group if this rule fails. Variable name &SDFLTGRP will be set.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails..	N/A
		BYPASS_USERID=(…, …)	Specifies RACF user-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_OWNER=(…, …)	Specifies RACF Owner-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_DFLTGRP=(…, …) or BYPASS_GROUPID=(…, …)	Specifies RACF default-group-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A

		BYPASS_ATTR=(…, …)	Specifies RACF attributes for user-IDs to be excluded from the validation process. Valid items are: ADSP NOADSP GRPACC NOGRPACC AUDITOR NOAUDITOR OIDCARD NOOIDCARD SPECIAL NOSPECIAL OPERATIONS NOOPERATIONS TIMEZONE NOTIMEZONE PROTECTED PWREQ NOPROTECTED REVOKED NOREVOKED Max. 8 attributes can be specified. Generic names are NOT supported.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0200 base record) will be bypassed for further processing.	N/A

	Note: • The key fields for rule checking support generic filtering. A key field can also contain the substitution character '?' e.g. IBM?A* • If all supplied rules fail, the program will create the necessary reports and RACF control cards to alter the profiles. • Excluding IDs from rules checking: ▪ There may be a need that certain IDs based on their user-ID and/or attributes are bypassed for rules checking. E.g. Rule +UR N='P390',U=*,SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL) will not be tested against any user profile having the attribute NOSPECIAL. This for

	example allows an installation to identify which new RACF user-IDs with the attribute SPECIAL have been added. Remember: for each valid user-ID in this case you need to have a rule defined. Sample: monitor user-IDS which have been added with SPECIAL or OPERATIONS +UR NAME='SPECIAL USERS',U=*, SPECIAL=YES,BYPASS_ATTR=(NOSPECIAL), BYPASS_USERID=(MOK001, MOK003, MOK007) +UR NAME='OPERATIONS USERS',U=*, OPERATIONS=YES,BYPASS_ATTR=(NOOPERATIONS), BYPASS_USERID=($AMEXCO, TEC10101,TEC12010)

+CLR - User classes record (0202)

The User Classes record defines the classes in which the user can create profiles. There is one record per user/class combination. *Defines the classes in which users can create profiles.*

DDname	Verbs	Keywords	Comment	Default
//USRRULES	*	N/A	Comment line	N/A
	+CLAUTH_RULE or +CLR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M. This rule applies to record type 0202 only.	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		USERID=	Specifies a RACF user-ID to be verified.	N/A
		CLAUTH=	Specifies RACF CLAUTH name to be assitned ti variable name &SCLAUTH. Max. 128 ID's can be specified. Generic names are supported.	N/A
		SET_CLAUTH=	Specifies a RACF CLAUTH name to be assigned to variable &SCLAUTH.	N/A
		BYPASS_USERID=(…, …)	Specifies RACF user-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails..	N/A
		Note: If CLAUTH= validation is requested and //IRRI0500 DD is present, the program will check if such a RACF class does exist. To check which classes a user-ID is authorized to you can specify e.g. +CLR USERID=AMEX*,CLAUTH=(TAPEVOL,DASDVOL,DATASET,$RA2SUB). The DDname //IRRI0202 is required to process the CLAUTH records. Record type is '0202' created by the IBM IRRDBU00 program.		

+NR - User NETVIEW segment record (0280)

The User NETVIEW segment record defines the information required by NetView. There is only one record per user profile that contains a NETVIEW segment.

DDname	Verbs	Keywords	Comment	Default
//USRRULES	*	N/A	Comment line	N/A
	+NETVIEW_RULE or +NR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M. This rule applies to record type 0280-282 only.	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		USERID=	Specifies a RACF user-ID to be verified.	N/A
		IC=	specifies the command or command list (up to 255 characters) to be processed by NetView for this operator when this operator logs on to NetView. Generic names are supported.	N/A
		CONSNAME=	Specifies the default Master Console Station (MCS) console name used for this operator. This default console name is used when the operator does not specify a console name on the NetView GETCONID command. Console-name is a 1-8 character identifier whose validity is checked by MVS processing when the operator tries to use it. Generic names are supported.	N/A
		CTL=	Specifies whether a security check is performed for this NetView operator when they try to use a span or try to do a cross-domain logon. GENERAL specifies that checking is done as described for SPECIFIC, and, in addition, that the operator is allowed to access devices that are not part of any span. GLOBAL specifies that no checking is done. SPECIFIC specifies that the operator is allowed to control only devices that are in spans the operator started, and that a security check is to be performed through RACROUTE REQUEST=AUTH whenever this operator attempts to use a span. Also, any cross-domain logon must be to a domain listed in the operator's NETVIEW segment with the DOMAINS keyword.	N/A

			Generic names are supported.	
		MSGRECVR= YES or NO	Specifies whether this operator is to receive unsolicited messages that are not routed to a specific NetView operator.	N/A
		NGMFADMN= YES or NO	Specifies whether a NetView operator has administrator authority to the NetView Graphic Monitor Facility (NGMF).	N/A
		NGMFVSPN=	Reserved for future use by the NetView Graphic Monitor Facility	N/A
		OPCLASS= (record type 0281 only)	NetView scope classes for which the operator has authority. The OPCLASS values are only used if NetView is doing the checking itself, rather than using SAF and the NETCMDS class that RACF provides. If the OPCLASS operand is not specified, the operator is considered to have authority in scope classes. Class is a number from 0001 to 2040 that specifies a NetView scope class.	N/A
		DOMAINS= (record type 0282 only)	Specifies the identifiers of NetView programs in another NetView domain where this operator can start a cross-domain session. The NetView program identifiers are coded on the NCCFID definition statement for the other domains, and represent the name given to that NetView program on the APPL statement. Domain-name is a 5 character identifier. The characters can be alphabetic, numeric, or national.	N/A
		BYPASS_USERID=(…, …)	Specifies RACF user-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails..	N/A
		The DDname //IRRI0280-0282 is required to process the NETVIEW records. Record type '0280-282' is created by the IBM IRRDBU00 program.		

DELTA Rules (Filter) Control Statements (//USRRULES DD *)

Following control statements can be utilized to perform the RACF user-ID verification:

DDname	Verbs	Keywords	Comment	Default
//USRRULES	*	N/A	Comment line	N/A
	+DELTA	USERID=(…, …) or U=	Specifies RACF user-ID(s) to be included for the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		GROUPID=(…, …) or G=	Specifies RACF Group-ID(s) to be included for the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		Note 1: Only //IRRIDELT DD is used as INPUT. For each record type 02XX separate output files will be used to report on the differences. Only 'failed' fields will be reported. This allows a user to identify the differences between multiple user-ids. If you utilize G=, then all connected User-Ids pertaining to the selected group(s) will be compared. This may result in lots of output. The DELTA function may be very handy to check if a set of User-IDS belonging to the same ROLE/RULE have the same attributes. Note 2: Since this function allows a user to generate any RACF commands, the assistance of e.g. a systems programmer may be required to define the necessary commands with its variables, For each failed ID a command can be generated. The commands reside in //COMMANDS. For each record type there is one single member starting with $DLT????, where ???? represents the record type. It is the responsibility of a user to create the required (e.g. RACF) commands. The generated output will be written to //DELT???? And this for each failed record. If written to a flat file the DCB format must be: RECFM=FB,LRECL=80 and DISP=(MOD). If DISP=MOD is not used, the previously generated data will be overwritten. If the +DELTA function is used, the program generates VARAIBLE NAMES which in turn can be used in the command members $DLT????. The variable names are the same as listed in the RA/2 manual and can be found at the end of this section.	

Sample: User-IDs which failed the rules checking

```
DEB$SW51-10 RACF USER-IDS WHICH FAILED RULES CHECKING    ALS(C) V3R4M1 11/30/05 00.45  RACF VERS 2608              PAGE:         1
                                                                                                                  DATE:2005-11-30
            JOBNAME :XRZP0017 STEPNAME:RA2RULES PROCNAME:                                                         TIME:   0:46:00
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME             RULE NAMES / COMMENT
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- --------          ----------------------------
TECTRAMM  PERFORMANCE TEST    2006-07-10 MAXTECH  N N N Y N      UMAXTECH 2006-03-11 19:07:25          'TECH USERIDS'
                                                                                                        'LJDATE      '
```

Sample: User-IDs for which a matching rule was found

```
DEB$SW51-20 RACF USER-IDS WHERE A DEFINED RULE MATCHED   ALS(C) V3R4M1 11/30/05 00.45  RACF VERS                    PAGE:         1
                                                                                                                    DATE:2005-11-30
           JOBNAME :XRZP0017 STEPNAME:RA2RULES PROCNAME:                                                            TIME:   0:46:00
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME             RULE NAMES / COMMENT
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- --------          ------------------------------

$START    ##################### 2001-09-04 MAXFTUID N N N N Y      MAXFTUID 2005-11-23 00:36:15          'LJDATE      '
```

Command generation

For each failing rule it is possible to generate any kind of commands. Below are the variable names listed which can be used in a command member (DDname //COMMANDS). Each time a rule fails and the keyword COMMAND=member_name has been specified, the member will be read from //COMMANDS PDS file and all variables replaced. The output will be written to //USRC0200, 0202, 0280 etc. . This facility works similar to the ISPF FTINCL function, however it does not support e.g.)SEL,)IM logic.

Variable names filled in by the IRRI0200 record:

The following variables can be used to generate commands related to user-Ids:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT
NAME	&R20NAME	
CREATE_DATE	&R20AUTHD	YYYY-MM-DD
OWNER_ID	&R20OWNER	
ADSP	&R20ADSP	Y OR N (?)
SPECIAL	&R20SPEC	Y OR N (?)
OPER	&R20OPER	Y OR N (?)
REVOKE	&R20REV	Y OR N (?)
GRPACC	&R20GRPA	Y OR N (?)
PWD_INTERVAL	&R20PWI	
PWD_DATE	&R20PWL	YYYY-MM-DD
PROGRAMMER	&R20PGMN	
DEFGRP_ID	&R20DEFGR	
LASTJOB_TIME	&R20TIME	HH:MM:SS
LASTJOB_DATE	&R20DATE	YYYY-MM-DD
INSTALL_DATA	&R20DATA	
UAUDIT	&R20UAUD	Y OR N (?)
AUDITOR	&R20AUDIT	Y OR N (?)
NOPWD	&R20PWREQ	Y OR N (?)
OIDCARD	&R20IODC	Y OR N (?)
PWD_GEN	&R20GENPW	
REVOKE_CNT	&R20FAIL	
MODEL	&R20MODEL	
SECLEVEL	&R20SECL	
REVOKE_DATE	&R20REVD	YYYY-MM-DD
RESUME_DATE	&R20RESD	YYYY-MM-DD
ACCESS_SUN	&R20WDSUN	Y OR N (?)
ACCESS_MON	&R20WDMON	Y OR N (?)
ACCESS_TUE	&R20WDTUE	Y OR N (?)
ACCESS_WED	&R20WDWED	Y OR N (?)
ACCESS_THU	&R20WDTHU	Y OR N (?)
ACCESS_FRI	&R20WDFRI	Y OR N (?)
ACCESS_SAT	&R20WDSAT	Y OR N (?)
START_TIME	&R20WTSTR	HH:MM:SS
END_TIME	&R20WTEND	HH:MM:SS
SECLABEL	&R20SECLA	
STARTHH_TIME	&R20TIHHS	HH:MM:SS
ENDHH_TIME	&R20TIHHE	HH:MM:SS

Variable names filled in by the failing rule:

OPTIONS KEYWORD	OPTIONS VARIABLE	FORMAT
SET_DFLTGRP=NAME	&SUDFLTGRP	MAX. 8 CHAR
SET_OWNER=NAME	&SUOWNER	MAX. 8 CHAR

```
RULE KEYWORD                  RULE VARIABLE      FORMAT
------------------            -------------      ----------
SET_DFLTGRP=NAME              &SDFLTGRP          MAX. 8 CHAR
SET_OWNER=NAME                &SOWNER            MAX. 8 CHAR
```

Sample: Command member

```
/* THIS IS A TEST MEMBER TO SHOW HOW COMMANDS WILL WORK */
 ALTUSER  (&R20NAME)           +
          NAME('&R20PGMN ')    +
          DFLTGRP(&R20DEFGR)   +
          SPECIAL              +
          OPERATIONS           +
          NOPASSWORD NOOIDCARD +
          RESTRICTED           +
          OWNER(&R20OWNER)
 PASSWORD INTERVAL(&R20PWI) USER(&R20NAME)
/*                                                    */
 SDFLTGRP    &SDFLTGRP
 SOWNER      &SOWNER
 SUDFLTGRP   &SUDFLTGRP
 SUOWNER     &SUOWNER
```

Variable names filled in by the IRRI0202 record (CLAUTH):

The following variables can be used to generate commands related to user-Ids:

```
RACF IRRDBU00 NAME            RA/2 VARIABLE   FORMAT
------------------            -------------   ----------
NAME                          &R22NAME
CLASS                         &R22CLAUT
```

```
RULE KEYWORD                  RULE VARIABLE      FORMAT
------------------            -------------      ----------
SET_CLAUTH                    &SCLAUTH           MAX. 8 CHAR
```

Sample: Rule and command member

```
+CLR NAME='AAAA CLAUTH ',U=A*,CLAUTH=(TAPEVOL,A,B,C,D),COMMAND=USRR0002,
  SET_CLAUTH=DASDVOL


/* OLD CONNECT CLAUTH VALUE           AUTH(&R22CLAUT    */
 ALTUSER  (&R22NAME)  CLAUTH(&SCLAUTH)
/*                                                    */
```

Variable names (for +DELTA command processing and other rule command processing)

Below is a list of all variable names (USERIDX and Y) which can be utilized to create commands. The same variable names (USERIDX only) can be utilized

- The variable &FIELDNAME shows a user, which field was in error ‘between’ USERIDX and USERIDY.
- The +DELTA template processing does not support additional selection processing as e.g. known under ISPF (file tailoring services). Control card images can be max. 80 bytes. This restricts the use of variable field names longer that 80 bytes.
- A command could look like this:
  - ALU &R20NAME OWNER(&Y20NAME)

```
BROWSE    RA2002.V3R6M0.COMMANDS($DLT0200) - 01.08   Line 00
Command ===> ____________________________________________________
******************************** Top of Data ***************
/*                                                          */
/* SAMPLE TO GENERATE CONTROL CARDS                         */
/*                                                          */
/* FIELDNAME IN ERROR: &FIELDNAME */
/*                                                          */
CONTROL   NOFLUSH  NOLIST NOCONLIST NOSYMLIST   MSG ASIS
ALTUSER (&R20NAME) +
                    NAME('&R20PGMN ') +
                    DFLTGRP(&R20DEFGR)  +
```

RECORD TYPE IRRI0200	USERIDX	USERIDY	LENGTH
USBD_NAME	&R20NAME	&Y20NAME	0008
USBD_CREATE_DATE	&R20AUTHD	&Y20AUTHD	0010
USBD_OWNER_ID	&R20OWNER	&Y20OWNER	0008
USBD_ADSP	&R20ADSP	&Y20ADSP	0001
USBD_SPECIAL	&R20SPEC	&Y20SPEC	0001
USBD_OPER	&R20OPER	&Y20OPER	0001
USBD_REVOKE	&R20REV	&Y20REV	0001
USBD_GRPACC	&R20GRPA	&Y20GRPA	0001
USBD_PWD_INTERVAL	&R20PWI	&Y20PWI	0003
USBD_PWD_DATE	&R20PWL	&Y20PWL	0010
USBD_PROGRAMMER	&R20PGMN	&Y20PGMN	0020
USBD_DEFGRP_ID	&R20DEFGR	&Y20DEFGR	0008
USBD_LASTJOB_TIME	&R20TIME	&Y20TIME	0008
USBD_LASTJOB_DATE	&R20DATE	&Y20DATE	0010
USBD_INSTALL_DATA	&R20DATA	&Y20DATA	0254
USBD_UAUDIT	&R20UAUD	&Y20UAUD	0001
USBD_AUDITOR	&R20AUDIT	&Y20AUDIT	0001
USBD_NOPWD	&R20PWREQ	&Y20PWREQ	0001
USBD_OIDCARD	&R20IODC	&Y20IODC	0001
USBD_PWD_GEN	&R20GENPW	&Y20GENPW	0003
USBD_REVOKE_CNT	&R20FAIL	&Y20FAIL	0003
USBD_MODEL	&R20MODEL	&Y20MODEL	0044
USBD_SECLEVEL	&R20SECL	&Y20SECL	0003
USBD_REVOKE_DATE	&R20REVD	&Y20REVD	0010
USBD_RESUME_DATE	&R20RESD	&Y20RESD	0010
USBD_ACCESS_SUN	&R20WDSUN	&Y20WDSUN	0001
USBD_ACCESS_MON	&R20WDMON	&Y20WDMON	0001
USBD_ACCESS_TUE	&R20WDTUE	&Y20WDTUE	0001
USBD_ACCESS_WED	&R20WDWED	&Y20WDWED	0001
USBD_ACCESS_THU	&R20WDTHU	&Y20WDTHU	0001
USBD_ACCESS_FRI	&R20WDFRI	&Y20WDFRI	0001
USBD_ACCESS_SAT	&R20WDSAT	&Y20WDSAT	0001
USBD_START_TIME	&R20WTSTR	&Y20WTSTR	0008
USBD_END_TIME	&R20WTEND	&Y20WTEND	0008
USBD_SECLABEL	&R20SECLA	&Y20SECLA	0008
USBD_ATTRIBS	&R20ATTRI	&Y20ATTRI	0008
USBD_PWDENV_EXIST	&R20PWENV	&Y20PWENV	0001
USBD_PWD_ASIS	&R20PASIS	&Y20PASIS	0001
USBD_PHR_DATE	&R20PDATE	&Y20PDATE	0010
USBD_PHR_GEN	&R20PGEN	&Y20PGEN	0003
USBD_CERT_SEQN	&R20SEQN	&Y20SEQN	0010
USBD_PPHENV_EXISTS	&R20PPHEN	&Y20PPHEN	0001
USBD_STARTHH_TIME	&R20TIHHS	&Y20TIHHS	0004
USBD_ENDHH_TIME	&R20TIHHE	&Y20TIHHE	0004

RECORD TYPE IRRI0201	USERIDX	USERIDY	LENGTH
USCAT_NAME	&R21NAME	&Y21NAME	0008
USCAT_CATEGORY	&R21CATNO	&Y21CATNO	0005

RECORD TYPE IRRI0202	USERIDX	USERIDY	LENGTH
USCLA_NAME	&R22NAME	&Y22NAME	0008
USCLA_CLASS	&R22CLAUT	&Y22CLAUT	0008

RECORD TYPE IRRI0203	USERIDX	USERIDY	LENGTH
USGCON_NAME	&R23NAME	&Y23NAME	0008
USGCON_GRP_ID	&R23CONG	&Y23CONG	0008

RECORD TYPE IRRI0204	USERIDX	USERIDY	LENGTH
USINSTD_NAME	&R24NAME	&Y24NAME	0008
USINSTD_USR_NAME	&R24USER	&Y24USER	0008
USINSTD_USR_DATA	&R24DATA	&Y24DATA	0254
USINSTD_USR_FLAG	&R24FLAG	&Y24FLAG	0008

RECORD TYPE IRRI0205	USERIDX	USERIDY	LENGTH
USCON_NAME	&R30NAME	&Y30NAME	0008
USCON_GRP_ID	&R30GROUP	&Y30GROUP	0008
USCON_CONNECT_DATE	&R30AUTHD	&Y30AUTHD	0010
USCON_OWNER_ID	&R30OWNER	&Y30OWNER	0008
USCON_LASTCON_TIME	&R30TIME	&Y30TIME	0008
USCON_LASTCON_DATE	&R30DATE	&Y30DATE	0010
USCON_UACC	&R30UACC	&Y30UACC	0008
USCON_INIT_CNT	&R30INIT	&Y30INIT	0005
USCON_GRP_ADSP	&R30FLAG1	&Y30FLAG1	0001
USCON_GRP_SPECIAL	&R30FLAG2	&Y30FLAG2	0001
USCON_GRP_OPER	&R30FLAG3	&Y30FLAG3	0001
USCON_REVOKE	&R30FLAG4	&Y30FLAG4	0001
USCON_GRP_ACC	&R30FLAG5	&Y30FLAG5	0001
USCON_NOTERMUACC	&R30TRM	&Y30TRM	0001
USCON_GRP_AUDIT	&R30GRPAU	&Y30GRPAU	0001
USCON_REVOKE_DATE	&R30REVD	&Y30REVD	0010
USCON_RESUME_DATE	&R30RESD	&Y30RESD	0010

RECORD TYPE IRRI0206	USERIDX	USERIDY	LENGTH
USRSF_NAME	&R206NAME	&Y206NAME	0008
USRSF_TARG_NODE	&R206NODE	&Y206NODE	0008
USRSF_TARG_USER_ID	&R206USER	&Y206USER	0008
USRSF_VERSION	&R206VERS	&Y206VERS	0003
USRSF_PEER	&R206PEER	&Y206PEER	0001
USRSF_MANAGING	&R206MANA	&Y206MANA	0001
USRSF_MANAGED	&R206MANT	&Y206MANT	0001
USRSF_REMOTE_PEND	&R206RPEN	&Y206RPEN	0001
USRSF_LOCAL_PEND	&R206LPEN	&Y206LPEN	0001
USRSF_PWD_SYNC	&R206PSYN	&Y206PSYN	0001
USRSF_REM_SYSERR	&R206RSYS	&Y206RSYS	0001
USRSF_DEFINE_DATE	&R206DEFD	&Y206DEFD	0010
USRSF_DEFINE_TIME	&R206DEFT	&Y206DEFT	0015
USRSF_ACCEPT_DATE	&R206ACCD	&Y206ACCD	0010
USRSF_ACCEPT_TIME	&R206ACCT	&Y206ACCT	0015
USRSF_CREATOR_ID	&R206CRID	&Y206CRID	0008

RECORD TYPE IRRI0207	USERIDX	USERIDY	LENGTH
USCERT_NAME	&R207NAME	&Y207NAME	0008
USCERT_CERT_NAME	&R207CNAM	&Y207CNAM	0246
USCERT_CERTLABL	&R207CLAB	&Y207CLAB	0032

RECORD TYPE IRRI0208	USERIDX	USERIDY	LENGTH
USNMAP_NAME	&R208NAME	&Y208NAME	0008
USNMAP_LABEL	&R208LABL	&Y208LABL	0032
USNMAP_MAP_NAME	&R208MAPN	&Y208MAPN	0246

RECORD TYPE IRRI0210	USERIDX	USERIDY	LENGTH
USDFP_NAME	&R25NAME	&Y25NAME	0008
USDFP_DATAAPPL	&R25ACLAS	&Y25ACLAS	0008
USDFP_DATACLAS	&R25DCLAS	&Y25DCLAS	0008
USDFP_MGMTCLAS	&R25MCLAS	&Y25MCLAS	0008
USDFP_STORCLAS	&R25SCLAS	&Y25SCLAS	0008

RECORD TYPE IRRI0220	USERIDX	USERIDY	LENGTH
USTSO_NAME	&R26NAME	&Y26NAME	0008
USTSO_ACCOUNT	&R26TACC	&Y26TACC	0040
USTSO_COMMAND	&R26CMD	&Y26CMD	0080
USTSO_DEST	&R26TDEST	&Y26TDEST	0008
USTSO_HOLD_CLASS	&R26THCLA	&Y26THCLA	0001
USTSO_JOB_CLASS	&R26TJCLA	&Y26TJCLA	0001
USTSO_LOGON_PROC	&R26TPROC	&Y26TPROC	0008
USTSO_LOGON_SIZE	&R26TLSIZ	&Y26TLSIZ	0007
USTSO_MSG_CLASS	&R26TMCLA	&Y26TMCLA	0001
USTSO_LOGON_MAX	&R26TMSIZ	&Y26TMSIZ	0007
USTSO_PERF_GROUP	&R26TPERF	&Y26TPERF	0010
USTSO_SYSOUT_CLASS	&R26TSCLA	&Y26TSCLA	0001
USTSO_USER_DATA	&R26TDATA	&Y26TDATA	0004
USTSO_UNIT_NAME	&R26TUNIT	&Y26TUNIT	0008
USTSO_SECLABEL	&R26SECLA	&Y26SECLA	0008

RECORD TYPE IRRI0230	USERIDX	USERIDY	LENGTH
USCICS_NAME	&R27NAME	&Y27NAME	0008
USCICS_OPIDENT	&R27OPI	&Y27OPI	0003
USCICS_OPPRTY	&R27PRTY	&Y27PRTY	0003
USCICS_NOFORCE	&R27NFORC	&Y27NFORC	0001
USCICS_TIMEOUT	&R27TOUT	&Y27TOUT	0005

RECORD TYPE IRRI0231	USERIDX	USERIDY	LENGTH
USCOPC_NAME	&R28NAME	&Y28NAME	0008
USCOPC_OPCLASS	&R28OPCL	&Y28OPCL	0002

RECORD TYPE IRRI0232	USERIDX	USERIDY	LENGTH
USCRSL_NAME	&R232NAME	&Y232NAME	0008
USCRSL_KEY	&R232KEY	&Y232KEY	0005

RECORD TYPE IRRI0233	USERIDX	USERIDY	LENGTH
USCTSL_NAME	&R233NAME	&Y233NAME	0008
USCTSL_KEY	&R233KEY	&Y233KEY	0005

RECORD TYPE IRRI0240	USERIDX	USERIDY	LENGTH
USLAN_NAME	&R29NAME	&Y29NAME	0008
USLAN_PRIMARY	&R29PRIM	&Y29PRIM	0003
USLAN_SECONDARY	&R29SECO	&Y29SECO	0003

RECORD TYPE IRRI0250	USERIDX	USERIDY	LENGTH
USOPR_NAME	&R2ANAME	&Y2ANAME	0008
USOPR_STORAGE	&R2ASTOR	&Y2ASTOR	0004
USOPR_MASTERAUTH	&R2AMAST	&Y2AMAST	0001
USOPR_ALLAUTH	&R2AALLA	&Y2AALLA	0001
USOPR_SYSAUTH	&R2ASAUTH	&Y2ASAUTH	0001
USOPR_IOAUTH	&R2AIOAUT	&Y2AIOAUT	0001
USOPR_CONSAUTH	&R2ACAUTH	&Y2ACAUTH	0001
USOPR_INFOAUTH	&R2AIAUTH	&Y2AIAUTH	0001
USOPR_TIMESTAMP	&R2ATSTAM	&Y2ATSTAM	0001
USOPR_SYSTEMID	&R2ASID	&Y2ASID	0001
USOPR_JOBID	&R2AJOBID	&Y2AJOBID	0001
USOPR_MSGID	&R2AMSGID	&Y2AMSGID	0001
USOPR_X	&R2AOPRX	&Y2AOPRX	0001
USOPR_WTOR	&R2AWTOR	&Y2AWTOR	0001
USOPR_IMMEDIATE	&R2AIMME	&Y2AIMME	0001
USOPR_CRITICAL	&R2ACRIT	&Y2ACRIT	0001
USOPR_EVENTUAL	&R2AEVEN	&Y2AEVEN	0001
USOPR_INFO	&R2AINFO	&Y2AINFO	0001
USOPR_NOBRODCAST	&R2ANBROD	&Y2ANBROD	0001
USOPR_ALL	&R2AALL	&Y2AALL	0001
USOPR_JOBNAMES	&R2AJOBN	&Y2AJOBN	0001
USOPR_JOBNAMEST	&R2AJOBNT	&Y2AJOBNT	0001
USOPR_SESS	&R2ASESS	&Y2ASESS	0001
USOPR_SESST	&R2ASESST	&Y2ASESST	0001
USOPR_STATUS	&R2ASTAT	&Y2ASTAT	0001
USOPR_ROUTECODE001	&R2ARC001	&Y2ARC001	0001
USOPR_ROUTECODE002	&R2ARC002	&Y2ARC002	0001
USOPR_ROUTECODE003	&R2ARC003	&Y2ARC003	0001
USOPR_ROUTECODE004	&R2ARC004	&Y2ARC004	0001
USOPR_ROUTECODE005	&R2ARC005	&Y2ARC005	0001
USOPR_ROUTECODE006	&R2ARC006	&Y2ARC006	0001
USOPR_ROUTECODE007	&R2ARC007	&Y2ARC007	0001
USOPR_ROUTECODE008	&R2ARC008	&Y2ARC008	0001
USOPR_ROUTECODE009	&R2ARC009	&Y2ARC009	0001
USOPR_ROUTECODE010	&R2ARC010	&Y2ARC010	0001
USOPR_ROUTECODE011	&R2ARC011	&Y2ARC011	0001
USOPR_ROUTECODE012	&R2ARC012	&Y2ARC012	0001
USOPR_ROUTECODE013	&R2ARC013	&Y2ARC013	0001
USOPR_ROUTECODE014	&R2ARC014	&Y2ARC014	0001
USOPR_ROUTECODE015	&R2ARC015	&Y2ARC015	0001
USOPR_ROUTECODE016	&R2ARC016	&Y2ARC016	0001
USOPR_ROUTECODE017	&R2ARC017	&Y2ARC017	0001
USOPR_ROUTECODE018	&R2ARC018	&Y2ARC018	0001
USOPR_ROUTECODE019	&R2ARC019	&Y2ARC019	0001
USOPR_ROUTECODE020	&R2ARC020	&Y2ARC020	0001
USOPR_ROUTECODE021	&R2ARC021	&Y2ARC021	0001
USOPR_ROUTECODE022	&R2ARC022	&Y2ARC022	0001
USOPR_ROUTECODE023	&R2ARC023	&Y2ARC023	0001
USOPR_ROUTECODE024	&R2ARC024	&Y2ARC024	0001
USOPR_ROUTECODE025	&R2ARC025	&Y2ARC025	0001
USOPR_ROUTECODE026	&R2ARC026	&Y2ARC026	0001
USOPR_ROUTECODE027	&R2ARC027	&Y2ARC027	0001

USOPR_ROUTECODE028	&R2ARC028	&Y2ARC028	0001
USOPR_ROUTECODE029	&R2ARC029	&Y2ARC029	0001
USOPR_ROUTECODE030	&R2ARC030	&Y2ARC030	0001
USOPR_ROUTECODE031	&R2ARC031	&Y2ARC031	0001
USOPR_ROUTECODE032	&R2ARC032	&Y2ARC032	0001
USOPR_ROUTECODE033	&R2ARC033	&Y2ARC033	0001
USOPR_ROUTECODE034	&R2ARC034	&Y2ARC034	0001
USOPR_ROUTECODE035	&R2ARC035	&Y2ARC035	0001
USOPR_ROUTECODE036	&R2ARC036	&Y2ARC036	0001
USOPR_ROUTECODE037	&R2ARC037	&Y2ARC037	0001
USOPR_ROUTECODE038	&R2ARC038	&Y2ARC038	0001
USOPR_ROUTECODE039	&R2ARC039	&Y2ARC039	0001
USOPR_ROUTECODE040	&R2ARC040	&Y2ARC040	0001
USOPR_ROUTECODE041	&R2ARC041	&Y2ARC041	0001
USOPR_ROUTECODE042	&R2ARC042	&Y2ARC042	0001
USOPR_ROUTECODE043	&R2ARC043	&Y2ARC043	0001
USOPR_ROUTECODE044	&R2ARC044	&Y2ARC044	0001
USOPR_ROUTECODE045	&R2ARC045	&Y2ARC045	0001
USOPR_ROUTECODE046	&R2ARC046	&Y2ARC046	0001
USOPR_ROUTECODE047	&R2ARC047	&Y2ARC047	0001
USOPR_ROUTECODE048	&R2ARC048	&Y2ARC048	0001
USOPR_ROUTECODE049	&R2ARC049	&Y2ARC049	0001
USOPR_ROUTECODE050	&R2ARC050	&Y2ARC050	0001
USOPR_ROUTECODE051	&R2ARC051	&Y2ARC051	0001
USOPR_ROUTECODE052	&R2ARC052	&Y2ARC052	0001
USOPR_ROUTECODE053	&R2ARC053	&Y2ARC053	0001
USOPR_ROUTECODE054	&R2ARC054	&Y2ARC054	0001
USOPR_ROUTECODE055	&R2ARC055	&Y2ARC055	0001
USOPR_ROUTECODE056	&R2ARC056	&Y2ARC056	0001
USOPR_ROUTECODE057	&R2ARC057	&Y2ARC057	0001
USOPR_ROUTECODE058	&R2ARC058	&Y2ARC058	0001
USOPR_ROUTECODE059	&R2ARC059	&Y2ARC059	0001
USOPR_ROUTECODE060	&R2ARC060	&Y2ARC060	0001
USOPR_ROUTECODE061	&R2ARC061	&Y2ARC061	0001
USOPR_ROUTECODE062	&R2ARC062	&Y2ARC062	0001
USOPR_ROUTECODE063	&R2ARC063	&Y2ARC063	0001
USOPR_ROUTECODE064	&R2ARC064	&Y2ARC064	0001
USOPR_ROUTECODE065	&R2ARC065	&Y2ARC065	0001
USOPR_ROUTECODE066	&R2ARC066	&Y2ARC066	0001
USOPR_ROUTECODE067	&R2ARC067	&Y2ARC067	0001
USOPR_ROUTECODE068	&R2ARC068	&Y2ARC068	0001
USOPR_ROUTECODE069	&R2ARC069	&Y2ARC069	0001
USOPR_ROUTECODE070	&R2ARC070	&Y2ARC070	0001
USOPR_ROUTECODE071	&R2ARC071	&Y2ARC071	0001
USOPR_ROUTECODE072	&R2ARC072	&Y2ARC072	0001
USOPR_ROUTECODE073	&R2ARC073	&Y2ARC073	0001
USOPR_ROUTECODE074	&R2ARC074	&Y2ARC074	0001
USOPR_ROUTECODE075	&R2ARC075	&Y2ARC075	0001
USOPR_ROUTECODE076	&R2ARC076	&Y2ARC076	0001
USOPR_ROUTECODE077	&R2ARC077	&Y2ARC077	0001
USOPR_ROUTECODE078	&R2ARC078	&Y2ARC078	0001
USOPR_ROUTECODE079	&R2ARC079	&Y2ARC079	0001
USOPR_ROUTECODE080	&R2ARC080	&Y2ARC080	0001

USOPR_ROUTECODE081	&R2ARC081	&Y2ARC081	0001
USOPR_ROUTECODE082	&R2ARC082	&Y2ARC082	0001
USOPR_ROUTECODE083	&R2ARC083	&Y2ARC083	0001
USOPR_ROUTECODE084	&R2ARC084	&Y2ARC084	0001
USOPR_ROUTECODE085	&R2ARC085	&Y2ARC085	0001
USOPR_ROUTECODE086	&R2ARC086	&Y2ARC086	0001
USOPR_ROUTECODE087	&R2ARC087	&Y2ARC087	0001
USOPR_ROUTECODE088	&R2ARC088	&Y2ARC088	0001
USOPR_ROUTECODE089	&R2ARC089	&Y2ARC089	0001
USOPR_ROUTECODE090	&R2ARC090	&Y2ARC090	0001
USOPR_ROUTECODE091	&R2ARC091	&Y2ARC091	0001
USOPR_ROUTECODE092	&R2ARC092	&Y2ARC092	0001
USOPR_ROUTECODE093	&R2ARC093	&Y2ARC093	0001
USOPR_ROUTECODE094	&R2ARC094	&Y2ARC094	0001
USOPR_ROUTECODE095	&R2ARC095	&Y2ARC095	0001
USOPR_ROUTECODE096	&R2ARC096	&Y2ARC096	0001
USOPR_ROUTECODE097	&R2ARC097	&Y2ARC097	0001
USOPR_ROUTECODE098	&R2ARC098	&Y2ARC098	0001
USOPR_ROUTECODE099	&R2ARC099	&Y2ARC099	0001
USOPR_ROUTECODE100	&R2ARC100	&Y2ARC100	0001
USOPR_ROUTECODE101	&R2ARC101	&Y2ARC101	0001
USOPR_ROUTECODE102	&R2ARC102	&Y2ARC102	0001
USOPR_ROUTECODE103	&R2ARC103	&Y2ARC103	0001
USOPR_ROUTECODE104	&R2ARC104	&Y2ARC104	0001
USOPR_ROUTECODE105	&R2ARC105	&Y2ARC105	0001
USOPR_ROUTECODE106	&R2ARC106	&Y2ARC106	0001
USOPR_ROUTECODE107	&R2ARC107	&Y2ARC107	0001
USOPR_ROUTECODE108	&R2ARC108	&Y2ARC108	0001
USOPR_ROUTECODE109	&R2ARC109	&Y2ARC109	0001
USOPR_ROUTECODE110	&R2ARC110	&Y2ARC110	0001
USOPR_ROUTECODE111	&R2ARC111	&Y2ARC111	0001
USOPR_ROUTECODE112	&R2ARC112	&Y2ARC112	0001
USOPR_ROUTECODE113	&R2ARC113	&Y2ARC113	0001
USOPR_ROUTECODE114	&R2ARC114	&Y2ARC114	0001
USOPR_ROUTECODE115	&R2ARC115	&Y2ARC115	0001
USOPR_ROUTECODE116	&R2ARC116	&Y2ARC116	0001
USOPR_ROUTECODE117	&R2ARC117	&Y2ARC117	0001
USOPR_ROUTECODE118	&R2ARC118	&Y2ARC118	0001
USOPR_ROUTECODE119	&R2ARC119	&Y2ARC119	0001
USOPR_ROUTECODE120	&R2ARC120	&Y2ARC120	0001
USOPR_ROUTECODE121	&R2ARC121	&Y2ARC121	0001
USOPR_ROUTECODE122	&R2ARC122	&Y2ARC122	0001
USOPR_ROUTECODE123	&R2ARC123	&Y2ARC123	0001
USOPR_ROUTECODE124	&R2ARC124	&Y2ARC124	0001
USOPR_ROUTECODE125	&R2ARC125	&Y2ARC125	0001
USOPR_ROUTECODE126	&R2ARC126	&Y2ARC126	0001
USOPR_ROUTECODE127	&R2ARC127	&Y2ARC127	0001
USOPR_ROUTECODE128	&R2ARC128	&Y2ARC128	0001
USOPR_LOGCMDRESP	&R2ALOG	&Y2ALOG	0008
USOPR_MIGRATIONID	&R2AMIG	&Y2AMIG	0001
USOPR_DELOPERMSG	&R2ADELOP	&Y2ADELOP	0008
USOPR_RETRIEVE_KEY	&R2ARETRK	&Y2ARETRK	0008
USOPR_CMDSYS	&R2ACMDSY	&Y2ACMDSY	0008

USOPR_UD	&R2AUD	&Y2AUD	0001
USOPR_ALTGRP_ID	&R2AALTID	&Y2AALTID	0008
USOPR_AUTO	&R2AAUTO	&Y2AAUTO	0001

RECORD TYPE IRRI0251	USERIDX	USERIDY	LENGTH
USOPRP_NAME	&R2BNAME	&Y2BNAME	0008
USOPRP_SYSTEM	&R2BSYST	&Y2BSYST	0008
RECORD TYPE IRRI0260	USERIDX	USERIDY	LENGTH
USWRK_NAME	&R2CNAME	&Y2CNAME	0008
USWRK_AREA_WANAME	&R2CANAME	&Y2CANAME	0060
USWRK_BUILDING	&R2CBLDG	&Y2CBLDG	0060
USWRK_DEPARTMENT	&R2CDEPT	&Y2CDEPT	0060
USWRK_ROOM	&R2CROOM	&Y2CROOM	0060
USWRK_ADDR_LINE1	&R2CLINE1	&Y2CLINE1	0060
USWRK_ADDR_LINE2	&R2CLINE2	&Y2CLINE2	0060
USWRK_ADDR_LINE3	&R2CLINE3	&Y2CLINE3	0060
USWRK_ADDR_LINE4	&R2CLINE4	&Y2CLINE4	0060
USWRK_ACCOUNT	&R2CACCT	&Y2CACCT	0254

RECORD TYPE IRRI0270	USERIDX	USERIDY	LENGTH
USOMVS_NAME	&R270NAME	&Y270NAME	0008
USOMVS_UID	&R270UID	&Y270UID	0010
USOMVS_HOME_PATH	&R270HPAT	&Y270HPAT	1023
USOMVS_PROGRAM	&R270PROG	&Y270PROG	1023
USOMVS_CPUTIMEMAX	&R270CMAX	&Y270CMAX	0010
USOMVS_ASSIZEMAX	&R270AMAX	&Y270AMAX	0010
USOMVS_FILEPROCMAX	&R270FMAX	&Y270FMAX	0010
USOMVS_PROCUSERMAX	&R270PMAX	&Y270PMAX	0010
USOMVS_THREADSMAX	&R270TMAX	&Y270TMAX	0010
USOMVS_MMAPAREAMAX	&R270MMAX	&Y270MMAX	0010
USOMVS_MEMLIMIT	&R270MLIM	&Y270MLIM	0009
USOMVS_SHMEMAX	&R270MEMX	&Y270MEMX	0009

RECORD TYPE IRRI0280	USERIDX	USERIDY	LENGTH
USNV_NAME	&R280NAME	&Y280NAME	0008
USNV_IC	&R280IC	&Y280IC	0255
USNV_CONSNAME	&R280CONS	&Y280CONS	0008
USNV_CTL	&R280CTL	&Y280CTL	0008
USNV_MSGRECVR	&R280MSGR	&Y280MSGR	0001
USNV_NGMFADMN	&R280NGMF	&Y280NGMF	0001
USNV_NGMFVSPN	&R280VSPN	&Y280VSPN	0008

RECORD TYPE IRRI0281	USERIDX	USERIDY	LENGTH
USNVOP_NAME	&R281NAME	&Y281NAME	0008
USNVOP_OPCLASS	&R281OPC	&Y281OPC	0004

RECORD TYPE IRRI0282	USERIDX	USERIDY	LENGTH
USNVDM_NAME	&R282NAME	&Y282NAME	0008
USNVDM_DOMAINS	&R282DOM	&Y282DOM	0005

RECORD TYPE IRRI0290	USERIDX	USERIDY	LENGTH
USDCE_NAME	&R290NAME	&Y290NAME	0008
USDCE_UUID	&R290UUID	&Y290UUID	0036
USDCE_DCE_NAMES	&R290DCEN	&Y290DCEN	1023
USDCE_HOMECELL	&R290HCEL	&Y290HCEL	1023
USDCE_HOMEUUID	&R290HUID	&Y290HUID	0036
USDCE_AUTOLOGIN	&R290LOGI	&Y290LOGI	0001

RECORD TYPE IRRI02A0	USERIDX	USERIDY	LENGTH
USOVM_NAME	&R2A0NAME	&Y2A0NAME	0008
USOVM_UID	&R2A0UID	&Y2A0UID	0010
USOVM_HOMEPATH	&R2A0HPAT	&Y2A0HPAT	1023
USOVM_PROGRAM	&R2A0PROG	&Y2A0PROG	1023
USOVM_FSROOT	&R2A0ROOT	&Y2A0ROOT	1023

RECORD TYPE IRRI02B0	USERIDX	USERIDY	LENGTH
USLNOT_NAME	&R2B0NAME	&Y2B0NAME	0008
USLNOT_SNAME	&R2B0SNAM	&Y2B0SNAM	0064

RECORD TYPE IRRI02C0	USERIDX	USERIDY	LENGTH
USNDS_NAME	&R2C0NAME	&Y2C0NAME	0008
USNDS_UNAME	&R2C0UNAM	&Y2C0UNAM	0246

RECORD TYPE IRRI02D0	USERIDX	USERIDY	LENGTH
USKERB_NAME	&R2D0NAME	&Y2D0NAME	0008
USKERB_KERBNAME	&R2D0KNAM	&Y2D0KNAM	0250
USKERB_MAX_LIFE	&R2D0MLIF	&Y2D0MLIF	0010
USKERB_KEX_VERS	&R2D0KVER	&Y2D0KVER	0003

RECORD TYPE IRRI02E0	USERIDX	USERIDY	LENGTH
USPROXY_NAME	&R2E0NAME	&Y2E0NAME	0008
USPROXY_LDAP_HOST	&R2E0LDAP	&Y2E0LDAP	1023
USPROXY_BIND_DN	&R2E0BIND	&Y2E0BIND	1023

RECORD TYPE IRRI02F0	USERIDX	USERIDY	LENGTH
USEIM_NAME	&R2F0NAME	&Y2F0NAME	0008
USEIM_LDAPPROF	&R2F0LDAP	&Y2F0LDAP	0246

Sample: +DELTA output file //DLTF0205

- Below you find a typical report for a user who inherited access rights when e.g. moving from the operations group (user-id NOM120) to the systems group. To fix the problem you can define the following in the member $DLT0205 of the //COMMANDS file:
  - REMOVE &R30NAME GROUP(&R30GROUP)

```
1DEB$SW51-2D DELTA USER_IDS - FAILED FIELDS:TYPE=0205     V3R6M0 RACFRA2.COM(C) 06/18/08 RACF VERS2608               PAGE:         1

                                                                                                                  DATE:2008-06-19

            JOBNAME :XRZP0015 STEPNAME:RA2VERIF PROCNAME:          R-NAME:BASE                                    TIME:   1:58:31

0IRRDBU00 FIELD NAME       USERID-X CONTENT                          USERID-Y CONTENT                         KEY

 ------------------------- -------- -------------------------------- -------- -------------------------------- ----------------
 GRP_ID                    NOM120   ADS05C2P                         NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   B31P01AD                         NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   B31P01US                         NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC1P                            NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC1W                            NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC10P3                          NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC10P4                          NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC11P3                          NOM780   *** MISSING ITEM? PLS VERIFY
 GRP_ID                    NOM120   CIC11P4                          NOM780   *** MISSING ITEM? PLS VERIFY
```

DEB$SW1C – RACF Connect Verification (RRE)

Purpose:

- Verify RACF connect profiles.

JCL required to run DEB$SW1C

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1C) to create the reports:

```
//RA2VERIF   EXEC PGM=DEB$SW1C
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* COMMANDS
//*
//COMMANDS   DD   DISP=SHR,DSN=RA2002.V?R?M?.COMMANDS
//*
//* INPUT FILES
//*
//IRRI0100   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0100.VB(0)
//IRRI0200   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0200.VB(0)
//IRRI0205   DD   DISP=SHR,DSN=XRZP001.IRRDBU.IRRI0205.VB(0)
//*
//* OUTPUT FILES
//*
//CONPRINT   DD SYSOUT=*   * PRINT RESNAME_RULE CONTROL STATEMENTS
//CONC0205   DD SYSOUT=*   * RACF CONNECTS - GENERATED COMMANDS
//CONG0205   DD SYSOUT=*   * RACF CONNECTS - MATCHING RULES
//CONF0205   DD SYSOUT=*   * RACF CONNECTS - FAILED RULES
//CONX0205   DD SYSOUT=*   * RACF CONNECTS - NO RULES APPLY
//CONT0205   DD SYSOUT=*   * RACF CONNECTS - SUMMARY
//*
//CONRULES   DD *          * RACF BASE USERID RULES
*
* DEFINE RULES FOR CONNECT PROFILES RECORD TYPE 0205
* +OPTIONS SPECIFIES THE DEFAULT VALUES TO BE ASSIGNED IF
*  NO RULE DOES MATCH
*
*
+OPTIONS     SET_OWNER=CONWON,SET_GROUPID=SETGRPID SET_REVOKE=YES
+CONNECT_RULE NAME='THIS IS A CONNECT TEST',
             OWNER=FCT*,GROUPID=SYS1,
             REVOKE=YES,
             SET_OWNER=SETOWNER,SET_GROUPID=SETGRPG
+CONNECT_RULE NAME='TEST ON $KIINC',
             OWNER=$KIINC,GROUPID=$KIINC,
             SET_OWNER=$KINCC,SET_GROUPID=SETKINCC
+CONNECT_RULE NAME='TEST ON REVOKED',
             REVOKE=YES,
             SET_OWNER=$REVCC,SET_GROUPID=REVKINCC
/*
```

DDnames:

- //IRRIxxxx must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. These files are used as input to the program. In case the files are not split by record type then define on all //IRRIxxxx the offloaded RACF database flat file.
  - xxxx = IRRDBU00 record type.
- //COMMANDS must be RECFM=FB, LRECL=80, DSORG=PO. This file contains the product-supplied commands and the user defined commands. The first letter '$' is reserved for product supplied commands.
- //SORTxxxxx DDNAMES are required when the option VERIFY=YES is set.
- //SYSOUT is required by the SORT program
- //???Cxxxx must be RECFM=FB, LRECL=80, DSORG=PS.
  - ??? = USR, GRP, DSN, CON or RES
  - xxxx = IRRDBU00 record type

Note:

Control cards generated by this program must reside in separate flat files and not e.g. in one common PDS, otherwise you will encounter the following ABEND:

IEC143I 213-30,IFG0194D,MYJOBID,RA2VERIF,CONC0205. This is due to the fact that multiple files are open at the same time to generate control cards.

Connect-ID Rules (Filter) Control Statements (//CONRULES DD *)

Following control statements can be utilized to perform the RACF connect-user-ID verification:

DDname	Verbs	Keywords	Comment	Default
//CONRULES	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	SET_OWNER=	Assign new default owner if all rules fail. Assign new group-ID if all rules fail. The global variable name &SCOWNER can be used in the command member for the failing rule.	N/A
		SET_GROUPID=	Assign new group-ID if all rules fail. The global variable name &SCGROUP can be used in the command member for the failing rule.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0205 base record) will be bypassed for further processing.	N/A
		Note: all generated RACF control statements must be reviewed prior executing them. This utility does not automatically update the RACF DB.		

+CR - Connect data record (0205)

The User Connect Data record defines the relationships between users and groups. There is one record per user connection.

	+CONNECTID_RULE or +CONNECT_RULE or +CON_RULE or +CR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record Type=0205	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		USERID=	Specifies a RACF user-ID to be verified.	N/A
		OWNERID=	Specifies a RACF Owner-ID to be verified.	N/A
		GROUPID=(A,…,)	Specifies a RACF connect-group-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		REVOKE=YES or NO	Specifies that the connect-user-ID must be revoked ('Y') or not.	N/A
		SPECIAL=YES or NO	Specifies that the connect-user-ID must have the special attribute ('Y') or not.	N/A
		OPERATIONS=YES or NO	Specifies that the connect-user-ID must have the operations attribute ('Y') or not.	N/A
		UACC=	Specifies default universal access	N/A

			authority for all new resources the user defines while connected to the specified group. Valid values are NONE, READ, UPDATE, CONTROL, and ALTER.	
		AUTHDATE=(yyyy-mm-dd,??) or AUTH_DATE=(yyyy-mm-dd,??)	AUTHDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. AUTHDATE=(2000,GE) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. AUTHDATE is the date when a profile was 'defined' to RACF.	N/A
		SET_OWNER=	Assign new connect-owner if this rule fails. The variable name &SOWNER can be used in the command member for the failing rule.	N/A
		SET_GROUPID=	Assign new connect-group if this rule fails. The variable name &SGROUP can be used in the command member for the failing rule.	N/A
		BYPASS_USERID=(…, …)	Specifies RACF user-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_OWNER=(…, …)	Specifies RACF Owner-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_GROUP=(…, …)	Specifies RACF group-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_ATTR=(…, …)	Specifies RACF attributes for user-IDs to be excluded from the validation process. Valid items are: ADSP NOADSP GRPACC NOGRPACC AUDITOR NOAUDITOR SPECIAL NOSPECIAL OPERATIONS NOOPERATIONS REVOKED NOREVOKED Max. 8 attributes can be specified. Generic names are NOT supported.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records	N/A

			(type=0205 base record) will be bypassed for further processing.	
	Note: • The key fields for rule checking support generic filtering. A key field can also contain the substitution character '?' e.g. IBM?A* • If all supplied rules fail, the program will create the necessary reports and RACF control cards to alter the profiles.			

Sample: obtain all Group-Specials

Using the following rules will list all User-Ids with a 'valid' GROUP-SPECIAL. The ones which do not pass the rules test will be listed as 'failed'. This allows you to control all these userids.

```
*
* GROUP-SPECIAL Z/OS
*
+CR N='MOKXX1',U=MOKXX1,GROUPID=(*),
    BYPASS_GROUP=(FCTOMIE,FCTOS0B,FCTOS0E,FCTRACE,FCTSTBE,FCTTSOE,
                  RESOS0A,RESOS0D,RESOS0T),
    SPECIAL=NO
+CR N='MOKXX2',U=MOKXX2,GROUPID=(*),
    BYPASS_GROUP=(FCTOS0B,FCTOS0E,FCTRACE,FCTSTBE,FCTTSOE,FCTUSSE,
                  RESOS0A,RESOS0D,RESOS0T),
    SPECIAL=NO
+CR N='MOKXX3',U=MOKXX3,GROUPID=(*),
    BYPASS_GROUP=(FCTOS0B,FCTOS0E,FCTRACE,FCTSTBE,FCTTSOE,FCTUSSE,
                  FCTWASE,RESOS0A,RESOS0D,RESOS0T),
    SPECIAL=NO
*
* GROUP-SPECIAL CICS-ENGINEERING
*
+CR N='MOKXXX',U=MOKXXX,GROUPID=(*),
    BYPASS_GROUP=(ASCIC,FCTCICE,FCTIMSE,IMSS2,STCI),
    SPECIAL=NO
+CR N='MOKXXY',U=MOKXXY,GROUPID=(*),
    BYPASS_GROUP=(ASCIC,FCTCICE,FCTIMSE,IMSS2,STCI),
    SPECIAL=NO
```

Sample: Failing Connect-IDs

```
DEB$SW52-10 RACF CONNECT-IDS WHICH FAILED RULES CHECKING ALS(C) V3R6M0 10/26/05 14.18  RACF VERS 2608                PAGE:         1
                                                                                                                  DATE:2005-10-27
           JOBNAME :XRZP001A STEPNAME:RA2RULES PROCNAME:                                                          TIME:   9:37:07
USERID   GROUP-ID AUTHDATE   T OWNER    S O R CON.-DATE  TIME                                  FAILING RULE NAME(S)
-------- -------- ---------- - -------- - - - ---------- --------                              --------------------------

$$$$USER $$$$TEST 2001-11-09 G $$$$TEST N N N                                                 -> ALL RULES FAILED
A        $$$$TEST 2005-05-22 U XRZP001  N N N                                                 -> ALL RULES FAILED
A        SYS1     2005-05-22 U AAAUSER  N N N                                                 'THIS IS A CONNECT TEST'
AOF01    SYS1     2004-09-18 U AAAUSER  N N N                                                 'THIS IS A CONNECT TEST'
```

Command generation

For each failing rule it is possible to generate any kind of commands. Below are the variable names listed which can be used in a command member (DDname //COMMANDS). Each time a rule fails and the keyword COMMAND=member_name has been specified, the member will be read from //COMMANDS PDS file and all variables replaced. The output will be written to //CONC0205. This facility works similar to the ISPF FTINCL function, however it does not support e.g.)SEL,)IM logic.

Variable names filled in by the IRRI0205 record:

The following variables can be used to generate commands related to connect user-Ids:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT	SAMPLE DATA
NAME	&R30NAME		A
GRP_ID	&R30GROUP		
CONNECT_DATE	&R30AUTHD	YYYY-MM-DD	
OWNER_ID	&R30OWNER		
LASTCON_TIME	&R30TIME	HH:MM:SS	
LASTCON_DATE	&R30DATE	YYYY-MM-DD	
UACC	&R30UACC		NONE
INIT_CNT	&R30INIT		
GRP_ADSP	&R30FLAG1	Y OR N (?)	
GRP_SPECIAL	&R30FLAG2	Y OR N (?)	
GRP_OPER	&R30FLAG3	Y OR N (?)	
REVOKE	&R30FLAG4	Y OR N (?)	
GRP_ACC	&R30FLAG5	Y OR N (?)	
NOTERMUACC	&R30TRM	Y OR N (?)	
GRP_AUDIT	&R30GRPAU	Y OR N (?)	
REVOKE_DATE	&R30REVD	YYYY-MM-DD	
RESUME_DATE	&R30RESD	YYYY-MM-DD	

Variable names filled in by the failing rule:

OPTIONS KEYWORD	OPTIONS VARIABLE	FORMAT
SET_GROUP=NAME	&SCGROUP	MAX. 8 CHAR
SET_OWNER=NAME	&SCOWNER	MAX. 8 CHAR

RULE KEYWORD	RULE VARIABLE	FORMAT
SET_GROUP=NAME	&SGROUP	MAX. 8 CHAR
SET_OWNER=NAME	&SOWNER	MAX. 8 CHAR

DEB$SW1D – RACF Dataset Verification (RRE)

Purpose:

- Verify RACF dataset profiles.

JCL required to run DEB$SW1D

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1D) to create the reports:

```
//RA2VERIF   EXEC PGM=DEB$SW1D
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* SORT WORK AREAS AND OPTIONS
//*
//SYSPRINT     DD   SYSOUT=*
//SYSOUT       DD   SYSOUT=*
//SORTWK01     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK02     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK03     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK04     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//$ORTPARM DD *
NORC16
//SORTCNTL DD *
 DEBUG         NOABEND
 OPTION        VLSHRT
//*
//* COMMANDS
//*
//COMMANDS   DD   DISP=SHR,DSN=RA2002.V?R?M?.COMMANDS
//*
//* INPUT FILES
//*
//IRRI0100   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0100.VB(0)
//IRRI0200   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0200.VB(0)
//IRRI0400   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0400.VB(0)
//IRRI0402   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0402.VB(0)
//IRRI0404   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0404.VB(0)
//*
//* OUTPUT FILES
//*
//DSNPRINT   DD SYSOUT=*   * PRINT DATASET_RULE CONTROL STATEMENTS
//DSNC0400   DD SYSOUT=*   * RACF DATASETS - GENERATED COMMANDS
//DSNG0400   DD SYSOUT=*   * RACF DATASETS - MATCHING RULES
//DSNF0400   DD SYSOUT=*   * RACF DATASETS - FAILED RULES
//DSNX0400   DD SYSOUT=*   * RACF DATASETS - NO RULES APPLY
//DSNT0400   DD SYSOUT=*   * RACF DATASETS - SUMMARY
//*
//DSNC0402   DD SYSOUT=*   * RACF DATASETS - GENERATED COMMANDS
//DSNG0402   DD SYSOUT=*   * RACF DATASETS - MATCHING RULES
//DSNF0402   DD SYSOUT=*   * RACF DATASETS - FAILED RULES
//DSNX0402   DD SYSOUT=*   * RACF DATASETS - NO RULES APPLY
//DSNT0402   DD SYSOUT=*   * RACF DATASETS - SUMMARY
//*
//*
//DSNC0404   DD SYSOUT=*   * RACF DATASETS - GENERATED COMMANDS
//DSNG0404   DD SYSOUT=*   * RACF DATASETS - MATCHING RULES
//DSNF0404   DD SYSOUT=*   * RACF DATASETS - FAILED RULES
//DSNX0404   DD SYSOUT=*   * RACF DATASETS - NO RULES APPLY
//DSNT0404   DD SYSOUT=*   * RACF DATASETS - SUMMARY
//*
//DSNRULES   DD *          * RACF BASE USERID RULES
*
* DEFINE RULES FOR GROUP PROFILES RECORD TYPE 0100
* +OPTIONS SPECIFIES THE DEFAULT VALUES TO BE ASSIGNED IF
*  NO RULE DOES MATCH
*
*
+OPTIONS     SET_OWNER=SWISSOWN,SET_UACC=NONE
+DATASET_RULE NAME='THIS IS A DATASET TEST',
             DATASET=SYS1.**,OWNER=*,
             SET_OWNER=SETOWNER,SET_UACC=READ
+DATASET_RULE NAME='THIS IS A DATASET TES2',
             DATASET=XRZP00?.**,OWNER=*,
             SET_OWNER=NEWOWNER,SET_UACC=ALTER
+DATASET_RULE NAME='FIX DATASET NAME      ',
             DATASET=XRZP001.**,OWNER=*,
             SET_OWNER=SHITFIX,SET_UACC=ALTER
+DATASET_RULE NAME='GLOBAL ERROR  OR  UACC',
             DATASET=**,OWNER=OWN*,UACC=NONE,
             SET_OWNER=,SET_UACC=NONE
/*
```

DDnames:

- //IRRIxxxx must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. These files are used as input to the program. In case the files are not split by record type then define on all //IRRIxxxx the offloaded RACF database flat file.
  - xxxx = IRRDBU00 record type.

- //COMMANDS must be RECFM=FB, LRECL=80, DSORG=PO. This file contains the product-supplied commands and the user defined commands. The first letter '$' is reserved for product supplied commands.
- //SORTxxxxx DDNAMES are required when the option VERIFY=YES is set.
- //SYSOUT is required by the SORT program
- //???Cxxxx must be RECFM=FB, LRECL=80, DSORG=PS.
  - ??? = USR, GRP, DSN, CON or RES
  - xxxx = IRRDBU00 record type

Note:

Control cards generated by this program must reside in separate flat files and not e.g. in one common PDS, otherwise you will encounter the following ABEND:
IEC143I 213-30,IFG0194D,MYJOBID,RA2VERIF,DSNC0400. This is due to the fact that multiple files are open at the same time to generate control cards.

Dataset Rules (Filter) Control Statements (//DSNRULES DD *)

Following control statements can be utilized to perform the RACF dataset verification:

DDname	Verbs	Keywords	Comment	Default
//DSNRULES	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	SET_OWNER=	Assign new default owner if all rules fail. Variable name which can be used in the command members is: &SDOWNER	N/A
		SET_UACC=	Assign new UACC if all rules fail. Variable name which can be used in the command members is: &SDUACC	N/A
		SET_NOTIFY=	Assign new NOTIFY if all rules fail. Variable name which can be used in the command members is: &SDNOTIFY	N/A
		HLQ=TSO or USERID or OWNER	Verify that a user dataset profile has as owner the high level qualifier of the RACF profile (HQL = OWNER). Group dataset profiles will be ignored. For non-matching items, the relevant ALTDSD commands will be generated to change the owner to the HLQ. If the owner for a user dataset is 'SYS1', the verification will be skipped. The commands will be written to the Ddname //DSNC04OW, which must be of RECFM=FB, LRECL=80.	N/A
		UDSN or USERID_DATASET=NO or YES	Do not process dataset profiles where the high level qualifier is a user-ID. Record type 0200 will be checked = //IRRI0200 DD DSN=	YES
		GDSN or GROUPID_DATASET=NO or YES	Do not process dataset profiles where the high level qualifier is a group-ID. Record type 0100 will be checked = //IRRI0100 DD DSN=	YES
		CAT=YES or NO	Verify if for the RACF profile name datasets do exists on the system. If dataset names have been found the "T" column will be marked with a "+" (e.g. "+U" or "+G". The "T"	NO

			column indicates if the HLQ is a group or user dataset.	
		USERID_UACC=NO or YES	If set to "NO", the UACC for all user dataset profiles will be ignored. UACC= checking is normally only required for group dataset profiles e.g. UACC=NONE.	YES
		GROUPID_UACC=NO or YES	If set to "NO", the UACC for all group dataset profiles will be ignored.	YES
		VERIFY=YER or NO	Independent of any dataset RULES defined: If the VERIFY option is set to 'YES', then in addition the OWNER, NOTIFY, ACCESS list(s) and the catalog (CAT=YES must be set too) will be checked. For dataset profiles, where the HLQ appears as the second qualifier again e.g. IBMUSER.IBMUSER.** a DELDSD command will be generated. Each failing entry will be reported on //DSNVERIF. Pre-defined ALTDSD and PE xxx DEL commands will be written to the file //DSNCLEAN. The pre-defined commands will be invoked from the PDS file //COMMANDS. All supplied (pre-defined) command members start with $04xxyzz and should not be altered.	NO
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0400 base record) will be bypassed for further processing.	N/A
		Note: all generated RACF control statements must be reviewed prior executing them. This utility does not automatically update the RACF DB.		

+DR - Dataset basic data record (0400)

The Dataset Basic Data record defines the basic information about a dataset resource. There is one record per dataset profile.

	+DATASET_RULE or +DSNAME_RULE or +DSN_RULE or +DR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M	NAME=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		DATASET=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FDATASET selection.	N/A
		FDATASET=	NON-Generic profile checking:	N/A

	Record Type=0400		Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 44 bytes.	
		OWNER=	Specifies a RACF Owner-ID to be verified.	N/A
		UACC=	Specifies universal access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A
		DATA=YES or NO	Specifies that installation data must be present.	N/A
		AUDIT_OKQUAL=	Specifies the successful access audit qualifier to be verified. This is set to blanks if AUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		AUDIT_FAQUAL=	Specifies the failing access audit qualifier to be verified. This is set to blanks if AUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		GAUDIT_OKQUAL=	Specifies the auditor-specified successful access audit qualifier to be verified. This is set to blanks if GAUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		GAUDIT_FAQUAL=	Specifies the auditor-specified failing access audit qualifier to be verified. This is set to blanks if GAUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		AUDIT_LEVEL=	Specifies the audit level to be verified. This indicates the level of resource-owner-specified auditing that is performed. Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		GAUDIT_LEVEL=	Specifies the global audit level to be verified. This indicates the level of auditor-specified auditing that is performed. Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		AUTHDATE=(yyyy-mm-dd,??) or AUTH_DATE=(yyyy-mm-dd,??)	AUTHDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. AUTHDATE=(2000,GE) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. AUTHDATE is the date when a profile was 'defined' to RACF.	N/A

		SET_OWNER=	Assign new dataset-owner if this rule fails. Variable name which can be used in the command members is: &SOWNER	N/A
		SET_UACC=	Assign new UACC if this rule fails. Variable name which can be used in the command members is: &SUACC	N/A
		SET_NOTIFY=	Assign new NOTIFY if this rule fails. Variable name which can be used in the command members is: &SNOTIFY	N/A
		SET_AUDIT_OKQUAL=	Assign new audit attributes if this rule fails. Variable name which can be used in the command members is: &SAUDIT_OKQUAL	N/A
		SET_AUDIT_FAQUAL=	Assign new audit if this rule fails. Variable name which can be used in the command members is: &SAUDIT_FAQUAL	N/A
		SET_GAUDIT_OKQUAL=	Assign new global audit if this rule fails. Variable name which can be used in the command members is: &SGAUDIT_OKQUAL	N/A
		SET_GAUDIT_FAQUAL=	Assign new global audit if this rule fails. Variable name which can be used in the command members is: &SGAUDIT_FAQUAL	N/A
		SET_AUDIT_LEVEL=	Assign new audit level if this rule fails. Variable name which can be used in the command members is: &SAUDIT_LEVEL	N/A
		SET_GAUDIT_LEVEL=	Assign new global audit level if this rule fails. Variable name which can be used in the command members is: &SGAUDIT_LEVEL	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		BYPASS_OWNER=(…, …)	Specifies RACF Owner-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_UACC=(…, …)	Specifies RACF UACC(s) to be excluded from the validation process. Max. 8 ID's can be specified.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0400 base record) will be bypassed for further processing.	N/A
		LEVEL=nn or LVL=nn	Specifies a level indicator, where nn is an integer between 00 and 99.	N/A

			Your installation assigns the meaning of the value.	
	Note: • The key fields for rule checking support generic filtering. A key field can also contain the substitution character '?' e.g. DATASET=IBM?A*.** . • To check 'as is' on a complete profile name use the keyword FDATASET. • If all supplied rules fail, the program will create the necessary reports and RACF control cards to alter the profiles. • If CAT=YES is specified, care should be taken to review the reports where RACF profiles are listed indicating that there are no 'real' datasets present. IGGCSI000 from IBM does not always return all datasets (check for any open IBM APARS).			

+CAR - Dataset conditional access record (0402)

The Dataset Conditional Access record defines the users or groups that are allowed to access data. There is one record per dataset/authorization combination.

	+COND_ACCESS_RULE or +CAR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type = 0402	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		DATASET= or DSN=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FDATASET selection.	N/A
		FDATASET= or FDSN=	NON-Generic profile checking: Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 44 bytes.	N/A
		ACCESSID=(… ,…)	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		ACCESS=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A
		BYPASS_ACCESSID=(… ,…)	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_ACCESS=(… , …)	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank. Max. 8 items can be specified.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members	N/A

			are only invoked if a rule fails.	
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the permit/access list. Up to 128 IDs can be specified. - xyz = READ or WRITE etc. e.g. (READ,IBM*) or (*,IBMUSER) You should specify a fully qualified name (file name) when utilizing this keyword. E.g. FDSN= This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A
		CATYPE=	The type of conditional access checking that is being performed. Valid values are APPCPORT, PROGRAM, CONSOLE, TERMINAL, JESINPUT, and SERVAUTH.	N/A
		CANAME=	The name of a conditional access element that is permitted access.	N/A

+AR - Dataset access record (0404)

The Dataset Access record defines the users or groups that are allowed to access data. There is one record per dataset/authorization combination.

	+ACCESS_RULE or +AR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type = 0404	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		DATASET= or DSN=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FDATASET selection.	N/A
		FDATASET= or FDSN=	NON-Generic profile checking: Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 44 bytes.	N/A
		ACCESSID=(… ,…)	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		ACCESS=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A
		BYPASS_ACCESSID=(… ,…)	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_ACCESS=(… , …)	Specifies the access of this data set. Valid values are NONE,	N/A

			EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank. Max. 8 items can be specified.	
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the permit/access list. Up to 128 IDs can be specified. - xyz = READ or WRITE etc. e.g. (READ,IBM*) or (*,IBMUSER) You should specify a fully qualified name (file name) when utilizing this keyword. E.g. FDSN= This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A

Sample: Failing Dataset profiles

```
DEB$SW53-10 DATASETS WHICH FAILED RULES CHECKING         ALS(C) V3R4M1 12/14/05 11.48  RACF VERS 2608                PAGE:         1
                                                                                                                       DATE:2005-12-14
           JOBNAME :XRZP001C STEPNAME:RA2RULES PROCNAME:                                                               TIME:  11:48:34
DATASET NAME                              T AUTHDATE   O OWNER    U ASFASF W E INSTDATA                   RULE NAME(S)/COMMENT
----------------------------------------- - ---------- - -------- - ------ - - -------------------------- --------------------------

A.*                                        U 2003-06-27 U A        N F RN   N N                         'BAD OWNER AND OR  UACC'
A.**                                      +U 2003-06-27 U A        N F RN   N N                         'BAD OWNER AND OR  UACC'
ACFNCP.**                                  G 2001-04-10 G ACFNCP   N F RN   N N ACF/SSP INSTALLATION     'BAD OWNER AND OR  UACC'
ADSM.*.**                                  G 1997-06-10 G SYS1     R F RN   N N                         'BAD OWNER AND OR  UACC'
ANF.*.**                                   G 1997-06-10 G SYS1     R F RN   N N                         'BAD OWNER AND OR  UACC'
```

Field names

Field name	Explanation	Comments
T	U = user dataset name G = group dataset name. A "+" in front indicates if any catalogued datasets exists for a given RACF profile name.	
U	UACC	Defines the universal access authority to be associated with the data sets. The universal access authorities are A=ALTER, C=CONTROL, R=READ, U=UPDATE, E=EXECUTE, and N=NONE.
O	Owner is a group- or user-Id. U = user; G = group	
ASF (first)	Audit attributes A = audit level S = success F= failures	The first character of the attribute will be shown as for the UACC.
ASF (second)	Global audit attributes A = audit level S = success F= failures	The first character of the attribute will be shown as for the UACC.
W	Warning attribute	

E	Erase on scratch attribute	
Rule name	Specifies the rule name, which matched.	

Command generation

For each failing rule it is possible to generate any kind of commands. Below are the variable names listed which can be used in a command member (DDname //COMMANDS). Each time a rule fails and the keyword COMMAND=member_name has been specified, the member will be read from //COMMANDS PDS file and all variables replaced. The output will be written to //DSNC0400. This facility works similar to the ISPF FTINCL function, however it does not support e.g.)SEL,)IM logic.

Variable names filled in by the IRRI0400 record:

The following variables can be used to generate commands related to dataset profiles:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT	SAMPLE DATA
NAME	&R40NAME		BBO.**
VOL	&R40VOL		
GENERIC	&R40GEND	Y OR N (?)	Y
CREATE_DATE	&R40AUTHD	YYYY-MM-DD	
OWNER_ID	&R40OWNER		
LASTREF_DATE	&R40LREF	YYYY-MM-DD	*
LASTCHG_DATE	&R40CREF	YYYY-MM-DD	*
ALTER_CNT	&R40ALTR		
CONTROL_CNT	&R40CNTL		
UPDATE_CNT	&R40UPDT		
READ_CNT	&R40READ		
UACC	&R40UACC		NONE
GRPDS	&R40FLAG1	Y OR N (?)	
AUDIT_LEVEL	&R40AUDIT		
GRP_ID	&R40GRPN		*
DS_TYPE	&R40DSTYP		
LEVEL	&R40DSLVL		
DEVICE_NAME	&R40DTYPX		
GAUDIT_LEVEL	&R40GAUD		*
INSTALL_DATA	&R40DATA		
AUDIT_OKQUAL	&R40AQS		*
AUDIT_FAQUAL	&R40AQF		*
GAUDIT_OKQUAL	&R40GQS		*
GAUDIT_FAQUAL	&R40GQF		*
WARNING	&R40WARN	Y OR N (?)	N
SECLEVEL	&R40SECL		*
NOTIFY_ID	&R40NOTIF		
RETENTION	&R40RETPD		
ERASE	&R40ERASE	Y OR N (?)	
SECLABEL	&R40SECLA		*

Variable names filled in by the failing rule:

OPTIONS KEYWORD	OPTIONS VARIABLE	FORMAT
SET_UACC=VALUE	&SDUACC	MAX. 8 CHAR
SET_OWNER=NAME	&SDOWNER	MAX. 8 CHAR
SET_NOTIFY=NAME	&SDNOTIFY	MAX. 8 CHAR

RULE KEYWORD	RULE VARIABLE	FORMAT
SET_UACC=VALUE	&SUACC	MAX. 8 CHAR
SET_OWNER=NAME	&SOWNER	MAX. 8 CHAR
SET_NOTIFY=NAME	&SNOTIFY	MAX. 8 CHAR

DEB$SW1R – RACF General Resources Verification (RRE)

Purpose:

- Verify RACF general resource profiles.

JCL required to run DEB$SW1R

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SW1R) to create the reports:

```
//RA2VERIF   EXEC PGM=DEB$SW1R
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* SORT WORK AREAS AND OPTIONS
//*
//SYSPRINT     DD   SYSOUT=*
//SYSOUT       DD   SYSOUT=*
//SORTWK01     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK02     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK03     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//SORTWK04     DD   UNIT=SYSDA,SPACE=(CYL,(9,9))
//$ORTPARM DD *
NORC16
//SORTCNTL DD *
 DEBUG         NOABEND
 OPTION        VLSHRT
//*
//* COMMANDS
//*
//COMMANDS   DD   DISP=SHR,DSN=RA2002.V?R?M?.COMMANDS
//*
//* INPUT FILES
//*
//IRRI0100   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0100.VB(0)
//IRRI0200   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0200.VB(0)
//IRRI0500   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0500.VB(0)
//IRRI0503   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0503.VB(0)
//IRRI0505   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0505.VB(0)
//IRRI0507   DD   DISP=SHR,DSN=???????.IRRDBU.IRRI0507.VB(0)
//*
//* OUTPUT FILES
//*
//RESPRINT   DD SYSOUT=*   * PRINT RESNAME_RULE CONTROL STATEMENTS
//RESC0500   DD SYSOUT=*   * RACF RESNAMES - GENERATED COMMANDS
//RESG0500   DD SYSOUT=*   * RACF RESNAMES - MATCHING RULES
//RESF0500   DD SYSOUT=*   * RACF RESNAMES - FAILED RULES
//RESX0500   DD SYSOUT=*   * RACF RESNAMES - NO RULES APPLY
//REST0500   DD SYSOUT=*   * RACF RESNAMES - SUMMARY
//*
//RESC0503   DD SYSOUT=*   * RACF RESNAMES - GENERATED COMMANDS
//RESG0503   DD SYSOUT=*   * RACF RESNAMES - MATCHING RULES
//RESF0503   DD SYSOUT=*   * RACF RESNAMES - FAILED RULES
//RESX0503   DD SYSOUT=*   * RACF RESNAMES - NO RULES APPLY
//REST0503   DD SYSOUT=*   * RACF RESNAMES - SUMMARY
//*
//*
//RESC0505   DD SYSOUT=*   * RACF RESNAMES - GENERATED COMMANDS
//RESG0505   DD SYSOUT=*   * RACF RESNAMES - MATCHING RULES
//RESF0505   DD SYSOUT=*   * RACF RESNAMES - FAILED RULES
//RESX0505   DD SYSOUT=*   * RACF RESNAMES - NO RULES APPLY
//REST0505   DD SYSOUT=*   * RACF RESNAMES - SUMMARY
//*
//RESC0507   DD SYSOUT=*   * RACF RESNAMES - GENERATED COMMANDS
//RESG0507   DD SYSOUT=*   * RACF RESNAMES - MATCHING RULES
//RESF0507   DD SYSOUT=*   * RACF RESNAMES - FAILED RULES
//RESX0507   DD SYSOUT=*   * RACF RESNAMES - NO RULES APPLY
//REST0507   DD SYSOUT=*   * RACF RESNAMES - SUMMARY
//RESRULES   DD *          * RACF BASE USERID RULES
*
* DEFINE RULES FOR GROUP PROFILES RECORD TYPE 0500
* +OPTIONS SPECIFIES THE DEFAULT VALUES TO BE ASSIGNED IF
*  NO RULE DOES MATCH
*
*
+OPTIONS     SET_OWNER=SGRESOWN,SET_UACC=VOGTUACC
+RESNAME_RULE NAME='RA2002 RULE           ',
             CLASS=FACILITY,RESNAME=RA2*.**,OWNER=*,
             SET_OWNER=SETGOWNR,SET_UACC=READ
+RESNAME_RULE NAME='THIS IS A DATASET TES2',
             RESNAME=XRZP00?.**,OWNER=*,
             SET_OWNER=NEWOWNER,SET_UACC=ALTER
+RESNAME_RULE NAME='BAD OWNER AND OR  UACC',
             RESNAME=**,OWNER=OWN*,UACC=NONE,
             SET_OWNER=SETNEW,SET_UACC=NONE
/*
```

DDnames:

- //IRRIxxxx must be RECFM=VB as outlined by the IBM RACF IRRDBU00 offload program. These files are used as input to the program. In case the files are not split by record type then define on all //IRRIxxxx the offloaded RACF database flat file.

  - xxxx = IRRDBU00 record type.
- //COMMANDS must be RECFM=FB, LRECL=80, DSORG=PO. This file contains the product-supplied commands and the user defined commands. The first letter '$' is reserved for product supplied commands.
- //SORTxxxxx DDNAMES are required when the option VERIFY=YES is set.
- //SYSOUT is required by the SORT program
- //???Cxxxx must be RECFM=FB, LRECL=80, DSORG=PS.
  - ??? = USR, GRP, DSN, CON or RES
  - xxxx = IRRDBU00 record type

Note:

Control cards generated by this program must reside in separate flat files and not e.g. in one common PDS, otherwise you will encounter the following ABEND:
IEC143I 213-30,IFG0194D,MYJOBID,RA2VERIF,DSNC0400. This is due to the fact that multiple files are open at the same time to generate control cards.

General Resource Rules (Filter) Control Statements (//RESRULES DD *)

Following control statements can be utilized to perform the RACF general resource verification:

DDname	Verbs	Keywords	Comment	Default
//RESRULES	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	SET_OWNER=	Assign new default owner if all rules fail. Variable name which can be used in the command members is: &SROWNER	N/A
		SET_UACC=	Assign new UACC if all rules fail. Variable name which can be used in the command members is: &SRUACC	N/A
		SET_NOTIFY=	Assign new NOTIFY if all rules fail. Variable name which can be used in the command members is: &SRNOTIFY	N/A
		BYPASS_CLASS=(…, …)	Specifies RACF CLASS(es) to be excluded from the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?CIC* . If defined, the RACF class(es) will be bypassed for further rule verification. It affects +RR BYPASS_CLASS and +AR BYPASS_CLASS rules as this control statement gets processed first. If the BYPASS_CLASS processing fails or is not defined, the remaining BYPASS_CLASS definitions in +RR and +RR will be processed.	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0500 base record) will be bypassed for further processing.	N/A
		SELECT_CLASS= or SC=	Specifies RACF CLASSES to be included for the validation process. Max. 128 classes can be specified.	N/A

			Generic names are supported e.g. ?OWN* . All non-matching records (type=05xx records) will be bypassed for further processing.	
		VERIFY=	Validate access list, owner and notify. Specify YES or NO. To be able to perform access list, owner and notify validation: the group- (IRRI0100) and user-ID (IRRI0200) files are required	NO
		Note: all generated RACF control statements must be reviewed prior executing them. This utility does not automatically update the RACF DB.		

+RR - General resource basic data record (0500)

The General Resource Basic Data record defines the basic information about a general resource. There is one record per general resource profile.

	+RESNAME_RULE or +RESOURCE_RULE or +RES_RULE or +RR or Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type=0500	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		CLASS= OR C=	Specifies a RACF class to be verified.	N/A
		RESNAME= or R=	Generic profile checking: Specifies a RACF general resource to be verified. To check fixed names refer to FRESNAME selection.	N/A
		FRESNAME= or FR= or FRES	NON-Generic profile checking: Specifies a RACF general resource to be verified. The supplied profile name will be checked in its entire length of 246 bytes.	N/A
		OWNER= or O=	Specifies a RACF Owner-ID to be verified.	N/A
		UACC=	Specifies universal access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A
		DATA=YES or NO Or D=	Specifies that installation data must be present.	N/A
		AUDIT_OKQUAL= or AO=	Specifies the successful access audit qualifier to be verified. This is set to blanks if AUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		AUDIT_FAQUAL= or AF=	Specifies the failing access audit qualifier to be verified. This is set to blanks if AUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		GAUDIT_OKQUAL= or	Specifies the auditor-specified	N/A

		GAO=	successful access audit qualifier to be verified. This is set to blanks if GAUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	
		GAUDIT_FAQUAL= or GAF=	Specifies the auditor-specified failing access audit qualifier to be verified. This is set to blanks if GAUDIT_LEVEL is NONE. Otherwise, it is set to either READ, UPDATE, CONTROL, or ALTER.	N/A
		AUDIT_LEVEL= or AL=	Specifies the audit level to be verified. This indicates the level of resource-owner-specified auditing that is performed. Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		GAUDIT_LEVEL= or GAL=	Specifies the global audit level to be verified. This indicates the level of auditor-specified auditing that is performed. Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		AUTHDATE=(yyyy-mm-dd,??) or AUTH_DATE=(yyyy-mm-dd,??)	AUTHDATE= allows to restrict a rule to a given date range. The compare will be done in the length of the supplied date e.g. AUTHDATE=(2000,GE) will only compare the first 4 digits. ??= EQ, GE, GT, LE or LT. AUTHDATE is the date when a profile was 'defined' to RACF.	N/A
		SET_OWNER=	Assign new dataset-owner if this rule fails. Variable name which can be used in the command members is: &SOWNER	N/A
		SET_UACC=	Assign new UACC if this rule fails. Variable name which can be used in the command members is: &SUACC	N/A
		SET_NOTIFY=	Assign new NOTIFY if this rule fails. Variable name which can be used in the command members is: &SNOTIFY	N/A
		SET_AUDIT_OKQUAL=	Assign new audit attributes if this rule fails. Variable name which can be used in the command members is: &SAUDIT_OKQUAL	N/A
		SET_AUDIT_FAQUAL=	Assign new audit if this rule fails. Variable name which can be used in the command members is: &SAUDIT_FAQUAL	N/A
		SET_GAUDIT_OKQUAL=	Assign new global audit if this rule fails. Variable name which can be used in the command members is: &SGAUDIT_OKQUAL	N/A
		SET_GAUDIT_FAQUAL=	Assign new global audit if this rule fails. Variable name which can be used in the command members is: &SGAUDIT_FAQUAL	N/A
		SET_AUDIT_LEVEL=	Assign new audit level if this rule	N/A

			fails. Variable name which can be used in the command members is: &SAUDIT_LEVEL	
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		BYPASS_OWNER=(…, …) or BO=	Specifies RACF Owner-ID(s) to be excluded from the validation process. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_UACC=(…, …)	Specifies RACF UACC(s) to be excluded from the validation process. Max. 8 ID's can be specified.	N/A
		BYPASS_CLASS=(…, …)	Specifies RACF CLASS(es) to be excluded from the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?CIC*	N/A
		SELECT_OWNER= or SO=	Specifies RACF OWNERS to be included for the validation process. Max. 128 owners can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=0500 base record) will be bypassed for further processing.	N/A
		SELECT_CLASS= or SC=	Specifies RACF CLASSES to be included for the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=05xx records) will be bypassed for further processing.	N/A
		LEVEL=nn or LVL=nn	Specifies a level indicator, where nn is an integer between 00 and 99. Your installation assigns the meaning of the value.	N/A
	Note: • The key fields for rule checking support generic filtering. A key field can also contain the substitution character '?' e.g. RESOURCE=IBM?A*.** . • To check 'as is' on a complete profile name use the keyword FRESOURCE. • If all supplied rules fail, the program will create the necessary reports and RACF control cards to alter the profiles.			

+MAR - General resource members record (0503)

The General Resource Members record defines the members of a general resource profile group. There is one record per general resource/member combination.

	+MEMBER_ACCESS_RULE	NAME= or N= or RN=	Specifies a rule name, which	N/A

	or +MAR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type=0503		can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	
		RESNAME= or R= or RES=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FRESNAME selection.	N/A
		FRESNAME= or FR= or FRES=	NON-Generic profile checking: Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 246 bytes.	N/A
		CLASS= OR C=	Specifies a RACF class to be verified.	N/A
		MEMBER_NAME= or MEMBER= or MN= or M=	Specifies a RACF member name to be verified. Generic names are supported e.g. ?CIC* Please refer as well to FMEMBER_NAME= Max. 64 characters.	N/A
		FMEMBER_NAME= or FMEMBER= or FMN= or FM=	NON-Generic member name checking: Specifies a RACF member name to be verified. Please refer as well to MEMBER_NAME= Max. 64 characters.	N/A
		PADS_DATA= or PADS= or PAD=	Specifies the padding characters to be verified. Max. 8 characters.	N/A
		VOL_NAME= or VOLUME= or VOLSER= or VOL= or V=	Specifies the volume serial to be verified. Max. 6 characters.	N/A
		SECLEVEL= or SECL= or SEC=	Specifies the security level to be verified. Max. 3 characters.	N/A
		CATEGORY= or CATEG= or CAT=	Specifies the category to be verified. Max. 3 characters.	N/A
		VM_EVENT_DATA= or VMEVENT= or VME=	Specifies the VM event data to be verified. Max. 5 characters.	N/A
		BYPASS_CLASS=(…, …) or BC=	Specifies RACF CLASS(es) to be excluded from the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?CIC*	N/A
		GLOBAL_ACCESS= or GACC= or GAC=	Specifies the conditional access of this resource to be verified.	N/A

		BYPASS_GLOBAL_ACCESS=(… , …) or or BGAC=	Specifies the conditional access of this resource. Valid values are e.g. NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank. Max. 8 items can be specified.	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		SELECT_CLASS= or SC=	Specifies RACF CLASSES to be included for the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=05xx records) will be bypassed for further processing.	N/A

+AR - General resource access record (0505)

The General Resource Access record defines the users or groups who have specific access to general resources. There is one record per general resource/authorization combination.

	+ACCESS_RULE or +AR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type=0505	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		RESNAME= or R=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FRESNAME selection.	N/A
		FRESNAME= or FR= or FRES=	NON-Generic profile checking: Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 246 bytes.	N/A
		CLASS= or C=	Specifies a RACF class to be verified.	N/A
		BYPASS_CLASS=(…, …) or BC=	Specifies RACF CLASS(es) to be excluded from the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?CIC*	N/A
		ACCESSID=(… ,…) or ACCID= or AC=	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		ACCESS= or ACC= or ACS= or AC=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A

		BYPASS_ACCESSID=(… ,…) or BA=	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_ACCESS=(… , …) or BAC=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank. Max. 8 items can be specified.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the permit/access list. Up to 128 IDs can be specified. - xyz = READ or WRITE etc. e.g. (READ,IBM*) or (*,IBMUSER) You should specify a fully qualified name (file name) when utilizing this keyword. E.g. FRES= This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		SELECT_CLASS= or SC=	Specifies RACF CLASSES to be included for the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=05xx records) will be bypassed for further processing.	N/A

+CAR - General resource conditional access record(0507)

The General Resource Conditional Access record defines the conditional access to a general resource. There is one record per general resource/access combination.

	+COND_ACCESS_RULE or +CAR Note: you can define as many rule statements as required. Make sure the region size is set to e.g. REGION=0M Record type=0507	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		RESNAME= or R=	Generic profile checking: Specifies a RACF dataset to be verified. To check fixed names refer to FRESNAME selection.	N/A
		FRESNAME= or FR= or FRES=	NON-Generic profile checking: Specifies a RACF dataset to be verified. The supplied profile name will be checked in its entire length of 246 bytes.	N/A

		CLASS= or C=	Specifies a RACF class to be verified.	N/A
		BYPASS_CLASS=(…, …) or BC=	Specifies RACF CLASS(es) to be excluded from the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?CIC*	N/A
		ACCESSID=(… ,…) or ACCID= or AC=	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		ACCESS= or ACC= or ACS= or AC=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank.	N/A
		BYPASS_ACCESSID=(… ,…) or BA=	Specifies a RACF access-ID to be verified. Max. 128 ID's can be specified. Generic names are supported.	N/A
		BYPASS_ACCESS=(… , …) or BAC=	Specifies the access of this data set. Valid values are NONE, EXECUTE, READ, UPDATE, CONTROL, and ALTER or blank. Max. 8 items can be specified.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the permit/access list. Up to 128 IDs can be specified. - xyz = READ or WRITE etc. e.g. (READ,IBM*) or (*,IBMUSER) You should specify a fully qualified name (file name) when utilizing this keyword. E.g. FRES= This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A
		COMMAND=	Invoke command member from //COMMANDS if a rule fails. Command members can be used to fix problems. Command members are only invoked if a rule fails.	N/A
		SELECT_CLASS= or SC=	Specifies RACF CLASSES to be included for the validation process. Max. 128 classes can be specified. Generic names are supported e.g. ?OWN* . All non-matching records (type=05xx records) will be bypassed for further processing.	N/A
		CATYPE=	The type of conditional access checking that is being performed. Valid values are CONSOLE, TERMINAL, JESINPUT, SYSID, APPCPORT,SERVAUTH,andPROGRAM.	N/A
		CANAME=	The name of a conditional access	N/A

			element that is permitted access.	

Sample: Member access rule for a general resource

```
+MAR          NAME='GAC RULE        ',
              CLASS=OPERCMDS,
              FRESNAME=MVS.STOP.**,
              GAC=READ,
              BGAC=(ALTER,EXECUTE),
              CATEG=CAT,
              SECL=SEC,
              VOLUME=MYTAPE,
              PADS=******,
              VM_EVENT_DATA=VMEVT,
              COMMAND=,
              DDNAME=
```

Sample: Conditional access rule for a general resource

```
+CAR          NAME='COND_ACCESS',
               CLASS=PROGRAM,
               FRESNAME=DFS*,
               CATYPE=SYSID,
               CANAME=I1,
               RACFIDS=((READ,SYSTCI))
```

Sample: Failing general resources

```
DEB$SW54-10 RESOURCES WHICH FAILED RULES CHECKING        ALS(C) V3R4M1 12/13/05 21.50  RACF VERS 2608                PAGE:         1
                                                                                                                       DATE:2005-12-14
           JOBNAME :XRZP001C STEPNAME:RA2RULES PROCNAME:                                                               TIME:  11:48:34
CLASS    GENERAL RESOURCE NAME                      AUTHDATE   T OWNER    U ASFASF W INSTDATA             RULE NAME(S)/COMMENT
-------- ------------------------------------------ ---------- - -------- - ------ - -------------------- -------------------------

FACILITY RA2002.DEC$CG10                            2001-10-13 U XRZP001  N F RN   N                       'RA2002 FIXED RULE     '


DEB$SW54-10 RESOURCES WHICH FAILED RULES CHECKING        ALS(C) V3R4M1 12/13/05 21.50  RACF VERS 2608                PAGE:         2
                                                                                                                       DATE:2005-12-14
           JOBNAME :XRZP001C STEPNAME:RA2RULES PROCNAME:                                                               TIME:  11:48:34
CLASS    GENERAL RESOURCE NAME                      AUTHDATE   T OWNER    U ASFASF W INSTDATA             RULE NAME(S)/COMMENT
-------- ------------------------------------------ ---------- - -------- - ------ - -------------------- -------------------------


===> TOTAL NUMBER OF RESOURCES  READ   :        11.873



===> TOTAL NUMBER OF RULES      FAILED :            1



===> TOTAL NUMBER OF RULES      MATCHED:           48
```

Field names

Field name	Explanation	Comments
U	UACC	Defines the universal access authority to be associated with the datasets. The universal access authorities are A=ALTER, C=CONTROL, R=READ, U=UPDATE, E=EXECUTE, and N=NONE or NOTRUST (class=DIGICERT), T=TRUST.
O	Owner is a group- or user-Id. U = user; G = group	

ASF (first)	Audit attributes A = audit level S = success F= failures	The first character of the attribute will be shown as for the UACC.
ASF (second)	Global audit attributes A = audit level S = success F= failures	The first character of the attribute will be shown as for the UACC.
W	Warning attribute (Y or N)	
Rule name	Specifies the rule name, which matched.	

Command generation

For each failing rule it is possible to generate any kind of commands. Below are the variable names listed which can be used in a command member (DDname //COMMANDS). Each time a rule fails and the keyword COMMAND=member_name has been specified, the member will be read from //COMMANDS PDS file and all variables replaced. The output will be written to //RESC0500. This facility works similar to the ISPF FTINCL function, however it does not support e.g.)SEL,)IM logic.

Variable names filled in by the IRRI0500 record:

The following variables can be used to generate commands related to general resource profiles:

RACF IRRDBU00 NAME	RA/2 VARIABLE	FORMAT	SAMPLE DATA
NAME	&R50NAME		**
CLASS_NAME	&R50CLASS		
GENERIC	&R50GEND	Y OR N (?)	N
CLASS	&R50CDT		041
CREATE_DATE	&R50AUTHD	YYYY-MM-DD	
OWNER_ID	&R50OWNER		
LASTREF_DATE	&R50LREF	YYYY-MM-DD	2003-01-15
LASTCHG_DATE	&R50CREF	YYYY-MM-DD	2003-01-15
ALTER_CNT	&R50ALTR		00000
CONTROL_CNT	&R50CNTL		00000
UPDATE_CNT	&R50UPDT		00000
READ_CNT	&R50READ		00000
UACC	&R50UACC		READ
AUDIT_LEVEL	&R50AUDIT		
LEVEL	&R50DSLVL		
GAUDIT_LEVEL	&R50GAUD		NONE
INSTALL_DATA	&R50DATA		
AUDIT_OKQUAL	&R50AQS		
AUDIT_FAQUAL	&R50AQF		READ
GAUDIT_OKQUAL	&R50GQS		
GAUDIT_FAQUAL	&R50GQF		
WARNING	&R50WARN	Y OR N (?)	N
SINGLEDS	&R50RESFL	Y OR N (?)	
AUTO	&R50AUTO	Y OR N (?)	
TVTOC	&R50TVTOC	Y OR N (?)	
NOTIFY_ID	&R50NOTIF		
ACCESS_SUN	&R50WDSUN	Y OR N (?)	
ACCESS_MON	&R50WDMON	Y OR N (?)	
ACCESS_TUE	&R50WDTUE	Y OR N (?)	
ACCESS_WED	&R50WDWED	Y OR N (?)	
ACCESS_THU	&R50WDTHU	Y OR N (?)	
ACCESS_FRI	&R50WDFRI	Y OR N (?)	
ACCESS_SAT	&R50WDSAT	Y OR N (?)	
START_TIME	&R50TIMES	HH:MM:SS	
END_TIME	&R50TIMEE	HH:MM:SS	
ZONE_OFFSET	&R50ZONEO		
ZONE_DIRECT	&R50ZONED	Y OR N (?)	
SECLEVEL	&R50SECL		000
APPL_DATA	&R50APPL		
SECLABEL	&R50SECLA		000
STARTHH_TIME	&R50TIHHS	HH:MM:SS	
ENDHH_TIME	&R50TIHHE	HH:MM:SS	

Variable names filled in by the failing rule:

```
OPTIONS KEYWORD              OPTIONS VARIABLE   FORMAT
------------------           ----------------   ----------
SET_UACC=VALUE               &SRUACC            MAX. 8 CHAR
SET_OWNER=NAME               &SROWNER           MAX. 8 CHAR
SET_NOTIFY=NAME              &SRNOTIFY          MAX. 8 CHAR
```

```
RULE KEYWORD                 RULE VARIABLE      FORMAT
------------------           -------------      ----------
SET_UACC=VALUE               &SUACC             MAX. 8 CHAR
SET_OWNER=NAME               &SOWNER            MAX. 8 CHAR
SET_NOTIFY=NAME              &SNOTIFY           MAX. 8 CHAR
```

Command member sample (//COMMANDS):

```
RALTER  &R50CLASS +
         &R50NAME1 +
         OWNER(&SROWNER)
```

Variable names filled in by the IRRI0503 record:

The following variables can be used to generate commands related to general resource profiles (MEMBER access elements):

```
RACF IRRDBU00 NAME           RA/2 VARIABLE   FORMAT       SAMPLE DATA
------------------           -------------   ----------   ------------
NAME                         &R54NAME                     CATEGORY
CLASS_NAME                   &R54CLASS
MEMBER                       &R54MEMBN
MEMBER2                      &R54MEMB2
MEMBER3                      &R54MEMB3
MEMBER4                      &R54MEMB4
GLOBAL_ACC                   &R54GACC
PADS_DATA                    &R54PAD
VOL_NAME                     &R54VOL
VMEVENT_DATA                 &R54VMEVT
SECLEVEL                     &R54SECL
CATEGORY                     &R54CATEG
```

Command member sample (//COMMANDS):

```
RALTER &R54CLASS +
  &R54NAME1 +
  DELMEM(&R54MEMBN)
```

Variable names filled in by the IRRI0505 record:

The following variables can be used to generate commands related to general resource profiles (access elements):

```
RACF IRRDBU00 NAME           RA/2 VARIABLE   FORMAT       SAMPLE DATA
------------------           -------------   ----------   ------------
NAME                         &R55NAME                     ACCT#
CLASS_NAME                   &R55CLASS
AUTH_ID                      &R55GRPUS
ACCESS                       &R55ACS                      READ
```

Command member sample (//COMMANDS):

```
PERMIT &R55NAME1 +
        CLASS(&R55CLASS) ID(&R55GRPUS) DELETE
```

DEB$SR10 - RACF SETROPTS - verification

Purpose:

- Verify the RACF SETROPTS settings

- This feature allows an installation to detect any changes made to e.g. the classes, options. This program makes no modification to the RACF database. Make sure the latest IBM APARs for IRRSEQ00 from 28.2.2006 have been applied, otherwise this program will not work under RACF 7709 or higher.

JCL required to run DEB$SR10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SR10) to create the SETROPTS verification reports:

```
//EXECSETR EXEC PGM=DEB$SR10
//STEPLIB  DD   DISP=SHR,DSN=YOUR.RA2002.LINKLIB         MUST BE APF DEFINED!
//VERPRINT DD   SYSOUT=*
//SETROPTS DD   SYSOUT=*
//SETERROR DD   SYSOUT=*
//SETMATCH DD   SYSOUT=*
//SETTOTAL DD   SYSOUT=*
//VERINPUT DD   *
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+SETROPTS   CLASSACT=(,
ETC.
```

DDnames:

- //VERPRINT lists the control cards (rules) to perform the verfication based on the defined field names. The field names utilized by this program are the same as documented by IBM under the callable function r_admin (setropts).

- //VERINPUT contains the 'rules' to verify the resources.

- //SETROPTS contains a standard SETROPTS LIST output.
- //SETERROR lists all the rules, which failed the verification process.
- //SETMATCH lists all the rules, which passed the verification process.
- //SETTOTAL lists the summary of processed items.

Note:

Each +SETROPTS statement is considered as one rule. You can specify as many rules as required. Only the specified verbs will be compared against the SETROPTS settings.

Verification Rules (Filter) Control Statements (//VERINPUT DD *)

Following control statements can be utilized to perform the RACF SETROPTS verification:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS	HEADING=YES or NO	Print headings (title lines)	YES
		MISSING_RULES=YES or NO	Print items for which no rule was found/defined. If set to 'YES' – all items for which no rule was defined will be printed as an error. Refer to //MODERROR, //SETERROR and //DSNERROR print files.	YES
	+SETROPTS	NAME= OR N=	Specify a rule name (max. 64 chars)	N/A
		ADDCREAT=	YES or NO	N/A

		APPLAUDT=	YES or NO	N/A
		AUDIT=	(classname, ...)	N/A
		CATDSNS	(YES,'value') or NO	N/A
		CLASSACT=	(classname, ...)	N/A
		CLASSTAT=	(classname, ...)	N/A
		CMDVIOL=	YES or NO	N/A
		COMPMODE=	YES or NO	N/A
		EGN=	YES or NO	N/A
		ERASE=	YES or NO	N/A
		ERASEALL=	YES or NO	N/A
		ERASESEC=	(YES,'value') or NO	N/A
		GENCMD=	(classname, ...)	N/A
		GENERIC=	(classname, ...)	N/A
		GENLIST=	(classname, ...)	N/A
		GENOWNER=	YES or NO	N/A
		GLOBAL=	(classname, ...)	N/A
		GRPLIST=	YES or NO	N/A
		HISTORY=	(YES,value) or NO	N/A
		INACTIVE=	(YES, value) or NO	N/A
		INITSTAT=	(classname, ...)	N/A
		INTERVAL=	(YES,value) or NO	N/A
		JESBATCH=	YES or NO	N/A
		JESEARLY=	YES or NO	N/A
		JESNJE=	(YES,'value') or NO	N/A
		JESUNDEF=	(YES,'value') or NO	N/A
		JESXBM=	YES or NO	N/A
		KERBLVL=	(YES,value) or NO	N/A
		LOGALWYS=	(classname, ...)	N/A
		LOGDEFLT=	(classname, ...)	N/A
		LOGFAIL=	(classname, ...)	N/A
		LOGNEVER=	(classname, ...)	N/A
		LOGSUCC=	(classname, ...)	N/A
		MINCHANG=	(YES,value) or NO	N/A
		MIXDCASE=	YES or NO	N/A
		MLACTIVE=	(YES,'value') or NO	N/A
		MLFS=	(YES,'value') or NO	N/A
		MLIPC=	(YES,'value') or NO	N/A
		MLNAMES=	YES or NO	N/A
		MLQUIET=	YES or NO	N/A
		MLS=	(YES,'value') or NO	N/A
		MLSTABLE=	YES or NO	N/A
		MODEL=	YES or NO	N/A
		MODGDG=	YES or NO	N/A
		MODGROUP=	YES or NO	N/A
		MODUSER=	YES or NO	N/A
		OPERAUDT=	YES or NO	N/A
		PREFIX=	(YES,'value') or NO	N/A
		PRIMLANG=	(YES,'value') or NO	N/A
		PROTALL=	(YES,'value') or NO	N/A
		RACLIST=	(classname, ...)	N/A
		REALDSN=	YES or NO	N/A
		RETPD=	(YES,'value') or NO	N/A
		REVOKE=	(YES,value) or NO	N/A
		RULE1=	(YES,'value') or NO	N/A
		RULE2=	(YES,'value') or NO	N/A
		RULE3=	(YES,'value') or NO	N/A
		RULE4=	(YES,'value') or NO	N/A
		RULE5=	(YES,'value') or NO	N/A
		RULE6=	(YES,'value') or NO	N/A
		RULE7=	(YES,'value') or NO	N/A
		RULE8=	(YES,'value') or NO	N/A

		RVARSTPW=	(YES,'value') or NO Note: the value must be defined as 7 characters e.g. (YES,'INSTLN ') until IBM fixes the problem and returns the correct length of this field.	N/A
		RVARSWPW =	(YES,'value') or NO Note: the value must be defined as 7 characters e.g. (YES,'INSTLN ') until IBM fixes the problem and returns the correct length of this field.	N/A
		SAUDIT=	YES or NO	N/A
		SECLABCT=	YES or NO	N/A
		SECLANG=	(YES,'value') or NO	N/A
		SESSINT=	(YES,'value') or NO	N/A
		SLABAUDT=	YES or NO	N/A
		SLBYSYS=	YES or NO	N/A
		SLEVAUDT=	(classname, ...)	N/A
		TAPEDSN=	YES or NO	N/A
		TERMINAL=	(YES,value) or NO	N/A
		WARNING=	(YES,value) or NO	N/A
		WHENPROG=	YES or NO	N/A
		NOTE: When specifying the 'Y' flag, the data supplied in the RULEn field consists of a length field and a character sequence, separated by a blank. The length field can be either a single numeric value, or two numeric values separated by a colon (:) to denote a minimum and maximum length. The character sequence conforms to the format of the output of the SETROPTS LIST command. It is a string of 1 to 8 characters, where each position of the string contains a character that indicates the valid characters that can occupy that position: A - Alphabetic ; C - Consonant ; c - Mixed consonant ; L - Alphanumeric ; m - Mixed numeric ; N - Numeric ; V - Vowel ; v - Mixed vowel ; W - Non-vowel ; * - Any character ; $ - National For example: RULE1 field is specified with field data of RULE1=(YES,'3:6 A*NV*A'). Field names which have not been verified will be listed as well in the output file //SETERROR. This has been implemented to make sure "ALL" existing field names obtained from RACF are examined.		

Sample: control card (rules) input //VERINPUT

```
1VERPRINT-10 CONTROL STATEMENTS (VALIDATE SECURITY OPTIONS)          ALS(C) V3R4M1 02/28/06 04.08   RACF VER:7709      PAGE:         1
                                                                                                                    DATE:2006-02-28
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:  14:55:58

 CONTROL CARD(S) READ VIA //VERINPUT                                    ERROR MESSAGE
 ------------------------------------------------------------------------ --------------------------------------------------

 *-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
 *   VERIFY INSTALLATION STANDARDS
 *-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
 +SETROPTS   CLASSACT=(,
 DATASET,USER,GROUP,$$EQQ3,ACCTNUM,ACICSPCT,APPL,BCICSPCT,
 Etc. . . . ),
  CLASSTAT=,
  GENCMD=(,
 DATASET,ACCTNUM,ACICSPCT,AIMS,ALCSAUTH,APPCLU,
 APPCPORT,APPCSERV,APPCSI,APPCTP,APPL,CACHECLS,
 CBIND,CCICSCMD,CIMS,CONSOLE,CPSMOBJ,CPSMXMP, etc……
 ADDCREAT=YES,
  ADSP=NO,
  CATDSN=YES,
  COMPMODE=NO,
  EGN=YES,
  GENOWNER=YES,
  GRPLIST=YES,
  MLACTIVE=NO,
  MLQUIET=NO,
  MLS=NO,
  MLSTABLE=NO,
  MLNAMES=YES,
  SLBSYS=YES,
  MLIPC=(YES,INACTIVE),
  MLFS=(YES,INACTIVE),
  PREFIX=NO,
  PROTALL=(YES,WARNING),
  REALDSN=NO,
  RETPD=(YES,00000),
  RVARSWPW=(YES,'INSTLN '),
  RVARSTPW=(YES,'INSTLN '),
  SECLABCT=NO,
  SESSINT=(YES,00012),
  TAPEDSN=NO,
  WHENPROG=YES,
  MODGDG=NO,
  MODGROUP=NO,
  MODUSER=NO,
  MODEL=YES,
  ERASE=YES,
  ERASEALL=YES,
  ERASESEC=YES,
  PRIMLANG=(YES,ENU),
  SECLANG=(YES,ENU),
  JESBATCH=YES,
  JESEARLY=NO,
  JESXBM=YES,
  JESNJE=(YES,A???????),
  JESUNDEF=(YES,B+++++++)
```

Output sample: failing rules //SETERROR

```
1SETERROR-10 DEFINED RULES WHICH DO NOT MATCH "SETROPTS" SETTINGS   ALS(C) V3R4M1 02/28/06 04.10   RACF VER:7709      PAGE:         1
                                                                                                                     DATE:2006-02-28
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  14:55:58

 FIELD NAME REQUESTED  VALUE                SETROPTS   CURRENT VALUE(S)     COMMENT/ACTIONS
 ---------- ---------- -------------------- ---------- -------------------- --------------------------------

 GENCMD     VMNODE     * NOT FOUND
            VMPOSIX    * NOT FOUND

            XFACILIT   * NOT FOUND

 GENERIC    VMNODE     * NOT FOUND
            VMPOSIX    * NOT FOUND

            WRITER     * NOT FOUND
            XFACILIT   * NOT FOUND

 GLOBAL     NODES      * NOT FOUND
            SECLABEL   * NOT FOUND

            VXMBR      * NOT FOUND

 AUDIT      VMNODE     * NOT FOUND
            VMPOSIX    * NOT FOUND

            XFACILIT   * NOT FOUND

 LOGDEFLT   VMNODE     * NOT FOUND
            VMPOSIX    * NOT FOUND
            VMRDR      * NOT FOUND


 SESSINT    YES        00012                Y          00030                * ITEM(S) DID NOT MATCH
1SETERROR-10 DEFINED RULES WHICH DO NOT MATCH "SETROPTS" SETTINGS   ALS(C) V3R4M1 02/28/06 04.10   RACF VER:7709      PAGE:         2
                                                                                                                     DATE:2006-02-28
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  14:55:58

 FIELD NAME REQUESTED  VALUE                SETROPTS   CURRENT VALUE(S)     COMMENT/ACTIONS
 ---------- ---------- -------------------- ---------- -------------------- --------------------------------

 JESNJE     YES        A???????             Y          ????????             * ITEM(S) DID NOT MATCH
 JESUNDEF   YES        B+++++++             Y          ++++++++             * ITEM(S) DID NOT MATCH
```

Output sample: SETROPTS Standard list //SETROPTS

```
1SETROPTS-10 STANDARD "SETROPTS" SETTINGS                          ALS(C) V3R4M1 02/28/06 04.10   RACF VER:7709      PAGE:         1
                                                                                                                     DATE:2006-02-28
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  14:55:58

 SETROPTS LIST
 ------------------------------------------------------------------------------------------------------------------------------------

 ATTRIBUTES = INITSTATS WHEN(PROGRAM -- ENHANCED WARNING) SAUDIT CMDVIOL OPERAUDIT
 STATISTICS = NONE
 AUDIT CLASSES = DATASET USER GROUP ACCTNUM ACICSPCT AIMS ALCSAUTH APPCLU
                 APPCPORT APPCSERV APPCSI APPCTP APPL BCICSPCT CACHECLS
                 CBIND CCICSCMD CDT CIMS CONSOLE CPSMOBJ CPSMXMP CSFKEYS
                 CSFSERV DASDVOL DBNFORM DCEUUIDS DCICSDCT DEVICES DIGTCERT
                 DIGTCRIT DIGTNMAP DIGTRING DIMS DIRACC DIRAUTH DIRECTRY
etc
```

Output sample: matching SETROPTS related rules //SETMATCH

```
1SETMATCH-10 RULES WHICH MATCH SETROPTS SETTINGS                    ALS(C) V3R4M1 02/28/06 04.10   RACF VER:7709      PAGE:          1
                                                                                                                      DATE:2006-02-28
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  14:55:58

 FIELD NAME REQUESTED  VALUE                SETROPTS   CURRENT VALUE(S)     COMMENT/ACTIONS
 ---------- ---------- -------------------- ---------- -------------------- --------------------------------

 CLASSACT                                                                   * ALL ITEMS MATCHED
 RACLIST                                                                    * ALL ITEMS MATCHED
 TERMINAL   YES        READ                 Y          READ                 * ITEM MATCHED
 CMDVIOL    YES                             Y                               * ITEM MATCHED
 OPERAUDT   YES                             Y                               * ITEM MATCHED
 SAUDIT     YES                             Y                               * ITEM MATCHED
 APPLAUDT   NO                              N                               * ITEM MATCHED
 SLABAUDT   NO                              N                               * ITEM MATCHED
 KERBLVL    YES        001                  Y          001                  * ITEM MATCHED
 LOGALWYS                                                                   * ALL ITEMS MATCHED
 LOGNEVER                                                                   * ALL ITEMS MATCHED
 HISTORY    YES        012                  Y          012                  * ITEM MATCHED
 INTERVAL   YES        090                  Y          090                  * ITEM MATCHED
 WARNING    YES        005                  Y          005                  * ITEM MATCHED
 REVOKE     YES        006                  Y          006                  * ITEM MATCHED
 RULE1      YES        6:8 ALLLLLLL         Y          6:8 ALLLLLLL         * ITEM MATCHED
 RULE2      NO                              N                               * ITEM MATCHED
 RULE3      NO                              N                               * ITEM MATCHED
 RULE4      NO                              N                               * ITEM MATCHED
 RULE5      NO                              N                               * ITEM MATCHED
```

Output sample: SETROPTS summary //SETTOTAL

```
1SETTOTAL-10 SUMMARY OF PROCESSED SETROPTS RELATED ITEMS            V3R4M3 RACFRA2.COM(C) 12/06/07 RACF VER:7709 MTI2 PAGE:          1
                                                                                                                      DATE:2007-12-06
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  23:42:48
 --------------------------------------------------------------------------------------------------------------------------------------



  ===> TOTAL NUMBER OF SETROPTS ITEMS          READ           :          55
  ===> TOTAL NUMBER OF SETROPTS RULES          READ           :           2  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF SETROPTS RULE/ITEM(S)  FAILED         :          65
  ===> TOTAL NUMBER OF SETROPTS RULE/ITEM(S)  MATCHED        :          53
  ===> TOTAL NUMBER OF SETROPTS RULE/ITEM(S)  UNDEFINED      :           3


 *** END OF LIST
```

Note:

The total of FAILED and/or MATCHED and/or UNDEFINED RULE/ITEM(S) will almost never match the total number of RULES READ or SETROPTS items read from CORE. This is due to the fact, e.g. that a RULE can contain ONE or more SETROPTS keywords to be verified . If for example only some selected items of a keyword match, the rule is counted as ONE error plus the failed items. Hence you may see TOTAL NUMBER OF SETROPTS RULE/ITEM(S) FAILED: 4, but only 3 error messages in the listing. Correct the rule so no errors are listed at all.

DEB$SD10 - Dataset verification (LNK, APF, SMF, LPA and user defined)

Purpose:

- Verify APF-, LNK-, LPA-, SMF- , CATALOG- and user defined datasets

JCL required to run DEB$SD10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SD10) to create the verification reports:

```
//EXECSETR EXEC PGM=DEB$SD10
//STEPLIB  DD   DISP=SHR,DSN=RA2002.V3R6M0.LINKLIB
//DSNSLIST DD   SYSOUT=*
//DSNERROR DD   SYSOUT=*
//DSNMATCH DD   SYSOUT=*
//DSNTOTAL DD   SYSOUT=*
//VERPRINT DD   SYSOUT=*
//VERINPUT DD   *
//
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES,MISSING_RULES=YES,
         BYPASS_APFLIST=NO,
         BYPASS_CATLIST=NO,
         BYPASS_CHKLIST=NO,
         BYPASS_LNKLIST=NO,
         BYPASS_LPALIST=NO,
         BYPASS_SMFLIST=NO
+LPALIST NAME='RACF    ',D=SYS1.ISAMLPA,
                         P=SYS1.ISAM*.**,
                         O=XRZP001,
                         V=OS39R8,
                         U=ALTER
ETC,
```

DDnames:

- //VERPRINT lists the control cards (rules) to perform the verfication based on the defined field names.

- //VERINPUT contains the 'rules' to verify the resources.

- //DSNSLIST contains a standard LIST of the APF-, LNK-, LPA- and SMF datasets. The information is extracted from the internal IBM tables.
- //DSNERROR lists all the rules, which failed the verification process.
- //DSNMATCH lists all the rules, which passed the verification process.
- //DSNTOTAL lists the summary of processed items.

System defined variables can be used for DATASET= and VOLUME= keywords. To obtain defined symbolises use the "D SYMBOLS" system command:

&SYSALVL.
&SYSCLONE.
&SYSNAME.
&SYSPLEX.
&SYSR1.
&CNMNETID.
&CNMTCPN.
&DEVSUP.
&GRSCNF.
&GRSRNL.
&IOS.
&LOGCLS.
&MAXUSR.
&MCAD.

Sample:
+APFLIST N=RAPF100,D=SYS1.LINKLIB,V=&SYSR1.,P=SYS1.*,UACC=READ

Verification Rules (Filter) Control Statements (//VERINPUT DD *)

Following control statements can be utilized to perform the RACF SETROPTS verification:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS	HEADING=YES or NO	Print headings (title lines)	YES
		MISSING_RULES=YES or NO	Print items for which no rule was found/defined. If set to 'YES' – all items for which no rule was defined will be printed as an error. Refer to //MODERROR, //SETERROR and //DSNERROR print files.	YES
		BYPASS_APFLIST=YES OR NO	If set to 'YES' APFLIST rule validation will be bypassed.	NO
		BYPASS_CATLIST=YES OR NO	If set to 'YES' CATLIST rule validation will be bypassed.	NO
		BYPASS_CHKLIST=YES OR NO	If set to 'YES' CHKLSIT rule validation will be bypassed.	NO
		BYPASS_LNKLIST=YES OR NO	If set to 'YES' LNKLIST rule validation will be bypassed.	NO
		BYPASS_LPALIST=YES OR NO	If set to 'YES' LPALIST rule validation will be bypassed.	NO
		BYPASS_SMFLIST=YES OR NO	If set to 'YES' SMFLIST rule validation will be bypassed.	NO
		WARNING=YES or NO	If set, the RACF warning flag is checked. This rule applies to all dataset related rules e.g. +APFLIST, +LNKLIST etc. Normally all RACF profiles should be in NOWARNING mode.	N/A

	+APFLIST OR +LNKLIST OR +LPALIST OR +SMFLIST OR +CHKLIST OR +CATLIST	N[AME]=	Specifies a user defined rule name (max. 64 chars). +APFLIST etc. verifies any defined DATASET= against the active APFLIST, LNKLIST, LPALIST, SMF, CATALOGS etc.. This allows an auditor to verify the current settings versus the defined rules. Note: +CHKLIST N=xxx,D=MY.FILE allows to specify "user defined datasets" which ought to be on your current system. Normally you utilize this statement to make sure e.g. a dataset resides on a given volume and/or has specific RACF attributes. This keyword statement is very helpful since most datasets cannot be found within a system table provided by systems programming.	N/A
		D[ATASET]=	Fully qualified dataset name to be verified. A DATASET= keyword must be present.	N/A
		V[OLSER]= optional	Fully qualified volume name to be verified e.g. V=&SYSR1. or V=SYSRES	N/A
		P[ROFILE]= optional	RACF dataset profile name retrieved must match DATASET name e.g. D=SYS1.LINKLIB,P=SYS1.*.**	N/A
		O[WNER]= optional	OWNER must match RACF profile owner retrieved e.g. OWNER=IBMUSER	N/A
		U[ACC]= optional	UACC must match RACF profile UACC retrieved e.g. UACC=NONE	N/A
		APF= YES or NO optional	DATASET= APF attribute must be YES or NO. APF= will be ignored where not applicable e.g. by +CHKLIST, +SMFLIST.	N/A
		WARNING=YES or NO	If set, the RACF warning flag is checked. There may be cases, where a dataset profile may be in warning mode. Normally no dataset profile should be in warning mode.	N/A

Sample: control card (rules) input //VERINPUT

```
1VERPRINT-10 CONTROL STATEMENTS (VALIDATE SECURITY OPTIONS)         ALS(C) V3R4M1 02/28/06 04.08   RACF VER:7709      PAGE:         1
                                                                                                                    DATE:2006-02-28
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:  14:55:58

 CONTROL CARD(S) READ VIA //VERINPUT                                    ERROR MESSAGE
 ------------------------------------------------------------------------ --------------------------------------------------

 *-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
 *   VERIFY INSTALLATION STANDARDS
 *-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---

+APFLIST N=RAPF100,D=SYS1.LINKLIB,V=&SYSR1.,P=SYS1.*,UACC=READ
+APFLIST N=RAPF101,D=DSN810.SDSNEXIT,P=DSN810.SDSNEXIT
+APFLIST N=RAPF102,D=DSN810.SDSNLINK,P=DSN810.SDSNLINK
+APFLIST N=RAPF103,D=DSN810.SDSNLOAD,P=DSN810.SDSNLOAD
*
*        CHECK LINKLIST
*
+LNKLIST N=RLNK101,D=CSQ531.SCSQANLE,P=CSQ531.SCSQANLE
+LNKLIST N=RLNK102,D=CSQ531.SCSQAUTH,P=CSQ531.SCSQAUTH
+LNKLIST N=RLNK103,D=CSQ531.SCSQLINK,P=CSQ531.SCSQLINK
*
*        CHECK LPA FILES
*
+LPALIST N=RLPA101,D=SYS1.LPALIB,P=SYS1.*,UACC=READ,OWNER=IBMUSER
*
*        CHECK SMF FILES
*
+SMFLIST N=RSMF101,D=SYS1.MAN1,UACC=ALTER,P=SYS1.MAN*.**,V=Z6SYS1
+SMFLIST N=RSMF102,D=SYS1.MAN2,UACC=ALTER,P=SYS1.MAN*.**,V=Z6SYS1
+SMFLIST N=RSMF103,D=SYS1.MAN3,UACC=ALTER,P=SYS1.MAN*.**,V=Z6SYS1
*
*        CHECK OTHER DATASETS
*
+CHKLIST N=RCHK100,D=SYS1.AACBCNTL
+CHKLIST N=RCHK101,D=SYS1.AADFMAC1
+CHKLIST N=RCHK102,D=SYS1.AADRLIB
```

Output sample: //DSNSLIST

```
DSNSLIST-10 DATASET INFORMATION EXTRACTED INTERNALLY "ASIS"       V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 MTI2 PAGE:         1
                                                                                                                    DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:   0:05:39

TYPE     DATASET NAME                                VOLUME O C A PROTECTING RACF PROFILE                    W OWNER    UACC
-------- ------------------------------------------- ------ - - - ------------------------------------------- - -------- --------

APFLIST  COK.SCOKLINK                                OS39R8 Y Y Y COK.*.**                                  N TEST1    READ
APFLIST  CPAC.LINKLIB                                OS39R8 Y Y Y CPAC.*.**                                 N SYS1     READ
APFLIST  CSF.SCSFMOD0                                OS39R8 Y Y Y CSF.**                                    N CSF      NONE
CATLIST  CATALOG.OS390.MASTER                        OS39M1 Y Y   CATALOG.**                                N $$SYSTEM UPDATE

LNKLIST  ASM.SASMMOD1                                OS39R8 Y Y N ASM.*.**                                  N SYS1     READ
LNKLIST  BFS.SBFSLMOD                                OS39R8 Y Y N BFS.*.**                                  N SYS1     READ
LNKLIST  CBC.SCBCCMP                                 OS39R8 Y Y N CBC.*.**                                  N SYS1     READ

LPALIST  SYS1.SISPLPA                                OS39R8 Y Y N SYS1.*.**                                 N P390     ALTER
LPALIST  SYS1.SORTLPA                                OS3R8A Y Y N SYS1.*.**                                 N P390     ALTER
```

Output sample: failed dataset related rules //DSNERROR

```
DSNERROR-10 DEFINED RULES WHICH DO NOT MATCH DATASET NAMES/ATTR.   V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 MTI2 PAGE:         1
                                                                                                                  DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                         TIME:   0:05:39

TYPE     DATASET NAME                             VOLUME A PROTECTING RACF PROFILE                  OWNER    U W COMMENT/RULE NAME
-------- ---------------------------------------- ------ - ---------------------------------------- -------- - - --------------------


*** END OF LIST



DSNERROR-20 DATASETS FOR WHICH NO VALID RULE NAME WAS FOUND        V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                  DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                         TIME:   0:05:39

TYPE     DATASET NAME                             VOLUME A PROTECTING RACF PROFILE                  OWNER    U W COMMENT/RULE NAME
-------- ---------------------------------------- ------ - ---------------------------------------- -------- - - --------------------

APFLIST  ADCD.Z16.LINKLIB                         Z6RES1 Y ADCD.**                                  SYS1     R N
APFLIST  ADCD.Z16.VTAMLIB                         Z6RES1 Y ADCD.**                                  SYS1     R N
APFLIST  CBC.SCLBDLL                              Z6RES1 Y CBC.*                                    IBMUSER  R N
APFLIST  CEE.SCEERUN                              Z6RES2 Y CEE.*                                    IBMUSER  R N
APFLIST  CSF.SCSFMOD0                             Z6RES2 Y CSF.*                                    IBMUSER  R N
```

Output sample: matching dataset related rules //DSNMATCH

```
DSNMATCH-10 DEFINED RULES WHICH MATCH +LNK-, +APF, +LPALIST V3R4M3 RACFRA2.COM(C) 10/18/07 16.01   RACF VER:2608 MTI2 PAGE:         1
                                                                                                                  DATE:2006-10-18
            JOBNAME :XRZP001A STEPNAME:EXECSETR PROCNAME:                                                         TIME:  17:14:52

TYPE     DATASET NAME                             VOLUME A PROTECTING RACF PROFILE                  OWNER    U COMMENT/RULE NAME
-------- ---------------------------------------- ------ - ---------------------------------------- -------- - --------------------

CATLIST  CATALOG.XXXXXX                           XXXXXX   CATALOG.**                              $$SYSTEM U 'CAT1 '
```

Output sample: dataset related summary //DSNTOTAL

```
1DSNTOTAL-10 SUMMARY OF PROCESSED DATASET RELATED ITEMS            V3R4M3 RACFRA2.COM(C) 12/06/07 RACF VER:7709 MTI2 PAGE:        1

                                                                                                                    DATE:2007-12-06
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                            TIME:  23:42:48

 ---------------------------------------------------------------------------------------------------------------------------------



  ===> TOTAL NUMBER OF APFLIST DATASETS       READ           :         94
  ===> TOTAL NUMBER OF APFLIST DATASETS       IN WARNING MODE:         25
  ===> TOTAL NUMBER OF APFLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF APFLIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF APFLIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF APFLIST RULES          UNDEFINED      :         94


  ===> TOTAL NUMBER OF CATALOG DATASETS       READ           :          6
  ===> TOTAL NUMBER OF CATLIST DATASETS       IN WARNING MODE:          0
  ===> TOTAL NUMBER OF CATLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF CATLIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF CATLIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF CATLIST RULES          UNDEFINED      :          6


  ===> TOTAL NUMBER OF CHKLIST DATASETS       IN WARNING MODE:          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF CHKLIST RULES          READ           :          0
  ===> TOTAL NUMBER OF CHKLIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF CHKLIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF CHKLIST RULES          UNDEFINED      :          0


  ===> TOTAL NUMBER OF LNKLIST DATASETS       READ           :         62
  ===> TOTAL NUMBER OF LNKLIST DATASETS       IN WARNING MODE:          0
  ===> TOTAL NUMBER OF LNKLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF LNKLIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF LNKLIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF LNKLIST RULES          UNDEFINED      :         62


  ===> TOTAL NUMBER OF LPALIST DATASETS       READ           :         20
  ===> TOTAL NUMBER OF LPALIST DATASETS       IN WARNING MODE:          0
  ===> TOTAL NUMBER OF LPALIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF LPALIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF LPALIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF LPALIST RULES          UNDEFINED      :         20



  ===> TOTAL NUMBER OF SMFLIST DATASETS       READ           :          3
  ===> TOTAL NUMBER OF SMFLIST DATASETS       IN WARNING MODE:          0
  ===> TOTAL NUMBER OF SMFLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
  ===> TOTAL NUMBER OF SMFLIST RULES          FAILED         :          0
  ===> TOTAL NUMBER OF SMFLIST RULES          MATCHED        :          0
  ===> TOTAL NUMBER OF SMFLIST RULES          UNDEFINED      :          3



 *** END OF LIST
```

DEB$SM10 - Module verification (IKJTSOxx, SVC, SSN, PPT(SCHED=))

Purpose:

- Verify PPT, SSN, SVC, IKJTSOxx definitions.

JCL required to run DEB$SM10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SM10) to create the verification reports:

```
//EXECSETR EXEC PGM=DEB$SM10
//STEPLIB  DD   DISP=SHR,DSN=RA2002.V3R6M0.LINKLIB
//VERPRINT DD   SYSOUT=*
//VERINPUT DD   *
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES,MISSING_RULES=YES,
         BYPASS_AUTHCMD=YES,
         BYPASS_AUTHPGM=YES,
         BYPASS_AUTHTSF=YES,
         BYPASS_NOTBKGND=YES,
         BYPASS_PLATCMD=YES,
         BYPASS_PLATPGM=YES,
         BYPASS_PPTLIST=YES,
         BYPASS_SVCLIST=NO,
         BYPASS_SSNLIST=YES
+SVCLIST N=SVC002,NUMBER=002,ACT=YES,UPD=NO,CNT=0,
               TYPE=1,APF=NO,ESR=NO,NP=NO,ASF=NO,AR=NO,LOCK=YES,
               TEXT='POST'
+SVCLIST N=SVC003,NUMBER=003,ACT=YES,UPD=NO,CNT=0,
               TYPE=1,APF=NO,ESR=NO,NP=NO,ASF=NO,AR=YES,LOCK=YES,
               TEXT='EXIT'
+SVCLIST N=SVC004,NUMBER=004,ACT=YES,UPD=NO,CNT=0,
               TYPE=1,APF=NO,ESR=NO,NP=NO,ASF=NO,AR=NO,LOCK=YES,
               TEXT='GETMAIN'

ETC.

*
+SSNLIST NAME='TESTSSNL',MX=0123456789ABCDEF
+SSNLIST NAME='CSQ1SSNL',MX=C3E2D8F1
+SSNLIST NAME='RACF    ',SSNNAME=RACF,ACTIVE=YES
+SSNLIST NAME='RACF    ',SSNNAME=RACF,ACTIVE=NO
+SSNLIST NAME='RACF    ',SSNNAME=RACF,ACTIVE=
+SSNLIST NAME='RACF    ',SSNNAME=RACF
+SSNLIST NAME='RACF    ',SSNNAME=RACF
+SSNLIST NAME='RACF    ',SSNNAME=RACF
+SSNLIST NAME='CSQ1    ',SSNNAME=CSQ1
+SSNLIST NAME='CSQ2    ',SSNNAME=CSQ2
```

```
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES,MISSING_RULES=YES,
         BYPASS_AUTHCMD=YES,
         BYPASS_AUTHPGM=YES,
         BYPASS_AUTHTSF=YES,
         BYPASS_NOTBKGND=YES,
         BYPASS_PLATCMD=YES,
         BYPASS_PLATPGM=YES,
         BYPASS_PPTLIST=NO,
         BYPASS_SSNLIST=YES
+PPTLIST NAME='ASBSCHWL',M=ASBSCHWL,NSWP=YES,N2LP=YES,DEFLT=YES
+PPTLIST NAME='ATBINITM',M=ATBINITM
+PPTLIST NAME='ATBSDFMU',M=ATBSDFMU
+PPTLIST NAME='AVFMNBLD',M=AVFMNBLD
+PPTLIST NAME='BPEINI00',M=BPEINI00
+PPTLIST NAME='BPXINIT ',M=BPXINIT
+PPTLIST NAME='BPXPINPR',M=BPXPINPR
+PPTLIST NAME='BPXVCLNY',M=BPXVCLNY
+PPTLIST NAME='CBRIIAS ',M=CBRIIAS
+PPTLIST NAME='CBROAM  ',M=CBROAM
+PPTLIST NAME='CNLSSDT ',M=CNLSSDT
+PPTLIST NAME='COFMINIT',M=COFMINIT
+PPTLIST NAME='COFMISDO',M=COFMISDO
```

DDnames:

- //VERPRINT lists the control cards (rules) to perform the verfication based on the defined field names.

- //VERINPUT contains the 'rules' to verify the resources.

- //PPTNLIST contains PPT related information extracted from the IBM tables.
- //PPTERROR lists all the rules, which failed the verification process.
- //PPTMATCH lists all the rules, which passed the verification process.
- //PPTTOTAL lists the summary of processed items.

- //SSNNLIST contains SSN related information extracted from the IBM tables.
- //SSNERROR lists all the rules, which failed the verification process.
- //SSNMATCH lists all the rules, which passed the verification process.
- //SSNTOTAL lists the summary of processed items.

- //SVCNLIST contains SVC related information extracted from the IBM tables.
- //SVCERROR lists all the rules, which failed the verification process.
- //SVCMATCH lists all the rules, which passed the verification process.
- //SVCTOTAL lists the summary of processed items.

- //TSONLIST contains TSO related information extracted from the IBM tables.
- //TSOERROR lists all the rules, which failed the verification process.
- //TSOMATCH lists all the rules, which passed the verification process.
- //TSOTOTAL lists the summary of processed items.

Note:

We suggest you create for each rule type (e.g. +PPTLIST, +TSOLIST) a separate batch job. This makes it simpler to maintain the rules. However a user can keep all 'rules' in one batch job if desired.

//VERINPUT	*	N/A	Comment line	N/A
	+PPTLIST	N[AME]=	Specify a rule name (max. 64 chars)	N/A
		MODNAME= or [PPT]NAME	PPT PROGRAM NAME (max. 8 chars)	N/A but is required
		[PPT]NCNCL=YES or NO	THIS PROGRAM IS NON-CANCELABLE	N/A
		[PPT]SKEY=YES or NO	THIS PROGRAM REQUIRES THE SPECIAL PROTECT. KEY IN PPTKEY	N/A
		[PPT]NSWP=YES or NO	THIS PROGRAM IS TO BE AUTHORIZED TO BE NON-SWAPPABLE	N/A
		[PPT]PRIV=YES or NO	THIS PROGRAM IS TO BE 'PRIVILEGED' (WITH RESPECT TO THE SYSTEM RESOURCES MANAGER)	N/A
		[PPT]SYSTK=YES or NO	THIS PROGRAM IS A SYSTEM TASK	N/A
		[PPT]NDSI=YES or NO	THIS PROGRAM IS NOT TO BE GIVEN DATA SET INTEGRITY	N/A
		[PPT]NOPAS=YES or NO	BYPASS PASSWORD PROTECTION	N/A
				N/A
		[PPT]KEY=value. The value can be 00 to 15	THIS KEY IS TO BE GIVEN TO THE PROGRAM BEING ATTACHED IF PPTSKEY IS ON	N/A
		[PPT]2LPU=YES or NO	2ND LEVEL PREFERRED USAGE	N/A
		[PPT]1LPU=YES or NO	1ST LEVEL PREFERRED USAGE	N/A
		[PPT]N2LP=YES or NO	NOT 2ND LEVEL PREFERRED USAGE	N/A
		[PPT]DEFLT=YES or	FROM IBM SUPPLIED DEFAULT TABLE	N/A

		NO		
		Note: PPTCPUA: BIT MASK OF CPU'S ON WHICH THIS PROGRAM CAN RUN (SHOULD BE X'FFFF' IF AFFINITY IS NOT REQUIRED) Above field will be displayed but is not a selection/verification field. The keywords can be defined without the prefix:PPT e.g. PRIV=YES. Define the rules with the keywords required. Apart from the MODNAME= are all other keywords optional.		

//VERINPUT continued	*	N/A	Comment line	N/A
	+AUTHPGM **+AUTHCMD** **+AUTHTSF** **+PLATPGM** **+PLATCMD** **+NOTBKGND**	N[AME]=	Specify a rule name (max. 64 chars)	N/A
		MODNAME=	IKJTSOxx parmlib member name (max. 8 chars)	N/A but is required
			IKJTSOxx is used at IPL time to define the authorized command list, the authorized program list, the not background command list, the authorized by the TSO service facility list, and to create the defaults the send command will use.	

//VERINPUT continued	*	N/A	Comment line	N/A
	+SSNLIST	N[AME]=	Specify a rule name (max. 64 chars)	N/A
		MODNAME= or For hex values: MODNAMEX= or MX=	Subsystem name (max. 4 chars) Up to 8 hex characters	N/A but is required

//VERINPUT continued	*	N/A	Comment line	N/A
	+SVCLIST	N[AME]=	Specify a rule name (max. 64 chars)	N/A
		[SVC]NUMBER= or SVC_NO=	SVC Number (max. 3 numeric chars). Valid range: 000-255	required
		SVC_A= SVC_ACTIVE= ACTIVE= ACT= A=	Verify if the SVC has to be active or inactive. Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_UPDATED= SVC_U= UPDATED= UPD= UP= U=	Verify if the SVC can be updated. Some SVC's get updated during or after the IPL process e.g. SVC 130. Some 3rd party products as well update the SVC table. Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_COUNT= SVC_C= COUNT= CNT=	Verify the SVC counter (max. 3 numeric chars). Valid range: 000-999	N/A

		C=		
		SVC_TYPE= SVC_T= TYPE= T=	Verify the SVC type (max. 4 numeric chars). e.g.: T=1 or 2 or ¾ or 6	N/A
		SVC_APF= APF=	Verify if the SVC is APF AUTHORIZED. Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_ESR= ESR=	Verify if the SVC is A PART OF THE ESR Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_NP= NP=	Verify if the SVC is NON-PREEMPTIVE Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_ASF= ASF=	Verify if the SVC can be ASSISTED. Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_AR= AR=	Verify if the SVC may be issued in AR ASC mode. Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_LOCK= LOCK=	Verify if the SVC holds any locks(local, cms, opt, salloc or disp). Valid values are: YES or NO. If nothing is defined, this field will not be validated against the SVC table.	N/A
		SVC_TEXT= TEXT=	This keyword allows an installation to overwrite the default description for an SVC. The max. length is 30 characters. This field is helpful for user SVC's e.g. CA/7 etc. The text will be not shown under //SVCNLIST but under //SVCERROR and //SVCMATCH.	N/A

Sample: control card (rules) input //VERINPUT

```
VERPRINT-10 CONTROL STATEMENTS (VALIDATE SECURITY OPTIONS)         V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:   1:19:36

CONTROL CARD(S) READ VIA //VERINPUT                                    ERROR MESSAGE
------------------------------------------------------------------------ --------------------------------------------------

 *** DEB$SM20-10I NOT ALL REQUIRED DDNAMES DEFINED. CHECK JOBLOG FOR DETAILS.


*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES,MISSING_RULES=YES,
         BYPASS_AUTHCMD=YES,
         BYPASS_AUTHPGM=YES,
         BYPASS_AUTHTSF=YES,
         BYPASS_NOTBKGND=YES,
         BYPASS_PLATCMD=YES,
         BYPASS_PLATPGM=YES,
         BYPASS_PPTLIST=NO,
         BYPASS_SSNLIST=YES
+PPTLIST NAME='COFMINIT',M=COFMINIT
+PPTLIST NAME='COFMISDO',M=COFMISDO
+PPTLIST NAME='CQSINIT0',M=CQSINIT0
+SSNLIST NAME='RACF    ',SSNNAME=RACF,ACTIVE=YES
```

Output sample: //PPTNLIST

```
PPTNLIST-10 PPT - MODULE MEMBERS EXTRACTED INTERNALLY "ASIS"       V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:   1:19:36

TYPE     MOD/NAME KEY   NCNCL SKEY  NSWP  PRIV  SYSTK NDSI  NOPAS 2LPU  1LPU  N2LP  DEFLT CPU AFFINITY
-------- -------- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----------------

PPTLIST  AHLGTF     00    Y     Y     Y     N     Y     N     N     N     N     Y     Y   1111111111111111
PPTLIST  AKPCSIEP   01    N     Y     Y     N     Y     Y     N     N     N     Y     Y   1111111111111111
PPTLIST  ANFFIEP    01    N     Y     Y     N     Y     Y     N     N     N     Y     N   1111111111111111
PPTLIST  APSPPIEP   01    N     Y     Y     N     Y     Y     N     N     N     Y     Y   1111111111111111
PPTLIST  ASBSCHIN   01    N     Y     Y     N     Y     N     N     Y     Y     N     Y   1111111111111111
PPTLIST  ASBSCHWL   01    N     Y     N     Y     N     N     N     N     N     N     Y   1111111111111111
PPTLIST  ATBINITM   01    N     Y     Y     N     Y     N     N     Y     Y     N     Y   1111111111111111
PPTLIST  ATBSDFMU   01    N     Y     N     Y     N     N     N     N     N     N     Y   1111111111111111
```

Output sample: failed PPT related rules //PPTERROR

```
PPTERROR-20 PPT - DEFINED RULES WHICH DO NOT MATCH.                V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:
                                                                                                                   DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:   1:19:36

TYPE     MOD/NAME KEY   NCNCL SKEY  NSWP  PRIV  SYSTK NDSI  NOPAS 2LPU  1LPU  N2LP  DEFLT CPU AFFINITY     COMMENT/RULE NAME
-------- -------- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ---------------- -------------------------

PPTLIST  ASBSCHWL                  ?Y                                        Y     Y                   'ASBSCHWL'

*** END OF LIST


PPTERROR-30 PPT - MODULE MEMBERS WITHOUT A DEFINED OR VALID RULE.  V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                           TIME:   1:19:36

TYPE     MOD/NAME KEY   NCNCL SKEY  NSWP  PRIV  SYSTK NDSI  NOPAS 2LPU  1LPU  N2LP  DEFLT CPU AFFINITY
-------- -------- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----------------

PPTLIST  AHLGTF     00    Y     Y     Y     N     Y     N     N     N     N     Y     Y   1111111111111111
PPTLIST  AKPCSIEP   01    N     Y     Y     N     Y     Y     N     N     N     Y     Y   1111111111111111
```

Output sample: matching dataset related rules //PPTMATCH

```
PPTMATCH-10 PPT - MATCHING RULE(S).                              V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     MOD/NAME KEY   NCNCL SKEY  NSWP  PRIV  SYSTK NDSI  NOPAS 2LPU  1LPU  N2LP  DEFLT CPU AFFINITY     COMMENT/RULE NAME
-------- -------- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- ---------------- ------------------------

PPTLIST  ATBINITM   01    N     Y     Y     N     Y     N     N     Y     Y     N     Y   1111111111111111 'ATBINITM'
PPTLIST  ATBSDFMU   01    N     Y     N     Y     N     N     N     N     N     N     Y   1111111111111111 'ATBSDFMU'
PPTLIST  AVFMNBLD   03    Y     Y     Y     N     Y     N     N     N     N     Y     Y   1111111111111111 'AVFMNBLD'
PPTLIST  BPEINI00   07    N     Y     Y     N     Y     N     N     N     N     N     Y   1111111111111111 'BPEINI00'
PPTLIST  BPXINIT    00    Y     Y     Y     N     Y     N     N     N     N     N     Y   1111111111111111 'BPXINIT '
```

Output sample: PPT related summary //PPTTOTAL

```
PPTTOTAL-10 SUMMARY OF PROCESSED PPT RELATED ENTRIES             V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   0:38:48

------------------------------------------------------------------------------------------------------------------------------------



 ===> TOTAL NUMBER OF PPTLIST MEMBERS        READ           :         81
 ===> TOTAL NUMBER OF PPTLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
 ===> TOTAL NUMBER OF PPTLIST RULES          FAILED         :          0
 ===> TOTAL NUMBER OF PPTLIST RULES          MATCHED        :          0
 ===> TOTAL NUMBER OF PPTLIST RULES          UNDEFINED      :         81

*** END OF LIST
```

Output sample: //SSNNLIST

```
SSNNLIST-10 SSN - MEMBER NAMES EXTRACTED  INTERNALLY "ASIS"      V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     MOD/NAME ACTIVE
-------- -------- --------

SSNLIST  CSQ1     Y
SSNLIST  CSQ2     N
SSNLIST  DB8G     Y
SSNLIST  DJ8G     N
SSNLIST  FFST     N
SSNLIST  IRLM     N
SSNLIST  JES2     Y
Etc.
```

Output sample: SSN related summary //SSNTOTAL

```
SSNTOTAL-10 SUMMARY OF PROCESSED SSN RELATED ENTRIES             V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                   DATE:2008-01-26
            JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

------------------------------------------------------------------------------------------------------------------------------------



 ===> TOTAL NUMBER OF SSNLIST MEMBERS        READ           :         23
 ===> TOTAL NUMBER OF SSNLIST RULES          READ           :         34  RULES PROCESSING BYPASSED:Y
 ===> TOTAL NUMBER OF SSNLIST RULES          FAILED         :          0
 ===> TOTAL NUMBER OF SSNLIST RULES          MATCHED        :          0
 ===> TOTAL NUMBER OF SSNLIST RULES          UNDEFINED      :          0

*** END OF LIST
```

Output sample: //SVCNLIST

```
SVCNLIST-10 SVC - MEMBER NAMES EXTRACTED  INTERNALLY "ASIS"       V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                  DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     SVC ACTIVE ENTRY-PT MOD-NAME AREA  DESCRIPTION   (IEASVC00)       UPD CNT U-DATE TYPE APF ESR NP ASF AR LOCK SVCMEMBR
-------- --- ------ -------- -------- ----- ------------------------------ --- --- ------ ---- --- --- -- --- -- ---- --------

SVCLIST  000 Y      80FDD3E0 IECVEXCP NUCL  EXCP/XDAP                      N   000        1    N   N   N  N   N  Y
SVCLIST  001 Y      80FF7114 IEAVEWAT NUCL  WAIT/WAITR/PRTOV               N   000        1    N   N   N  N   N  Y
SVCLIST  002 Y      80FEC028 IEAVEPST NUCL  POST                          N   000        1    N   N   N  N   N  Y
SVCLIST  003 Y      815235A8 IGC003   NUCL  EXIT                          N   000        1    N   N   N  N   Y  Y
SVCLIST  004 Y      8157B032 IGVVSM24 NUCL  GETMAIN,LOC=BELOW             N   000        1    N   N   N  N   N  Y
SVCLIST  005 Y      8157B032 IGVVSM24 NUCL  FREEMAIN,LOC=BELOW            N   000        1    N   N   N  N   N  Y
SVCLIST  006 Y      814C3170 CSVLINK  NUCL  LINK/LINKX                    N   000        2    N   N   N  N   N  Y
SVCLIST  007 Y      814D4008 CSVXCTL  NUCL  XCTL/XCTLX                    N   000        2    N   N   N  N   N  Y
SVCLIST  008 Y      814C3408 CSVLOAD  NUCL  LOAD                          N   000        2    N   N   N  N   N  Y
SVCLIST  009 Y      814C1450 CSVDELET NUCL  DELETE                        N   000        2    N   N   N  N   N  Y
SVCLIST  010 Y      8157BFF8 IGVVSM24 NUCL  GETMAIN/FREEMAIN ,LOC=BELOW    N   000        1    N   N   N  N   N  Y
```

Output sample: failed SVC related rules //SVCERROR

```
SVCERROR-20 SVC - DEFINED RULES WHICH DO NOT MATCH.               V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:
                                                                                                                  DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     SVC ACTIVE MOD-NAME AREA  DESCRIPTION                    UPD CNT TYPE APF ESR NP ASF AR LOCK COMMENT/RULE NAME
-------- --- ------ -------- ----- ------------------------------ --- --- ---- --- --- -- --- -- ---- ------------------------------


*** END OF LIST



SVCERROR-30 SVC - MEMBER NAMES   WITHOUT A DEFINED OR VALID RULE.  V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                  DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     SVC ACTIVE ENTRY-PT MOD-NAME AREA  DESCRIPTION   (IEASVC00)       UPD CNT U-DATE TYPE APF ESR NP ASF AR LOCK
-------- --- ------ -------- -------- ----- ------------------------------ --- --- ------ ---- --- --- -- --- -- ----

SVCLIST  000 Y      80FDD3E0 IECVEXCP NUCL  EXCP/XDAP                      N   000        1    N   N   N  N   N  Y
SVCLIST  001 Y      80FF7114 IEAVEWAT NUCL  WAIT/WAITR/PRTOV               N   000        1    N   N   N  N   N  Y
SVCLIST  002 Y      80FEC028 IEAVEPST NUCL  POST                          N   000        1    N   N   N  N   N  Y
SVCLIST  003 Y      815235A8 IGC003   NUCL  EXIT                          N   000        1    N   N   N  N   Y  Y
SVCLIST  004 Y      8157B032 IGVVSM24 NUCL  GETMAIN,LOC=BELOW             N   000        1    N   N   N  N   N  Y
```

Output sample: matching dataset related rules //SVCMATCH

```
SVCMATCH-10 SVC - MATCHING RULE(S).                               V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                  DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

TYPE     SVC ACTIVE MOD-NAME AREA  DESCRIPTION                    UPD CNT TYPE APF ESR NP ASF AR LOCK COMMENT/RULE NAME
-------- --- ------ -------- ----- ------------------------------ --- --- ---- --- --- -- --- -- ---- ------------------------------


*** END OF LIST
```

Output sample: SVC related summary //SVCTOTAL

```
SVCTOTAL-10 SUMMARY OF PROCESSED SVC RELATED ENTRIES              V3R4M4 RACFRA2.COM(C) 01/25/08 RACF VER:7709 ALS2 PAGE:         1
                                                                                                                  DATE:2008-01-26
           JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:   1:19:36

--------------------------------------------------------------------------------------------------------------------------------



 ===> TOTAL NUMBER OF SVCLIST MEMBERS        READ           :        256
 ===> TOTAL NUMBER OF SVCLIST RULES          READ           :          0  RULES PROCESSING BYPASSED:N
 ===> TOTAL NUMBER OF SVCLIST RULES          FAILED         :          0
 ===> TOTAL NUMBER OF SVCLIST RULES          MATCHED        :          0
 ===> TOTAL NUMBER OF SVCLIST RULES          UNDEFINED      :        256

*** END OF LIST
```

SVCs - A primary source of integrity exposures

The z/OS(MVS) architecture provides one non-privileged instruction which allows an ordinary application program (one which executes in problem state) to pass control to another program which is intended to execute in supervisor state. This specialized instruction is called Supervisor Call or SVC. z/OS(MVS) contains over 100 SVCs, which are defined to pass control to parts of the operating system to perform specialized and/or restricted functions which require the use of privileged instructions and also involve use of system key (0-7) storage areas. MVS also allows a customer to define new SVC's to perform their own specialized functions; such extensions are called **User SVCs**.

Sample: IBM SVC-CODE ASSIGNMENTS

SVC NO	MVS, OS/390, z/OS	SVC NO	MVS, OS/390, z/OS
X'00'	EXCP XDAP	X'80'	
X'01'	PRTOV WAIT WAITR	X'81'	
X'02'	POST	X'82'	RACHECK
X'03'	EXIT	X'83'	RACINIT
X'04'	GETMAIN R,LOC=BELOW	X'84'	RACLIST
X'05'	FREEMAIN R,LOC=BELOW	X'85'	RACDEF
X'06'	LINK LINKX	X'86'	
X'07'	XCTL XCTLX	X'87'	
X'08'	LOAD	X'88'	
X'09'	DELETE	X'89'	ESR
X'0A'	GETMAIN R,LOC=BELOW FREEMAIN LOC=BELOW	X'8A'	PGSER
X'0B'	TIME	X'8B'	CVAF CVAFDIR CVAFDSM CVAFSEQ CVAFVOL CVAFVRF
X'0C'	SYNCH SYNCHX	X'8C'	
X'0D'	ABEND	X'8D'	
X'0E'	SPIE	X'8E'	
X'0F'	ERREXCP	X'8F'	CIPHER EMK (type 4) GENKEY RETKEY
X'10'	PURGE	X'90'	no macro
X'11'	RESTORE	X'91'	
X'12'	BLDL ,,D FIND ,,D	X'92'	BPESVC
X'13'	OPEN	X'93'	
X'14'	CLOSE	X'94'	
X'15'	STOW	X'95'	
X'16'	OPEN TYPE=J	X'96'	
X'17'	CLOSE TYPE=T	X'97'	
X'18'	DEVTYPE	X'98'	
X'19'	TRKBAL	X'99'	
X'1A'	CATALOG INDEX LOCATE	X'9A'	
X'1B'	OBTAIN	X'9B'	
X'1C'		X'9C'	
X'1D'	SCRATCH	X'9D'	
X'1E'	RENAME	X'9E'	
X'1F'	FEOV	X'9F'	
X'20'	ALLOC REALLOC	X'A0'	
X'21'	IOHALT	X'A1'	
X'22'	MGCR MGCRE QEDIT	X'A2'	
X'23'	WTO WTOR	X'A3'	
X'24'	WTL	X'A4'	
X'25'	SEGLD	X'A5'	

	SEGWT		
X'26'		X'A6'	
X'27'	LABEL	X'A7'	
X'28'	EXTRACT	X'A8'	
X'29'	IDENTIFY	X'A9'	
X'2A'	ATTACH ATTACHX	X'AA'	
X'2B'	CIRB	X'AB'	
X'2C'	CHAP	X'AC'	
X'2D'	OVLYBRCH	X'AD'	
X'2E'	STIMERM CANCEL STIMERM TEST TTIMER	X'AE'	
X'2F'	STIMER STIMERM SET	X'AF'	
X'30'	DEQ	X'B0'	
X'31'		X'B1'	
X'32'		X'B2'	
X'33'	SDUMP SDUMPX SNAP SNAPX	X'B3'	
X'34'	RESTART	X'B4'	
X'35'	RELEX	X'B5'	
X'36'	DISABLE	X'B6'	
X'37'	EOV	X'B7'	
X'38'	ENQ RESERVE	X'B8'	
X'39'	FREEDBUF	X'B9'	
X'3A'	RELBUF REQBUF	X'BA'	
X'3B'	OLTEP	X'BB'	
X'3C'	STAE ESTAE STAI ESTAI	X'BC'	
X'3D'	IKJEGS6A (TSO)	X'BD'	
X'3E'	DETACH	X'BE'	
X'3F'	CHKPT	X'BF'	
X'40'	RDJFCB	X'C0'	
X'41'		X'C1'	
X'42'	BTAMTEST	X'C2'	
X'43'		X'C3'	
X'44'	SYSNADAF SYNADRLS	X'C4'	
X'45'	BSP	X'C5'	
X'46'	GSERV	X'C6'	
X'47'	ASGNBFR BUFINQ RLSEBFR	X'C7'	
X'48'	No macro;	X'C8'	
X'49'	SPAR	X'C9'	
X'4A'	DAR	X'CA'	
X'4B'	DQUEUE	X'CB'	
X'4C'	IFBSTAT	X'CC'	
X'4D'		X'CD'	
X'4E'	LSPACE	X'CE'	
X'4F'	STATUS	X'CF'	
X'50'			
X'51'	SETDEV SETPRT		
X'52'			
X'53'	SMFWTM BRANCH=NO SMFEWTM BRANCH=NO		
X'54'	GRAPHICS		
X'55'	DDRSWAP		
X'56'	ATLAS		
X'57'	DOM		
X'58'			
X'59'			
X'5A'			
X'5B'	VOLSTAT		
X'5C'	TCBEXCP TCPEXCP		
X'5D'	TGET TPG TPUT		

X'5E'	GTDEVSIZ GTSIZE RTAUTSRM STATTN STAUTOCP STAUTOLN STAUTSRM STBREAK STCC STCLEAR STCOM STFSMODE STLINENO STSIZE STTIMEOU STTMPPMD STTRAN TCABEND TCLEARQ TCSEND TSEND TSTGTTRM TSTTPMD		
X'5F'	SYSEVENT		
X'60'	STAX		
X'61'	IKJEGS9G		
X'62'	PROTECT		
X'63'	DYNALLOC		
X'64'	IKJEFFIB		
X'65'	QTIP		
X'66'	AQCTL		
X'67'	XLATE		
X'68'	TOPCTL		
X'69'	IMGLIB		
X'6A'			
X'6B'	MODESET		
X'6C'			
X'6D'	ESPIE IFAUSAGE MFDATA (RMF) MFSTART (RMF) MSGDISP OUTADD OUTDEL		
X'6E'			
X'6F'	no macro		
X'70'	PGRLSE		
X'71'	PGANY PGFIX PGFREE PGLOAD PGOUT		
X'72'	EXCPVR		
X'73'			
X'74'	CALLDISP CHNGNTRY IECTATNR IECTCHGA IECTRDTI RESETPL		
X'75'	DEBCHK		
X'76'			
X'77'	TESTAUTH		
X'78'	GETMAIN LOC=ABOVE FREEMAIN LOC=ABOVE		
X'79'	no macro (VSAM)		
X'7A'	EVENTS (type 2) extended LINK extended LOAD extended XCTL Service Processor Call STIMERE VALIDATE		
X'7B'	PURGEDQ		
X'7C'	TPIO		
X'7D'	EVENTS (type 1)		
X'7E'	MSS		
X'7F'			

DEB$OM10 - OMVS(HFS) verification

Purpose:

- List and/or verify OMVS
- Generate OMVS related commands if required

RACF currently provides the IRRDBU00 utility to unload the contents of the RACF database into a Flat File format. No such capability exists for the security data contained within the Hierarchical File System (HFS). To obtain the OMVS(HFS) data utilise “IRRHFSU - the HFS Unload Utility” from IBM, written by Bruce R. Wells.

Where to get the offload utility IRRHFSU?

```
//HFSUNLD EXEC PGM=BPXBATCH,
// PARM='PGM IRRHFSU -F //SYS1.IRRDBU00.OUTPUT /'
//STDERR DD PATH='/U/BRWELLS/HFSUERR',
// PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
// PATHMODE=SIRWXU
```

To obtain detailed information about the IRRHFSU utility please refer to: http://www-03.ibm.com/servers/eserver/zseries/zos/racf/IRRHFSU.html . This tool cannot be found on the racfra2.com product libraries.

In the OS/390 UNIX environment, the Hierarchical File System (HFS) contains files and directories. The security information for these files and directories resides within the file system itself, not within the RACF database. Thus, the RACF Database Unload Utility (IRRDBU00) cannot be used to report on HFS security data.
The IRRHFSU utility will report on the HFS security data in a manner consistent with IRRDBU00. For each file and directory in the currently mounted file system structure, a record will be created which contains security data: permissions bits, owner, audit settings, etc. The format of this record is documented the same way as IRRDBU00 output is documented in OS/390 Security Server (RACF) Macros and Interfaces.
The IRRHFSU utility can be invoked as a UNIX command, or from BATCH using the BPXBATCH program. It can be run against the entire file system, or a list of subtrees within the file system.
The IRRHFSU utility consists of these files:

- documentation for IRRHFSU in PDF format (54K)
- C source code for the utility (24K)
- Sample DB2 load statement (4K)
- Sample DB2 table statements (9K)

You can download these files either by using your browser or by using anonymous file transfer protocol (FTP). From your browser, select "file" and "save as". For anonymous ftp, use the site ftp.software.ibm.com. IRRHFSU can be found in the directory /eserver/zseries/zos/racf/IRRHFSU/. Full installation instructions are in the HFSUnloadReadMe.pdf file.

JCL required to run DEB$OM10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$OM10) to create the verification reports.

```
//EXECSETR EXEC PGM=DEB$OM10
//STEPLIB  DD   DISP=SHR,DSN=RA2002.V3R6M0.LINKLIB
//*
//*        SORTIN= OFFLOADED HFS DIRECTORY INFORMATION
//*
//SORTIN   DD DISP=SHR,DSN=RA2002.MYCORP.IRRIHFS
//SYSOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SORTCNTL DD *
 DEBUG          NOABEND
 OPTION         VLSHRT
//$ORTPARM DD *
NORC16
//*
//SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//SORTWK04 DD   UNIT=SYSDA,SPACE=(CYL,(5,5))
//*
//COMMANDS DD DISP=SHR,DSN=RA2002.MYCORP.COMMANDS
//*
//HFSC0900 DD SYSOUT=*
//HFSC0901 DD SYSOUT=*
//HFSC0902 DD SYSOUT=*
//HFSC0903 DD SYSOUT=*
//*
//HFSE0900 DD SYSOUT=*
//HFSE0901 DD SYSOUT=*
//HFSE0902 DD SYSOUT=*
//HFSE0903 DD SYSOUT=*
//HFSL0900 DD SYSOUT=*
//HFSL0901 DD SYSOUT=*
//HFSL0902 DD SYSOUT=*
//HFSL0903 DD SYSOUT=*
//HFSM0900 DD SYSOUT=*
//HFSM0901 DD SYSOUT=*
//HFSM0902 DD SYSOUT=*
//HFSM0903 DD SYSOUT=*
//HFSTOTAL DD SYSOUT=*
//HFSX0900 DD SYSOUT=*
//HFSX0901 DD SYSOUT=*
//HFSX0902 DD SYSOUT=*
//HFSX0903 DD SYSOUT=*

//VERPRINT DD   SYSOUT=*
//VERINPUT DD   *
*
+OPTIONS HEADING=YES
+INCLUDE_HFSBD FILENAME=**MOKKEG**,RN=INCL2
+INCLUDE_HFACC FILENAME=**SWD**
+HFACC_RULE N=ACCTVOGT,FILENAME=**,
            READ=YES,WRITE=YES,EXEC=YES
//
```

Filter Control Statements (//VERINPUT DD)

To create any reports you must have at least one +INCLUDE or +EXCLUDE statement for each supported record type e.g. HFSBD(0900), HFACC(0901), HFACF(0902) and HFACD(0903).
To create any validation reports based on 'rules' (not roles) you must specify additional control statements e.g. +HFSBD_RULE, +HFACC_RULE, +HFACF_RULE and +HFACD_RULE . These rule statements have the same KEYWORDS as the +INCLUDE/+EXCLUDE statements with one exception, that a rule statement can have a rule name: e.g. NAME= or RN= .

Following control statements can be utilised to obtain the necessary verification reports:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	HEADING=YES or NO	Print headings (title lines)	YES

- **Include/exclude Record Type 0900 – HFS File Basic Data record(s)**

Continued	+INCLUDE_HFSBD and/or +EXCLUDE_HFSBD Note: you can define as many +INCLUDE_ or +EXCLUDE_ statements as required. Make sure the region size is set to e.g. REGION=0M	FILENAME= or FN=	Path name of file or directory	N/A
		INODE=	Inode (file serial number)	N/A
		FILE_TYPE=	What type of file is this? Valid values are FILE, DIR, SOCKET, EXTLINK, SYMLINK, FIFO, BLOCK, and CHAR.	N/A
		OWN_UID=	The owner's z/OS UNIX user identifier (UID) associated with the file.	N/A
		OWN_UNAME=	The owner's RACF user ID	N/A
		OWN_GID=	The owner z/OS UNIX group identifier (GID) associated with the file.	N/A
		OWN_GNAME=	The RACF group name corresponding to this GID	N/A
		S_ISUID=	Is the S_ISGID (set-gid) bit on for this file? Values: YES or NO	N/A
		S_ISGID=	Is the S_ISUID (set-uid) bit on for this file? Values: YES or NO	N/A
		S_ISVTX=	Is the S_ISVTX (sticky) bit on for this file? Values: YES or NO	N/A
		OWN_READ=	Is the owner read bit on for this file? Values: YES or NO	N/A
		OWN_WRITE=	Is the owner write bit on for this file? Values: YES or NO	N/A
		OWN_EXEC=	Is the owner execute bit on for this file? Values: YES or NO	N/A
		GRP_READ=	Is the group read bit on for this file? Values: YES or NO	N/A
		GRP_WRITE=	Is the group write bit on for this file? Values: YES or NO	N/A
		GRP_EXEC=	Is the group execute bit on for this file? Values: YES or NO	N/A
		OTH_READ=	Is the other read bit on for this file? Values: YES or NO	N/A
		OTH_WRITE=	Is the other write bit on for this file? Values: YES or NO	N/A
		OTH_EXEC=	Is the other execute bit on for this file? Values: YES or NO	N/A
		APF=	Is the APF bit on for this file? Values: YES or NO	N/A
		PROGRAM=	Is the program-control bit on for this file? Values: YES or NO	N/A
		SHAREAS=	Is the SHAREAS bit on for this file? Values: YES or NO	N/A
		AAUD_READ=	What are the auditor audit options for READ actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		AAUD_WRITE=	What are the auditor audit options for WRITE actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		AAUD_EXEC=	What are the auditor audit options for EXECUTE actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		UAUD_READ=	What are the user audit options for READ actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		UAUD_WRITE=	What are the user audit options for WRITE actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A
		UAUD_EXEC=	What are the user audit options for EXECUTE actions? Valid values are ALL, SUCCESS, FAIL, and NONE.	N/A

		AUDIT_ID=	RACF audit id	N/A
		FID=	FID	N/A
		CREATE_DATE=	Date the file was created.	N/A
		CREATE_TIME=	Time the file was created.	N/A
		LASTREF_DATE=	Date of last access	N/A
		LASTREF_TIME=	Time of last access	N/A
		LASTCHG_DATE=	Date of last file status change	N/A
		LASTCHG_TIME=	Time of last file status change	N/A
		LASTDAT_DATE=	Date of last data modification	N/A
		LASTDAT_TIME=	Time of last data modification	N/A
		NUMBER_LINKS=	Number of links	N/A
		SHARELIB=	Is the shared library extended attribute bit on for this file? Values: YES or NO	N/A
		ACCESS_ACL=	Does an access ACL exist for this file or directory? Values: YES or NO	N/A
		FILEMOD_ACL=	Does a file default ACL exist for this directory? Values: YES or NO	N/A
		DIRMOD_ACL=	Does a directory default ACL exist for this directory? Values: YES or NO	N/A
		SECLABEL=	The security label (SECLABEL)	N/A
		COMMAND=x or CMD=	Invoke command member from //COMMANDS if a rule fails or an INCLUDE matched. Command members can be used to fix problems. The output will be written to //HFSC0900.	N/A
	Note: 4. Generic names are supported for all KEYWORDS with the exception for field names which can contain only YES or NO. 5. Keywords without an assigned value will be ignored. 6. Date and time fields can be tested for "GT,GE,LT,LE,EQ". e.g. LASTDAT_DATE=(2008-10-09,EQ) or you can still use a generic name e.g. LASTDAT_DATE=(2008-10*)			

- **Include/exclude Record Type 0901 – HFS File Access record(s)**

Continued	+INCLUDE_HFACC and/or +EXCLUDE_HFACC Note: you can define as many +INCLUDE_ or +EXCLUDE_ statements as required. Make sure the region size is set to e.g. REGION=0M	FILENAME= or FN=	Path name of file or directory	N/A
		INODE=	Inode (file serial number)	N/A
		TYPE=	'USER' or 'GROUP'	N/A
		ID=	UID or GID	N/A
		ID_NAME=	RACF user ID or group name	N/A
		READ=	Does the user or group have read access to this file? Values: YES or NO	N/A
		WRITE=	Does the user or group have write access to this file? Values: YES or NO	N/A
		EXEC=	Does the user or group have search/execute access to this file? Values: YES or NO	N/A
		COMMAND=x or CMD=	Invoke command member from //COMMANDS if a rule fails or an INCLUDE matched. Command members can be used to fix problems. The output will be written to //HFSC0901.	N/A

- **Include/exclude Record Type 0902 – HFS File Default Access record(s)**

Continued	+INCLUDE_HFACF and/or +EXCLUDE_HFACF Note: you can define as many +INCLUDE_ or +EXCLUDE_	FILENAME= or FN=	Path name of file or directory	N/A
		INODE=	Inode (file serial number)	N/A
		TYPE=	'USER' or 'GROUP'	N/A
		ID=	UID or GID	N/A
		ID_NAME=	RACF user ID or group name	N/A
		READ=	Does the user or group have read access to this file? Values: YES or NO	N/A

	statements as required. Make sure the region size is set to e.g. REGION=0M	WRITE=	Does the user or group have write access to this file? Values: YES or NO	N/A
		EXEC=	Does the user or group have search/execute access to this file? Values: YES or NO	N/A
		COMMAND=x or CMD=	Invoke command member from //COMMANDS if a rule fails or an INCLUDE matched. Command members can be used to fix problems. The output will be written to //HFSC0902.	N/A

- **Include/exclude Record Type 0903 – HFS Directory Default Access record(s)**

Continued	+INCLUDE_HFACD and/or +EXCLUDE_HFACD Note: you can define as many +INCLUDE_ or +EXCLUDE_ statements as required. Make sure the region size is set to e.g. REGION=0M	FILENAME= or FN=	Path name of file or directory	N/A
		INODE=	Inode (file serial number)	N/A
		TYPE=	'USER' or 'GROUP'	N/A
		ID=	UID or GID	N/A
		ID_NAME=	RACF user ID or group name	N/A
		READ=	Does the user or group have read access to this file? Values: YES or NO	N/A
		WRITE=	Does the user or group have write access to this file? Values: YES or NO	N/A
		EXEC=	Does the user or group have search/execute access to this file? Values: YES or NO	N/A
		COMMAND=x or CMD=	Invoke command member from //COMMANDS if a rule fails or an INCLUDE matched. Command members can be used to fix problems. The output will be written to //HFSC0900.	N/A

- **Rule statement: Record Type 0900 – HFS File Basic Data record**

Continued	+ HFSBD_RULE Note: you can define as many rule statements as required.	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		Note: All other keywords to specify a rule are the same as found under +INCLUDE_HFSBD		

- **Rule statement: Record Type 0901 – HFS File Access record**

Continued	+ HFACC_RULE Note: you can define as many rule statements as required.	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the ACL. Up to 128 IDs can be specified. X = 'R' (read) or '–' (noread) y = 'W' (write) or '–' (nowrite) z = 'X' (execute) or '–' (noexecute) e.g. (RWX,IBM*) Attributes like 'R', 'W', 'X' can be replaced by a '*' if they have not to be checked. E.g. (*-X,AXA*) You should specify a fully qualified path name (file name) when utilizing this keyword.	N/A

			This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	
		Note: All other keywords to specify a rule are the same as found under +INCLUDE_HFACC		

- **Rule statement: Record Type 0902 – HFS File Default Access record**

Continued	+ HFACF_RULE Note: you can define as many rule statements as required.	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the ACL. Up to 128 IDs can be specified. X = 'R' (read) or '–' (noread) y = 'W' (write) or '–' (nowrite) z = 'X' (execute) or '–' (noexecute) e.g. (RWX,IBM*) Attributes like 'R', 'W', 'X' can be replaced by a '*' if they have not to be checked. E.g. (*-X,AXA*) You should specify a fully qualified path name (file name) when utilizing this keyword. This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A
		Note: All other keywords to specify a rule are the same as found under +INCLUDE_HFACF		

- **Rule statement: Record Type 0903 – HFS Directory Default Access record**

Continued	+ HFACD_RULE Note: you can define as many rule statements as required.	NAME= or N= or RN=	Specifies a rule name, which can be up to 32 characters. This rule name will appear on the generated listings as a reference. We recommend assigning for each rule a meaningful name.	N/A
		RACFIDS=((xyz,racfid) ,(xyz,racfid) …. … …)	This keyword allows verifying if a set of RACF IDs (group and or users) exist in the ACL. Up to 128 IDs can be specified. X = 'R' (read) or '–' (noread) y = 'W' (write) or '–' (nowrite) z = 'X' (execute) or '–' (noexecute) e.g. (RWX,IBM*) Attributes like 'R', 'W', 'X' can be replaced by a '*' if they have not to be checked. E.g. (*-X,AXA*) You should specify a fully qualified path name (file name) when utilizing this keyword. This keyword is most valuable to find out if an access list has been changed (no longer matches the installation standards).	N/A
		Note:		

		All other keywords to specify a rule are the same as found under +INCLUDE_HFACD

Sample:

```
//VERINPUT   DD *
*
+OPTIONS HEADING=YES
+INCLUDE_HFSBD FILENAME=**AMXKEG**,RN=INCL1
+INCLUDE_HFSBD FILENAME=**ATIKEG**,RN=INCL2
+INCLUDE_HFSBD FILENAME=**,
        OWN_UNAME=ATIKEG,RN=INCL3
+INCLUDE_HFSBD FILENAME=**,
        OWN_UNAME=AMXKEG,RN=INCL4
+HFSBD_RULE RN=ZURKEG2,
            FILENAME=/HOME/AMXKEG/LDAPSRVE/ACL.LDIF,
            OWN_READ=YES,OWN_WRITE=YES,OWN_EXEC=YES,
            GRP_READ=NO,GRP_WRITE=NO,GRP_EXEC=NO,
            OTH_READ=NO,OTH_WRITE=NO,OTH_EXEC=YES
+HFSBD_RULE RN=MOKKEG3,
            FILENAME=/HOME/ATIKEG/LDAPSRVE/ACL.LDIF,
            OWN_READ=YES,OWN_WRITE=YES,OWN_EXEC=YES,
            GRP_READ=NO,GRP_WRITE=NO,GRP_EXEC=NO,
            OTH_READ=NO,OTH_WRITE=NO,OTH_EXEC=NO
```

```
+INCLUDE_HFSBD FILENAME=**SAMKEG**,RN=INCL2
+INCLUDE_HFACC FILENAME=**SAM**
+INCLUDE_HFACF FILENAME=**SAM**
+INCLUDE_HFACD FILENAME=**SAM**
+HFACC_RULE N=TESTAXA1,FILENAME=/HOME/SYSDPL/SAM,
RACFIDS=((RWX,S*),(RWX,A*),(RWX,VOGT*),
(RWX,3*),(RWX,2*),(RWX,1OGT*))
```

DDNAMES related to the OMVS(HFS) extract and verification process

DDNAME	Description
VERINPUT	Input file - Control statements. DCB=(RECFM=FB,LRECL=80)
VERPRINT	Print file – lists all //VERINPUT control statements. If errors occur, please review this output.
COMMANDS	Input file – contains the template(s) to generate commands based on +INCLUDE and/or RULE statements. //COMMANDS is a PDS file using the DCB format LRECL=80.
HFSC0900	Punch file – generated commands etc. for record type 0900 = HFS FILE BASIC DATA. DCB=(RECFM=FB,LRECL=80)
HFSC0901	Punch file – generated commands etc. for record type 0901 = HFS FILE ACCESS RECORD. DCB=(RECFM=FB,LRECL=80)
HFSC0902	Punch file – generated commands etc. for record type 0902 = HFS FILE DEFAULT ACCESS RECORD. DCB=(RECFM=FB,LRECL=80)
HFSC0903	Punch file – generated commands etc. record type 0903 = HFS DIRECTORY DEFAULT ACCESS RECORD. DCB=(RECFM=FB,LRECL=80)
HFSE0900	Print file – lists failing rules for record type 0900 = HFS FILE BASIC DATA
HFSE0901	Print file – lists failing rules for record type 0901 = HFS FILE ACCESS RECORD
HFSE0902	Print file – lists failing rules for record type 0902 = HFS FILE DEFAULT ACCESS RECORD
HFSE0903	Print file – lists failing rules for record type 0903 = HFS DIRECTORY DEFAULT ACCESS RECORD
HFSM0900	Print file – lists matching rules for record type 0900 = HFS FILE BASIC DATA
HFSM0901	Print file – lists matching rules for record type 0901 = HFS FILE ACCESS RECORD
HFSM0902	Print file – lists matching rules for record type 0902 = HFS FILE DEFAULT ACCESS RECORD
HFSM0903	Print file – lists matching rules for record type 0903 = HFS DIRECTORY DEFAULT ACCESS RECORD
HFSL0900	Print file – lists selected record type 0900 = HFS FILE BASIC DATA
HFSL0901	Print file – lists selected record type 0901 = HFS FILE ACCESS RECORD
HFSL0902	Print file – lists selected record type 0902 = HFS FILE DEFAULT ACCESS RECORD
HFSL0903	Print file – lists selected record type 0903 = HFS DIRECTORY DEFAULT ACCESS RECORD
HFSTOTAL	Print file – lists statistics (total records processed etc.)
HFSX0900	Print file – lists record type 0900 = HFS FILE BASIC DATA, where the RACF-ID is unresolved
HFSX0901	Print file – lists record type 0901 = HFS FILE ACCESS RECORD, where the RACF-ID is unresolved
HFSX0902	Print file – lists record type 0902 = HFS FILE DEFAULT ACCESS RECORD, where the RACF-ID is unresolved
HFSX0903	Print file – list record type 0903 = HFS DIRECTORY DEFAULT ACCESS RECORD, where the RACF-ID is unresolved
SYSOUT	Print file – list SORT messages
SYSIN	Input file to the SORT program, which contains the data of the 'IRRHFSU' program from IBM.
SORTWK01-04	Working files for the SORT program
SORTCNTL	Input file – control statements for the IBM SORT DEBUG NOABEND OPTION VLSHRT
$ORTPARM	Input file – control statements for the SYNSORT

Output Samples:

- **HSFX0900 – unresolved RACF-IDS**

```
1HFSX0900-10 OWNER'S RACF GROUP- OR USERID CANNOT BE MAPPED.        V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2 PAGE:         1
                                                                                                                DATE:2008-02-22
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                      TIME:  16:22:52



 FILETYPE  SUI SGI VTX OWNER-UID  OWNER-GID  OWNGRPOTH AAC FAC DAC SHR-L INODE           CREA.-DATE ACC.-DATE  FSTAT-DATE MODIF-DATE
 SECLABEL  APF SHR PGM RACF-UID   RACF-GID   ARD-AWR-AEX-  URD-UWR-UEX-  FID             CREA.-TIME ACC.-TIME  FSTAT-TIME MODIF-TIME
 --------- --- --- --- ---------- ---------- --------- --- --- --- ----- ---------------- ---------- ---------- ---------- ----------
 DIR       NO  NO  NO  0000000000 0000000100 RWXR-XR-X NO  YES YES NO    0000000003       2000-11-28 2007-12-01 2007-08-14 2007-08-14
           NO  YES NO                       NONENONENONE  FAILFAILFAIL  0000000000000003 09:24:41   09:52:10   13:51:00   13:51:00
/
```

- **HSFX0901 – unresolved RACF-IDS**

```
1HFSX0901-10 RACF GROUP- OR USERID CANNOT BE MAPPED. * FIX-IT*      V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2 PAGE:         1

                                                                                                                DATE:2008-02-22

             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                      TIME:  16:22:52



 PATH NAME                                                      TYPE     UID OR GID RACF-ID  READ WRITE EXEC INODE

 -------------------------------------------------------------- -------- ---------- -------- ---- ----- ---- ----------
 /.setup                                                        GROUP    1000010900          NO   NO    YES  0000000304
```

- **HSFL0900 – selected HFSBD records**

```
1HFSL0900-10 RECORD TYPE 0900 - HFS FILE BASIC DATA RECORD          V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2 PAGE:         1

                                                                                                                DATE:2008-02-22

             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                      TIME:  16:22:52



 FILETYPE  SUI SGI VTX OWNER-UID  OWNER-GID  OWNGRPOTH AAC FAC DAC SHR-L INODE           CREA.-DATE ACC.-DATE  FSTAT-DATE MODIF-DATE
 SECLABEL  APF SHR PGM RACF-UID   RACF-GID   ARD-AWR-AEX-  URD-UWR-UEX-  FID             CREA.-TIME ACC.-TIME  FSTAT-TIME MODIF-TIME
 PATH NAME
 --------- --- --- --- ---------- ---------- --------- --- --- --- ----- ---------------- ---------- ---------- ---------- ----------

 DIR       NO  NO  NO  0000000000 0000000100 RWXR-XR-X NO  NO  NO  NO    0000000001       2007-08-25 2007-11-30 2007-11-30 2007-11-30
           NO  YES NO  TICSA601   FCTOS001   NONENONENONE  FAILFAILFAIL  0000000100000001 17:51:33   07:20:55   12:36:36   12:36:36
 /home/AMXTAT
```

- **HSFL0901 – selected HFACC file access records**

```
1HFSL0901-10 RECORD TYPE 0901 - HFS FILE ACCESS RECORD              V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2 PAGE:         1
                                                                                                                DATE:2008-02-22
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                      TIME:  16:22:52

 PATH NAME                                                      TYPE     UID OR GID RACF-ID  READ WRITE EXEC INODE

 -------------------------------------------------------------- -------- ---------- -------- ---- ----- ---- ----------
 /home/sysdpl/sox                                               USER     1000090301 TICsox01 YES  YES   YES  0000000375
 /home/sysdpl/sox                                               GROUP    1000090410 soxADMIN YES  YES   YES  0000000375
 /home/sysdpl/sox/e                                             USER     1000090301 TICsox01 YES  YES   YES  0000000376
 /home/sysdpl/sox/e                                             GROUP    1000090410 soxADMIN YES  YES   YES  0000000376
```

- **HSFM0901 – matching rules for HFACC file access records**

```
1HFSM0901-10 MATCHING RULES - HFS FILE ACCESS RECORD                V3R4M4 RACFRA2.COM(C) 02/22/08 RACF VER:7709 IBM2 PAGE:         1
                                                                                                                DATE:2008-02-22
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                      TIME:  16:22:52

 PATH NAME                                                      TYPE     UID OR GID RACF-ID  READ WRITE EXEC RULE NAME/COMMENT
 -------------------------------------------------------------- -------- ---------- -------- ---- ----- ---- ----------------------
 /home/sysdpl/sox                                               USER     1000090301 TICsox01 YES  YES   YES  ACCTVOGT
 /home/sysdpl/sox                                               GROUP    1000090410 soxADMIN YES  YES   YES  ACCTVOGT
```

- **HSFE0901 – failed rules for HFACC file access records**

```
1HFSE0901-10 FAILING  RULES - HFS FILE ACCESS RECORD               V3R4M4 RACFRA2.COM(C) 02/22/08 RACF VER:7709 IBM2 PAGE:         1
                                                                                                                    DATE:2008-02-22
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:  16:22:52

 PATH NAME                                                      TYPE     UID OR GID RACF-ID  READ WRITE EXEC RULE NAME/COMMENT
 -------------------------------------------------------------- -------- ---------- -------- ---- ----- ---- ----------------------
 /home/sysdpl/sox/e/a                                           USER     1000090301 TICsox01 YES  YES  ?NO   ACCTVOGT
 /home/sysdpl/sox/e/a                                           GROUP    1000090410 soxADMIN YES  YES  ?NO   ACCTVOGT
```

- **HSFTOTAL – summary of processed records**

```
1HFSTOTAL-10 SUMMARY OF PROCESSED HFS/OMVS RELATED ITEMS           V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2 PAGE:         1
                                                                                                                    DATE:2008-02-22
             JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                          TIME:  16:22:52

 ------------------------------------------------------------------------------------------------------------------------------------

  ===> TOTAL NUMBER OF HFS FILE BASIC DATA            SORTIN         :      2.753  RECORD TYPE=0900
  ===> TOTAL NUMBER OF HFSBD RECORDS FINALLY SELECTED ---->          :         67
  ===> TOTAL NUMBER OF HFSBD INCLUDE STATEMENTS                      :          1
  ===> TOTAL NUMBER OF HFSBD EXCLUDE STATEMENTS                      :          0
  ===> TOTAL NUMBER OF HFSBD RULE STATEMENTS                         :          0
  ===> TOTAL NUMBER OF S_ISUID (SET-UID) BIT ON. . .                 :          0
  ===> TOTAL NUMBER OF S_ISGID (SET-GID) BIT ON. . .                 :          0
  ===> TOTAL NUMBER OF S_ISVTX (STICKY) BIT ON . . .                 :          0
  ===> TOTAL NUMBER OF OWNER READ BIT ON . . . . . .                 :      2.753
  ===> TOTAL NUMBER OF OWNER WRITE BIT ON  . . . . .                 :      2.752
  ===> TOTAL NUMBER OF OWNER EXECUTE BIT ON  . . . .                 :        198
  ===> TOTAL NUMBER OF GROUP READ BIT ON . . . . . .                 :      2.697
  ===> TOTAL NUMBER OF GROUP WRITE BIT ON. . . . . .                 :        858
  ===> TOTAL NUMBER OF GROUP EXECUTE BIT ON. . . . .                 :        160
  ===> TOTAL NUMBER OF OTHER READ BIT ON . . . . . .                 :      2.638
  ===> TOTAL NUMBER OF OTHER WRITE BIT ON  . . . . .                 :        855
  ===> TOTAL NUMBER OF OTHER EXECUTE BIT ON  . . . .                 :        149
  ===> TOTAL NUMBER OF APF BIT ON  . . . . . . . . .                 :          0
  ===> TOTAL NUMBER OF PROGRAM-CONTROL BIT ON. . . .                 :          0
  ===> TOTAL NUMBER OF SHAREAS BIT ON. . . . . . . .                 :      2.753
  ===> TOTAL NUMBER OF SHARED LIBRARY EXT. ATTRIB. ON                :          0
  ===> TOTAL NUMBER OF ACCESS ACL EXISTS . . . . . .                 :      1.161
  ===> TOTAL NUMBER OF FILE DEFAULT ACL EXISTS . . .                 :         20
  ===> TOTAL NUMBER OF DIRECTORY DEFAULT ACL EXISTS.                 :         14

  ===> TOTAL NUMBER OF HFS FILE ACCESS RECORD         SORTIN         :      1.277  RECORD TYPE=0901
  ===> TOTAL NUMBER OF HFACC RECORDS FINALLY SELECTED --->           :        182
  ===> TOTAL NUMBER OF HFACC INCLUDE STATEMENTS                      :          1
  ===> TOTAL NUMBER OF HFACC EXCLUDE STATEMENTS                      :          0
  ===> TOTAL NUMBER OF HFACC RULE STATEMENTS                         :          1
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  READ          :      1.260
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  WRITE         :      1.260
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  EXECUTE       :          1

  ===> TOTAL NUMBER OF HFS FILE DEFAULT ACCESS RECORD SORTIN         :         37  RECORD TYPE=0902
  ===> TOTAL NUMBER OF HFACF RECORDS FINALLY SELECTED --->           :          0
  ===> TOTAL NUMBER OF HFACF INCLUDE STATEMENTS                      :          0
  ===> TOTAL NUMBER OF HFACF EXCLUDE STATEMENTS                      :          0
  ===> TOTAL NUMBER OF HFACF RULE STATEMENTS                         :          0
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  READ          :         24
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  WRITE         :         24
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  EXECUTE       :          1

  ===> TOTAL NUMBER OF HFS DIRECTORY DEFAULT ACCESS   SORTIN         :         29  RECORD TYPE=0903
  ===> TOTAL NUMBER OF HFACD RECORDS FINALLY SELECTED --->           :          0
  ===> TOTAL NUMBER OF HFACD INCLUDE STATEMENTS                      :          0
  ===> TOTAL NUMBER OF HFACD EXCLUDE STATEMENTS                      :          0
```

```
1HFSTOTAL-10 SUMMARY OF PROCESSED HFS/OMVS RELATED ITEMS           V3R4M4 RACFRA2.COM(C) 02/21/08 RACF VER:7709 IBM2sox PAGE:         2
                                                                                                                    DATE:2008-02-22
               JOBNAME :XRZP001C STEPNAME:EXECSETR PROCNAME:                                                        TIME:  16:22:52

------------------------------------------------------------------------------------------------------------------------------------


  ===> TOTAL NUMBER OF HFACD RULE STATEMENTS                         :          0
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  READ          :         16
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  WRITE         :         16
  ===> TOTAL NUMBER OF ACCESS (YES). . . . . .  EXECUTE       :          1

 *** END OF LIST
```

Generating commands

If required a user can generate any kind of commands. This can be very useful e.g. to clean-up OMVS or to perform mass-changes.

Sample: COMMAND=

```
//VERINPUT DD *
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   SELECT ALL HFSBD RECORDS: SET_UID, SET_GID, STICKY BIT AND APF
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+INCLUDE_HFSBD FILENAME=**,CMD=$0900M01
+INCLUDE_HFACC FILENAME=**,CMD=$0901M01
+INCLUDE_HFACF FILENAME=**,CMD=$0902M01
+INCLUDE_HFACD FILENAME=**,CMD=$0903M01
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
//
```

```
//VERINPUT DD *
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   VERIFY INSTALLATION STANDARDS
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+OPTIONS HEADING=YES
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   SELECT ALL HFSBD RECORDS: SET_UID, SET_GID, STICKY BIT AND APF
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
+INCLUDE_HFSBD FILENAME=**
+INCLUDE_HFACC FILENAME=**
+INCLUDE_HFACF FILENAME=**
+INCLUDE_HFACD FILENAME=**
+HFSBD_RULE    FILENAME=**,CMD=$0900M01,
            OWN_READ=YES,OWN_WRITE=YES,OWN_EXEC=YES,
            GRP_READ=NO,GRP_WRITE=NO,GRP_EXEC=NO,
            OTH_READ=NO,OTH_WRITE=NO,OTH_EXEC=YES
+HFACC_RULE    FILENAME=**,CMD=$0901M01,READ=YES,EXEC=YES
+HFACF_RULE    FILENAME=**,CMD=$0902M01,READ=YES,EXEC=YES
+HFACD_RULE    FILENAME=**,CMD=$0903M01,READ=YES,EXEC=YES
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
//
```

Generating commands for record type 0900

The variable names which can be used to generated commands are the same as outlined in the IBM manual or under the +INCLUDE and rule statement of the program DEB$OM10. A sample can be found in the supplied COMMANDS file: Member RA2002.VxRxMx.COMMANDS($0900M01).
Due to the length of the path name and the output line limit, a user may have to use up to 16 variable names for it (&HFSBD_NAME0-F).

- **Template sample**

```
HFSBD_AAUD_EXEC     :&HFSBD_AAUD_EXEC
HFSBD_AAUD_READ     :&HFSBD_AAUD_READ
HFSBD_AAUD_WRITE    :&HFSBD_AAUD_WRITE
HFSBD_ACCESS_ACL    :&HFSBD_ACCESS_ACL
HFSBD_APF           :&HFSBD_APF
HFSBD_AUDITID       :&HFSBD_AUDITID
HFSBD_CREATE_DATE   :&HFSBD_CREATE_DATE
HFSBD_CREATE_TIME   :&HFSBD_CREATE_TIME
HFSBD_DIRMOD_ACL    :&HFSBD_DIRMOD_ACL
HFSBD_FID           :&HFSBD_FID
HFSBD_FILE_TYPE     :&HFSBD_FILE_TYPE
HFSBD_FILEMOD_ACL   :&HFSBD_FILEMOD_ACL
HFSBD_GRP_EXEC      :&HFSBD_GRP_EXEC
HFSBD_GRP_READ      :&HFSBD_GRP_READ
HFSBD_GRP_WRITE     :&HFSBD_GRP_WRITE
HFSBD_INODE         :&HFSBD_INODE
HFSBD_LASTCHG_DATE  :&HFSBD_LASTCHG_DATE
HFSBD_LASTCHG_TIME  :&HFSBD_LASTCHG_TIME
HFSBD_LASTDAT_DATE  :&HFSBD_LASTDAT_DATE
HFSBD_LASTDAT_TIME  :&HFSBD_LASTDAT_TIME
HFSBD_LASTREF_DATE  :&HFSBD_LASTREF_DATE
HFSBD_LASTREF_TIME  :&HFSBD_LASTREF_TIME
HFSBD_NAME          :&HFSBD_NAME
HFSBD_NUMBER_LINKS  :&HFSBD_NUMBER_LINKS
HFSBD_OTH_EXEC      :&HFSBD_OTH_EXEC
HFSBD_OTH_READ      :&HFSBD_OTH_READ
HFSBD_OTH_WRITE     :&HFSBD_OTH_WRITE
HFSBD_OWN_EXEC      :&HFSBD_OWN_EXEC
HFSBD_OWN_GID       :&HFSBD_OWN_GID
HFSBD_OWN_GNAME     :&HFSBD_OWN_GNAME
HFSBD_OWN_READ      :&HFSBD_OWN_READ
HFSBD_OWN_UID       :&HFSBD_OWN_UID
HFSBD_OWN_UNAME     :&HFSBD_OWN_UNAME
HFSBD_OWN_UNAME     :&HFSBD_OWN_UNAME
HFSBD_PROGRAM       :&HFSBD_PROGRAM
HFSBD_S_ISGID       :&HFSBD_S_ISGID
HFSBD_S_ISUID       :&HFSBD_S_ISUID
HFSBD_S_ISVTX       :&HFSBD_S_ISVTX
HFSBD_SECLABEL      :&HFSBD_SECLABEL
HFSBD_SHAREAS       :&HFSBD_SHAREAS
HFSBD_SHARELIB      :&HFSBD_SHARELIB
HFSBD_UAUD_EXEC     :&HFSBD_UAUD_EXEC
HFSBD_UAUD_READ     :&HFSBD_UAUD_READ
HFSBD_UAUD_WRITE    :&HFSBD_UAUD_WRITE
-- --- --- --- --- --- --- --- --- ---
   PATH NAME (16 SLOTS BY 64 BYTES)
-- --- --- --- --- --- --- --- --- ---
HFSBD_NAME          :&HFSBD_NAME0
                    :&HFSBD_NAME1
                    :&HFSBD_NAME2
                    :&HFSBD_NAME3
                    :&HFSBD_NAME4
                    :&HFSBD_NAME5
                    :&HFSBD_NAME6
                    :&HFSBD_NAME7
                    :&HFSBD_NAME8
                    :&HFSBD_NAME9
                    :&HFSBD_NAMEA
                    :&HFSBD_NAMEB
                    :&HFSBD_NAMEC
                    :&HFSBD_NAMED
                    :&HFSBD_NAMEE
                    :&HFSBD_NAMEF
```

Generating commands for record type 0901-903

The variable names which can be used to generated commands are the same as outlined in the IBM manual or under the +INCLUDE and rule statement of the program DEB$OM10. A sample can be found in the supplied COMMANDS file: Member RA2002.VxRxMx.COMMANDS($0901M01, $0902M01, $0903M01).
Due to the length of the path name and the output line limit, a user may have to use up to 16 variable names for it (&HFACC_NAME0-F or &HFACF_NAME0-F or &HFACD_NAME0-F).

Note: for the variable name &HFACC_ID_NAME and &HFACF_ID_NAME and &HFACD_ID_NAME you must utilise &HFACC_IDNAME and &HFACF_IDNAME and &HFACD_IDNAME instead. If you specify *xxxxx_ID_NAME you will receive the data from the field xxxxx_ID!*

- **Template sample**

```
TEST IT 0901
 HFACC_NAME     :&HFACC_NAME.
 HFACC_INODE    :&HFACC_INODE.
 HFACC_TYPE     :&HFACC_TYPE.
 HFACC_ID       :&HFACC_ID.
 HFACC_IDNAME   :&HFACC_IDNAME.
 HFACC_READ     :&HFACC_READ.
 HFACC_WRITE    :&HFACC_WRITE.
 HFACC_EXEC     :&HFACC_EXEC.
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*   PATH NAME (16 SLOTS BY 64 BYTES)
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
 HFACC_NAME         :&HFACC_NAME0
                    :&HFACC_NAME1
                    :&HFACC_NAME2
                    :&HFACC_NAME3
                    :&HFACC_NAME4
                    :&HFACC_NAME5
                    :&HFACC_NAME6
                    :&HFACC_NAME7
                    :&HFACC_NAME8
                    :&HFACC_NAME9
                    :&HFACC_NAMEA
                    :&HFACC_NAMEB
                    :&HFACC_NAMEC
                    :&HFACC_NAMED
                    :&HFACC_NAMEE
                    :&HFACC_NAMEF
```

- **Output sample for type 0900:**

```
/* GENERATED COMMANDS (IF ANY) - DEB$SW10/0900 */
TEST IT 0900 :
 HFSBD_AAUD_EXEC    :NONE
 HFSBD_AAUD_READ    :NONE
 HFSBD_AAUD_WRITE   :NONE
 HFSBD_ACCESS_ACL   :N
 HFSBD_APF          :N
 HFSBD_AUDITID      :01C8C6E2F0F2F2001E07000000000003
 HFSBD_CREATE_DATE  :2000-11-28
 HFSBD_CREATE_TIME  :09:24:41
 HFSBD_DIRMOD_ACL   :Y
 HFSBD_FID          :0000000000000003
 HFSBD_FILE_TYPE    :DIR
 HFSBD_FILEMOD_ACL  :Y
 HFSBD_GRP_EXEC     :Y
 HFSBD_GRP_READ     :Y
 HFSBD_GRP_WRITE    :N
 HFSBD_INODE        :0000000003
 HFSBD_LASTCHG_DATE :2007-08-14
 HFSBD_LASTCHG_TIME :13:51:00
 HFSBD_LASTDAT_DATE :2007-08-14
 HFSBD_LASTDAT_TIME :13:51:00
 HFSBD_LASTREF_DATE :2007-12-01
 HFSBD_LASTREF_TIME :09:52:10
 HFSBD_NAME         :/
 HFSBD_NUMBER_LINKS :0000000025
 HFSBD_OTH_EXEC     :Y
 HFSBD_OTH_READ     :Y
 HFSBD_OTH_WRITE    :N
 HFSBD_OWN_EXEC     :Y
 HFSBD_OWN_GID      :0000000100
 HFSBD_OWN_GNAME    :
 HFSBD_OWN_READ     :Y
 HFSBD_OWN_UID      :0000000000
```

- **Output sample for type 0901:**

```
 /* GENERATED COMMANDS (IF ANY) - DEB$SW10/0901 */
TEST IT 0901
 HFACC_NAME      :/.PROFILE
 HFACC_INODE     :00000003
 HFACC_TYPE      :GROUP
 HFACC_ID        :1000010900
 HFACC_IDNAME    :IHSSRV
 HFACC_READ      :N
 HFACC_WRITE     :N
 HFACC_EXEC      :Y
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
*    PATH NAME (16 SLOTS BY 64 BYTES)
*-- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---
 HFACC_NAME          :/.PROFILE
                     :
                     :
                     :
```

DEB$SO10 - Business-, Application- and systems owner verification

Swiss Re: Every Risk Needs An '**Owner**'. | National Underwriter ... - [Diese Seite übersetzen]
Swiss Re: Every Risk Needs An **Owner**. from National Underwriter Property & Casualty-Risk & Benefits Management in Business provided by Find Articles.
findarticles.com/p/articles/ mi_hb5249/is_200104/ai_n20027237 - 38k -
Im Cache - Ähnliche Seiten

Purpose:

- Create and maintain ownership environment
  - Verify ownership tree: user defined flat file against RACF and vice versa.

Many installations have registered their applications (incl. TSO, IMS, CICS, etc.) in a repository like DB2 or by other means. Such systems can be queried to find out who is in charge (=OWNER) of such applications.

Many RACF installations still do not utilize the OWNER field in RACF profiles (group, user, connect, dataset, general resources) to assign the responsibility of such resources. Hence for large IT installations it is very difficult to find out who is the owner of the RACF resources defined e.g. datasets, general resources, technical user-Ids etc.

The program DEB$SO10 allows an installation, based on user defined information, to create an owner tree in the first place. The owner tree consists always of ONE owner group and ONE connected user-ID. The owner (always a group-ID) shall be defined in the relevant resources e.g. datasets, general resources, technical user-IDs etc. Hence when listing a RACF resource a user can see who is the owner and who is connected to that group-ID. This connected user is responsible for that resource and its access rights.

Main Group	Subgroup	Sub-Subgroup
OWNHOME	OWNHOMEB	Business ownership
	OWNHOMEA	IT application ownership (standard applications e.g. COBOL, PL1, ASM etc.)
	OWNHOMES	IT Systems ownership (system related items e.g. TSO, CICS, IMS, SMS, MQS, DB2, TWS etc.)

In DEB$SO10 we differentiate between three different types of OWNER: BUSINESS, APPLICATION (DEVELOPMENT) and SYSTEMS (Operations, Storage, z/OS, CICS, MQS, IMS, etc.). This means that under normal circumstances for 'application production' files/resources a BUSINESS owner has to be assigned.

- A BUSINESS owner must know who can access/utilize the resources e.g. based on roles.
- An APPLICATION owner is responsible for his/her resources in the development environment.
- A SYSTEMS owner is responsible e.g. for software products e.g. z/OS, TWS, OMEGAMON, RA/2, TSO, MQS, CICS, IMS, WebSphere, HSM/SMS etc.

```
INFORMATION FOR GROUP OWNHOME
     SUPERIOR GROUP=MTI          OWNER=XRZP001
     INSTALLATION DATA=TEST ONWERSHIP
     NO MODEL DATA SET
     TERMUACC
     SUBGROUP(S)= OWNHOMEB OWNHOMES OWNHOMEA
     NO USERS
```

Ownership concept used by DEB$SO10 is as follows:

- The owner field of a RACF profile is always a RACF group-Id and never a RACF userID.
- Each 'owner' group-Id has always only ONE connected user-Id to it. This shows the user who is fully in charge of that profile.
- The user-ID connected to the 'owner' group-Id controls and manages the ACCESS list. This program does not handle any access lists or performs any verifications thereof.

JCL required to run DEB$SO10

Run the following JCL (refer to the RA2002.SAMPLIB member DEB$SO10) to create the verification reports. Before you can execute the following JCL, you must make sure you have prepared an input file which in turn can be utilized by DEB$SO10. Please refer to "How to build your own //OWNI0200 file" below in this section.

```
//APPLOWNR   EXEC PGM=DEB$SO10
//STEPLIB    DD   DISP=SHR,DSN=RA2002.V?R?M?.LINKLIB
//*
//* INPUT FILES
//*
//IRRI0100   DD   DISP=SHR,DSN=MY.IRRI0100
//IRRI0200   DD   DISP=SHR,DSN=MY.IRRI0200
//IRRI0205   DD   DISP=SHR,DSN=MY.IRRI0205
//OWNI0200   DD   DISP=SHR,DSN=MY.YOUR APPLICATION.FILE1
//           DD   DISP=SHR,DSN=MY.YOUR APPLICATION.FILE2
//*
//VERRAUSR DD SYSOUT=*  LIST ALL USERIDS
//VERRAREV DD SYSOUT=*  LIST REVOKED USERIDS
//VERRAPRO DD SYSOUT=*  LIST PROTECTED USERIDS
//VERRAOPR DD SYSOUT=*  LIST OPERATION USERIDS
//VERRANEV DD SYSOUT=*  LIST NEVER USED USERIDS
//VEROWLST DD SYSOUT=*  LIST OWNI0200 "AS IS"
//VERRACON DD SYSOUT=*  LIST CONNECTED USERIDS
//VERRAGRP DD SYSOUT=*  LIST ALL GROUPIDS
//VERAPMIS DD SYSOUT=*  USERIDS MISSING IN RACF
//VERRCADG DD SYSOUT=*  GENNED RACF COMMANDS TO ADD MISSING GRPIDS
//VERGRMIS DD SYSOUT=*  GROUPIDS MISSING IN RACF
//VERRCDEG DD SYSOUT=*  RACF COMMANDS TO DELETE MISSING USERIDS/GRPIDS
//VEROWMIS DD SYSOUT=*  APPLICATIONS MISSING IN OWNI0200
//*
//VERPRINT   DD SYSOUT=*   * PRINT CONTROL STATEMENTS
//VERINPUT   DD *
```

DDNAMES related to the OWNERSHIP verification process

DDNAME	Description
VERGRMIS	Print file – lists missing RACF group-Ids. Such group-Ids are defined in the //OWNI0200 dataset but could not be located in //IRRI0100.
VERINPUT	Input file - Control statments
VEROWLST	Print file – lists all items "AS IS" based on //OWNI0200.
VEROWMIS	Print file – lists missing application-Ids. Such application-Ids exist in RACF e.g. OWNA???? But no such id could be found in //OWNI0200.
VERPRINT	Print file – lists all //VERINPUT control statements. If an error occurred please review this output.
VERRACON	Print file – lists all items "AS IS" based on //OWNI0205. Connected user-Ids.
VERRAGRP	Print file – lists all items "AS IS" based on //OWNI0100. Group-Ids.
VERRANEV	Print file – lists all userids "AS IS" based on //IRRI0200, which have the attribute 'never used'.
VERRAOPR	Print file – lists all userids "AS IS" based on //IRRI0200, which have the attribute 'operations'.
VERRAPRO	Print file – lists all userids "AS IS" based on //IRRI0200, which have the attribute 'protected'.
VERRAREV	Print file – lists all userids "AS IS" based on //IRRI0200, which have the attribute 'revoked'.
VERRAUSR	Print file – lists all userids "AS IS" based on //IRRI0200
VERRCADG	Punch file – contains AG RACF commands due to missing group-Ids.
VERRCDEG	Punch file – contains DG RACF commands due to missing group-Ids.
VERRPMIS	Print file – lists missing RACF user-Ids. Such user-Ids are defined in the //OWNI0200 dataset but could not be located in //IRRI0200.

How to build your own //OWNI0200 file?

RRE does not know any of your application systems, as the inventory may not even reside on the IBM Host. You can build e.g. via REXX or DB2 searches or by other means the //OWNI0200 input file.

The //OWNI0200 file must have the same record format as the IRRDBU00 from IBM: RECFM=VB, LRECL=4096. The record layout is as follows:

Pos. 1 – 4	record type	BASO	This is a fix value.
Pos. 5	reserved		
Pos. 6 – 13	Application prefix	e.g. AXA	
Pos. 14	reserved		
Pos. 15 – 22	Business owner	e.g. IBMUSER	
Pos. 23	reserved		
Pos. 24 – 31	Application owner	e.g. IBMUSER	
Pos. 32	reserved		
Pos. 33 – 40	System owner	e.g. IBMUSER	
Pos. 41	reserved		
Pos. 42 – 63	Application Status	e.g. ACTIVE	
Pos. 64	reserved		
Pos. 65 – 96	Description		

```
****** ******************************* Top of Data ********************************
=COLS> ----+----1----+----2----+----3----+----4----+----5----+----6----+----7--
000001 BASO ARC      TTTHRG   TTTCYD   ######## In Productive use
000002 BASO BGP      MAXRUL   MAXPKI   ######## In Productive use
****** ***************************** Bottom of Data ******************************
```

Filter Control Statements (//VERINPUT DD)

Following control statements can be utilized to obtain the necessary HR versus RACF verification reports:

DDname	Verbs	Keywords	Comment	Default
//VERINPUT	*	N/A	Comment line	N/A
	+OPTIONS Note: only one statement allowed	HEADING=YES or NO	Print headings (title lines)	YES
		SEL_GROUPID=	Process specific group-Ids from theinput file //IRRI0205. Normally you would set this to SEL_GROUPID=OWN* as you are only interested in OWNA*, OWNB* and OWNS* connect records. In case you utilize different prefixes for application, business or systems, then you can use other filtering statements e.g. SEL_B_GROUPID, SEL_A_GROUPID or SEL_S_GROUPID statements.	Blanks=all.
		SEL_B_GROUPID=	Filter RACF connect records for further processing.	* use this statement if the global filter SEL_GROUPID= cannot be used.
		SEL_A_GROUPID=	Filter RACF connect records for further processing.	* use this statement if the global filter SEL_GROUPID= cannot be used.
		SEL_S_GROUPID=	Filter RACF connect records for further processing.	* use this statement if the global filter SEL_GROUPID= cannot be used.

		SET_B_GROUPID=	Assign a group-ID prefix in case an application prefix exists in //OWNI0200. In such a case the program will generate the necessary ADDGROUP commands e.g. OWNBxxxx.	* e.g. OWNB
		SET_A_GROUPID=	Assign a group-ID prefix in case an application prefix exists in //OWNI0200. In such a case the program will generate the necessary ADDGROUP commands e.g. OWNAxxxx.	* e.g. OWNA
		SET_S_GROUPID=	Assign a group-ID prefix in case an application prefix exists in //OWNI0200. In such a case the program will generate the necessary ADDGROUP commands e.g. OWNSxxxx.	* e.g. OWNS
		SET_B_HOMEGRP=	Assign superior group and owner name for the ADDGROUP command. This relates to the RACF group which contains all business owner group-IDS. (used in file //VERRCADG)	
		SET_A_ HOMEGRP =	Assign superior group and owner name for the ADDGROUP command. This relates to the RACF group which contains all application owner group-IDS. (used in file //VERRCADG)	
		SET_S_ HOMEGRP =	Assign superior group and owner name for the ADDGROUP command. This relates to the RACF group which contains all systems owner group-IDS. (used in file //VERRCADG)	
	+IC (include connect records) This control statement allows a user to define 'additional' filter statements. This applies only to the input file //IRRI0205.	USERID=	Select a user-ID. Generic Ids are supported incl. The '?' as substitution character.	Blanks=all
		GROUPID=	Select a group-ID. Generic Ids are supported incl. The '?' as substitution character.	Blanks=all
	+IA (include application records) This control statement	PREFIX=	Select an application prefix-ID. Generic Ids are supported incl. The '?' as substitution character. E.G. 1001	Blanks=all

	allows a user to filter/select specific application prefix records. This applies only to the input file //OWNI0200.	B_OWNER=	Select record by business user-ID. Generic Ids are supported incl. The '?' as substitution character. E.G. IBM*	Blanks=all
		A_OWNER=	Select record by application user-ID. Generic Ids are supported incl. The '?' as substitution character. E.G. IBM*	Blanks=all
		S_OWNER=	Select record by systems user-ID. Generic Ids are supported incl. The '?' as substitution character. E.G. IBM*	Blanks=all
	* For B_, A_ and S_OWNER refer as well to the record layout "How to build your own //OWNI0200 file".			

Input Sample //VERINPUT:

```
//VERINPUT   DD *
*
* SELECT ALL RECORDS FROM //IRRI0200 AND SET PREFIXES FOR COMMANDS
+OPTIONS HEADING=YES,
         SET_B_GROUPID=OWNB,SET_A_GROUPID=OWNA,SET_S_GROUPID=OWNS,
         SEL_B_GROUPID=*,SEL_A_GROUPID=*,SEL_S_GROUPID=*
* SELECT ALL RECORDS FROM //OWNI0200
+IA      PREFIX=*,B_OWNER=*,A_OWNER=*,S_OWNER=*
```

Output Sample //VERRAUSR:

```
DEB$SO14-10 RACF IRRDBU00 TYPE 0200 USER RECORDS (ALL)   V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608                PAGE:         1
                                                                                                                     DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                             TIME:  16:04:11
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
$1234567  SCHNITTSTELLEN USER 2001-09-04 OWNAALG  N N N N Y      USRTEC01 2008-06-11 00:00:12 JDBC-ZUGRIFF STARNET OLTP INFRASTR. VE
```

Output Sample //VERRAREV:

```
DEB$SO30-10 RACF IRRDBU00 TYPE 0200 REVOKED USERIDS      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608                PAGE:         1
                                                                                                                     DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                             TIME:  16:04:11
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
$1234567  TECHN. USERID       2002-10-29 OWNS1006 N N N Y N      USRTEC01 2005-04-08 02:05:22         INTERFACE FOR PARS FROM DB2 TO
```

Output Sample //VERRAPRO:

```
DEB$SO32-10 RACF IRRDBU00 TYPE 0200 PROTECTED USER-IDS   V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:         1
                                                                                                                     DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                             TIME:  16:04:11
USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
-------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
$12345   TWS PRODUCTION USER  1995-09-21 OWNS1102 P N Y N N      USRTECH  2008-06-11 07:31:13 TWS
```

Output Sample //VERRAOPR:

```
 DEB$SO33-10 RACF IRRDBU00 TYPE 0200 OPER/SPEC USER-IDS   V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:         1
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
 -------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
 $12345   TWS PRODUCTION USER  1995-09-21 OWNS1102 P N Y N N      USRTECH  2008-06-11 07:31:13 TWS
 TARM01   Nanno DAVID          1995-10-17 USRPERS1 N Y N N N      USRPERS1 2008-06-10 14:48:49

Etc.

 DEB$SO33-10 RACF IRRDBU00 TYPE 0200 OPER/SPEC USER-IDS   V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:         2
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
 -------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------


 ===> TOTAL NUMBER OF USER-IDS LISTED    :             9

 ===> TOTAL NUMBER OF USER-IDS PROTECTED :             2

 ===> TOTAL NUMBER OF USER-IDS SPECIAL   :             7

 ===> TOTAL NUMBER OF USER-IDS OPERATIONS:             2

 ===> TOTAL NUMBER OF USER-IDS REVOKED   :             0

 ===> TOTAL NUMBER OF USER-IDS NEVER USED:             0
```

Output Sample //VERRANEV:

```
 DEB$SO31-10 RACF IRRDBU00 TYPE 0200 NEVER USED USER-IDS  V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:         1
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 USERID   USER NAME            AUTHDATE   OWNER    P S O R G ATTR DFLTGRP. LAST-LOGON TIME     INSTALLATION DATA
 -------- -------------------- ---------- -------- - - - - - ---- -------- ---------- -------- ----------------------------------------
 irrcerta CERTAUTH Anchor      2000-02-19 irrcerta N N N Y N
 irrmulti Criteria Anchor      2001-01-13 irrmulti N N N Y N
 irrsitec SITE Anchor          2000-02-19 irrsitec N N N Y N
 XCV123   DB2-SECOND-AUTH-ID   2002-07-11 STDSDB2  P N N N N      MAXSDB2
```

Output Sample //VEROWLST:

```
DEB$SO15-10 OWNERSHIP ENTRIES "AS IS" FROM //OWNI0200    V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608              PAGE:         1
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 APPL.    BUSINESS APPLICATION SYSTEM   STATUS            DESCRIPTION          INFORMATION (ERROR MESSAGES ETC.)
 PREFIX   OWNER    OWNER       OWNER
 -------- -------- ----------- -------- ----------------- -------------------- ----------------------------------------------
 ACC      AMXHRG   AMXCYD      ######## In Productive use ACCPAC USED BY CONNI
 AGP      MIXRUL   MIXPKI      ######## In Productive use AGENDA+
 ALG      MIXXFN   MIXXFN      ######## In Productive use ALLG C/S-ARCHIT
 ALI      MIXHOK   MIXPIJ      ######## Decommissioned    SSS LODS INTER
 1001     ######## ########    IBMUSER  ACTIVE            Z/OS System
 1002     ######## ########    IBMUSER  ACTIVE            SMS HSM
 1003     ######## ########    SNACK10  ACTIVE            DB2
 1004     ######## ########    CAT0111  ACTIVE            MQS
```

Output Sample //VERRACON (//IRRI0205 connect records):

```
 DEB$SO16-10 RACF IRRDBU00 TYPE 0205 CONNECT RECORDS      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:     5.608
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 USERID   GROUP-ID AUTHDATE   T OWNER    S O R CON.-DATE  TIME     INFORMATION (ERROR MESSAGES ETC.)
 -------- -------- ---------- - -------- - - - ---------- -------- ----------------------------------------------

 DEB$SO16-10 RACF IRRDBU00 TYPE 0205 CONNECT RECORDS      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:     5.610
                                                                                                                   DATE:2008-08-21
             JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                         TIME:  16:04:11
 USERID   GROUP-ID AUTHDATE   T OWNER    S O R CON.-DATE  TIME     INFORMATION (ERROR MESSAGES ETC.)
 -------- -------- ---------- - -------- - - - ---------- -------- ----------------------------------------------


 ===> TOTAL NUMBER OF USER-IDS PROCESSED :       280.405

 ===> TOTAL NUMBER OF USER-IDS SPECIAL   :           302

 ===> TOTAL NUMBER OF USER-IDS OPERATIONS:             0

 ===> TOTAL NUMBER OF USER-IDS REVOKED   :            45


 ===> TOTAL NUMBER OF RECORDS ACCEPTED   :       280.405
```

Output Sample //VERRAGRP (//IRRI0100 group records):

```
DEB$SO17-10 RACF IRRDBU00 TYPE 0100 GROUP RECORDS        V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608                PAGE:         1
                                                                                                                    DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                            TIME:  16:04:11
GROUPID  SUPGROUP AUTHDATE   OWNER    UACC     INSTALLATION DATA    INFORMATION (ERROR MESSAGES ETC.)
-------- -------- ---------- -------- -------- -------------------- ------------------------------------------------------------
@ALIL    INADPROJ 2001-09-13 INADPROJ NONE     FIN: SR_AMERICA/LODS
@ALIT    @ALIL    2001-09-13 @ALIL    NONE     FIN: SR_AMERICA/LODS
@ALIV    @ALIL    2001-09-13 @ALIL    NONE     FIN: SR_AMERICA/LODS
```

Output Sample //VERAPMIS:

```
DEB$SO20-10 OWNER-IDS (=USER-IDS) MISSING IN "RACF"      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608                PAGE:         1
                                                                                                                    DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                            TIME:  16:05:28
APPL.    BUSINESS APPLICATION SYSTEM   STATUS            DESCRIPTION          INFORMATION (ERROR MESSAGES ETC.)
PREFIX   OWNER    OWNER       OWNER
-------- -------- ----------- -------- ----------------- -------------------- ------------------------------------------------------
AA1      MIXSMC   SRYPUP      ######## Decommissioned    CTMF FOR MAN, Decomm U=MIXSMC   G=OWNBAA1  NOT FOUND IN //IRRI0205
                                                                               U=SRYPUP   G=OWNAAA1  NOT FOUND IN //IRRI0205
AA2      MIXSMC   SRYPUP      ######## Decommissioned    CTMF EBPM INTEG, Dec U=MIXSMC   G=OWNBAA2  NOT FOUND IN //IRRI0205
                                                                               U=SRYPUP   G=OWNAAA2  NOT FOUND IN //IRRI0205
AA3      MIXSMC   MIXBOH      ######## Decommissioned    CTMF FOR FINCO3 , De U=MIXSMC   G=OWNBAA3  NOT FOUND IN //IRRI0205
                                                                               U=MIXBOH   G=OWNAAA3  NOT FOUND IN //IRRI0205
ACC      AMXHRG   AMXCYD      ######## In Productive use ACCPAC USED BY CONNI U=AMXHRG   G=OWNBACC  NOT FOUND IN //IRRI0205
                                                                               U=AMXCYD   G=OWNAACC  NOT FOUND IN //IRRI0205
AC1      MIXFRF   MIXXXC      ######## In Productive use ACD                  U=MIXFRF   G=OWNBAC1  NOT FOUND IN //IRRI0205
                                                                               U=MIXXXC   G=OWNAAC1  NOT FOUND IN //IRRI0205
AGP      MIXRUL   MIXPKI      ######## In Productive use AGENDA+              U=MIXRUL   G=OWNBAGP  NOT FOUND IN //IRRI0205
                                                                               U=MIXPKI   G=OWNAAGP  NOT FOUND IN //IRRI0205
AIG      T60INVAL T60INVAL    ######## Decommissioned    AIG                 RACF USERID T60INVAL NOT FOUND IN //IRRI0200
                                                                               U=T60INVAL G=OWNBAIG  NOT FOUND IN //IRRI0205
                                                                               RACF USERID T60INVAL NOT FOUND IN //IRRI0200


DEB$SO20-10 OWNER-IDS (=USER-IDS) MISSING IN "RACF"      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608                PAGE:        42
                                                                                                                    DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                            TIME:  16:05:28
APPL.    BUSINESS APPLICATION SYSTEM   STATUS            DESCRIPTION          INFORMATION (ERROR MESSAGES ETC.)
PREFIX   OWNER    OWNER       OWNER
-------- -------- ----------- -------- ----------------- -------------------- ------------------------------------------------------


===> TOTAL NUMBER OF RECORDS  VERIFIED :          1.386

===> TOTAL NUMBER OF USER-IDS MISSING  :            212

===> TOTAL NUMBER OF CONNECTS MISSING  :          1.624

===> TOTAL NUMBER OF CONNECTS PROCESSED:        280.405

===> TOTAL NUMBER OF USER-IDS REVOKED  :            164
```

DEB$SO10 searches for a specified user-Id (business, application and/or systems owner) in the file //IRRI0200 and IRRI0205. Any missing user-ID in RACF but defined in //OWNI0200 will be reported. The same applies to group-Ids. In above sample there exits no RACF Group-ID for the application(APPL PREFIX) 'AA1'. Either such a group has never been defined to RACF or the file //OWNI0200 contains obsolete data.
The user either defines 2 new RACF groups called OWNAAA1 and OWNBAA1 or removes the application prefix record from //OWNI0200.

Output Sample //VERRCADG:

```
/* RACF OWNER GROUP MISSING FOR: B/A/S-OWNERS */
 ADDGROUP OWNBAA1   SUPGROUP(ALSHOMEB) OWNER(ALSHOMEB) +
            DATA('CTMF FOR MAN, DECOMM., MERGED TO')
 CONNECT SRZSMC   GROUP(OWNBAA1 ) OWNER(ALSHOMEB)
 ADDGROUP OWNAAA1   SUPGROUP(ALSHOMEA) OWNER(ALSHOMEA) +
            DATA('CTMF FOR MAN, DECOMM., MERGED TO')
```

DEB$SO10 generates ADDGROUP statements in case a RACF group was missing to cover an application prefix. By using +OPTIONS
SET_B_HOMEGRP=OWNB????,SET_A_HOMEGRP=OWNA????,SET_S_HOMEGRP=OWNS????
You can specify the default superior group and owner name you want to be assigned for the ADDGROUP command.

Output Sample //VERGRMIS (//IRRI0100):

```
DEB$SO21-10 APPLICATION GROUP-IDS MISSING IN "RACF"      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:         1
                                                                                                                   DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                           TIME:  16:05:28
APPL.    BUSINESS APPLICATION SYSTEM   STATUS            DESCRIPTION          INFORMATION (ERROR MESSAGES ETC.)
PREFIX   OWNER    OWNER       OWNER
-------- -------- ----------- -------- ----------------- -------------------- ------------------------------------------------------
AA1      MIXSMC   MIXPUP      ######## Decommissioned    CTMF FOR MAN, Decomm RACF GROUPID OWNBAA1  NOT FOUND IN //IRRI0100
                                                                               RACF GROUPID OWNAAA1  NOT FOUND IN //IRRI0100


DEB$SO21-10 APPLICATION GROUP-IDS MISSING IN "RACF"      V3R6M0 RACFRA2.COM(C) 06/19/08 RACF VERS2608               PAGE:        17
                                                                                                                   DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                           TIME:  16:05:28
APPL.    BUSINESS APPLICATION SYSTEM   STATUS            DESCRIPTION          INFORMATION (ERROR MESSAGES ETC.)
PREFIX   OWNER    OWNER       OWNER
-------- -------- ----------- -------- ----------------- -------------------- ------------------------------------------------------


===> TOTAL NUMBER OF RECORDS  VERIFIED :          1.386

===> TOTAL NUMBER OF GROUP-IDS MISSING :            747
```

Group-Ids missing based on the application prefix e.g. AA1 in this sample will be reported.

Output Sample //VERCDEG:

```
/* RACF OWNER GROUP MISSING FOR: B/A/S-OWNERS */
 REMOVE MIXAMA   GROUP(OWNAABA )
 DG OWNAABA  /* NO APPL.PREFIX EXISTS - DELETE OBSOLETE GROUP */
 REMOVE MIXWEK   GROUP(OWNAACI )
 DG OWNAACI  /* NO APPL.PREFIX EXISTS - DELETE OBSOLETE GROUP */
```

In case there a group records e.g. OWNAABA but there is no application prefix defined in //OWNI0200, the program generates the required RACF delete commands to remove a user and to delete the group.

Output Sample //VEROWMIS:

```
DEB$SO22-10 OWNER GROUP-IDS MISSING IN //OWNI0200        V3R6M0 RACFRA2.COM(C) 06/20/08 RACF VERS2608               PAGE:         1
                                                                                                                   DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                           TIME:  16:05:28
GROUPID  SUPGROUP AUTHDATE   OWNER    UACC     INSTALLATION DATA                 INFORMATION (ERROR MESSAGES ETC.)
-------- -------- ---------- -------- -------- --------------------------------- ------------------------------------------------------
OWNAABA  OWNHOMEA 2008-02-21 OWNHOMEA NONE     XXXXXX - SALARY MANAGERMENT SYST APPLICATION: ABA      NOT FOUND IN //OWNI0200
OWNAACI  OWNHOMEA 2008-02-04 OWNHOMEA NONE     MYCLAIMS                          APPLICATION: ACI      NOT FOUND IN //OWNI0200



DEB$SO22-10 OWNER GROUP-IDS MISSING IN //OWNI0200        V3R6M0 RACFRA2.COM(C) 06/20/08 RACF VERS2608               PAGE:         9
                                                                                                                   DATE:2008-08-21
           JOBNAME :XRZP001C STEPNAME:APPLOWNR PROCNAME:                                                           TIME:  16:05:28
GROUPID  SUPGROUP AUTHDATE   OWNER    UACC     INSTALLATION DATA                 INFORMATION (ERROR MESSAGES ETC.)
-------- -------- ---------- -------- -------- --------------------------------- ------------------------------------------------------


===> TOTAL NUMBER OF APPLICATION PREFIXES PROCESSED (//OWNI0200) :          1.386


===> TOTAL NUMBER OF GROUP-IDS VERIFIED FROM //IRRI0100          :          7.462


===> TOTAL NUMBER OF APPLICATION PREFIXES NOT FOUND IN //OWNI0200:            375
```

Above report shows the missing application prefix(es) in //OWNI0200.