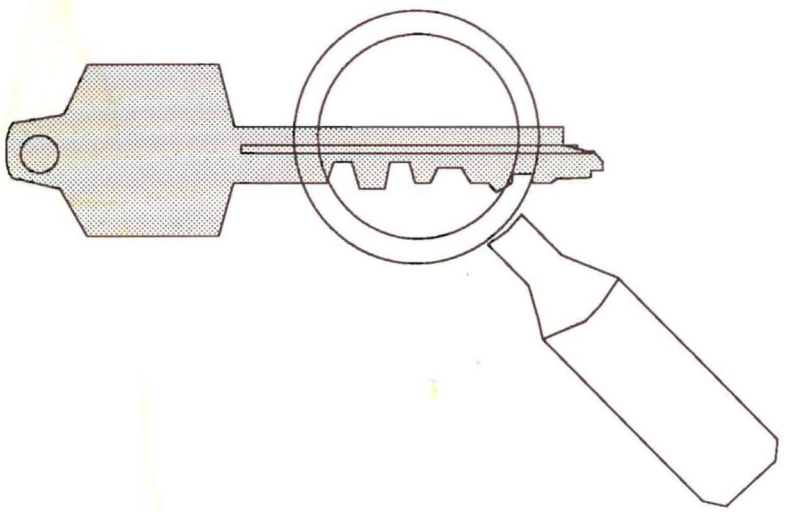
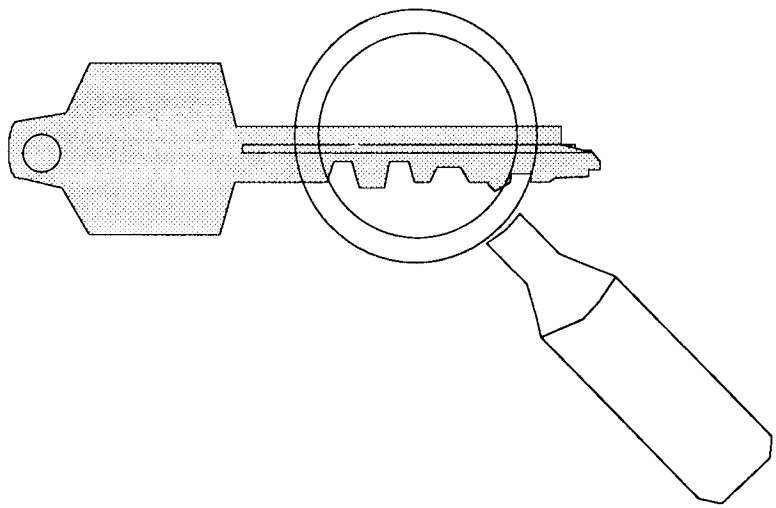
CONSUL/RACF



User Reference Manual

Version 1.1

CONSUL/RACF



User Reference Manual

Version 1.1

Sixth edition (October 4, 1991)

This manual GC14-5288-01 applies to modification level 3 of CONSUL/RACF version 1 release 1.
In some countries it is available through IBM. Program number 5786-HBN (Netherlands).

The examples in this manual are not meant to be representative or particularly applicable to another situation. Commands generated by CONSUL/RACF should never be executed without inspection. No guarantees are expressed or implied as to the accuracy, correctness, and applicability of any result produced by the CONSUL/RACF program.

The CONSUL/RACF and CONSUL/COLLECT programs are distributed by

CONSUL Risk Management B.V.
Veenweg 112
2631 RB Nootdorp
The Netherlands
tel: +31 15 146670 or +31 1731 7989
fax: +31 15 146697 or +31 1731 7433

Please direct requests for information or maintenance to CONSUL Risk Management B.V. or your software distributor.
Problem reports are accepted formally by fax or mail and must contain sufficient information (summary dump, joblog, JCL, and input commands if abended).
If purchased through IBM, normal IBM problem management procedures apply.

Table of Contents

Table of Figures

Summary of Amendments

Sixth edition for version 1.1.3

- Examples have been added for CMA-SPOOL and JES328X users.
- The usage guide now includes a section on CONSUL/COLLECT.
- Numerous editorial changes have been made to clarify the text.

From version 1.1.2 to version 1.1.3

- The package can now be installed with SMP/E (release 5 or higher) as well as without. The following changes were made to ease this:
- Sample JCL members have been renamed to adhere to the common prefix CNR. In addition, the 4th letter can be J for sample JCL, L for members previously starting with LIST, V for members previously starting with VER, X for members previously starting with SAS or LST, and Z for members previously starting with I$.
- Default datasetnames have been modified to let the last qualifier of datasets coincide with SMP/E DDDEF (i.e. CNRLOAD and CNRSAMP instead of LOAD and CNTL). In addition, the default first qualifier has been changed from SYS1 to CRM.
- Installation JCL has been moved to a separate library (CNRINST), replacing the sequential LOADJCL member.
- Consul/Collect is now installed in a separate library (CNFLOAD).
- A number of JCL procedures are now also shipped in the form of a procedure library (CNRPROC).
- Sample IPOUPDTE JCL and input is shipped to tailor the installation JCL before installation, and to tailor sample JCL, procedure, and CLIST libraries after installation.
- Defect support is now available through IBM SSC in some countries if the package has been purchased from IBM.

From version 1.1.1 to version 1.1.2

- The limit on the number of databases that can be read in parallel has been increased from 10 to 64.
- The listing of input statements now contains line numbers.
- The listing of input statements now lists the datasetname(s) containing the statements.
- The datasetname and volume of unloaded database input is now reported on message CNR004I.
- Command input processing has been enhanced to give clearer error messages for syntax errors. In addition, they now have 9 different message numbers instead of one number.
- A page skip is now generated after the last report to separate it from summary messages.
- Minor enhancements to the diagnostic summary dump.
- Bug fixes: PL=0 support, REPORT PERMIT= erroneously included GLOBAL accesses.

From version 1.1.0 to version 1.1.1

- The universal access column in the REPORT ACI report has been extended with the keywords READLPA, LOADEXE, HIDDEN, and COPY to correctly interpret the interplay of program profiles, dataset profiles, linklist residency, and LPA residency. In addition, the Lnk column contains a lower case "h" if the module is present in the linklist but hidden by a similar module concatenated in front.
- The REPORT SCOPE command can be tailored by including a SUPPRESS REASON= command with a list of reasons. The reasons that can be suppressed are UACC, WARNING, GLOBAL, and SELFCONNECT.
- The REPORT SENSITIVE and REPORT SCOPE commands now both consider profile warning mode and the effect of the global access table. An additional message is introduced to indicate excessive universal access.
- The selection SEGMENT=BASE is now always true in a non-RDS database. This helps to design reports that look the same in RDS and non-RDS environments. The sample reports include this selection where appropriate.
- Additional processing is done to pinpoint and correct problems due to inconsistent ICHCNX00 processing. The command DEBUG QUAL can be used to review all ICHCNX00 processing, the command SUPPRESS MSG=172 can be used to suppress excessive number of CNR172I error messages regarding ICHCNX00.
- Program profile ** is now supported.
- Input syntax checking now resynchronizes on each new command.
- The letters '/*' can now be incuded in a quoted string without starting a comment.
- A message is now issued for classes in the database that are not present in the Class Descriptor Table.

From version 1.0.2 to version 1.1.0

- CONSUL/COLLECT 2.0 support. CONSUL/COLLECT features parallel access to the disks to collect VTOC, VVDS, catalog, and PDS directory information, resulting in spectacular speed improvements for large DASD farms.
- Collection of ICF catalog data for auditing purposes has been made possible without the need of an ALTER permit, if authorized to a FACILITY profile $CNF.AUDIT. The same profile is tested for collection of APF dataset directory information: a READ permit would also allow the execution of the APF programs, with the $CNF.AUDIT permit you can audit all APF library contents and compare them with PROGRAM profiles without having the authority to execute dangerous utilities.
- Support for the Restructured Database format (RDS) of RACF 1.9
- Support for the conditional access list of general resource profiles (RACF 1.9), for instance in the REPORT SCOPE command.
- Addition of REMOVE USERID= command to generate REMOVE and DELUSER commands in addition to the REMOVE PERMIT functionality. Optionally the scope of the command can be limited to a list of groups with the FROMGROUP= parameter.

- Addition of MOVE USERID command to support transferral of userids to a different group. Optionally this may be used to transfer the userid to a holding group prior to removal, by specifying the keyword REVOKE.
- REMOVE PERMIT extended to optionally remove NOTIFY fields and profiles in the classes PROPCNTL, SURROGAT, JESJOBS, JESSPOOL, VMEVENT, VMXEVENT, and NODES.
- REMOVE NOTIFY= command and MOVE NOTIFY=, NEWNOTIFY= to remove or change notify fields.
- REMOVE GROUP= command to generate REMOVE and DELGROUP commands in addition to REMOVE PERMIT functionality.
- Formatting of LOGTIME, LJTIME, LOGDAYS, LOGZONE, TVTOCSEQ, ACL2RSVD, and RETPD fields.
- Formatting of MEMLST for VMEVENT, VMXEVENT and CATEGORY profiles.
- Support for alias names on the LIST/SORTLIST command (e.g. UACC for DATASET profiles).
- Command SHOW TEMPLATES to show all fields that can be listed, including the default length and output format. This helps to design your own reports without the SPL: RACF manual.
- Command SHOW CLASSES to show classes present in the database and the number of discrete and generic profiles in each class.
- Possibility to request a different output format than the default on the LIST/SORTLIST commands.
- Default now OVP=0 to prevent problems in using Xerox 9700 printers because OVP specification did not apply to the first output line.
- Command NEWLIST to start a new combination of SELECT/EXCLUDE, PRINT, and LIST/SORTLIST commands so you can produce more than one report in one run. Each NEWLIST is a further selection from the selections made before the first NEWLIST.
- Support for different output DDname and pagesize for LIST and SORTLIST behind a NEWLIST command. By default, LIST will generate no page headers and separators. Each NEWLIST can be directed to another file.
- Option DEFAULT OWNER= to be used instead of SYS1 as the new owner of general resource profiles with the command REMOVE PERMIT=.
- Support for RACF 1.9 generics (** anywhere in the profile name).
- Support to specify REPORT SCOPE=* to see the scope potentially available to all users (including undefined users).
- Overview of the protection of APF modules with AC=1 or privileged PPT attributes, indicating the PROGRAM profile (if any), and possibly multiple versions in different libraries.
- Extensive use of internal caching algorithms to reduce CPU usage.
- New LIST/SORTLIST keywords DB, RBA, and SEGMENT to print the location of the originating profile and the segment name.
- The unloaded database format has been changed to include the origin database number and RBA of the profiles.
- The FIELDVALUE= and SCAN= options of the SELECT command are now converted to uppercase if they are not enclosed in quotes. To search for lowercase values, the value can be enclosed in quotes.

- Lists CPU time used and elapsed time at termination.
- Enhanced support for discrete tape dataset profiles.
- GLOBALAUDIT taken into account on REPORT SENSITIVE.
- Allocation of the CMDOUT output file now automatically triggers command generation, no explicit REMOVE command is necessary.
- Upon normal completion of the program, all allocated storage is freed.
- It is now possible to unload an already unloaded file with further subselection. The complete trail of unloads is recorded in the unloaded file.
- The ISPF application can now display all class options by selecting the class on the class table panel. This includes numerous new RACF 1.9 options.
- The started task table display can now display old format (pre-1.7) tables as well.

Third edition for version 1.0.2

- IOCONFIG 1.5.6 messages and options now described.

From version 1.0.1 to version 1.0.2

- Support for bold text with either DCB=OPTCD=J for 3800-type laserprinters or overprinting for impact printers. With OPTCD=J, Table Reference Character 0 is used for normal text and TRC 1 for bold.
- New PRINT option OVP=*n* to set number of overstrikes. Can be set to 0 to disable.
- New PRINT option SUBTITLE='*text*' to add an extra page header line below the TITLE.
- Support for SYSPRINT with LRECL up to 255. Can be exploited by LIST and SORTLIST commands. Invoked by specifying LRECL on DD statement.
- IOCONFIG 1.5.6 is now shipped with fast catalog dump (10 times as fast on shared DASD)
- Enhanced support for multiple keyranges of a VSAM dataset on one volume.

From version 1.0.0 to version 1.0.1

- REPORT SCOPE extended to propagate *group authority* recursively to owned groups and to resources owned by group-owned users.
- REPORT SCOPE extended to consider WARNING mode of profiles equivalent with ALTER acess.
- Numeric fields containing the 'not present' value (all bytes high value) are now formatted as blanks instead of 255, 65535 etc.
- Various bug fixes: SELECT REVOKE problems, date field formatting problem, VERIFY CONNECT incorrect output due to database unload missing a profile in some very special cases, incorrect formatting of SECDATA member profiles.

From version 0.0.7 to version 1.0.0

- Discrete GDG generation profiles made redundant by the addition of a GDG model profile while MODEL(GDG) is active, are now identified by separate messages.
- The REPORT REDUNDANT command now also lists candidate profiles checked by redundant profiles.

- The REPORT REDUNDANT command now gives a disctinct 'First reason' for *undefined* ids. Hopefully this will help in identifying too restrictive use of the SELECT and EXCLUDE commands.
- A separate message is issued for missing self-describing records for catalogs. This is a normal condition for unconnected catalogs, since these cannot be opened in a normal way.
- The manual has been extended with a section giving guidance in case of abends.
- Various bug fixes: LIMIT DISCRETE problem.

From version 0.0.6 to version 0.0.7

- Addition of the SORTLIST command to give sorted equivalent of LIST. Limited by user region above and belowe 16Mb.
- Datasets reported by REPORT DATASETS sorted alphabetically
- Addition of expiry date mechanism.
- Addition of installation JCL.
- Option VERIFY NOTEMPTY meaning changed: only removes empty generics for which a less specific generic exists. Option VERIFY ALLNOTEMPTY added to remove empty generics even if they are not covered by a less specific generic (former meaning of VERIFY NOTEMPTY).
- Command generation support for removing volumes of multivolume discrete dataset profiles (instead of deleting the profile). Also separate messages distinguishing multivolume profiles from single-volume profiles.
- Better exception handling for VVDS datasets.
- Handling of VSAM alternate index (AIX) improved. REPORT DATASETS now reports AIX data and index components under the base cluster name.
- SELECT of repeated field now true if *any* of the repeated values equal to the FIELDVALUE.

From version 0.0.5 to version 0.0.6

- Catalog dump processing added to give VSAM support to VERIFY parameters INDICATED, PROTECTALL, ONVOLUME. This option requires IOCONFIG 1.5.6 supporting catalog dump.
- Support for LRECL=X,RECFM=VBS files as unload target and SYSUT1 input. This prevents truncation problems for sites with RACF profiles exceeding 32K.
- New options PRINT EGN and PRINT NOEGN to assist in debugging EGN problems without having to change the systemwide setting in RACF.
- Display of release and modification level for RACF and DFP.
- Enhancement of the SELECT FIELDVALUE selection based on profile field contents: support for quoted strings, support for internally coded variables (numeric variables, access levels, and audit levels, etc.). In addition, full support for comparison functions (less than, greater than, etc.)
- Support for RACF 1.8 new templates. New LIMIT OLDTEMPLATE option to use old templates instead.

From version 0.0.4 to version 0.0.5

- Better handling of system-generated temporary dataset names.
- NONREDUNDANT processing extended to identify redundant generic profiles.
- The parameters GENERIC and DISCRETE on the LIMIT command to regulate the scope of REPORT NONREDUNDANT.
- The parameter NOTEMPTY on the VERIFY command to generate commands to remove unused generic dataset profiles.
- Function of SUPPRESS VOLUME= extended to cover duplicate catalogs.
- LIST options CLASS and KEY added, not default anymore. This will enable you to specify a shorter length for printing the profile key.
- REPORT DATASETS implemented.
- SELECT REVOKE now also takes revoke by date and resume by date into account.
- SELECT ERASE implemented.
- REPORT SCOPE= partially implemented (no propagation of group authority yet).
- Former REPORT APF command implemented and changed to REPORT SENSITIVE. The command currently reports on the protection of APF, RACF, page, and swap datasets.

Introduction

The purpose of CONSUL/RACF is to provide you with support in the area of management, audit, and operation of a RACF system. In no way does it attempt to replace or duplicate functions provided by RACF itself, unless the poor performance of these functions can be prohibitive for their daily use (e.g. ICHUT100).

Typically, the people using CONSUL/RACF can be found in four areas:

- the *security administrator* responsible for maintaining consistency in the database and performing (bulk) updates;
- the *systems programmer* responsible for conversion projects and for synchronizing RACF with actual disk contents after recovery operations;
- the *security officer* responsible for defining and enhancing the local security policy;
- the *EDP-auditor* responsible for checking conformance of profile contents with the local security policy.

The product consists of three independent components: a *batch* component to collect MVS resource data (CONSUL/COLLECT), an *interactive* component to display incore RACF-related information, and a *batch* component to analyze the RACF database and protected resources in the system. An interactive interface to the database analysis is planned for a future release, as well as a collection tool for VM systems.

CONSUL/RACF by itself does not perform any RACF function. It merely produces reports for you and, if you request it, the batch component produces an output file containing RACF commands. It is up to you to submit a job to execute these commands. You can use an editor to modify or delete commands as you like. In this way you have full control over the modifications you want to make to your database. In addition, there is no risk that the internal structure of the RACF database is damaged or that authority checks to modify profiles can be circumvented: RACF checks the authority of the submittor for all the commands. Generally, the SPECIAL or group-SPECIAL attribute will be needed to perform the commands.

To use the RACF database functions of CONSUL/RACF, a read permit on the RACF datasets is required. No authorization checking is currently done on access to the profiles (this feature is however being considered for inclusion in a future release).

The remainder of this ***User Reference Manual*** is structured as follows:

Part 1 — A ***Usage Guide*** which discusses common problems in RACF Administration and shows how CONSUL/RACF can be used to solve them.

Part 2 — A ***Reference*** section containing detailed material on the syntax, restrictions and function of the JCL and commands necessary to use CONSUL/RACF.

Part 3 — A ***Message*** reference for CNR-prefix messages issued by CONSUL/RACF.

Appendix A — ***Installation Instructions***

Appendix B — ***Sample Templates*** for RACF 1.9 system.

It is recommended that you first read through Part 1 of this manual, then install the product as described in Appendix A. The reference material in Part 2 can be consulted later when you are ready to begin running CONSUL/RACF.

An additional product CONSUL/COLLECT is needed for full use of all CONSUL/RACF facilities. Instructions on how to install and run this product are to be found in the **CONSUL/COLLECT *User Reference Manual*** which you will find after the ***Index*** pages of this manual.

PART 1 Usage Guide

This Usage Guide contains four sections:

1.1 Unloading and Selecting RACF datasets (to process under CONSUL/RACF).

1.2 Common RACF problems and their solution - how CONSUL/RACF can assist in the problems of day-to-day RACF administration.

1.3 Standard Reporting Tasks - examples of the use of the LIST, SORTLIST, SELECT, and EXCLUDE commands including some special cases.

1.4 Common CONSUL/RACF problems and their solution - what to do if you encounter problems in running CONSUL/RACF. Also gives advice for installations using Hot-Standby Volumes and Alternate master Catalogs.

The usage guide is by no means a complete overview of the possible uses of CONSUL/RACF. However all special-purpose functions (mainly the VERIFY and REMOVE commands) are introduced under the heading of the problems they were designed to solve.

The usage guide contains some sections that have a title starting with *Background -*. These sections generally explain some RACF terminology and can be skipped by readers already familiar with RACF.

1.1 Unloading and selecting RACF datasets

Usually, when you extract information from the RACF database, it takes several runs to get exactly the report you need.

To minimize impact on the system (especially if you share the RACF database between systems), it is recommended that you make an unloaded version of the RACF database first, and direct subsequent processing to the unloaded version.

Some JCL procedures are included with CONSUL/RACF to perform specific functions. These procedures assume that you unload the database first, using a standard naming convention for your unloaded file.

Background - RACF datasets

Your RACF database can consist of one or more datasets each containing one or more *key ranges* of the RACF profiles. This is defined in two customizable RACF modules: the dataset names are defined in the dataset name table ICHRDSNT, the key ranges are defined in the range table ICHRRNG. The first of these datasets is called the *master* database, because it contains the systemwide options that are used by the system.

In addition, RACF supports a *primary* and a *backup* (or *secondary*) dataset for each key range. Generally, the backup dataset will not be updated for changes in profile statistics. In case of I/O errors, you may switch to the backup dataset. The backup dataset will then become the *active* dataset. Normally, the *primary* datasets are *active*.

In addition, your site may make regular backup copies of the RACF datasets. We will call this an *archive* copy. Generally, archive copies will be copies of all primary datasets.

1.1.1 Using the active primary datasets

Unload of the currently active and primary RACF datasets is the default action if file SYSUT2 is present. In addition, it may be requested explicitly by the command

```
UNLOAD
```

The JCL to use for this function is available in the sample procedure CNRCOPYR:

```
//CNRCOPYR PROC CPREF=SYS1,            Prefix for CONSUL/RACF libraries
//   CPROJ=CNR110,          Project for CONSUL/RACF libraries
//   CREG=4096K,            Region for CONSUL/RACF (at least 2M)
//   DSTAT=NEW,             Disposition of work datasets
//   DPREF=SYS1,            Prefix for work datasets
//   DUNIT=SYSDA,           Esoteric unit name for work datasets
//   DVOLSER=,              Optional volume serial for work datasets
//   OPT=                   Optional command like 'ALLOC BACKUP'
//*
//*********************************************************************
//* Name:     CNRCOPYR                        Version: CONSUL/RACF 1.1.0
//* Purpose: Unload RACF database to work dataset
//*********************************************************************
//*
//CNRACF EXEC PGM=CNRACF,REGION=&CREG,
//  PARM='&OPT.;SHOW CLASSES'
//STEPLIB  DD DISP=SHR,DSN=&CPREF..&CPROJ..LOAD
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DISP=(&DSTAT,CATLG),DSN=&DPREF..CNRACF.UNLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(150,150),RLSE,,ROUND)
//       PEND
//*
//CNRCOPYR EXEC CNRCOPYR
```

Fig 1. JCL procedure CNRCOPYR

The RACF datasets are allocated dynamically, and need not be allocated in the JCL.

Sample output of the unload function:

```
CNRACF 1.1.b 02/15/91 22.26   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   24 Feb 1991 23:34
page    1
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,
THE NETHERLANDS


CNR017I 00 Processing started for SYSRAC01 SHR101 SYS2.RACF.PRIM1
CNR017I 00 Processing started for SYSRAC02 SHR101 SYS2.RACF.PRIM2
           at 24 Feb 1991 23:34 running RACF 1.8.1
           Non-restructured database format

CNR033I 00  SYS2.RACF.PRIM1 has 28535 segments in use, 79345 segments free (26% used)
            Index uses 4%.  Space beyond 44% never used.

CNR033I 00  SYS2.RACF.PRIM2 has 107335 segments in use, 110281 segments free (49% used)
            Index uses 13%.  Free space completely fragmented.

CNR800I 00 Maximum profile length is 13978 bytes for GROUP #SDGSD

CNR005I 00 110428 profiles read, 110428 profiles selected (100%)
```

Fig 2. Sample UNLOAD output

For an explanation of the messages, refer to part 3 of this manual (the message reference).

1.1.2 Using an archive copy of the database

You might want to specify a different source for the unload operation. For instance, last week's backup copy may be a good candidate if you want to reconstruct the access list of accidentally deleted profiles. To accomplish this, you have to pre-allocate the SYSRAC*nn* DDnames to define an alternate source. You must use the DDname SYSRAC01 to point to the archive copy of the master dataset, and you can use SYSRAC02 and upwards for additional key ranges. The exact numbers beyond 01 have no meaning, but *nn* is incremented by 1 until no file SYSRAC*nn* is present. For example, if you specify SYRAC01, SYSRAC02, and SYSRAC03 in the JCL, then all 3 DDnames will be read. If you specify SYSRAC01 and SYSRAC03, then only SYSRAC01 will be read, since SYSRAC02 is not present.

An example of JCL to unload archive copies of a system with the RACF database split across two datasets:

```
//CNRCOPYR EXEC CNRCOPYR
//SYSRAC01 DD DISP=SHR,DSN=SYS1.RACF.DUMP1(0)
//SYSRAC02 DD DISP=SHR,DSN=SYS1.RACF.DUMP2(0)
```

Fig 3. JCL to unload archive copies

1.1.3 Using the active backup dataset

Sometimes reasons exist to process your active backup copies (those that are updated automatically by RACF). For instance, you might want to check the last access dates of user profiles before activating REVOKE INACTIVE to see whether it is necessary to copy the primaries to the backups first (last access dates are considered statistics by some RACF releases). Selecting the backup instead of the primary datasets is accomplished by the command:

```
ALLOC BACKUP
```

The listing will show which datasets were selected for input.

1.1.4 Using a subset of the database

If you have made a functional separation of the profiles into different RACF datasets by means of the range table, many functions may be performed on only a subset of the datasets, resulting in enhanced performance for these functions.

For instance, a very usable separation is to concentrate backup profiles (i.e. discrete dataset profiles that are created by a storage management system at the same time that a backup is made of the dataset) in a separate dataset. For many operations, you will not be interested in the profiles of archived datasets.

Selection can be accomplished by identifying the datasets you want by means of their sequence number. However, the master dataset must *always* be included, since it describes the system wide options that are used by a system. The following command

```
ALLOC DB=1
```

selects only the master dataset. The command

```
ALLOC DB=(1,3)
```

selects the master database as well as database sequence number 3. The relation between sequence numbers and datasetnames is defined in the RACF database name table ICHRDSNT. You can use the interactive component of CONSUL/RACF to review the range and name tables active at your system (see "2.1 Interactive component"). CONSUL/RACF will include the database sequence number *nn* in the DDname SYSRAC*nn* during dynamic allocation the datasets.

1.2 Collecting resource information

For many commands it is necessary to match resource information with the RACF profiles that protect them. The resource information is not present in the RACF database, but must be collected from many different sources (e.g. VTOC, VVDS, catalog, PDS directory). The process of collecting resource information is done by a separate program called CONSUL/COLLECT, which is included as part of the CONSUL/RACF package (but also part of other packages like CONSUL/SMS DSCAT, CONSUL/CONFIG, and CONSUL/CCW).

Since CONSUL/COLLECT can be used for different purposes, it is necessary to indicate the objective of the Collect run. This is done by setting the FOCUS parameter to AUDIT if the objective is CONSUL/RACF. To reduce processing time, additional parameters may be added to limit the information collected even further, depending on the CONSUL/RACF commands you want to use.

APF authorization is required to collect some information. The use of the APF authorized features of CONSUL/COLLECT needed for CONSUL/RACF is subject to authorization on the FACILITY profile $CNF.AUDIT - that is, you will only be able to execute CONSUL/COLLECT from an APF authorized library if you have at least READ access to the resource $CNF.AUDIT in the class FACILITY.

Full details on CONSUL/COLLECT are included in a separate manual, which is bound in one cover with the CONSUL/RACF manual (following the index of this manual).

The next sections give two examples.

1.2.1 Member-level information

The JCL to invoke CONSUL/COLLECT to collect resource information at the most detailed level (the member level needed for matching with PROGRAM profiles):

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCOPYV PROC REGSIZE=4096K,        Region for CONSUL/COLLECT
//  IPREF='CRM.',                    Prefix. for CONSUL/COLLECT libraries
//  IPROJ=CNF203,                    Project for CONSUL/COLLECT libraries
//  ILOAD=CNFLOAD,                   Last qualifier load library /COLLECT
//  DSTAT='NEW,CATLG',               Disposition of work datasets
//  DPREF='CRM.TEST',                Prefix for work datasets
//  DUNIT=SYSDA,                     Esoteric unit name for work datasets
//  DVOLSER=                         Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJCPYV     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Unload VTOC and VVDS for all volumes, and ICFcats if APF
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//IOCONFIG EXEC PGM=CNFCOLL,REGION=&REGSIZE,PARM='FOCUS=AUDIT'
//STEPLIB  DD DISP=SHR,DSN=&IPREF.&IPROJ..&ILOAD  !APF lib
//SYSPRINT DD SYSOUT=*
//IOCONFIG DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.IOCONFIG,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(30,30),RLSE,,ROUND)
//         PEND
//*
//CNRCOPYV EXEC CNRCOPYV
```

Fig 4. Sample JCL for CONSUL/COLLECT.

Note that the only parameter specified is FOCUS indicating that CONSUL/RACF postprocessing is the objective.

1.2.2 Dataset-level information

If member-level information is not needed because you only want to consider dataset profiles, then processing time may be reduced considerably by excluding PDS directories from the resource information to be collected. This can be accomplished by passing the parameter PDS=NO to CONSUL/COLLECT:

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCOPYV PROC REGSIZE=4096K,       Region for CONSUL/COLLECT
//  IPREF='CRM.',                Prefix. for CONSUL/COLLECT libraries
//  IPROJ=CNF203,                Project for CONSUL/COLLECT libraries
//  ILOAD=CNFLOAD,               Last qualifier load library /COLLECT
//  DSTAT='NEW,CATLG',           Disposition of work datasets
//  DPREF='CRM.TEST',            Prefix for work datasets
//  DUNIT=SYSDA,                 Esoteric unit name for work datasets
//  DVOLSER=                     Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJCPYV     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Unload VTOC and VVDS for all volumes, and ICFcats if APF
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//IOCONFIG EXEC PGM=CNFCOLL,REGION=&REGSIZE,PARM='FOCUS=AUDIT'
//STEPLIB  DD DISP=SHR,DSN=&IPREF.&IPROJ..&ILOAD  !APF lib
//SYSPRINT DD SYSOUT=*
//IOCONFIG DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.IOCONFIG,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(30,30),RLSE,,ROUND)
//         PEND
//*
//CNRCOPYV EXEC CNRCOPYV
//SYSIN DD *
PDS=NO
```

Fig 5. Sample JCL for CONSUL/COLLECT with additional parameters

The parameters may be specified on the PARM parameter of the EXEC statement or in the file SYSIN, this has the same result.

1.3 Common RACF problems and their solution

The standard RACF product includes utilities and commands to handle most of the situations which security administrators, auditors, and technical staff are likely to encounter. However, some of these facilities are difficult to use or take too long to consider using them on a daily basis. As a result of this some installations have been forced to develop their own procedures for dealing with these situations and others have been unable to afford the resources to deal with them on a regular basis.

This section of the manual shows how CONSUL/RACF can be used to provide fast solutions to some of the most common problems facing RACF users today in three major areas:

USER/GROUP maintenance

- Removing a user or group (including all PERMITs to resource profiles)
- Finding and removing *orphan permits* (i.e. resulting from past "partial" removal of users/groups)
- Finding user/group/connect inconsistencies (ensures USER, GROUP, and CONNECT profiles in synchronization)

PROGRAM Class Maintenance

- Checking for obsolete conditional access lists (when PROGRAM profiles have been removed)
- Checking for program existence (finds non-existent dataset/volume PROGRAM combinations)

DATASET Maintenance

- Finding and protecting unprotected datasets (checks for possible unrecorded access depending on the current PROTECTALL setting)
- Removing unused discrete profiles (resulting from volume-level operations)
- Finding and removing redundant discrete profiles (after conversion from ADSP to generic environment)
- Removing unused generic profiles (existing after "subject" datasets deleted)
- Converting to generic profiles (CONSUL/RACF commands to assist in conversion to generics)
- Finding and resetting unnecessary RACF-indicated bits (where no discrete profile exists)

1.3.1 Removing a user or group

Removing a user in RACF generally involves two kinds of actions:

1. Removing the user from all its connect groups except the default group. This is done by the REMOVE command. At the same time, all OWNER fields equal to the user for dataset profiles starting with this group are changed.
2. Deleting the user (from its default group). This is done by the DELUSER command.

It is not uncommon that these are the only RACF actions performed by the security administrator. If IBM's instructions are followed properly then the administrator should really have gone on to do the following after action number 1:

1b. Removing all references in access lists and owner fields to the user. This is usually accomplished by running ICHUT100 and manually editing the resulting report to a set of RACF commands.

This manual procedure is rather cumbersome, and in addition, running ICHUT100 typically takes from 5 to 60 minutes to complete, during which time normal RACF processing is severely impacted. This causes security administrators to omit the step. If it is omitted after the DELUSER, then a number of *orphan permits* will exist in the database.

CONSUL/RACF provides support for removing user and group references from the database as well as for handling orphan permits. The latter is the subject of the next section, this section describes removing references to known users and groups.

RACF commands to remove all references in the proper way are generated with the following command:

```
REMOVE PERMIT=id
```

where *id* is the user or group to be removed from access lists, owner fields, notify fields, and profile keys. CONSUL/RACF will not directly update the RACF database, but generate the necessary commands in file CMDOUT instead. After inspection and an opportunity to edit the file, you can run TSO in the batch or execute the file interactively. See "2.2 CNRACF Batch JCL" for examples of the JCL to use.

For an example of the output, see below.

Instead of using CONSUL/RACF only for removing references, commands can also be generated to remove the userid with all its connects. This is especially beneficial when the user has many connects. In addition, the default group is automatically taken into account to generate the right commands. The command to accomplish this is

```
REMOVE USER=id
```

This command implies all functionality of REMOVE PERMIT=. This is reflected in the default layout of the generated messages, which will be grouped under two two headers: REMOVE PERMIT and REMOVE USER.

The following figure shows sample output of the REMOVE USER command. Note by the way that the *timestamp* at the top of the first page differs from the subsequent pages: the first page shows the time the job was run, while subsequent pages show the time the database was unloaded (since the messages relate to the situation at that moment).


```
CNRACF 1.1.a 01/06/91 23.40   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  14 Jan 1991 13:19        page 1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: REMOVE  USER=SYSPAVB

CNR004I 00 Processing started for SYSUT1
          Unloaded by program CNRACF 1.1.a    01/06/91 23.40 job EUSRSCHA at 13 Jan 1991 11:44
          Source dataset   1 was SPRG19 SYS1.M9002.ICH.PRIMARY
          Non-restructured database format

CNR005I 00 5990 profiles read, 5990 profiles selected (100%)
CNR087I 00 Number of detail error messages is 51
```

```
M E S S A G E S   R E M O V E   P E R M I T          13 Jan 1991 11:44                                          page    2
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

CNR068I 00 Removing  id -   SYSPAVB  referenced      65 times
CNR248I 04 Removing  qualif SYSPAVB  of generic dataset profile        SYSPAVB.*                                - delete profile
CNR248I 00 Removing  qualif SYSPAVB  of generic dataset profile        SYSPAVB.CCWAN%%%.*                       - delete profile
CNR248I 04 Removing  qualif SYSPAVB  of generic dataset profile        SYSPAVB.MICSDOC.*                        - delete profile
CNR248I 04 Removing  qualif SYSPAVB  of generic dataset profile        SYSPAVB.PRIVATE.*                        - delete profile
CNR263I 04 Removing  notify SYSPAVB  general resource profile FACILITY $CNF.*
CNR063I 04 Removing  owner  SYSPAVB  general resource profile FACILITY $SUBMITBY.U.AVBCO01.EUSR*
CNR063I 04 Removing  owner  SYSPAVB  general resource profile PROGRAM  IOCNF*
CNR063I 04 Removing  owner  SYSPAVB  general resource profile PROGRAM  IOCNF156
CNR061I 04 Removing  owner  SYSPAVB  on group SYSPAVB1                                  - make SYS1
CNR063I 04 Removing  owner  SYSPAVB  general resource profile FACILITY $CNF.*
CNR063I 04 Removing  owner  SYSPAVB  general resource profile FACILITY $SUBMITBY.U.AVBCO012.*
CNR063I 04 Removing  owner  SYSPAVB  general resource profile FACILITY $SUBMITBY.U.AVBCO01.EUSR*
CNR060I 04 Removing  owner  SYSPAVB  on user  SYSPROX                                  - make SYS1
CNR064I 04 Removing  permit SYSPAVB  general resource profile PROGRAM  IOCNF*
CNR064I 04 Removing  permit SYSPAVB  general resource profile PROGRAM  IOCNF156
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOAUTH  ACCT
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOAUTH  JCL
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOAUTH  MOUNT
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOAUTH  OPER
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOAUTH  RECOVER
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOPROC  TSOPROC1
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOPROC  TSOSM1
CNR064I 04 Removing  permit SYSPAVB  general resource profile TSOPROC  TSOTEST1
CNR050I 04 Removing  permit SYSPAVB  in access list generic dataset    EUSRSCH.RACFTEST.WARN.*
CNR064I 04 Removing  permit SYSPAVB  general resource profile ACCTNUM  *
CNR064I 04 Removing  permit SYSPAVB  general resource profile FACILITY $CNF.*
```

```
M E S S A G E S   ( R E ) M O V E   U S E R / G R O U P  13 Jan 1991 11:44                                        page    3
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

CNR281I 00 Removing user    SYSPAVB  from     SYSPAVB1 as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSAPPL  as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSBASE  as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSBUDG  as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSDASD  as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSDB    as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSOPR   as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSTAPE  as requested
CNR281I 00 Removing user    SYSPAVB  from     SYSUSER  as requested
CNR283I 00 Deleting userid  SYSPAVB  group    SYSP     as requested
CNR039I 00 CNRACF used 3.0 CPU seconds and took 5 wall clock seconds
```

Fig 6. Sample REMOVE USER= output

You may also note that the number of messages is less than the number of references. This is because profiles to be deleted may have multiple references that are all removed by deleting the profile. No detail messages are generated for these references. You may suppress profile deletion by means of the SUPPRESS DELDSD command. In that case, the profile will not be deleted, but the references will be removed.

The commands generated if the CMDOUT file is allocated are shown in the figure below. The order of the commands is correct if you do not modify the message sort order with the VERIFY BY= option.

```
 /* Commands generated by REMOVE PERMIT */
dd 'SYSPAVB.*' generic
dd 'SYSPAVB.CCWAN%%%.*' generic
dd 'SYSPAVB.MICSDOC.*' generic
dd 'SYSPAVB.PRIVATE.*' generic
ralt FACILITY $CNF.* nonotify
ralt FACILITY $SUBMITBY.U.AVBCO01.EUSR* nonotify
ralt PROGRAM  IOCNF* owner(SYS1    )
ralt PROGRAM  IOCNF156 owner(SYS1    )
alg SYSPAVB1 owner(SYS1    )
ralt FACILITY $CNF.* owner(SYS1    )
ralt FACILITY $SUBMITBY.U.AVBCO012.* owner(SYS1    )
ralt FACILITY $SUBMITBY.U.AVBCO01.EUSR* owner(SYS1    )
alu SYSPROX  owner(SYS1    )
pe IOCNF* cl(PROGRAM ) delete id(SYSPAVB )
pe IOCNF156 cl(PROGRAM ) delete id(SYSPAVB )
pe ACCT cl(TSOAUTH ) delete id(SYSPAVB )
pe JCL cl(TSOAUTH ) delete id(SYSPAVB )
pe MOUNT cl(TSOAUTH ) delete id(SYSPAVB )
pe OPER cl(TSOAUTH ) delete id(SYSPAVB )
pe RECOVER cl(TSOAUTH ) delete id(SYSPAVB )
pe TSOPROC1 cl(TSOPROC ) delete id(SYSPAVB )
pe TSOSM1 cl(TSOPROC ) delete id(SYSPAVB )
pe TSOTEST1 cl(TSOPROC ) delete id(SYSPAVB )
pe 'EUSRSCH.RACFTEST.WARN.*' generic delete id(SYSPAVB )
pe * cl(ACCTNUM ) delete id(SYSPAVB )
pe $CNF.* cl(FACILITY) delete id(SYSPAVB )
 /* Commands generated by (RE)MOVE USER/GROUP */
remove SYSPAVB  group(SYSPAVB1)
remove SYSPAVB  group(SYSAPPL )
remove SYSPAVB  group(SYSBASE )
remove SYSPAVB  group(SYSBUDG )
remove SYSPAVB  group(SYSDASD )
remove SYSPAVB  group(SYSDB   )
remove SYSPAVB  group(SYSOPR  )
remove SYSPAVB  group(SYSTAPE )
remove SYSPAVB  group(SYSUSER )
deluser SYSPAVB  /* dfltgrp=SYSP     */
```

Fig 7. Sample REMOVE USER= output on CMDOUT

Some variations are possible. For instance, instead of removing NOTIFY fields, you may replace occurrences of the userid to be deleted in NOTIFY fields by including the NEWNOTIFY= option for each user to be deleted:

```
REMOVE USER=id, NEWNOTIFY=newid
```

In addition, groups can be removed. This will remove all connects that users have to the group, and adjust default groups as needed. However, the function currently fails if the group still has subgroups or if there are users without any other connects. The command to remove a group is:

```
REMOVE GROUP=id
```

For more information and examples, see the reference material in "2.3.9 (RE)MOVE".

1.3.2 Finding and removing orphan permits

As explained in the previous section, it is not uncommon for RACF administrators to delete users or groups from the RACF database without first removing all references to them. This results in *orphan permits*, i.e. permits to an identity that is neither present as user nor as group. More generally these can be called *undefined identities*, since they can be present in other fields than permits (access list entries), for instance OWNER fields and NOTIFY fields.

Depending on the procedural measures around the names of users and groups, this may or may not be a *security exposure*. The risk involved is that of re-issuance of a previously used *id* to a new user or group, resulting in the new user having access to all kinds of resources belonging to the department of the previous owner of the userid/group. This can easily happen if RACF user names are based on human names or on reusable personnel numbers, as was often done in pre-RACF times.

CONSUL/RACF provides a function specifically designed for reporting and/or removing orphan permits from the RACF database (in addition, orphan permits may be *prevented* as discussed in the previous section). The command to report on orphan permits is:

```
VERIFY PERMIT
```

The figure below gives an example of output of this command. A summary message is generated for each id with a count of the permits encountered, and for each field containing the id an error message.

```
CNRACF 0.0.3 01/31/90 14.47   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   3 Feb 1990 17:25
(C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: print pagelen=60
SYSIN: VERIFY PERMIT

CNR0041 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    01/31/90 14.47 job          at  3 Feb 1990 17:23
           Source dataset    1 was SHR101 SYS2.RACF.PRIM1
           Source dataset    2 was SHR101 SYS2.RACF.PRIM2


CNR0051 00 115029 profiles read, 115029 profiles selected (100%)
CNR0681 04 Undefined id -    @GD477   referenced       1 times as owner or permit
CNR0681 04 Undefined id -    @GD588   referenced       6 times as owner or permit
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM GDF101 DMSOS.DMSBACKP.D90006.THM0510.TSS3945
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM GDF101 DMSOS.DMSBACKP.D90013.THM0316.TSS2335
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM GDF101 DMSOS.DMSBACKP.D90020.THM0416.TSS0206
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM GDF101 DMSOS.DMSBACKP.D90027.THM0238.TSS1131
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM GDF101 DMSOS.DMSBACKP.D90034.THM0704.TSS3342
CNR0461 04 Undefined permit @GD477    in access list of non-VSAM GDF101 DMSOS.DMSIMARC.D89272.THM1843.TSS3914
CNR0461 04 Undefined permit @GD588    in access list of non-VSAM PRDS01 SYS4.PSBCICE
```

Fig 8. Sample VERIFY PERMIT output

To remove the orphan permits, CONSUL/RACF generates RACF commands in the CMDOUT file if allocated.

For access list entries, the orphan *id*s are simply deleted. For OWNER fields containing orphan *id*s, processing depends on the resource type. For dataset entries, ownership is changed to the 'natural' owner of the data, as indicated by the first qualifier (or the id returned by exit ICHCNX00). If this natural owner is also undefined, than it is changed to SYS1 instead. For connect profiles, ownership is changed to the connect group (connect ownership does not give authority anyway). For other resource types, no 'natural' owner exists, and the owner is changed to SYS1, or the default owner you selected with the DEFAULT OWNER= command.

SYS1 is used for ownership if nothing better can be constructed, because this is the only value that is guaranteed to give no access to people that do not have direct or indirect access already[1].

[1]To exploit group ownership, group-special or group-operations is needed. If anybody has this for SYS1 then this propagates to all RACF groups in the system if group-OWNER fields are groups (since all RACF groups resort hierarchically under SYS1). In addition, the user has access to system datasets that are required to be called SYS1.XXXX and which make it possible to circumvent RACF by updates since they are treated as APF authorized (e.g. SYS1.LPALIB).

1.3.3 Moving users to a different department

Having personal and individual userids is generally considered the safest way to manage userids. This introduces, however, the problem of personnel moving from one department to another, with the accompanying shift in authority. While adding authorities is generally not difficult, the RACF REMOVE command can change only ownership of profiles starting with a group as first qualifier, and cannot delete access list entries or NOTIFY fields. In addition, sites with an ICHCNX00 exit changing the first qualifier for authorization purposes only, will have noticed that ownership of those group datasets is not changed.

CONSUL/RACF provides support in this area by allowing selective removal of userids from the profiles of group datasets. This applies to access lists, ownership, and notify fields, and includes all profiles that have the group returned as a first qualifier by ICHCNX00. The command to accomplish this is:

```
REMOVE PERMIT=id, FROMGROUP=group
```

The FROMGROUP= parameter may specify a list of groups as well:

```
REMOVE PERMIT=id, FROMGROUP=(group1,group2,group3)
```

In addition, the RACF REMOVE commands can be generated automatically, together with an ALTUSER command to change the default group, if necessary:

```
REMOVE USER=id, FROMGROUP=(group1,group2,group3)
```

In addition, the RACF CONNECT command may also be generated by including the target group:

```
REMOVE USER=id, FROMGROUP=(group1,group2,group3), TOGROUP=newgroup
```

In fact, if the user has simply to be removed from all his current connect groups and moved to a new group, then the following command would suffice:

```
MOVE USER=id, TOGROUP=newgroup
```

It is even possible to keep references to some specific current connect group; the TOGROUP= parameter can be used both for new and for existing connect groups:

```
MOVE USER=id, TOGROUP=(newgroup,oldgroup)
```

In this example, the user will be removed from all other connect groups than those mentioned on the TOGROUP.

1.3.4 Checking for obsolete conditional access lists

RACF offers the possibility to allow access to datasets only through specific programs. This facility is called Program Access to Datasets (PADS). This is done by having an extra access list on the dataset profile called the *conditional access list.* The conditional access list consists of entries containing an *id,* an *access level* and a *program name.* For the entry to be effective, the program name must be defined as a profile in the class PROGRAM, and the user must call the program from a 'clean' environment.

The problem is that no check is made by RACF on the existence of the PROGRAM profile, neither when setting the conditional permit, nor when deleting a PROGRAM profile. Most users have found this out trying to debug 913 abends for PADS datasets, e.g. due to a typing error in the program name on the conditional access list. A more serious problem is introduced if the PROGRAM profiles (and presumably the programs, too) are removed from the system. A conditional access list entry will then exist that serves no function anymore, a so-called *obsolete conditional access list entry.*

Depending on the procedures around the use of the PROGRAM class, this may result in a *security exposure.* The exposure exists because a dataset profile gives access based on a program name that is undefined. Anybody with class authorization for PROGRAM can define the program profile, indicating a program in his own load library. He then has access to somebody else's dataset.

CONSUL/RACF provides a function specifically designed to report and remove obsolete conditional access list entries. The report is created by means of the following command:

```
VERIFY PADS
```

The following figure gives an example of the output:

```
CNRACF 0.0.3 01/31/90 14.47   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   3 Feb 1990 17:49        page    1
(C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: VERIFY PADS

CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    01/31/90 14.47 job ZGC501A1 at  3 Feb 1990 17:23
           Source dataset   1 was SHR101 SYS2.RACF.PRIM1
           Source dataset   2 was SHR101 SYS2.RACF.PRIM2

CNR005I 00 115029 profiles read, 115029 profiles selected (100%)
CNR045I 04 Conditional access list undefined program $CCFP002 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP003 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP004 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP005 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP006 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP007 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP009 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP010 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program $CCFP011 in         DGDCCF.ODI068.CCF.HISTMAST.*
CNR045I 04 Conditional access list undefined program ZA102#Z  in DCM102 CICS#.INTFACE.ROSDISOS.VSQ0#Z
CNR045I 04 Conditional access list undefined program $CCFP002 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP003 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP004 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP005 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP006 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP007 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP009 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP010 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
CNR045I 04 Conditional access list undefined program $CCFP011 in GDF101 DMSOS.DMSBACK2.D90024.THM2225.TSS5365
```

Fig 9. Sample VERIFY PADS output

To generate commands to remove the obsolete entries, the CMDOUT file must be allocated.

The commands generated will be longer than 72 characters. This implies that the CMDOUT file *must* have a record length longer than 80 (TSO ignores line number fields). Use of the default CLIST format (VB 255) is recommended.

1.3.5 Checking for program existence

Profiles in the class PROGRAM contain datasetname/volume combinations as members. RACF does not check on the existence of these datasets, either during profile creation, or during dataset deletion. This may result in unsolicited 913 abends if somebody moves the dataset to a different volume. Any program-accessed dataset (PADS) using the moved program will be affected.

In addition, a *security exposure* results if the program-protected library is part of the Authorized Program Facility (APF). It is common procedure to protect sensitive utilities in the MVS linklist with PROGRAM profiles. Utilities can be considered *sensitive* if they have the ability to circumvent RACF. To have this potential, utilities must be part of an APF library and linked with AC=1. Now if the APF library containing the sensitive utility is moved to another volume, then access to its function will not be restricted anymore based on the access list of a program profile, unless the program profile is updated to contain the new volume.

CONSUL/RACF provides a function designed to report and remove dataset/volume combinations that are non-existent. Note that for the APF problem described above, removal of the old combination is not enough - the new combination should be added as well.

Checking for existence of the datasets on the volumes as indicated by the PROGRAM profiles is done by the following command:

```
VERIFY PROGRAM
```

An IOCONFIG file is required for this function.

Sample output from this command:

```
CNRACF 0.0.0 09/27/89 12.21    C O N S U L / R A C F    D A T A B A S E    U T I L I T Y
(C) COPYRIGHT 1989, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: VERIFY PROGRAM

CNR004I 00 Processing started for SYSUT1

CNR044I 04 Dataset not found for program            *         - DCM201 DCOM.CBLIB#Z
CNR044I 04 Dataset not found for program            @IUTXPRT  - DCM201 DCOM.CBLIB#Z
CNR044I 04 Dataset not found for program            DBUTLTY   - DCM201 DCOM.CBLIB#Z
CNR044I 04 Dataset not found for program            IDBATCH   - DCM201 DCOM.CBLIB#Z
CNR044I 04 Dataset not found for program            SCPSUTIL  - DCM201 DCOM.CBLIB#Z
CNR044I 04 Dataset not found for program            IEBGENER  - EMV001 SYS1.LINKLIST.SSM3002
CNR044I 04 Dataset not found for program            *         - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            DSIOST    - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            ICHDSM00  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            ICHUT100  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            ICHUT200  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            ICHUT300  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            ICHUT400  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            IDCBD01   - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            IDCLA01   - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            IDCSC01   - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            IEBGENER  - FMVS01 SYS1.LINKLIB
CNR044I 04 Dataset not found for program            LOOKLOG   - FMVS01 SYS1.LINKLIST.SLU0660
CNR044I 04 Dataset not found for program            ADSAR003  - FMVS01 SYS1.LINKLIST.SSL7500
CNR044I 04 Dataset not found for program            *         - FMVS01 SYS1.LINKLIST.SSM3100
CNR044I 04 Dataset not found for program            ZE015JBN  - FMVS01 SYS1.LINKLIST.ZSE3822
CNR044I 04 Dataset not found for program            IEBGENER  - FMV001 SYS1.LINKLIST.SSM3002
CNR044I 04 Dataset not found for program            *         - GDF101 SYS2.DMSLINK
CNR044I 04 Dataset not found for program            ADSAR003  - GDF101 SYS2.DMSLINK
CNR044I 04 Dataset not found for program            ADSMI002  - GDF101 SYS2.DMSLINK
CNR044I 04 Dataset not found for program            *         - SHR101 SYS1.LINKLIST.NOSMP.DMS77LNK
CNR044I 04 Dataset not found for program            ADSAR003  - SHR101 SYS1.LINKLIST.NOSMP.DMS77LNK
CNR044I 04 Dataset not found for program            ADSMI002  - SHR101 SYS1.LINKLIST.NOSMP.DMS77LNK
CNR044I 04 Dataset not found for program            *         - SPG001 TZ.W207.ROSLINK

CNR005I 00 34873 profiles read, 34873 profiles selected (100%)
```

Fig 10. Sample VERIFY PROGRAM output

Commands can be generated to remove the non-existing dataset/volume combination from the PROGRAM profiles by allocating the CMDOUT file.

The commands generated will be longer than 72 characters. This implies that the CMDOUT file *must* have a record length larger than 80. Use of the default CLIST format (VB 255) is recommended.

1.3.6 Finding and protecting unprotected datasets

It is possible that datasets exist in the system that are not matched by any profile in RACF. Access to these datasets is governed by the setting of the system-wide RACF option PROTECTALL. If PROTECTALL(FAIL) is active, then access is allowed only for users with SPECIAL authority. We will call these datasets *inaccessible*. If PROTECTALL(FAIL) is not active, then access is allowed for anybody (even for users not defined to RACF). We will call these datasets *unprotected*.

In both cases it is desirable to be able to identify the datasets not protected by any profile. In a PROTECTALL environment, a generic profile may have been removed prematurely (i.e. without first deleting or renaming the datasets). In a non-PROTECTALL environment, datasets may be unprotected that should have been protected by your standards.

CONSUL/RACF provides a function to identify all unprotected datasets, and optionally generates commands to add profiles for these resources. Identification of datasets not protected by any profile is requested by the command:

```
VERIFY PROTECTALL
```

An IOCONFIG file is required for this function.

Sample output from this command:

```
CNRACF 0.0.3 02/09/90 17.43   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   9 Feb 1990 17:57
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN:   VERIFY PROTECTALL

CNR132I 00 Configuration for system 309E running MVS/SP2.1.7  (XA)
           created by program IOCNF154 1.5.4 11/21/89 22.13 job ZGC501N    9 Feb 1990 16:24:39.03
CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    02/08/90 17.58 job          at  9 Feb 1990 17:03
           Source dataset   1 was SHR101 SYS2.RACF.PRIM1
           Source dataset   2 was SHR101 SYS2.RACF.PRIM2

CNR005I 00 110505 profiles read, 110505 profiles selected (100%)
CNR087I 00 Number of detail error messages is 10
CNR097I 00 GCWORK has 2 inaccessible dataset(s) (not indicated, no profile)
CNR097I 00 SPG001 has 1 inaccessible dataset(s) (not indicated, no profile)
CNR097I 00 TST001 has 2 inaccessible dataset(s) (not indicated, no profile)
CNR097I 00 TST008 has 1 inaccessible dataset(s) (not indicated, no profile)
CNR082I 08 Inaccessible dataset (not indicated and no generic)  GCWORK @GD679.ICQNAMES.DIR
CNR082I 08 Inaccessible dataset (not indicated and no generic)  GCWORK @GD679.SPFLOG1.LIST
CNR082I 08 Inaccessible dataset (not indicated and no generic)  SPG001 @GD512.DSNCONV2.INPUT
CNR082I 08 Inaccessible dataset (not indicated and no generic)  TST001 #GDJPL.JPL111.AI206KVP.BSJMUL01
CNR082I 08 Inaccessible dataset (not indicated and no generic)  TST001 #GDJPL.JPL111.TOTAALJP.BSJMUL01
CNR082I 08 Inaccessible dataset (not indicated and no generic)  TST008 DG.S005.DATMUGIK.W081NR01
```

Fig 11. Sample VERIFY PROTECTALL output

Note that the error messages on dataset level are preceded by summary messages per volume. You may modify the order of the messages by means of the BY parameter of VERIFY. Refer to the reference section on VERIFY.

Currently, no commands are generated to generate profiles for datasets that are not RACF-indicated and not covered by any profile.

Note that generating discrete profiles would not have the intended effect as to reachability: no group has any access on the profiles unless the issuer of the commands is connected to the group indicated by the first qualifier and has the GRPACC attribute on his USER or CONNECT profile. It is better to define a generic profile and PERMIT the groups requiring access. A future version of CONSUL/RACF may provide better support in this area.

1.3.7 Removing unused discrete profiles

When operating an Automatic Dataset Protection (ADSP) environment, sooner or later there will exist a number of discrete profiles for which no corresponding dataset exists. This happens mostly through volume-level operations for which authorization is granted by APF and often in addition by checks in the DASDVOL class. These volume-level operations often do not support discrete profiles. The problem is not necessarily limited to ADSP environments, since by default (i.e. without special RACF exit processing) any user can specify PROTECT=YES in his JCL and/or issue the ADDSD command, at least for datasets starting with his userid, and for group datasets if he is connected with CREATE authority.

The unused profiles exist mostly unnoticed in the RACF database. If the volume name is reused and a user tries to allocate a dataset for which a profile was left over, then this will result in the error message RESOURCE ALREADY DEFINED.

The situation can be highly misleading if a dataset exists on the volume with the RACF indicated bit off (meaning that DFP will *not* request a search for a discrete profile), while at the same time a discrete profile does exist. Note that the LISTDSD command of RACF will list the discrete profile for the dataset in spite of the fact that the profile will *not* be used for access decision with regard to the dataset!

Some security specialists will call this a *security exposure*, since users may try to restrict access to the dataset by modifying the access list of the discrete profile, while in reality access is *not* restricted due to the fact that DFP only checks the generic profile.

CONSUL/RACF provides a function for reporting and removing the unused discrete profiles. The command to accomplish this is:

```
VERIFY DATASET
```

The function is called this way because it verifies that a dataset is present for each discrete profile. An equivalent (alias) command but easier to remember is:

```
VERIFY ONVOLUME
```

An IOCONFIG file is required for this function.

Sample output from this command is shown below. The error messages distinguish three cases: no dataset present on volume, volume not mounted, and dataset present but not RACF-indicated. In addition, summary messages per volume are printed.


```
CNRACF 0.0.1 09/27/89 12.21    C O N S U L / R A C F    D A T A B A S E    U T I L I T Y
(C) COPYRIGHT 1989, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: VERIFY ONVOLUME

CNR004I 00 Processing started for SYSUT1
CNR093I 04 EMVS01 has 1 discrete profile(s) for non-RACF indicated datasets
CNR094I 04 EMVS01 has 1 discrete profile(s) without dataset on the volume
CNR095I 04 EPGXX1 has 2 discrete profile(s) but volume not mounted
CNR094I 04 EXN001 has 2 discrete profile(s) without dataset on the volume
CNR093I 04 FMC001 has 4 discrete profile(s) for non-RACF indicated datasets
CNR094I 04 FMC001 has 6 discrete profile(s) without dataset on the volume
CNR090I 04 FMVS01 message limit exceeded - 82 detail message(s) suppressed
CNR095I 04 FMVS01 has 132 discrete profile(s) but volume not mounted
CNR090I 04 PRD008 message limit exceeded - 41 detail message(s) suppressed
CNR095I 04 PRD008 has 91 discrete profile(s) but volume not mounted
CNR094I 04 WORK01 has 3 discrete profile(s) without dataset on the volume
CNR041I 04 Discrete profile found but RACF indicator not set    EMVS01 SMPE.EMVS01.SMPTLOG
CNR041I 04 Discrete profile found but RACF indicator not set    FMC001 SMF1.SMFDUMPF.G0001V00
CNR041I 04 Discrete profile found but RACF indicator not set    FMC001 SMF1.SMFDUMPF.G0002V00
CNR041I 04 Discrete profile found but RACF indicator not set    FMC001 SMF1.SMFDUMPF.G0003V00
CNR041I 04 Discrete profile found but RACF indicator not set    FMC001 SMF1.SMFDUMPF.G0004V00
CNR042I 04 Discrete profile present but no dataset on volume    EMVS01 SMPE.EMVS01.SMPTLOG
CNR042I 04 Discrete profile present but no dataset on volume    EXN001 EEB.C010.VERKIEZA.TK890WD1
CNR042I 04 Discrete profile present but no dataset on volume    EXN001 TR.F013.RF154KNV.GD830812
CNR042I 04 Discrete profile present but no dataset on volume    WORK01 $GDMSA.MSA701.PRTGLN13
CNR042I 04 Discrete profile present but no dataset on volume    WORK01 $GDMSA.MSA701.PRTGLN14
CNR042I 04 Discrete profile present but no dataset on volume    WORK01 $GDMSA.MSA701.TOPTION2
CNR043I 04 Discrete profile present but volume not mounted      EPGXX1 SYS2.LOGREC#E.TRENDSDS.G0285V00
CNR043I 04 Discrete profile present but volume not mounted      EPGXX1 SYS2.LOGREC#E.TRENDSDS.G0289V00
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SMPE.FMVS01.SMPTLOG
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.ADFMAC1
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BLGFMT
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BLGPNLS
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BNJPNL1
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BNJPNL2
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BNJSRC1
CNR043I 04 Discrete profile present but volume not mounted      FMVS01 SYS1.BRODCAST
CNR043I 04 Discrete profile present but volume not mounted      PRD008 $EBGSB.ISE103.CM760E01.G0025V00
CNR043I 04 Discrete profile present but volume not mounted      PRD008 $EBGSB.ISE103.CM760E01.G0026V00
CNR043I 04 Discrete profile present but volume not mounted      PRD008 $EBGSB.ISE103.CM760E01.G0027V00

CNR005I 00 34873 profiles read, 34873 profiles selected (100%)
```

Fig 12. Sample VERIFY ONVOLUME output

Commands can be generated to remove the unused discrete dataset profiles by allocating the CMDOUT file.

For unused single volume discrete profiles, commands will be generated to remove the profile.

For *multivolume* discrete profiles, commands will be generated to remove from the profile the volumes that did not contain the dataset. If the dataset is not present on *any* of the volumes in the profile, then the last command generated will be refused by RACF, since the profile is not multi-volume anymore. A second VERIFY run will generate the command to remove this profile.

1.3.8 Removing unused generic profiles

While an environment with generic profiles contains considerably fewer profiles than an all-discrete environment, discrete profiles do have one big advantage over generics: they are deleted when the dataset is deleted. The complete ban on discrete profiles (as some sites have done) can lead to a proliferation of generic profiles that have been created for the purpose of setting a permit on one dataset, but that have not been deleted when the dataset was deleted.

CONSUL/RACF assists in this problem by providing a function that can automatically generate commands to remove *empty* generic dataset profiles, i.e. dataset profiles that are not used for the protection of any currently existing dataset. Two options exist for this function: removing *all* empty generic profiles, or only generic profiles also covered by a less specific profile. The last option is generally preferable in a PROTECTALL environment, since removal of the last generic profile potentially covering a new dataset will prevent allocation of that new dataset.

For the function to perform properly, an IOCONFIG file is required containing VTOC, VVDS, and BCS (catalog) information. *If the catalog information is not present, then VSAM datasets are left out, which can increase the number of profiles found to be empty.*

In addition, the following sources of datasets are not taken into account:

1. *Archived datasets*
2. *Tape datasets*
3. *GDG base names*

Support for those cases is planned for a future release.

You should not use this function in an ADSP environment with PROTECTALL active, since in that case all datasets will be covered by discrete profiles, resulting in all generics being considered empty. The generics are however required to be able to create new datasets with discrete profiles.

The function is invoked by the following command:

```
VERIFY NOTEMPTY
```

or, if you are sure that you want to have the *last* generic profile removed:

```
VERIFY ALLNOTEMPTY
```

To generate commands, allocate the CMDOUT file.

Sample output is shown below.


```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  23 Apr 1990 17:47
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: VERIFY NOTEMPTY

CNR132I 00 Configuration for system ASX1 running MVS/SP2.2.0  (XA)      with DFP 2.3.0
           created by program IOCNF155 1.5.5 03/26/90 21.17 job SYSPROBZ  26 Mar 1990 21:30:44.69
CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 23 Apr 1990 17:47 running RACF 1.8.1

CNR033I 00  HRF1802.Y00.PRIMARY has 6466 segments in use, 123738 segments free (4% used)
             Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5465 profiles read, 5465 profiles selected (100%)
CNR087I 00 Number of detail error messages is 98
CNR077I 04 Generic profile without matching datasets                CAT1.#00
CNR077I 04 Generic profile without matching datasets                EUSRROB.NEW.*
CNR077I 04 Generic profile without matching datasets                EUSRSCH.AB%D
CNR077I 04 Generic profile without matching datasets                EUSRSCH.AB%D.*
CNR077I 04 Generic profile without matching datasets                EUSRSCH.ABC%
CNR077I 04 Generic profile without matching datasets                EUSRSCH.RACF.*
CNR077I 04 Generic profile without matching datasets                SYSMBLN.GTFTRACE
CNR077I 04 Generic profile without matching datasets                SYSPROB.CCWAN###.*
CNR077I 04 Generic profile without matching datasets                SYSPROB.PRIVATE.*
CNR077I 04 Generic profile without matching datasets                SYSAPPL.CICS8803.WORK*.*
CNR077I 04 Generic profile without matching datasets                SYSAPPL.IDMS8804.*
CNR077I 04 Generic profile without matching datasets                SYSRACF1.CATIFC02
CNR077I 04 Generic profile without matching datasets                SYSRACF1.PASSWORD
CNR077I 04 Generic profile without matching datasets                SYS1.AFDBUD
CNR077I 04 Generic profile without matching datasets                SYS1.AFDIND
CNR077I 04 Generic profile without matching datasets                SYS1.AWINCACC
CNR077I 04 Generic profile without matching datasets                SYS1.BNJ*.*
CNR077I 04 Generic profile without matching datasets                SYS1.CNM.*
CNR077I 04 Generic profile without matching datasets                SYS1.CNM.SA21PARM
CNR077I 04 Generic profile without matching datasets                SYS1.CNM*.*
CNR077I 04 Generic profile without matching datasets                SYS1.COMACC
CNR077I 04 Generic profile without matching datasets                SYS1.COPYXX
CNR077I 04 Generic profile without matching datasets                SYS1.DSIPRF
CNR077I 04 Generic profile without matching datasets                SYS1.INCACC
CNR077I 04 Generic profile without matching datasets                SYS1.INDACC
CNR077I 04 Generic profile without matching datasets                SYS1.SA21.DSIPARM
CNR077I 04 Generic profile without matching datasets                SYS1.TUD.CNMCLST
CNR077I 04 Generic profile without matching datasets                SYS1.VS-FORT.*
CNR077I 04 Generic profile without matching datasets                SYS2.DSPRINT.*
CNR077I 04 Generic profile without matching datasets                SYS2.SYSPMRL.R3-0.*
```

Fig 13. Sample VERIFY NOTEMPTY output

1.3.9 Finding and resetting unnecessary RACF indicated bits

If a dataset is RACF indicated, DFP will request a search for a discrete profile. If the profile is not present, then generic processing will be invoked. The latter process is called *always-call* (i.e. RACF is called to search for a generic profile even if the dataset is not RACF indicated or if the dataset is indicated but no discrete profile was found)[2]. In this way an unnecessary I/O overhead is created by datasets that are RACF indicated but do not have a discrete profile. The normal situation would be that the dataset is *not* RACF indicated, avoiding the overhead of trying to find a discrete in the database before using a generic profile (generics are retained incore once referenced).

The situation is typically created by backup/restore operations without a proper RACF interface for discrete profiles, or by conversions to generic profiles that used DELDSD NOSET to remove the discrete profiles. In addition, a number of 'hard cases' can exist that can only be removed by using alternate IPL parameters, like page datasets and SMF datasets.

CONSUL/RACF provides a function to report on RACF indicated datasets without discrete profile, and can optionally generate commands to add discrete profiles for the datasets. To properly reset the indicators, the discrete profiles can be removed later on with the REMOVE REDUNDANT command discussed in "1.3.12 Finding and removing redundant discrete profiles". The command to invoke the report is:

```
VERIFY INDICATED
```

An IOCONFIG file is required for this function.

Sample output from this command:

```
CNRACF 0.0.0 09/27/89 12.21    C O N S U L / R A C F    D A T A B A S E     U T I L I T Y
(C) COPYRIGHT 1989, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: VERIFY INDICATED

CNR004I 00 Processing started for SYSUT1

CNR092I 04 DCM200 has 1 RACF indicated dataset(s) without profile
CNR092I 04 EMVS01 has 8 RACF indicated dataset(s) without profile
CNR092I 04 EMV001 has 1 RACF indicated dataset(s) without profile
CNR092I 04 EXN001 has 3 RACF indicated dataset(s) without profile
CNR092I 04 WORK01 has 2 RACF indicated dataset(s) without profile
CNR040I 04 RACF indicator set but no discrete profile found for DCM200 DCOM.CXXUNL#B
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 @GD501.TEST.LOAD
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LINKLIST.@SM3100
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LINKLIST.PDS401A
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LINKLIST.SSL7700
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LPALIST.SAVE.PCO2400
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LPALIST.SSL7700
CNR040I 04 RACF indicator set but no discrete profile found for EMVS01 SYS1.LPALIST.SU04800
CNR040I 04 RACF indicator set but no discrete profile found for EXN001 EEB.S003.BRANDGEG.DATA87D3
CNR040I 04 RACF indicator set but no discrete profile found for WORK01 SYS4.@GD553.IDCAMS.OUTPUT
CNR040I 04 RACF indicator set but no discrete profile found for WORK01 SYS4.IPO.@GD239.OUTPUT

CNR005I 00 34873 profiles read, 34873 profiles selected (100%)
```

Fig 14. Sample VERIFY INDICATED output

The messages in the example are generated in their default order (by message type). You can use the BY parameter of VERIFY to change the order.

Commands can be generated to add discrete dataset profiles by allocating the CMDOUT file.

[2] CONSUL/RACF terminology presupposes an *always-call* environment. All systems with DFP have always-call.

1.3.10 Finding user/group/connect inconsistencies

Information about the user/group structure is stored in the RACF database in a redundant way. To be precise, RACF database stores the information 3 times: on the USER profile, on the GROUP profile, and on the CONNECT profile (for non-restructured databases) or in a repeat group in the USER profile (for restructured databases).

Some events can destroy the consistency of these 3 information sources on connects: system down during a RACF command for example, or operating with inconsistent *database range tables* across shared systems.

CONSUL/RACF provides a function to verify the consistency of these 3 information sources. This is done by the following command:

```
VERIFY CONNECT
```

Sample output from this command:

```
CNRACF 0.0.0 09/27/89 12.21    C O N S U L / R A C F    D A T A B A S E    U T I L I T Y
(C) COPYRIGHT 1989, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: VERIFY CONNECT

CNR004I 00 Processing started for SYSUT1

CNR065I 04 Missing userid   @GDO01    on group DMSOS
CNR065I 04 Missing userid   @GD068    on group DMSOS
CNR065I 04 Missing userid   @GD252    on group DMSOS
CNR065I 04 Missing userid   @GD256    on group DMSOS
CNR065I 04 Missing userid   @GD258    on group DMSOS
CNR065I 04 Missing userid   @GD287    on group DMSOS
CNR065I 04 Missing userid   @GD591    on group DMSOS
CNR065I 04 Missing userid   DMSAR     on group DMSOS
CNR065I 04 Missing userid   DMSARCHP  on group DMSOS
CNR065I 04 Missing userid   DMSBACKP  on group DMSOS
CNR065I 04 Missing userid   DMSBACK1  on group DMSOS
CNR065I 04 Missing userid   DMSBACK2  on group DMSOS
CNR065I 04 Missing userid   DMSDISPO  on group DMSOS
CNR065I 04 Missing userid   DMSIMARC  on group DMSOS
CNR065I 04 Missing userid   DMSIXMT   on group DMSOS
CNR065I 04 Missing userid   DMSMERGE  on group DMSOS
CNR065I 04 Missing userid   DMSPACE   on group DMSOS
CNR065I 04 Missing userid   DMSPRDS   on group DMSOS
CNR065I 04 Missing userid   DMSRESTO  on group DMSOS
CNR065I 04 Missing userid   DMSVSAM   on group DMSOS
CNR065I 04 Missing userid   DMSVSAMP  on group DMSOS
CNR065I 04 Missing userid   DMSWORK   on group DMSOS
CNR065I 04 Missing userid   FDRDMP    on group DMSOS
CNR065I 04 Missing userid   FDRFRI    on group DMSOS
CNR065I 04 Missing userid   FDRMON    on group DMSOS
CNR065I 04 Missing userid   FDRRERUN  on group DMSOS
CNR065I 04 Missing userid   FDRSAT    on group DMSOS
CNR065I 04 Missing userid   FDRTHU    on group DMSOS
CNR065I 04 Missing userid   FDRTUE    on group DMSOS
CNR065I 04 Missing userid   FDRWED    on group DMSOS

CNR005I 00 34873 profiles read, 34873 profiles selected (100%)
```

Fig 15. Sample VERIFY CONNECT output

No commands are generated by CONSUL/RACF to remedy the situation, since the best course of action depends on a number of factors (it is for instance undesirable to delete a user profile and add it again if many dataset profiles have to be preserved). Sometimes, a sequence of REMOVE and CONNECT commands will do the job, sometimes however, the BLKUPD utility will be needed. If IBMUSER is one of the userids involved, you should carefully study the RACF documentation (each IPL, the IBMUSER userid will be recreated if it is missing, but this does not necessarily apply to its connects).

1.3.11 Converting to generic profiles

Conversion from an Automatic Dataset Protection (ADSP) environment to a PROTECTALL environment with mainly generic profiles generally includes the following sequence of steps:

1. Define generic profiles for all high-level qualifiers that have the group with ALTER in the access list if the qualifier is a group.
2. Activate PROTECTALL(FAIL).
3. Remove ADSP attributes synchronously with removing production JCL steps containing explicit PERMIT commands. This synchronization may be necessary if production JCL uses explicit PERMIT commands to tailor the access list of datasets (and hence, profiles) newly made in previous steps (poor man's profile modelling) and in addition aborts the production job sequence if a nonzero return code is issued (which would be the case if ADSP were turned off globally - the PERMIT command would issue return code 12 because no profile would exist). Using the system-wide SETROPTS NOADSP is possible if all these conversions can be scheduled at the same time or if your production JCL does not contain dependencies on correct execution of RACF commands on discrete profiles - see also the background item below.
4. Remove discrete profiles with the same properties as the generic profile. These profiles are *redundant.*
5. Define generic profiles for groups of similar profiles not covered by the standard generic profile, and remove the discrete profiles that have become redundant by the addition of the generic profile.

CONSUL/RACF provides assistance in this process by means of the SELECT ADSP (step 3), REMOVE REDUNDANT (step 4), and REPORT NONREDUNDANT (step 5) commands. These are discussed respectively in "1.4.3 Finding profiles with specific attributes", "1.4.6 Reporting non-redundancy reasons for profiles", and "1.3.12 Finding and removing redundant discrete profiles".

Background - removing ADSP

With respect to step 3, removing ADSP, you must be aware that this is difficult to perform on a group-by-group basis if you have activated list-of-group processing. The ADSP attribute used by RACF is *not* the ADSP attribute on the CONNECT for which a dataset profile is being CREATEd, but instead the ADSP attribute of the *current connect group.*

1.3.12 Finding and removing redundant discrete profiles

CONSUL/RACF provides a function specifically designed to aid in the conversion from ADSP to generic profiles. This is done by automatically generating commands to delete discrete profiles that give *similar* ("equivalent") access to that resulting from the combination of the most specific matching generic dataset profile and the most specific matching global table entry (if present).

For a profile to be considered *redundant*, three profile properties are checked:

1. Access requirements (access list, conditional access list, and universal access).
2. Audit requirements (failure audit level, success audit level).
3. Erase-on-scratch requirement.

The latter two items are simply compared to those of the most specific matching generic profile, called the *candidate* profile. They must be equal before the profile is considered redundant. However, the *erase-on-scratch* requirement is not checked if the system wide option ERASE(ALL) or NOERASE is active.

The access requirement comparison is more complicated. Simplest is the *universal access* (UACC) comparison. For a profile to be considered redundant, its UACC must either be equal to that of the candidate profile, or less than ALTER and at the same time less than or equal to the most specific matching entry of the Global Access Table (i.e. a member of the DATASET profile in class GLOBAL).

The *access list* and *conditional access list* comparison takes into account group membership of userids in the list. That is, a userid in the conditional list of a redundant profile may be missing from the conditional access list of the candidate profile, only if one of the connect groups is present with the same access (and the same program name), and no connect groups are present with a higher access (and the same program name).

The command generating function is requested by means of the command:

```
REMOVE REDUNDANT
```

Optionally the number of profiles processed can be limited to a group of datasets by means of SELECT commands, for instance:

```
SELECT QUAL=id
```

The way to display the profiles that are considered non-redundant with an explanation why they are non-redundant, is discussed in "1.4.6 Reporting non-redundancy reasons for profiles".

1.4 Standard reporting tasks

This section gives examples of the use of the REPORT, LIST, and SORTLIST commands, optionally together with SELECT parameters. The REPORT commands give reports with a predefined layout. The LIST command can be used to list exactly the information that you want from RACF profiles. The SELECT and EXCLUDE commands can be used to search for specific kinds of profiles.

1.4.1 Listing profile fields

To list the contents of profile fields that you indicate, CONSUL/RACF provides the LIST and SORTLIST commands. The parameters of both commands are field names. Two names, CLASS and KEY are predefined by CONSUL/RACF. The rest of the field names must be as defined in the *templates* in the RACF database. The most common field names are to be found in the description of the LIST command in section 2.3.5 LIST. A complete list of available field names can be found appendix B. Sample templates, or by issuing the SHOW TEMPLATES command.

As an example, the code to list profile class, name, and standard access list:

```
LIST CLASS, KEY, USERID, USERACS
```

This might for example give the following output:

```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  23 Apr 1990 17:19
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: LIST CLASS, KEY, USERID, USERACS
SYSIN: LIMIT OUT=25

CNR0171 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 23 Apr 1990 17:19 running RACF 1.8.1

USER     IBMUSER
GROUP    SYS1                                          IBMUSER  JOIN
                                                       SYSMSYS  USE
GROUP    VSAMDSET                                      IBMUSER  JOIN
GROUP    SYSCTLG                                       IBMUSER  JOIN
CONNECT  IBMUSER/SYS1
CONNECT  IBMUSER/VSAMDSET
CONNECT  IBMUSER/SYSCTLG
GROUP    SYSADM                                        SYSMFDH  USE
                                                       SYSMJLN  USE
                                                       SYSMJVE  USE
                                                       SYSMSYS  USE
USER     VLRUGVG
GROUP    SYSPROG
GROUP    SYSBASE                                       SYSMFDH  USE
                                                       SYSMJLN  USE
                                                       SYSMJVE  USE
                                                       SYSMSYS  USE
                                                       SYSPROB  USE
                                                       HDY1103  USE
GROUP    SYSNET                                        SYSMCAH  USE
                                                       SYSMHSM  USE
                                                       SYSPMAR  USE
GROUP    SYSDB                                         SYSPAVD  USE
                                                       SYSPMRL  USE
                                                       SYSPROB  USE
                                                       SYSMFDH  USE
                                                       SYSMJLN  USE
                                                       SYSMJVE  USE
GROUP    SYSSTC                                        SYSJES2  USE
                                                       SYSVTAM  USE
                                                       SYSCNM   USE
                                                       SYSTCAS  USE
                                                       SYSINIT  USE
                                                       SYSMNT   USE
                                                       SYSMCP   USE
GROUP    SYSOPR                                        SYSDUMP  USE
                                                       SYSEREP  USE
                                                       SYACMAN  USE
                                                       SYSMRMF  USE
                                                       SYACRST  USE
                                                       SYACDAG  USE
                                                       SYACREL  USE
                                                       SYSPAVD  USE
                                                       SYSPMRL  USE
                                                       SYSPROB  USE
                                                       SYSMFDH  USE
```

Fig 16. Sample LIST output with access list - group profiles

Note that this is probably not what you intended - the start of the database contains groups and users, and for a group profile, the access defines the group-authority, and not a regular resource access level.

In general, issuing just the LIST command will generate more output than you want, since it lists at least one line for each profile in the database. The way to limit the output to some profiles instead of all is by using the SELECT command to define a selection criterion or by means of the LIMIT command to limit the output numerically. For instance:

```
SELECT CLASS=DATASET
LIST CLASS, KEY, USERID, USERACS
```

Sample output for this function:

```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  23 Apr 1990 17:22
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: SELECT CLASS=DATASET
SYSIN: LIST CLASS, KEY, USERID, USERACS
SYSIN: LIMIT OUT=25

CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
          at 23 Apr 1990 17:22 running RACF 1.8.1

DATASET  SYSMJLN.*                                  SYSM      READ
                                                    SYSP      READ
                                                    SYAC      READ
                                                    SYSMSYS   ALTER
DATASET  SYSMSYS.*                                  SYSM      UPDATE
                                                    SYSP      READ
                                                    SYAC      READ
                                                    RCIV      READ
                                                    SYSMSYS   ALTER
                                                    SYSMJLN   ALTER
DATASET  SYACREC.*                                  SYSM      UPDATE
                                                    SYSP      UPDATE
                                                    SYAC      UPDATE
DATASET  SYAC000.*                                  SYSM      UPDATE
                                                    SYSP      UPDATE
                                                    SYAC      UPDATE
DATASET  SYS1.*                                     IBMUSER   ALTER
                                                    SYSBASE   UPDATE
                                                    SYACDAG   ALTER
                                                    SYSMFDH   ALTER
                                                    SYSMJLN   ALTER
                                                    SYSMCAH   UPDATE
                                                    SYS1      ALTER
                                                    SYSMSYS   ALTER
                                                    SYSPCJK   ALTER
DATASET  SYSRACF1.PASSWORD                          SYSMSYS   ALTER
DATASET  SYS1.AFDBUD                                IBMUSER   ALTER
                                                    SYSUSER   UPDATE
                                                    SYSOPR    ALTER
                                                    SYSMSYS   ALTER
DATASET  SYS1.AFDIND                                IBMUSER   ALTER
                                                    SYSUSER   UPDATE
                                                    SYSOPR    ALTER
                                                    SYSMSYS   ALTER
DATASET  SYS1.AWINCACC                              IBMUSER   ALTER
                                                    SYSUSER   UPDATE
                                                    SYSOPR    ALTER
                                                    SYSMSYS   ALTER
DATASET  SYS1.BRODCAST                              IBMUSER   ALTER
                                                    SYSMSYS   ALTER
DATASET  SYS1.CMDPROC                               IBMUSER   ALTER
                                                    SYSBASE   UPDATE
                                                    SYSDB     UPDATE
                                                    SYSAPPL   UPDATE
                                                    SYSMSYS   ALTER
DATASET  SYS1.CMDLIB                                IBMUSER   ALTER
                                                    SYSBASE   UPDATE
```

Fig 17. Sample LIST output with access list - dataset profiles

The width of each column is derived from the templates and from an internal table (see appendix B). You can modify this width by giving the length in parentheses. For instance, since you know that program names are not longer than 8 characters:

```
SELECT CLASS=GLOBAL
LIST CLASS, KEY(8), MEMLST
```


This would print all GLOBAL profiles with their member profiles. CONSUL/RACF automatically formats the option bytes in the members (in the case of GLOBAL, it is the universal access). The following examples shows such output:

```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  23 Apr 1990 17:32
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: SELECT CLASS=GLOBAL
SYSIN: LIST CLASS, KEY(8), MEMLST

CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 23 Apr 1990 17:32 running RACF 1.8.1

GLOBAL   DATASET  &RACUID.*/ALTER
                  SYS1.ISPF.*/READ
                  SYSAPPL.*/READ

CNR033I 00  HRF1802.Y00.PRIMARY has 6466 segments in use, 123738 segments free (4% used)
            Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5465 profiles read, 1 profiles selected (0%)
```

Fig 18. Sample LIST output with explicit length

There are major differences between LIST and SORTLIST with regard to the sequence in which the profiles are listed and in the use of main memory:

- The LIST command outputs the profiles in the order as present in the RACF database or unloaded input file.

  The SORTLIST command sorts the profiles in ascending order according to the sequence of field names on the command.

- The LIST command does not store the profiles in main memory.

  The SORTLIST command stores and sorts the profiles in main memory.

The LIST command is therefore more suited to tasks requiring further postprocessing by computer, while SORTLIST is ideal for making reports based on the selected field names.

The following command shows an example of the standard LISTPROG member supplied in CNR110.CNTL, combining the commands discussed before.


```
CNRACF 1.0.2 06/24/90 12.38   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  26 Jun 1990 17:41       page    1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: print    title='Program profile overview'
SYSIN: select   class=program
SYSIN: sortlist class, key(8), memlst, uacc, userid, useracs

CNR004I 00 Processing started for SYSUT1
          Unloaded by program CNRACF 1.0.2    06/24/90 12.38 job          at 26 Jun 1990 10:56
          Source dataset   1 was SHR101 SYS2.RACF.PRIM1

CNR005I 00 27564 profiles read, 62 profiles selected (0%)
PROGRAM  $CCFP001 SYS1.LINKLIST.NOSMP.SLI3801/SHR102/PADCHK                  READ
PROGRAM  *        SYS2.ROSLINKT/SHR105/NOPADCHK                              READ
                  SYS2.ROSLINK/SHR105/NOPADCHK
                  SYS1.LINKLIST.SSM3100/******/NOPADCHK
                  SYS2.PACOLIB2/SHR102/NOPADCHK
                  SYS1.LINKLIST.NOSMP.COBLIB/SHR101/NOPADCHK
                  SYS2.DMSLINK/GDF101/NOPADCHK
                  SYS1.ISPLOAD/******/NOPADCHK
                  SYS1.LINKLIST.PDF2000/******/NOPADCHK
                  SYS2.LIBRCCFX.V03L08P0.LOADTEST/SPG001/NOPADCHK
                  SYS1.LINKLIST.NOSMP.DMS.V7L7M0.DMSLINK/SHR101/NOPADCHK
PROGRAM  #CALCLIB SYS2.ROSLINKT/SHR105/PADCHK                               READ
                  SYS2.ROSLINK/SHR105/NOPADCHK
PROGRAM  #MAKELIB SYS2.ROSLINKT/SHR105/PADCHK                               READ
                  SYS2.ROSLINK/SHR105/NOPADCHK
PROGRAM  ADDGROUP SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDPRB    READ
PROGRAM  ADSAR003 SYS2.DMSLINK/GDF101/NOPADCHK                              READ
                  SYS1.LINKLIST.NOSMP.DMS.V7L7M0.DMSLINK/SHR101/NOPADCHK
PROGRAM  ADSMI002 SYS2.DMSLINK/GDF101/PADCHK                                READ
                  SYS1.LINKLIST.NOSMP.DMS.V7L7M0.DMSLINK/SHR101/PADCHK
PROGRAM  AG       SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ

CNRACF 1.0.2 06/24/90 12.38   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  26 Jun 1990 17:41       page    2
 Program profile overview
                                                                                     #GDPRB    READ
PROGRAM  DELGROUP SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDPRB    READ
PROGRAM  DELUSER  SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDPRB    READ
PROGRAM  DG       SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDPRB    READ
PROGRAM  DSIOST   SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
PROGRAM  DU       SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDPRB    READ
PROGRAM  EX       SYS1.CMDLIB/******/NOPADCHK                               READ     @GD501    ALTER
PROGRAM  EXEC     SYS1.CMDLIB/******/NOPADCHK                               READ     @GD501    ALTER
PROGRAM  ICHCAG00 SYS1.LINKLIB/******/PADCHK                                NONE
PROGRAM  ICHCDG00 SYS1.LINKLIB/******/PADCHK                                NONE
PROGRAM  ICHCDU00 SYS1.LINKLIB/******/PADCHK                                NONE
PROGRAM  ICHDSM00 SYS1.LINKLIB/******/PADCHK                                NONE     #GDAEP    READ
                                                                                     SYS1      READ
                                                                                     #GDICO    READ
PROGRAM  ICHUT100 SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
                                                                                     #GDICO    READ
                                                                                     #GDPRB    READ
PROGRAM  ICHUT200 SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
PROGRAM  ICHUT300 SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      READ
PROGRAM  ICHUT400 SYS1.LINKLIB/******/PADCHK                                NONE     #GDPRB    READ
                                                                                     SYS1      READ
                                                                                     #GDICO    READ
PROGRAM  ICKDSF   SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      ALTER
                                                                                     DMSOS     READ
PROGRAM  IEHATLAS SYS1.LINKLIB/******/PADCHK                                NONE     SYS1      ALTER
PROGRAM  IEHINITT SYS1.LINKLIB/******/PADCHK                                NONE     #GDTMS    READ
                                                                                     SYS1      ALTER
PROGRAM  LIBRFFR  SYS2.ROSLINK/SHR105/NOPADCHK                              READ     @GD501    ALTER
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  LIBSERV  SYS2.ROSLINK/SHR105/PADCHK                                READ
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  LIBSERVE SYS2.ROSLINK/SHR105/PADCHK                                READ
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  LIBUTIL  SYS2.ROSLINK/SHR105/PADCHK                                READ
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  LOOKLOG  SYS1.LINKLIST.SLU0660/******/PADCHK                       NONE     #GDAEP    READ
                                                                                     SYS1      READ
PROGRAM  ROSCOPY  SYS2.ROSLINK/SHR105/PADCHK                                READ
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  ROSDATA  SYS2.ROSLINK/SHR105/PADCHK                                READ
                  SYS2.ROSLINKT/SHR105/PADCHK
PROGRAM  RTDS6000 CICS.DISOSS34.DSVLOAD/CICS21/PADCHK                       READ
                  CICS.V01L07P0.LOADLIBZ/DCM102/PADCHK
PROGRAM  SASSBEND SYS2.LINKLIB/SHR102/NOPADCHK                              READ
PROGRAM  SASSBSTR SYS2.LINKLIB/SHR102/NOPADCHK                              READ
PROGRAM  SASSINCD SYS2.LINKLIB/SHR102/NOPADCHK                              READ
PROGRAM  STRBCCV  SYS1.LINKLIST.NOSMP.STROBE80/SHR102/NOPADCHK              NONE     #QSOND    ALTER
                                                                                     SYS1      ALTER
PROGRAM  STRBVPH1 SYS1.LINKLIST.NOSMP.STROBE80/SHR102/NOPADCHK              NONE     #QSOND    ALTER
                                                                                     SYS1      ALTER
PROGRAM  TLMAIN   SYS1.LINKLIST.STL3100/******/NOPADCHK                     READ     @GD254    ALTER
                  SYS2.PANLINK/SHR103/NOPADCHK
PROGRAM  TLTSO    SYS1.LINKLIST.STL3100/******/NOPADCHK                     READ     @GD254    ALTER
                  SYS2.PANLINK/SHR103/NOPADCHK
```

Fig 19. Sample output of LISTPROG command member

1.4.2 Finding specific profile field contents

To search for profiles with a specific value in some profile variable field (not a flag field), you can use the general FIELDVALUE selection mechanism (flag fields all have distinct SELECT keywords for the flags - see "1.4.3 Finding profiles with specific attributes"). The selection can generally be made with the following command:

```
SELECT fieldname=fieldvalue
```

Here *fieldname* must be the name of the profile field as present in the templates in the RACF database. The RACF manual called "SYSTEM PROGRAMMING LIBRARY: RACF" contains a listing of these names. However, you can also use the command SHOW TEMPLATES to obtain a listing of the field names that are defined in your database. Appendix B contains sample output of this command for RACF 1.9.

The value *fieldvalue* must be in a format consistent with the type of the field. For instance, an access level can have READ, UPDATE, etc. as values.

The operator field may be different from "=". The following operators are supported:

=	Equal
<	Less than
>	Greater than
<=	Less than or equal
>=	Greater than or equal
<>	Not equal

For example, to select datasets with a universal access greater than or equal to UPDATE, you might give the following commands:

```
SELECT CLASS=DATASET, UNIVACS>=UPDATE
LIST CLASS, KEY, UNIVACS
```

The following figure gives an example of the output:

```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  22 Apr 1990
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: SELECT CLASS=DATASET, UNIVACS>=UPDATE
SYSIN: LIST CLASS, KEY, UNIVACS

CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 22 Apr 1990 23:02 running RACF 1.8.1

DATASET  SYS1.BRODCAST                                  UPDATE
DATASET  SYS1.DIRACC                                    UPDATE
DATASET  EUSRROB.LOGREC                                 UPDATE
DATASET  CAT1.USER*                                     UPDATE
DATASET  SYS2.TPREG.*                                   CONTROL
DATASET  SYS2.ICES.STV4M0.DD2                           UPDATE
DATASET  SYS2.BD0211                                    UPDATE
DATASET  SYS2.PROCESS.*                                 UPDATE
DATASET  SYS2.MARCK2.*                                  UPDATE

CNR033I 00  HRF1802.Y00.PRIMARY has 6462 segments in use, 123742 segments free (4% used)
            Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5461 profiles read, 9 profiles selected (0%)
```

Fig 20. Sample SELECT with field value selection

1.4.3 Finding profiles with specific attributes

The RACF database can be scanned for users with specific attributes by means of keywords on the SELECT command. This cannot easily be done by the fieldvalue selection described in the previous section, since the attributes are flags in flag bytes. The SELECT command aims to provide two keyword parameters for each attribute (flag): one for selection if the flag is on, and one for selection if the flag is off. For example, to find users with the UAUDIT (user-audit) attribute, the command sequence would be:

```
SELECT CLASS=USER, UAUDIT
LIST CLASS, KEY
```

To find users *without* the UAUDIT attribute, the command sequence would be:

```
SELECT CLASS=USER, NOUAUDIT
LIST CLASS, KEY
```

To find users with group-SPECIAL attribute, the commands would be:

```
SELECT CLASS=CONNECT, SPECIAL
LIST CLASS, KEY
```

To select both system-wide and group-special users:

```
SELECT SPECIAL
```

For a complete overview of the numerous attributes supported as well as additional examples, see the reference material in "2.3.11 SELECT and EXCLUDE".

The following example shows part the result of the standard input member LISTPROG provided in the CNR110.CNTL library. It selects PADS datasets by means of the PADS keyword on the SELECT command combined with restriction to the DATASET class. The SORTLIST command requests a listing of the dataset type, volume serial, datasetname, universal access, standard access list, and conditional access list for the profiles selected. The sort order is defined by the order of the field names (in this case: dataset type, volume serial, dataset name).

```
CNRACF 1.1.b 02/03/91 15.51   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  12 Feb 1991 16:35
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN:  /*****************************************************************/
SYSIN:  /* EXEC CNRACFL,MEMBER=LISTPROG or concatenated   CONSUL/RACF 1.1.0 */
SYSIN:  /* Program Profile Overview                                      */
SYSIN:  /* Program Accessed Dataset Overview                             */
SYSIN:  /*****************************************************************/
SYSIN: newlist
SYSIN:    print    title='Program profile overview'
SYSIN:    select   class=program
SYSIN:    sortlist class, key(8), memlst, uacc, userid, useracs
SYSIN: newlist
SYSIN:    print    title='Program Accessed Dataset overview'
SYSIN:    select   pads, class=dataset
SYSIN:    sortlist key, volser, dstype, univacs, userid, useracs,
SYSIN:             user2acs, progacs, program

CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 1.1.B    02/03/91 15.51 job CFOASHC  at 26 Feb 1991 10:56
           Source dataset   1 was SHR101 SYS2.RACF.PRIM1
           Non-restructured database format

CNR005I 00 27564 profiles read, 18 profiles selected (0%)
```

```
CNROUPUT CNRACF 1.1.b 02/18/91 17.07 C O N S U L / R A C F   P R O F I L E   L I S T I N G  24 Feb 1991 23:34
 Program Accessed Dataset overview

DGDCCF.*.HISTMAST*                                NONE    #GDSMT   ALTER   *         UPDATE  $CCFP001
                                                                            *         UPDATE  $CCFB001
                                                                            *         UPDATE  $CCFB002
                                                                            *         UPDATE  $CCFB003
DGDCCF.IAD000.SYSTFILE                            NONE    @GD100   ALTER   *         UPDATE  $CCFB001
                                                                            *         UPDATE  $CCFB002
                                                                            *         UPDATE  $CCFB003
                                                                            *         UPDATE  $CCFB007
                                                                            *         UPDATE  $CCFB009
                                                                            *         UPDATE  $CCFB045
                                                                            *         UPDATE  $CCFB100
GM.W328.TMSDATA.LISTTAPE                PRD305    NONE    @GD545   READ    *         READ    ZL051
                                                          #GDSBH   ALTER
                                                          @GD258   ALTER
PM#.WJ50.UCC7.COMMDS                              NONE    #GDAEP   UPDATE  *         UPDATE  SASSINCD
                                                          #GDSBH   ALTER   *         UPDATE  SASSBEND
                                                          SYS1     ALTER   *         UPDATE  SASSBSTR
                                                          #GDUCC   ALTER
SYS2.DMSFILES*                                    NONE    SYS1     ALTER   *         UPDATE  ADSMI002
                                                          #GDDMS   UPDATE
                                                          #GDAEP   ALTER
SYS2.RACF*                                        NONE    SYS1     ALTER   #GDPRB    UPDATE  ICHUT400
                                                          #GDPRB   READ
                                                          DMSIXMT  READ
                                                          #GDICO   READ
                                                          #GDAEP   UPDATE
SYS2.ROSLIB*                                      NONE    SYS1     ALTER   *         UPDATE  ROSCOPY
                                                          #GDAEP   ALTER   *         UPDATE  #CALCLIB
                                                          #GDPRB   UPDATE  *         UPDATE  ROSDATA
                                                                           *         UPDATE  #MAKELIB
                                                                           *         UPDATE  LIBRFFR
                                                                           *         UPDATE  LIBSERVE
                                                                           *         UPDATE  ZA150
                                                                           #GDCVO    UPDATE  LIBUTIL
SYS2#.PANTSQ                                      NONE    @GD501   ALTER   *         UPDATE  TLMAIN
                                                          SYS1     ALTER   *         UPDATE  TLTSO
TZ.Y000.@GD501.TEST*                              NONE    @GD501   ALTER   #GDTST   UPDATE  LIBOPEN
                                                          #GDAEP   UPDATE
```

Fig 21. Sample output of LISTPROG command member PADS report

1.4.4 Reporting dataset access outside group

In order to highlight potential access by people outside the scope of the group to which datasets "belong", the following command is available:

```
REPORT OUTOFGROUP
```

This results in a display of group-dataset profiles that have a UACC other than NONE, or an OWNER or access list entry for a user or group outside the group indicated by the first qualifier (as modified by ICHCNX00).

The sample report below displays all "out-of-group" access for the 1st-qualifier group indicated on the QUAL parameter of the SELECT command. Note that if you use selection with the REPORT command, you should only exclude dataset profiles, but not any other profile types, because you must preserve the user/group structure and the GLOBAL class profiles (the SELECT and EXCLUDE processing takes place *before* the REPORT processing - hence the REPORT command can only take into consideration the information passed to it by SELECT and EXCLUDE).

```
CNRACF 0.0.3 02/19/90 13.42   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  20 Feb 1990 12:17                page    1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: print pagelen=60
SYSIN: report outofgroup
SYSIN: select qual=#gdaep

CNR0041 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    02/19/90 13.42 job 2GC501A1 at 20 Feb 1990 11:53
           Source dataset   1 was SHR101 SYS2.RACF.PRIM1
           Source dataset   2 was SHR101 SYS2.RACF.PRIM2

CNR0051 00 109116 profiles read, 17123 profiles selected (15%)
CNR1401 00 Number of profiles referring outside group is 9
```

```
L I S T   O F   P E R M I T S   O U T S I D E   G R O U P  #GDAEP   20 Feb 1990 11:53                                  page    2
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type    Volume Datasetname                          User/group  access  program  UACC    First reason
NONVSAM SHR102 #GDAEP.TESTACCT.GDG                     @GD501   OWNER            NONE    User not in group
                                                    -> @GD993   READ
                                                       #GDAEP   UPDATE
                                                       @GD501   ALTER
MODEL          #GDAEP.TESTACCT.GDG2                    #GDAEP   OWNER            NONE    User not in group
                                                    -> @GD993   READ
                                                       #GDAEP   UPDATE
                                                       @GD501   ALTER
GENERIC        CICS.*                                  #GDAEP   OWNER            READ    Universal access
                                                       #GDAEP   ALTER
NONVSAM CICS21 CICS.DISOSS34.JARS#A                    #GDAEP   OWNER            NONE    Other group
                                                    -> #GDPRB   ALTER
                                                       #GDAEP   UPDATE
GENERIC        CICS#.INTFACE.ROSDISOS.VSQ0#*           @GD151   OWNER            NONE    Other group
                                                    -> *        UPDATE  RTDS6000
                                                    -> *        UPDATE  ZA102#K
                                                    -> *        UPDATE  ZA102
                                                       @GD254   ALTER
                                                       #GDAEP   UPDATE
                                                       @GD151   ALTER
VSAM    DCM102 CICS#.SDFV5R00.SDFITFB                  @GD222   OWNER            NONE    User not in group
                                                    -> @GD302   UPDATE
                                                    -> @GD393   UPDATE
                                                    -> @GD245   UPDATE
                                                    -> @GD333   UPDATE
                                                    -> #GDMSA   UPDATE
                                                       #GDAEP   ALTER
                                                       @GD222   ALTER
GENERIC        DCOM.CBLIB#Y                            @GD099   OWNER            READ    Universal access
                                                       #GDAEP   ALTER
                                                       @GD099   ALTER
GENERIC        DCOM.CBLIB#Z                            @GD144   OWNER            NONE    Other group
                                                    -> #GDDBB   READ
                                                       #GDAEP   ALTER
                                                       @GD144   ALTER
GENERIC        DCOM.CXX#Y                              @GD099   OWNER            NONE    User not in group
                                                    -> @GD403   UPDATE
                                                    -> @GD114   UPDATE
                                                    -> @GD100   UPDATE
                                                    -> @GD132   UPDATE
                                                    -> @GD082   UPDATE
                                                    -> #GDDBB   UPDATE
                                                       #GDAEP   ALTER
                                                       @GD099   ALTER
```

Fig 22. Sample REPORT OUTOFGROUP output

The last column indicates the first reason why a profile was included in the report. There may be more reasons, but only one is spelled out. However, *all* access list entries referring outside the dataset 1st-qualifier group are marked with an arrow. The reasons that can be present are:

`Universal access`	UACC is unequal to NONE, giving access outside the group.
`User not in group`	A user in the access list or OWNER field is not connected to the dataset 1st-qualifier group. If it is a user in the access list, the user will be one of the entries marked with an arrow.
`Other group`	A group in the access list or OWNER field was present different from the dataset 1st-qualifier group. If the group is part of the access list, then it will be one of the entries marked with an arrow.

1.4.5 Reporting non-standard dataset access lists

A report of access granted outside the dataset 1st qualifier group is not very usable if your site distinguishes the grouping of datasets (resources, objects) and the grouping of users (subjects). RACF groups are used for both of these purposes. This distinction is called the concept of *dataset groups*:

> ***Dataset groups*** are normal RACF groups, but without *any* connected user.

This contrasts with *user groups*:

> ***User groups*** are normal RACF groups with users, but without *any* group-dataset profile (and hence, in a PROTECTALL environment, without any datasets).

The rest of the groups might be called *common groups*:

> ***Common groups*** are normal RACF groups, containing *both* user profiles and group-dataset profiles.

Access to the datasets in a dataset group is regulated by a generic profile. In this generic profile, the *user group* that currently "owns" the datasets is listed in the *access list* with ALTER, as well as in the OWNER field. This gives people with group-authority in the user-group (ultimately based on a connect to a user-group) also control over the "owned" dataset group, while eliminating the need for connects to the dataset group. It also makes it a lot easier to transfer ownership of groups of datasets from one owner (a user-group) to another owner (another user-group). Only the ownership of the dataset-group needs to be changed, not all dataset profiles for the dataset-group.

In order to support these kinds of dataset profiles as "default" or "standard", in addition to those profiles containing only the dataset first-level qualifier in the access list, the following report was designed:

```
REPORT NONDEFAULT
```

For group-datasets this will list "out-of-group" access as discussed in the previous section, as well as "out-of-group" access where the 1st qualifier is another group than the OWNER. However, userids in the access list are not checked for connects, but always marked as non-default, unless they are equal to either the OWNER or the 1st qualifier (an explicit PERMIT command must have been used to create this situation, and in this sense they are nondefault).

In addition, user dataset profiles will be listed where other people than the owner are in the access list or in the OWNER field (or UACC not equal to NONE).

If your security standards differ from those implied here you might prefer to see a different interpretation on the "default" settings and your comments on this aspect would be welcomed (direct those to the address shown on the back of the front page of this manual).

The sample output below gives an example of the output of this report. Note that the SELECT command has been used to restrict the scope of the report by excluding most of the dataset profiles (but not any other profile types).


```
CNRACF 0.0.3 02/19/90 13.42   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  20 Feb 1990 12:01                page    1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: print pagelen=60
SYSIN: select qual=#gdaep
SYSIN: REPORT nondefault

CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    02/19/90 13.42 job ZGC501A1 at 20 Feb 1990 11:53
           Source dataset    1 was SHR101 SYS2.RACF.PRIM1
           Source dataset    2 was SHR101 SYS2.RACF.PRIM2

CNR005I 00 109116 profiles read, 17123 profiles selected (15%)
CNR141I 00 Number of non-default profiles found is 17

L I S T   O F   N O N D E F A U L T   P E R M I T S   T O   D A T A S E T S   O F  #GDAEP   20 Feb 1990 11:53              page    1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type    Volume Datasetname                          User/group  access  program  UACC    First reason
NONVSAM SHR102 #GDAEP.TESTACCT.GDG                     @GD501   OWNER            NONE    Not owner or group
                                                    -> @GD993   READ
                                                       #GDAEP   UPDATE
                                                       @GD501   ALTER
MODEL          #GDAEP.TESTACCT.GDG2                    #GDAEP   OWNER            NONE    Not owner or group
                                                    -> @GD993   READ
                                                    -> #GDAEP   UPDATE
                                                    -> @GD501   ALTER
GENERIC        CICS.*                                  #GDAEP   OWNER            READ    Universal access
                                                       #GDAEP   ALTER
NONVSAM CICS21 CICS.STAIRS43.ACCOUNT                   @GDICZ   OWNER            NONE    Missing access
NONVSAM CICS21 CICS.SYSOUT#I                           #GDAEP   OWNER            NONE    Owner access not ALTER
                                                    -> #GDAEP   UPDATE
NONVSAM DCM102 CICS.V01L07P0.GCEITB#I                  @GD222   OWNER            NONE    Not owner or group
                                                       @GD222   ALTER
                                                       #GDAEP   UPDATE
                                                    -> @GD151   ALTER
GENERIC        CICS#.*                                 #GDAEP   OWNER            READ    Universal access
                                                       #GDAEP   ALTER
VSAM    DCM102 CICS#.DISOSS34.CAAPIQ                   @GDICA   OWNER            NONE    Missing access
VSAM    DCM102 CICS#.DISOSS34.CASOCB                   @GDICA   OWNER            NONE    Missing access
VSAM    DCM102 CICS#.DISOSS34.CASRDS                   @GDICA   OWNER            NONE    Missing access
GENERIC        CICS#.INTFACE.ROSDISOS.VSQ0#*           @GD151   OWNER            NONE    Conditional access
                                                    -> *        UPDATE  RTDS6000
                                                    -> *        UPDATE  ZA102#K
                                                    -> *        UPDATE  ZA102
                                                    -> @GD254   ALTER
                                                       #GDAEP   UPDATE
                                                       @GD151   ALTER
VSAM    DCM102 CICS#.INTFACE.ROSDISOS.VSQ0#K           @GD151   OWNER            NONE    Conditional access
                                                    -> *        UPDATE  RTDS6000
                                                    -> *        UPDATE  ZA102#K
                                                    -> *        UPDATE  ZA102
                                                       #GDAEP   UPDATE
                                                       @GD151   ALTER
VSAM    DCM102 CICS#.SDFV5R00.SDFITF                   @GD222   OWNER            NONE    Not owner or group
                                                    -> #GDMSA   UPDATE
                                                    -> @GD302   UPDATE
                                                    -> @GD393   UPDATE
                                                    -> @GD245   UPDATE
                                                    -> @GD333   UPDATE
                                                       #GDAEP   ALTER
                                                       @GD222   ALTER
NONVSAM DCM102 DCOM.CBLIB                              @GD099   OWNER            READ    Universal access
                                                       @GD099   ALTER
                                                    -> #GDVAS   READ
                                                    -> #GDPRS   READ
                                                       #GDAEP   ALTER
                                                    -> @GD029   ALTER
NONVSAM SHR102 GZ.W302.NCP.V05L02P0.LOAD3745           #GDAEP   OWNER            NONE    Owner access not ALTER
                                                    -> #GDAEP   UPDATE
MODEL          TZ.Y000.TSOGC512.MODEL                  #GDAEP   OWNER            NONE    Owner access not ALTER
                                                    -> #GDAEP   UPDATE
                                                    -> @GD512   ALTER
GENERIC        TZ.Y000.TSOGC545.*                      @GD545   OWNER            NONE    Group access
                                                    -> #GDAEP   READ
                                                       @GD545   ALTER
```

Fig 23. Sample REPORT NONDEFAULT output

The last column indicates the first reason why a profile was included in the report. There may be more reasons, but only one is spelled out. However, *all* access list entries present that are considered non-default are marked with an arrow. The reasons that can be present are:

Conditional access	An entry in the conditional access list is always considered non-default.
Group access	The owning group or else the dataset 1st qualifier group must be in the access list of a group dataset profile to be considered default. In addition, the access must be either ALTER or UPDATE. ALTER is the preferred way, especially in a generic profile environment with PROTECTALL active, since otherwise dataset creation is impossible. However, the RACF default set by the GRPACC attribute is UPDATE, therefore this is considered default, too[3]. The entry will be marked with an arrow unless it is missing from the access list.
Missing access	To be considered default, access must be granted either to the first qualifier identity or to the owner, either implicitly (e.g. 1st qualifier is a userid) or explicitly (by means of an access list entry).
More than 1 group	For an access list to be considered default, it may contain only one group, either the 1st qualifier group or the owning group. The groups are marked with an arrow.
Not owner or group	An identity in the access list is only considered default if it is the owner or if it is a group *and* equal to the 1st qualifier. An undefined identity is never considered default. The identity is one of the entries marked with an arrow.
Owner access not ALTER	To be considered default, the identity owning the profile must have ALTER access to his data.
Owner not in group	The owner of a group-dataset profile is a user without connect to the group. Note that the notion of dataset-groups presupposes that ownership is a user-group, not a user.
Universal access	UACC is unequal to NONE, giving access outside the group.
User not owner	The dataset name starts with a userid. To be considered default, the dataset profile owner must be that userid.

Fig 24. Non-default reasons

[3] In most commercial sites with ADSP, the UPDATE is probably overruled by an entry in the Global Access Table giving ALTER access to the group. Users interested in taking this into account should consider use of the REPORT NONREDUNDANT command.

1.4.6 Reporting non-redundancy reasons for profiles

The methods CONSUL/RACF uses to determine profile redundancy were explained in "1.3.12 Finding and removing redundant discrete profiles" - how to use the command: REMOVE REDUNDANT.

It is also possible to request a report indicating the *reason* why profiles were considered non-redundant and this is done by menas of the command:

```
REPORT NONREDUNDANT
```

The SELECT command can be used to restrict the scope of the report by excluding most of the dataset profiles.

> ***You must not exclude profiles from the classes USER, GROUP, and GLOBAL, since these may all be necessary to determine redundancy.***

A convenient way is to restrict the dataset profiles based on the 1st qualifier (or whatever may be returned by ICHCNX00) by adding the command:

```
SELECT QUAL=id
```

If you want to *exclude* specific qualifiers, you must take care not to exclude any non-DATASET profiles. You can do this by limiting the exclude command to dataset profiles:

```
EXCLUDE QUAL=id, CLASS=DATASET
```

An example of the output is given in the next figure (reasons are explained nehind the example).


```
CNRACF 0.0.3 01/31/90 14.47   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   3 Feb 1990 17:31                    page    1
(C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: print pagelen=60
SYSIN: select qual=#gdaep
SYSIN: report nonredundant

CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 0.0.3    01/31/90 14.47 job          at  3 Feb 1990 17:23
           Source dataset   1 was SHR101 SYS2.RACF.PRIM1
           Source dataset   2 was SHR101 SYS2.RACF.PRIM2

CNR005I 00 115029 profiles read, 16437 profiles selected (14%)
CNR900I 00 Of the 362 profiles tested 82 are redundant (22%)
```

```
L I S T   O F   N O N - R E D U N D A N T   D A T A S E T   P R O F I L E S   O F  #GDAEP    3 Feb 1990 17:23              page    2
(C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type    Volume Datasetname                              User/group  access  program  UACC    Success Failure Erase First reason
GENERIC        #GDAEP.*                                    #GDAEP   OWNER            NONE            READ         - candidate -
                                                           #GDAEP   ALTER
NONVSAM SHR102 #GDAEP.TESTACCT.GDG                         @GD501   OWNER            NONE            READ         User no connect
                                                        -> @GD993   READ
                                                           #GDAEP   UPDATE
                                                           @GD501   ALTER
GENERIC        CICS.*                                      #GDAEP   OWNER            READ            READ         - candidate -
                                                           #GDAEP   ALTER
NONVSAM CICS20 CICS.BACKUP.SYSOUT#L                        #GDAEP   OWNER            NONE            READ         User no connect
                                                        -> DMSOS    UPDATE
                                                        -> DMSBACK1 ALTER
                                                           #GDAEP   UPDATE
NONVSAM CICS21 CICS.DISOSS34.JARS#A                        #GDAEP   OWNER            NONE            READ         Extra group
                                                        -> #GDPRB   ALTER
                                                           #GDAEP   UPDATE
NONVSAM CICS21 CICS.DISOSS34.JARS#K                        #GDAEP   OWNER            NONE            READ         Extra group
                                                        -> #GDPRB   ALTER
                                                           #GDAEP   UPDATE
NONVSAM CICS21 CICS.STAIRS43.ACCOUNT                       @GDICZ   OWNER            NONE            READ         Missing group
NONVSAM CICS20 CICS.UCSLIB                                 #GDAEP   OWNER            NONE            READ         Extra group
                                                        -> #GDPRB   READ
                                                           @GD151   ALTER
                                                           #GDAEP   UPDATE
GENERIC        CICS#.*                                     #GDAEP   OWNER            READ            READ         - candidate -
                                                           #GDAEP   ALTER
VSAM    DCM102 CICS#.DISOSS34.CAAPIQ                       @GDICA   OWNER            NONE            READ         Missing group
VSAM    DCM102 CICS#.DISOSS34.CASOCB                       @GDICA   OWNER            NONE            READ         Missing group
VSAM    DCM102 CICS#.DISOSS34.CASRDS                       @GDICA   OWNER            NONE            READ         Missing group
GENERIC        CICS#.INTFACE.ROSDISOS.VSQ0#*               @GD151   OWNER            NONE            READ         - candidate -
                                                           *        UPDATE   RTDS6000
                                                           *        UPDATE   ZA102#K
                                                           *        UPDATE   ZA102
                                                           @GD254   ALTER
                                                           #GDAEP   UPDATE
                                                           @GD151   ALTER
VSAM    DCM102 CICS#.INTFACE.ROSDISOS.VSQ0#K               @GD151   OWNER            NONE            READ         Missing user
                                                           *        UPDATE   RTDS6000
                                                           *        UPDATE   ZA102#K
                                                           *        UPDATE   ZA102
                                                           #GDAEP   UPDATE
                                                           @GD151   ALTER
VSAM    DCM102 CICS#.INTFACE.ROSDISOS.VSQ0#O               @GD366   OWNER            NONE            READ         User privileged
                                                        -> *        UPDATE   ZA102#O
                                                        -> #GDAEP   ALTER
                                                        -> @GD366   ALTER
VSAM    DCM102 CICS#.INTFACE.ROSDISOS.VSQ0#Z               @GD151   OWNER            NONE            READ         Extra group
                                                        -> *        UPDATE   ZA102#Z
                                                           #GDAEP   UPDATE
```



```
L I S T   O F   N O N - R E D U N D A N T   D A T A S E T   P R O F I L E S   O F  #GDAEP    3 Feb 1990 17:23            page    3
(C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type     Volume Datasetname                              User/group  access  program  UACC     Success Failure Erase First reason
                                                            @GD151   ALTER
VSAM     DCM102 CICS#.TSFILE#C                              @GD454   OWNER             NONE              READ         User restricted
                                                         -> @GDICC   UPDATE
                                                            #GDAEP   UPDATE
                                                            @GD454   ALTER
GENERIC          DCOM.$IDOBJ#A                             @GD099   OWNER             NONE              READ         - candidate -
                                                            #GDDBB   UPDATE
                                                            #GDAEP   ALTER
                                                            @GD099   ALTER
NONVSAM  DCM102 DCOM.CBLIBS.V01L04P0                       @GD099   OWNER             READ              READ         Universal acces
                                                            #GDDBB   READ
                                                            #GDAEP   UPDATE
                                                            @GD099   ALTER
GENERIC          DCOM.LOADLIB                              @GD099   OWNER             READ              READ         - candidate -
                                                            #GDAEP   ALTER
                                                            @GD099   ALTER
NONVSAM  DCM102 DCOM.LOADLIBS.V01L04P0                     @GD099   OWNER             READ              READ         Universal acces
                                                            #GDDBB   READ
                                                            #GDAEP   UPDATE
                                                            @GD099   ALTER
NONVSAM  DCM102 DCOM.LOADLIBX.V02L00P0                     @GD029   OWNER             READ              READ         Universal acces
                                                            #GDAEP   UPDATE
                                                            @GD029   ALTER
GENERIC          TZ.*                                      #GDAEP   OWNER             NONE              READ         - candidate -
                                                            #GDAEP   ALTER
NONVSAM  GDRESY TZ.W207.@GD254.RACFTEST                    @GD254   OWNER             NONE              READ     YES Erase
                                                            #GDAEP   UPDATE
                                                            @GD254   ALTER
VSAM     SHR101 TZ.W207.GC090.VIRTFLOP                     SYS1     OWNER             NONE              READ         Extra group
                                                         -> #GDHNI   UPDATE
                                                            #GDAEP   UPDATE
GENERIC          TZ.Y000.TSOGC545.*                        @GD545   OWNER             NONE              READ         - candidate -
                                                            #GDAEP   READ
                                                            @GD545   ALTER
NONVSAM  SPG001 TZ.Y000.TSOGC545.ISPLLIB                   @GD545   OWNER             NONE              READ         Access
                                                         -> #GDAEP   UPDATE
                                                            @GD545   ALTER
NONVSAM  SPG001 TZ.Y000.TSOGC545.ISPPLIB                   @GD545   OWNER             NONE              READ         Access
                                                         -> #GDAEP   UPDATE
                                                            @GD545   ALTER
GENERIC          TZ#.*                                     #GDAEP   OWNER             NONE              READ         - candidate -
                                                            #GDAEP   ALTER
VSAM     SHR101 TZ#.W310.TSOGC144.TESTDMS6                 @GD072   OWNER             NONE              READ         User restricted
                                                         -> @GD099   READ
                                                         -> @GD029   UPDATE
                                                            @GD144   ALTER
                                                            #GDAEP   UPDATE
VSAM     SHR101 TZ#.W310.TSOGC144.TESTDMS7                 @GD072   OWNER             NONE              READ         User restricted
                                                         -> @GD099   READ
                                                         -> @GD029   UPDATE
                                                            @GD144   ALTER
                                                            #GDAEP   UPDATE
```

Fig 25. Sample REPORT NONREDUNDANT output

The column marked 'First reason' helps to determine the reason why the profile is included in the report. In addition, the user/group column contains arrows to point to entries in the access list (or the OWNER field) that make the profile different from the most specific matching generic. While all access list entries are considered for marking with an arrow, the 'first reason' field gives only the first condition that caused inclusion on the report. This condition does not necessarily result in an arrow; neither need the arrow correspond with the first reason listed.

The reasons that can appear in the redundancy report are described below:

- candidate -	This is a generic profile or entry in the global access table that was the most specific matching generic for one of the profiles considered non-redundant.
- redundant -	This is only present if you requested REPORT REDUNDANT. It marks all profiles that were considered redundant.
Access	The access level of a user or group in the access list of this profile was different from the access level in the access list of the candidate profile for the same identity (user or group), and it was not overruled anyway by an entry in the global access table. The entry is one of the entries marked with an arrow.
Audit	The audit requirements are different from the candidate generic profile.
Erase	The erase-on-scratch requirement is different from that of the candidate generic profile.
Extra group	The access list of this profile contains a group that is not present on the access list of the candidate generic profile. It will be one of the entries marked with an arrow.
Missing group	The access list of this profile does not contain a group that is present in the access list of the candidate generic profile. You must look up the candidate profile access list to see which group. No arrow is present.
Missing user	The access list of this profile does not contain a userid that is present in the access list of the candidate generic profile. Nor is that userid given the same access anyway by means of one of his connects. You must look up the candidate profile access list to see which group. No arrow is present.
No generic	There is no matching generic to serve as candidate.
Undefined id	There is an extra user or group present in the access list that is not present in the access list of the candidate generic profile. However, the user or group was not defined.
Used as model	The profile is used as a model on a USER or GROUP profile and hence not redundant.
User no connect	A userid is present in the access list that is not present in the access list of the candidate profile. Nor does that user have a connect to any of the groups present in the access list of the candidate generic profile. The userid will be one of the entries marked by an arrow.
User privileged	A userid is present in the access list that is given more access than the equivalent entry in the access list of the candidate profile (either the userid or the connect group giving the highest access). The userid will be one of the entries marked by an arrow. You will have to look at the candidate profile access list to find the access level involved.
User restricted	A userid is present in the access list that is given less access than the equivalent entry in the access list of the candidate profile (either the userid or the connect group giving the highest access). The userid will be one of the entries marked by an arrow. You will have to look at the candidate profile access list to find the access level involved.
Universal access	UACC is different from the candidate profile's UACC.

Fig 26. Non-redundancy reasons

The REPORT REDUNDANT command can be used to display all profiles, redundant as well as non-redundant, with the dataset resources they cover. The following example shows use of this command to display JES328X queue authorities. The authority to issue JES328X commands or view the message log for specific printers is derived from the authority to either UPDATE or READ a log dataset existing for that remote terminal. The complete authorization structure can be displayed with CONSUL/RACF by the following commands:

```
SELECT CLASS=DATASET, MASK=SYS1.JSXLOG.**
SELECT CLASS=USER; SELECT CLASS=GROUP; SELECT CLASS=CONNECT
REPORT REDUNDANT, DATASET
```


The following figure shows sample output of this command:

```
CNRACF 1.1.2 05/20/91 00.26   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  20 Sep 1991 16:36                page    1
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Input:  SYSIN    JES2.JOB01278.SI000101

     1 |  /* show all profiles and datasets beginning with sys1.jsxlog */
     2 |   select class=dataset, mask=sys1.jsxlog.**
     3 |   select class=user;  select class=group;  select class=connect
     4 |   report redundant, dataset

CNR132I 00 Configuration for system THD1 running MVS/SP2.2.3  (XA)     with DFP 3.1.1
           created by program CNFCOLL 2.0.3 06/22/91 22.19  job RCOPSECR  20 Sep 1991 16:35:45.49
CNR017I 00 Processing started for SYSRAC01 SPRG19 SYS1.M9002.ICH.PRIMARY
           at 20 Sep 1991 16:36 running RACF 1.8.1
           Non-restructured database format

CNR033I 00  SYS1.M9002.ICH.PRIMARY has 10805 segments in use, 119395 segments free (8% used)
           Index uses 0%.  Space beyond 8% never used.
CNR168I 00 Maximum profile length is 1350 bytes for TAPEVOL DFHSM0

CNR005I 00 8121 profiles read, 5471 profiles selected (67%)
CNR142I 00 Of the 4 profiles tested 0 are redundant (0%)
CNR087I 00 Number of detail error messages is 6

L I S T   O F   N O N - R E D U N D A N T   D A T A S E T   P R O F I L E S   O F  ^@^@^@^@^@^@^@^@ 20 Sep 1991 16:36
page    2
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type    Volume Datasetname                              User/group  access  program  UACC    Success Failure Erase First reason
GENERIC        SYS1.JSXLOG.*.**                            SYS1     OWNER            READ            READ        - candidate -
 nvsam  DASD05 SYS1.JSXLOG.JES328X                         SYSNET   ALTER
 nvsam  DASD05 SYS1.JSXLOG.RMT133                          SYSBASE  ALTER
 nvsam  DASD05 SYS1.JSXLOG.RMT134                          JES328X  UPDATE
 nvsam  DASD05 SYS1.JSXLOG.RMT135
 nvsam  DASD05 SYS1.JSXLOG.RMT136
 nvsam  DASD05 SYS1.JSXLOG.RMT137
 nvsam  DASD05 SYS1.JSXLOG.RMT138
 nvsam  DASD05 SYS1.JSXLOG.RMT139
 nvsam  DASD05 SYS1.JSXLOG.RMT140
 nvsam  DASD05 SYS1.JSXLOG.RMT141
 nvsam  DASD05 SYS1.JSXLOG.RMT142
 nvsam  DASD05 SYS1.JSXLOG.RMT143
 nvsam  DASD05 SYS1.JSXLOG.RMT144
 nvsam  DASD05 SYS1.JSXLOG.RMT145
 nvsam  DASD05 SYS1.JSXLOG.RMT146
 nvsam  DASD05 SYS1.JSXLOG.RMT149
 nvsam  DASD05 SYS1.JSXLOG.RMT150
 nvsam  DASD05 SYS1.JSXLOG.RMT151
 nvsam  DASD05 SYS1.JSXLOG.RMT152
 nvsam  DASD05 SYS1.JSXLOG.RMT153
 nvsam  DASD05 SYS1.JSXLOG.RMT154
 nvsam  DASD05 SYS1.JSXLOG.RMT155
 nvsam  DASD05 SYS1.JSXLOG.RMT156
 nvsam  DASD05 SYS1.JSXLOG.RMT157
 nvsam  DASD05 SYS1.JSXLOG.RMT158
 nvsam  DASD05 SYS1.JSXLOG.RMT159
 nvsam  DASD05 SYS1.JSXLOG.RMT160
 nvsam  DASD05 SYS1.JSXLOG.RMT161
 nvsam  DASD05 SYS1.JSXLOG.RMT162
 nvsam  DASD05 SYS1.JSXLOG.RMT163
 nvsam  DASD05 SYS1.JSXLOG.RMT164
 nvsam  DASD05 SYS1.JSXLOG.RMT165
 nvsam  DASD05 SYS1.JSXLOG.RMT166
 nvsam  DASD05 SYS1.JSXLOG.RMT167
 nvsam  DASD05 SYS1.JSXLOG.RMT168
 nvsam  DASD05 SYS1.JSXLOG.RMT169
 nvsam  DASD05 SYS1.JSXLOG.RMT170
 nvsam  DASD05 SYS1.JSXLOG.RMT171
 nvsam  DASD05 SYS1.JSXLOG.RMT172
 nvsam  DASD05 SYS1.JSXLOG.RMT173
 nvsam  DASD05 SYS1.JSXLOG.RMT174
 nvsam  DASD05 SYS1.JSXLOG.RMT175
 nvsam  DASD05 SYS1.JSXLOG.RMT176
 nvsam  DASD05 SYS1.JSXLOG.RMT177
 nvsam  DASD05 SYS1.JSXLOG.RMT178
 nvsam  DASD05 SYS1.JSXLOG.RMT179
GENERIC        SYS1.JSXLOG.RMT1                            RCOP     OWNER            NONE            READ        Universal acces
 nvsam  DASD05 SYS1.JSXLOG.RMT1                            JES328X  UPDATE
                                                           RCOPTRY  UPDATE
GENERIC        SYS1.JSXLOG.RMT147                          SYS1     OWNER            READ            READ        Extra group
 nvsam  DASD05 SYS1.JSXLOG.RMT147                       -> ETST     UPDATE
                                                           SYSNET   ALTER
                                                           SYSBASE  ALTER
                                                           JES328X  UPDATE
GENERIC        SYS1.JSXLOG.RMT148                          SYS1     OWNER            NONE            READ        Universal acces
 nvsam  DASD05 SYS1.JSXLOG.RMT148                          SYSNET   ALTER
                                                           SYSBASE  ALTER
                                                           JES328X  UPDATE
                                                           WWBOTJS  UPDATE
                                                           RCCSHSM  UPDATE
```

Fig 27. Sample REPORT REDUNDANT output JES328X

1.4.7 Reporting user or group scope

The scope of a user or group can be requested by the following command:

```
REPORT SCOPE=id
```

where *id* is the user or group. The philosophy of the REPORT SCOPE command is to show the profiles and optionally the resources to which the user or group *directly* or *indirectly* has access (indirect access is for instance the ability to modify a profile to give oneself direct access). Because of this, group-special and group-operations authority are taken into account, but group-audit is *not* taken into account. To prevent inadvertent listing of all or almost all profiles in the system, the system-wide special and operations attributes are *not* taken into account. Access through universal access and warning mode of profiles is taken into account. The following is an example of the output:

```
CNRACF 0.0.3 03/11/90 01.00   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   19 Mar 1990 22:34
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: REPORT SCOPE=EUSRSCH

CNR017I 00 Processing started for SYSRAC01 SPRG15 HRP1802.Y00.PRIMARY
           at 19 Mar 1990 22:34

CNR033I 00  HRP1802.Y00.PRIMARY has 6397 segments in use, 123807 segments free (4% used)
            Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5425 profiles read, 5425 profiles selected (100%)
CNR143I 00 Number of profiles in selected scope is 95
```

```
S C O P E   R E P O R T   F O R   I D  EUSRSCH  19 Mar 1990 22:34                    page    2
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

Class    Profile name                                  Volume Access  Via       Program
DATASET  CAT1.*                                              READ    - UACC -
DATASET  CAT1.USER*                                          UPDATE  - UACC -
DATASET  CNM.*                                               READ    - UACC -
DATASET  EUSRHOU.APF.LOAD                                    UPDATE  EUSR
DATASET  EUSRROB.*                                           ALTER   SYSPROB1
DATASET  EUSRROB.LOGREC                                      ALTER   SYSPROB1
DATASET  EUSRROB.SPF.LOAD                                    ALTER   SYSPROB1
DATASET  EUSRSCH.*                                           OWNER   EUSRSCH
DATASET  EUSRSCH.*.CNTL                                      OWNER   EUSRSCH
DATASET  EUSRSCH.AB%D                                        OWNER   EUSRSCH
DATASET  EUSRSCH.AB%D.*                                      OWNER   EUSRSCH
DATASET  EUSRSCH.ABC%                                        OWNER   EUSRSCH
DATASET  EUSRSCH.RACF.*                                      OWNER   EUSRSCH
FACILITY $SUBMITBY.R.CPOASX1.R19                             UPDATE  EUSR
DATASET  GIM.*                                               READ    - UACC -
DATASET  ICQ.*                                               READ    - UACC -
DATASET  IPO1.*                                              READ    - UACC -
DATASET  ISP.*                                               READ    - UACC -
DATASET  ISR.*                                               READ    - UACC -
DATASET  SYACREC.*                                           READ    - UACC -
DATASET  SYAC000.*                                           READ    - UACC -
DATASET  SYSMBLN.*                                           READ    - UACC -
DATASET  SYSPMCS.*.*.*.LOAD                                  READ    - UACC -
DATASET  SYSPMCS.*.*.LOAD                                    READ    - UACC -
DATASET  SYSPMCS.*.LOAD                                      READ    - UACC -
DATASET  SYSPROB.*                                           ALTER   SYSPROB1
DATASET  SYSAPPL.*                                           READ    - UACC -
DATASET  SYSAPPL.CICS8803.*                                  READ    - UACC -
DATASET  SYS1.*                                              READ    - UACC -
DATASET  SYS1.AFDBUD                                         READ    - UACC -
DATASET  SYS1.AFDIND                                         READ    - UACC -
DATASET  SYS1.AWINCACC                                       READ    - UACC -
DATASET  SYS1.BNJ*.*                                         READ    - UACC -
DATASET  SYS1.BRODCAST                                       UPDATE  - UACC -
DATASET  SYS1.CMDLIB                                         READ    - UACC -
DATASET  SYS1.CMDPROC                                        READ    - UACC -
DATASET  SYS1.CNM.*                                          READ    - UACC -
DATASET  SYS1.CNM.SA21PARM                                   READ    - UACC -
DATASET  SYS1.CNM*                                           READ    - UACC -
DATASET  SYS1.CNM*.*                                         READ    - UACC -
DATASET  SYS1.COBLIB                                         READ    - UACC -
```

Fig 28. Sample REPORT SCOPE= output

The column "Via" shows the userid or group in the access list entry that gave the access indicated. It can also be the OWNER, "- UACC -", or "- WARN -".

The dataset resources covered by the profiles can be added to the report by adding the DATASETS keyword:

```
REPORT SCOPE=id, DATASETS
```

The DATASETS options requires IOCONFIG file input. *VSAM datasets are not listed if no catalog dump was included.* The following figure illustrates the DATASETS keyword.

```
CNRACF 1.1.b 02/03/91 15.51   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y   12 Feb 1991 16:35
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: report scope=ccis, datasets

CNR132I 00 Configuration for system IPO1 running MVS/SP2.2.3  (XA)     with DFP 3.1.1
           created by program CNFCOLL 2.0.0 01/19/91 18.09  job GRACTSOA  12 Feb 1991 15:54:48.22
CNR004I 00 Processing started for SYSUT1
           Unloaded by program CNRACF 1.1.a    01/26/91 23.45 job GRACTSOA at 12 Feb 1991 15:54
           Source dataset   1 was SYSV19 SYS1.M9002.ICH.PRIMARY
           Non-restructured database format

CNR005I 00 6026 profiles read, 6026 profiles selected (100%)
CNR143I 00 Number of profiles in selected scope is 579
```

```
S C O P E   R E P O R T   F O R   I D  CCIS     12 Feb 1991 15:54                          page    2
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

Class     Type     Profile name                                  Volume Access  Via       When
FACILITY           $JOBCLASS.*                                          READ    - UACC -
FACILITY           $JOBCLASS.P                                          ALTER   - WARN -
DATASET   GENERIC  CAT1.*                                              READ    - UACC -
           clustr  CAT1.N9006.#00                                TSO005
            index  CAT1.N9006.#00.CATINDEX                       TSO005
            data   CAT1.N9006.#00                                TSO005
           clustr  CAT1.SMS1                                     SYSV22
            index  CAT1.SMS1.CATINDEX                            SYSV22
            data   CAT1.SMS1                                     SYSV22
DATASET   GENERIC  CAT1.#00                                             READ    - UACC -
DATASET   GENERIC  CAT1.USER*                                           UPDATE  - UACC -
           clustr  CAT1.USER1                                    TSO006
            index  CATINDEX.T33B69F0.VID87085.T9C7A4FC           TSO006
            data   CAT1.USER1                                    TSO006
           clustr  CAT1.USER2                                    TSO006
            index  CATINDEX.TCF0EB1E.VID89046.T9FDFC7F           TSO006
            data   CAT1.USER2                                    TSO006
PROGRAM            IOCNF156                                             READ    - UACC -
DATASET   GENERIC  IPO1.*                                               CONTROL CCIS
DATASET   GENERIC  ISP.*                                                READ    - UACC -
           nvsam   ISP.V3R1M0.ISPLOAD                            SYSV19
           nvsam   ISP.V3R1M0.ISPMLIB                            SYSV19
           nvsam   ISP.V3R1M0.ISPPLIB                            SYSV19
           nvsam   ISP.V3R1M0.ISPSLIB                            SYSV19
           nvsam   ISP.V3R1M0.ISPTLIB                            SYSV19
DATASET   GENERIC  SYSPMCS.*.*.*.LOAD                                   READ    - UACC -
           nvsam   SYSPMCS.P.MICS.USER.LOAD                      MICS00
           nvsam   SYSPMCS.T.MICS.USER.LOAD                      MICS00
           nvsam   SYSPMCS.V.MICS.USER.LOAD                      MICS00
DATASET   GENERIC  SYSPMCS.*.*.LOAD                                     READ    - UACC -
           nvsam   SYSPMCS.MICS.PSP.LOAD                         MICS00
           nvsam   SYSPMCS.MICS.TEST.LOAD                        MICS00
           nvsam   SYSPMCS.MICS.USER.LOAD                        MICS00
DATASET   GENERIC  SYSPMCS.*.LOAD                                       READ    - UACC -
           nvsam   SYSPMCS.MICS.LOAD                             MICS00
           nvsam   SYSPMCS.UGA.LOAD                              MICS00
DATASET   GENERIC  SYS1.*                                               READ    - UACC -
           clustr  SYS1.PAGE.OVFL00                              SYSV22
            data   SYS1.PAGE.OVFL00.DATA                         OVFL00
           clustr  SYS1.PAGE.VSYSV22.COMMON                      SYSV22
            data   SYS1.T995545C.VDD90164.TA23F2FD               SYSV22
           clustr  SYS1.PAGE.VSYSV22.LOCAL1                      SYSV22
            data   SYS1.TB5441AA.VDD90164.TA23F2FE               SYSV22
           clustr  SYS1.PAGE.VSYSV22.PLPA                        SYSV22
            data   SYS1.T25DBB72.VDD90170.TA24683F               SYSV22
           clustr  SYS1.STGINDEX                                 SYSV22
            index  SYS1.T7EBBE9A.VID90164.TA23F2FD               SYSV22
            data   SYS1.T7EBBA70.VDD90164.TA23F2FD               SYSV22
TSOPROC            TSOPROC1                                             READ    - UACC -
TSOPROC            TSOSM1                                               READ    - UACC -
```

Fig 29. Sample REPORT DATASETS output

1.4.8 Verifying the protection of sensitive datasets

A common audit function is to verify that sensitive datasets are adequately protected. For some datasets, updates must be tightly controlled (like APF datasets, RACF datasets, and page/swap datasets), for other datasets read access to the information must be tightly controlled (RACF database, page/swap datasets). CONSUL/RACF recognizes some sensitive datasets automatically, and verifies protection according to either *confidentiality* or *integrity* demands. These are:

- APF datasets (integrity)
- Page datasets (confidentiality and integrity)
- Swap datasets (confidentiality and integrity)
- RACF datasets (confidentiality and integrity)
- JES2 checkpoint datasets (confidentiality and integrity)
- SMF datasets (confidentiality and integrity)
- Other system datasets: SYS1.NUCLEUS and SYS1.LPALIB (integrity).

More system datasets will be added in future releases.

The command to verify that adequate protection exists is:

```
REPORT SENSITIVE
```

The security policy adhered to for *confidential* datasets is:

1. Confidential datasets must be protected against READ.
2. READ access must be audited.
3. Information must be erased physically when deleted.

The security policy adhered to for *integrity-sensitive* datasets is:

1. Integrity-sensitive datasets must be protected against UPDATE.
2. UPDATE access must be audited.

For the check on audit trail, both AUDIT and GLOBALAUDIT are checked. The check is passed if any of them provides the required level of auditing.

The following figure gives an example of the output.


```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  22 Apr 1990 22:32                    page    1
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: REPORT SENSITIVE

CNR132I 00 Configuration for system ASX1 running MVS/SP2.2.0  (XA)      with DFP 2.3.0
           created by program IOCNF155 1.5.5 03/26/90 21.17 job SYSPROBZ  26 Mar 1990 21:30:44.69
CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 22 Apr 1990 22:32 running RACF 1.8.1

CNR033I 00  HRF1802.Y00.PRIMARY has 6462 segments in use, 123742 segments free (4% used)
            Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5461 profiles read, 5461 profiles selected (100%)
CNR087I 00 Number of detail error messages is 29
```

```
S E N S I T I V E   D A T A S E T   P R O T E C T I O N   O V E R V I E W 22 Apr 1990 22:32                                 page    2
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Type     Volume Datasetname                          User/group  access  program  UACC    Success Failure Erase Shortcomings
GENERIC         EUSRHOU.APF.LOAD                       SYSMJLN  OWNER            NONE            READ          No update audit
 nvsam   DASD05 EUSRHOU.APF.LOAD                       SYSMSYS  ALTER
                                                       SYSM     UPDATE
                                                       SYSP     UPDATE
                                                       EUSR     UPDATE
GENERIC         HRF1802.*                              SYSMGT   OWNER            NONE            READ          No read audit
 nvsam   SPRG15 HRF1802.Y00.PRIMARY                    SYSPSEC  READ                                            No update audit
                                                       SYSPAUD  READ                                            No erase
                                                       SYSPBRP  READ
                                                       SYSMSYS  ALTER
                                                       SYSMJLN  ALTER
GENERIC         M8904I.*                               SYSMGT   OWNER            NONE            READ          No update audit
 nvsam   SPRG14 M8904I.LINKLIB                         SYSMSYS  ALTER
                                                       RCIV     READ
                                                       SYSM     READ
                                                       SYAC     READ
                                                       SYSBASE  UPDATE
GENERIC         SYS1.*                                 SYS1     OWNER            READ            READ          No update audit
 nvsam   SPRG13 SYS1.SVCLIB                            SYSPCJK  ALTER
 nvsam   SPRG13 SYS1.LINKLIB                           SYSMSYS  ALTER
 nvsam   SPRG15 SYS1.TEST.LINKLIB                      SYS1     ALTER
 nvsam   DASD06 SYS1.ASM2.V310.LOAD                    SYSMCAH  UPDATE
 nvsam   DASD06 SYS1.TALEN.LINKLIB                     SYSMJLN  ALTER
 nvsam   SPRG16 SYS1.ISPF.M8904.ISPLLIB.RC             SYSMFDH  ALTER
 nvsam   SPRG16 SYS1.ISPF.M8904.ISPLLIB                SYACDAG  ALTER
 nvsam   SPRG14 SYS1.VSF2LOAD                          SYSBASE  UPDATE
 nvsam   SPRG14 SYS1.VSF2COMP                          IBMUSER  ALTER
 nvsam   SPRG14 SYS1.GDDMLOAD
GENERIC         SYS1.CNM*                              SYSMJLN  OWNER            READ            READ          No update audit
 nvsam   SPRG14 SYS1.CNMLINK                           SYSMSYS  ALTER
                                                       SYSNET   UPDATE
                                                       SYSBASE  UPDATE
                                                       SYSMJLN  ALTER
GENERIC         SYS1.NCPLIB                            IBMUSER  OWNER            READ            UPDATE        No update audit
 nvsam   SPRG16 SYS1.NCPLIB                            SYSMSYS  ALTER
                                                       SYSBASE  UPDATE
                                                       SYSNET   UPDATE
                                                       SYSMJLN  ALTER
GENERIC         SYS1.NLDMLIB                           IBMUSER  OWNER            READ            UPDATE        No update audit
 nvsam   SPRG14 SYS1.NLDMLIB                           SYSMSYS  ALTER
                                                       SYSBASE  UPDATE
                                                       SYSNET   UPDATE
                                                       SYSMJLN  ALTER
GENERIC         SYS1.NPDALIB                           IBMUSER  OWNER            READ            UPDATE        No update audit
 nvsam   SPRG14 SYS1.NPDALIB                           SYSMSYS  ALTER
                                                       SYSBASE  UPDATE
                                                       SYSNET   UPDATE
                                                       SYSMJLN  ALTER
GENERIC         SYS1.TEST.NCPLIB                       IBMUSER  OWNER            READ            UPDATE        No update audit
 nvsam   SPRG16 SYS1.TEST.NCPLIB                       SYSMSYS  ALTER
                                                       SYSBASE  UPDATE
                                                       SYSNET   UPDATE
```

Fig 30. Sample REPORT SENSITIVE output

1.4.9 Verifying the protection of AC=1 APF modules

In a safe system, the security policy regarding the use of utilities that have the potential to bypass RACF and MVS protection mechanisms must be one of protection-by-default. Only after a risk analysis has been performed, should universal access be allowed to such utilities.

In an MVS system, it can prove surprisingly difficult to determine the exact protection of AC=1 APF modules. For instance, it is not uncommon that more than one module with the same name is available in the system. In addition, a number of questions need to be answered to get a full picture of the protection for AC=1 modules. These are:

1. Do the DATASET profiles covering the APF datasets containing the module have UACC(NONE)?
2. Is the module covered by a PROGRAM profile in all datasets?
3. Does the PROGRAM profile have UACC(NONE)?
4. Is one of the APF datasets part of the linklist concatenation?
5. Which (APF or non-APF) dataset is the first in the linklist concatenation to contain the module?
6. Is the module present in LPA?
7. Is the module present in MLPA?
8. Is the dataset covered by a global access table entry with READ or higher?
9. Is the dataset profile in warning mode?

CONSUL/RACF can determine the answers to these questions, and deduce the resulting universal access to the module, showing EXECUTE or higher if anybody (not explicitly denied access) can execute it. The command to accomplish this is:

```
REPORT AC1
```

The IOCONFIG file must include PDS directory information for this function. This means that CONSUL/COLLECT must either be run APF authorized with FOCUS=AUDIT or non-APF authorized with FOCUS=AUDIT,PDS=YES and sufficient access on *all* linklist and APF dataset directories.

The following figure shows sample output from the REPORT AC1 command.


```
CNRACF 1.1.b 02/03/91 15.51   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  13 Feb 1991 16:58        page    1
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

SYSIN: report AC1

CNR132I 00 Configuration for system ASX1 running MVS/SP2.2.3  (XA)     with DFP 3.1.1
           created by program CNFCOLL 2.0.0 01/19/91 18.09  job CFOASCHZ   3 Feb 1991 12:06:38.32
CNR017I 00 Processing started for SYSRAC01 SPRG19 SYS1.M9002.ICH.PRIMARY
           at 13 Feb 1991 16:58 running RACF 1.8.1
           Non-restructured database format

CNR033I 00  SYS1.M9002.ICH.PRIMARY has 10011 segments in use, 120189 segments free (7% used)
           Index uses 0%.  Space beyond 7% never used.
CNR168I 00 Maximum profile length is 33102 bytes for GROUP SYS1

CNR005I 00 7611 profiles read, 7611 profiles selected (100%)
```

```
A P F   M O D U L E   P R O T E C T I O N   O V E R V I E W   13 Feb 1991 16:58                                  page    3
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP, THE NETHERLANDS

Module   UACC    AuthAttr Member   Datasetname                                    Volser xLPA Lnk PROGRAM  DATASET profile
AD       READ    AC=1     ICHCAD00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ADDGROUP NONE    AC=1     ICHCAG00 SYS1.LINKLIB                                   SPRG18        1 ADDGROUP SYS1.*
ADDSD    READ    AC=1     ICHCAD00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ADDUSER  READ    AC=1     ICHCAU00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ADFMDF03 READ    AC=1     ADFMDF03 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ADRDSSU  READ    AC=1     ADRDSSU  SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ADRRELVL READ    AC=1     ADRRELVL SYS1.LINKLIB                                   SPRG18        1          SYS1.*
AG       NONE    AC=1     ICHCAG00 SYS1.LINKLIB                                   SPRG18        1 AG       SYS1.*
AHLGTF   READ    AC=1     AHLGTF   SYS1.LINKLIB                                   SPRG18        1          SYS1.*
                 Key 0
AHLVCOFF READ    AC=1     AHLVCOFF SYS1.LPALIB                                    SPRG18 P  1              SYS1.*
AHLVCON  READ    AC=1     AHLVCON  SYS1.LPALIB                                    SPRG18 P  1              SYS1.*
ALD      READ    AC=1     ICHCCD00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ALG      READ    AC=1     ICHCCG00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ALTDSD   READ    AC=1     ICHCCD00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ALTER    READ    AC=1     IDCAM01  SYS1.CMDLIB                                    SPRG18        5          SYS1.CMDLIB
ALTGROUP READ    AC=1     ICHCCG00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ALTUSER  NONE    AC=1     ALTUSER  CFOASYS.APF.LOAD                               DASD05          *        CFOASYS.APF.LOAD
ALTUSER  READ    AC=1     ICHCCU00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
ALU      READ    AC=1     ICHCCU00 SYS1.LINKLIB                                   SPRG18        1          SYS1.*
AMASPZAP NONE    AC=1     AMASPZAP SYS1.LINKLIB                                   SPRG18        1 AMASPZAP SYS1.*
```

Fig 31. Sample REPORT AC1 output

The columns contain the following information:

Module	This is the name of an entry point in the module.	
UACC	This is the consolidated universal access to the module, taking into account PROGRAM profile UACC, DATASET profile UACC, global access table, linklist residency, LPA residency, and warning mode. Special values are:	
	READLPA	The dataset UACC does not allow READ, but module can be read in LPA.
	LOADEXE	The dataset UACC does not allow READ, but the module can be executed, and it can be read by issuing LOAD (this requires that you know the module name).
	HIDDEN	The module UACC is NONE, because it is hidden by a similar-named module concatenated in front of the LPA or linklist concatenation.
	COPY	The module can be read, but it cannot be executed. If its operation does not depend on APF or library residence (PADS), then anyone can access its functionality by copying it to his own load library.
AuthAttr	This column indicates the source of authorization for the module. It can be **AC=1**, **Bypass** (bypass RACF and password security), or **Key=*n*** (system key). The last two come from the Program Property Table (PPT).	
Member	This is the name of the member in the PDS. This differs from the module name in case the module is an alias name.	
Datasetname	This column contains the name of the dataset containing the member. It may contain ***** LPA **** if the module was in the LPA, but could not be found in any of the PDS directories, or if its incore AC=1 attribute is different.	
Volser	Volume serial containing the dataset in the previous column.	
xLPA	This column contains the letter P or M for PLPA or M/FLPA, respectively. In addition, the sequence number in the LPA list concatenation of the dataset is displayed.	
	Between the LPA and Lnk columns, the letter h may be displayed to indicate that the module is not accessible through the LPAlist or linklist concatenation ('hidden').	
Lnk	This column contains the sequence number of the dataset in the linklist concatenation. If the dataset is not part of the linklist, the field will be blank.	
PROGRAM	This column contains the name of the program profile covering the module (if any).	
DATASET profile	This column contains the name of the dataset profile covering the dataset. It may be truncated if your line length is less than 153. In addition to the dataset profile, it may also contain an entry from the global access table.	

1.4.10 Finding all occurrences of a string

A quick scan of the entire RACF database for a string is performed by the following command sequence:

```
SELECT SCAN=string
LIST CLASS, KEY
```

The order of the profiles listed is the order in which they are present in input dataset. Sample output:

```
CNRACF 0.0.6 04/22/90 19.14   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  23 Apr 1990 17:37
 (C) COPYRIGHT 1989, 1990, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: SELECT SCAN=EUSRSCH
SYSIN: LIST CLASS, KEY

CNR017I 00 Processing started for SYSRAC01 SPRG15 HRF1802.Y00.PRIMARY
           at 23 Apr 1990 17:37 running RACF 1.8.1

GROUP     EUSR
USER      EUSRSCH
CONNECT   EUSRSCH/EUSR
DATASET   EUSRSCH.RACF.*
GROUP     SYSPROB1
CONNECT   EUSRSCH/SYSPROB1
DATASET   EUSRSCH.*
DATASET   EUSRSCH.AB%D
DATASET   EUSRSCH.ABC%
DATASET   EUSRSCH.AB%D.*
DATASET   EUSRSCH.*.CNTL

CNR033I 00  HRF1802.Y00.PRIMARY has 6466 segments in use, 123738 segments free (4% used)
            Index uses 0%.  Space beyond 5% never used.

CNR005I 00 5465 profiles read, 11 profiles selected (0%)
```

Fig 32. Sample LIST output with SELECT SCAN

1.4.11 Reporting on "user fields"

Some IBM and third-party software packages store information in the RACF database in the "user fields" USRNM and USRDATA that may present on each profile in each class. No RACF commands exist to display this information. The following example shows how to display information stored in user-fields on profiles in the class USER by CMA-SPOOL.

```
CNRACF 1.1.2 05/20/91 00.26   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  20 Sep 1991 16:41
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

Input:  SYSIN    JES2.JOB01279.SI000101

     1 |    /* show user profiles for CMA-SPOOL */
     2 |    select class=user, usrcnt>0
     3 |    sortlist class, key(8), usrcnt, usrnm, usrflg, usrdata

CNR017I 00 Processing started for SYSRAC01 SPRG19 SYS1.M9002.ICH.PRIMARY
           at 20 Sep 1991 16:41 running RACF 1.8.1
           Non-restructured database format

CNR033I 00  SYS1.M9002.ICH.PRIMARY has 10805 segments in use, 119395 segments free (8% used)
            Index uses 0%.  Space beyond 8% never used.
CNR168I 00 Maximum profile length is 1350 bytes for TAPEVOL DFHSM0

CNR005I 00 8121 profiles read, 19 profiles selected (0%)
```

```
CNROUPUT CNRACF 1.1.0 03/22/91 14.53 C O N S U L / R A C F   P R O F I L E   L I S T I N G  20 Sep 1991

USER     CCRPD26      1 PRNTINFO      ESFDPRT(P656003)
USER     CCSPS07      1 PRNTINFO      ESFDPRT(PCH674)
USER     CCSPS24      1 PHONE         81266
USER     CCSPS35      1 PRNTINFO      ESFDPRT(P053G904)
USER     CCSPS46      1 PRNTINFO      ESFDPRT(P053G904)
USER     CCSPS47      1 PRNTINFO      ESFDPRT(RMT28)
USER     CCSPS48      2               ESFDPRT(P053GB04)
                      PRNTINFO        ESFDPRT(P053GB04)
USER     CCSPS52      1 PRNTINFO      ESFDPRT(P053GB04)
USER     CCSPS55      1 PRNTINFO      ESFDPRT(P053GB04)
USER     CCSPS64      2 PHONE         81265
                      PRNTINFO        ESFDPRT(P053GB04)
USER     CCSPS65      1 PRNTINFO      ESFDPRT(P053GB04)
USER     CCSPS66      2 PHONE         81265
                      PRNTINFO        ESFDPRT(P053G904)
USER     CCSPS67      3 PHONE         81265
                      ACTCODE         K-00758-XXOB-001
                      PRNTINFO        ESFDPRT(P053G904)
USER     CCSPS78      1 PRNTINFO      ESFDPRT(P053G904)
USER     CCTSS25      1 PRNTINFO      ESFDPRT(P722001)
USER     CCTSS34      6               ESFDPRT(P053GB01)
                                      ESFDPRT(P053GB02)
                                      ESFDPRT(P053GB03)
                                      ESFDPRT(P053GB04)
                                      ESFDPRT(P053GB05)
                                      ESFDPRT(P053GB06)
USER     CCTSS44      1 PRNTINFO      ESFDPRT(P722001)
USER     DEMONTC      1 PRNTINFO      ESFDPRT(P506071)
USER     DEMONTN      1 PPDATA        02708DEMONS

CNR039I 00 CNRACF used 1.3 CPU seconds and took 4 wall clock seconds
```

Fig 33. Sample LIST output user-fields CMA-SPOOL

1.5 Handling problems and abends in CONSUL/RACF

Provided that the CONSUL/RACF Installation Instructions are followed and all batch JCL and parameters are correctly tailored to reflect the configuration and standards at your own installation you should not experience any problems in normal usage of CONSUL/RACF. If abend codes *are* encountered whilst executing CONSUL/RACF please refer to "1.5.5 Abends and other problems" for further advice.

If your installation uses hot-standby volumes which are online when CONSUL/RACF is being executed please see "1.5.1 Handling hot-standby volumes" for details of the SUPPRESS command which you can use to cause CONSUL/RACF to ignore those volumes.

Installations using Alternate Master Catalogs should review sections "1.5.2 Handling alternate master catalogs" and "1.5.3 Handling catalog/VVDS inconsistencies" to understand the problems which can occur in DFP, RACF, and CONSUL/RACF in this type of environment.

Finally, the possibility exists that past system failures, software errors, etc. may have caused inconsistencies in the RACF database which you are unaware of but which CONSUL/RACF will highlight for you. Advice on the necessary actions to take can be found in "1.5.4 Handling database layout problems".

1.5.1 Handling hot-standby volumes

A difficulty in the processing of CONSUL/RACF is introduced by the use of *hot-standby* volumes. These volumes contain a track image copy of another volume, only the volume label is different (to be able to make the copy online). Typically, no profiles are copied. This automatically would result in a large number of error messages by the VERIFY PROTECTALL and VERIFY INDICATED functions.

For this kind of volume, CONSUL/RACF provides a way to suppress all error messages. This is done by means of the following command:

```
SUPPRESS VOLUME=volume
```

More than one command may be present indicating different volumes.

Sample output from this command:

```
CNRACF 0.0.0 09/27/89 12.21    C O N S U L / R A C F    D A T A B A S E     U T I L I T Y
(C) COPYRIGHT 1989, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP

SYSIN: VERIFY INDICATED
SYSIN: SUPPRESS VOL=GDRES1
SYSIN: SUPPRESS VOL=GDRES2
SYSIN: SUPPRESS VOL=GDRES3
SYSIN: SUPPRESS VOL=GDRES4

CNR004I 00 Processing started for SYSUT1

CNR092I 04 DCM200 has 1 RACF indicated dataset(s) without profile
CNR092I 04 EMV001 has 1 RACF indicated dataset(s) without profile
CNR092I 04 EXN001 has 3 RACF indicated dataset(s) without profile
CNR090I 04 GDRES1 suppress request - 166 detail message(s) suppressed
CNR090I 04 GDRES2 suppress request - 104 detail message(s) suppressed
CNR090I 04 GDRES3 suppress request - 70 detail message(s) suppressed
CNR090I 04 GDRES4 suppress request - 270 detail message(s) suppressed
CNR092I 04 WORK01 has 2 RACF indicated dataset(s) without profile
CNR040I 04 RACF indicator set but no discrete profile found for DCM200 DCOM.CXXUNL#B
CNR040I 04 RACF indicator set but no discrete profile found for EMV001 SYS1.VTOCIX.EMV001
CNR040I 04 RACF indicator set but no discrete profile found for EXN001 EEB.S003.BRANDGEG.DATA87D3
CNR040I 04 RACF indicator set but no discrete profile found for WORK01 SYS4.@GD553.IDCAMS.OUTPUT
CNR040I 04 RACF indicator set but no discrete profile found for WORK01 SYS4.IPO.@GD239.OUTPUT

CNR005I 00 34873 profiles read, 34873 profiles selected (100%)
```

Fig 34. Sample SUPPRESS VOLUME= output

1.5.2 Handling alternate master catalogs

Alternate master catalogs generally cause difficulties since by their purpose they provide the same function as the *real* master catalog. This implies that VSAM datasets cataloged in the master catalog are also cataloged in the alternate master catalog. However, DFP and RACF do not support the situation that a VSAM dataset is cataloged in two catalogs. Hence, on the DFP side, a DIAGNOSE of the alternate master catalog will complain: the VVDS points to only one catalog, the other catalog disturbs the one-to-one relation between VVDS and catalogs.

Why is this a problem for RACF, too? This can be explained by considering which profile RACF will check for access to a VSAM dataset: the volume name in the profile is the catalog volume. If the master catalog and its alternate reside on different volumes (this is generally the case to be able to use the alternate if the disk volume of the master catalog fails), then the profile checked by RACF will depend on whether the intended master catalog is used or its alternate. Therefore, a possible *security exposure* may be created if some day the alternate master catalog needs to be used - datasets that were protected by discrete profile when they were created, will *not* be protected anymore by this profile if the alternate master catalog is used, since RACF will look for a different profile.

For CONSUL/RACF, alternate master catalogs pose problems, too. Which profile must be considered the right one? You will have to take special care to look into error messages generated for VSAM datasets cataloged in a master catalog. Often, the problems are made worse by the CBIPO installation process, if this is not followed by the proper IDCAMS commands to merge the IPO master catalog with the master catalog to be used in production. For instance, page datasets are typically cataloged in the wrong catalog, making it impossible for this release of CONSUL/RACF to report correctly on the proper protection. For access without any explicit catalog name (like dataset accesses without STEPCAT), RACF will use the current master catalog volume.

Some problems may be prevented by using the SUPPRESS option to indicate which catalog should *not* be considered:

```
SUPPRESS CAT=datasetname
```


1.5.3 Handling catalog/VVDS inconsistencies

As discussed in the previous section, catalog/VVDS inconsistencies can be expected for VSAM datasets cataloged in the master catalog. Sometimes, other problems exist. CONSUL/RACF will generate error messages for these problems. These should be checked by using the IDCAMS DIAGNOSE command on the catalog and VVDS involved. Special care should be taken with these inconsistencies in relation to some VERIFY commands to identify *unused* generic or discrete profiles. A dataset listed in one of the VVDS inconsistency messages may in some cases (for instance, when using an alternate master catalog) be protected by one of the profiles considered unused by CONSUL/RACF.

1.5.4 Handling database layout problems

Due to system failures, software errors, or other calamities, the RACF database may contain inconsistencies in its internal layout. A number of these inconsistencies will be identified in error messages by CONSUL/RACF. Some inconsistencies may not be intercepted, but may still provide you with unexpected output from CONSUL/RACF.

For instance, CONSUL/RACF may report on a profile that you cannot display with RACF commands. If this is the case, or if you receive messages on invalid segment identifiers or BAM block problems, you can run the RACF utility ICHUT200 (or IRRUT200 for restructured databases) to check the consistency of the database.

CONSUL/RACF includes sample JCL to run ICHUT200 in a fast way. The speed advantage is obtained by copying the RACF database once to VIO and performing the direct access processing of ICHUT200 on the VIO copy. This only works if you are allowed to allocate sufficiently large VIO datasets. The figure below shows the sample JCL.

```
//ICHUT200 PROC DSN=,        RACF database you want to check
//      VIO=VIO              Esoteric unit anme for VIO processing
//*
//*********************************************************************
//* Name:    ICHUT200                          Version: CONSUL/RACF 1.1.0
//* Purpose: Fastest way to check for BAM conflicts in non-RDS database
//* Warning: You must be allowed to allocate sufficiently large VIO ds
//*********************************************************************
//*
//COPYVIO  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=&DSN,DCB=BUFNO=255
//SYSUT2   DD DISP=(NEW,PASS),UNIT=&VIO,SPACE=(CYL,(10,10))
//SYSIN    DD DUMMY
//ANALYZE EXEC PGM=ICHUT200
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD SYSOUT=*
//SYSRACF  DD DISP=OLD,DSN=*.COPYVIO.SYSUT2
//         PEND
// EXEC ICHUT200
//ANALYZE.SYSIN DD *
MAP ALL
END
```

Fig 35. Sample ICHUT200 JCL

Typically, you will find BAM block conflicts for the segments containing the profiles that are giving you problems. If you have a BAM block conflict that shows segments as allocated, while the index does not point to the segments, and CONSUL/RACF erroneously processes the (partially) deleted profile, then you can instruct CONSUL/RACF to skip the profile. To do this, you must know the database number and RBA of the profile. The LIST keywords DB and RBA can help here:

```
SELECT ... /* the offending profile */
LIST CLASS, KEY, DB, RBA
```

To skip the offending profile in subsequent runs, you can explicitly exclude it from processing:

```
EXCLUDE DB=num, RBA=hexnum
```

If your problem is the other way around - you miss a profile that RACF can find but CONSUL/RACF cannot find, then this can only be solved by repairing the BAM block conflict with BLKUPD.

1.5.5 Abends and other problems

The most common abend codes encountered with CONSUL/RACF are listed below with a suggestion for the possible cause and remedy (for CNFCOLL abends, see the CONSUL/COLLECT reference manual). Of course your first check should be the appropriate message manual for your operating system, that will tell you the exact meaning of the abend and reason code.

001 Probably problems with blocksize. Look at the message in your joblog to determine the DDname. If you used a concatenation for this DDname, make sure the largest blocksize comes first, or specify the larger blocksize on a DCB=BLKSIZE= parameter on the first DD statement.

002 Problems with the DCB parameters of a file. Look at the message in your job log to determine the DDname. Check your specification for DCB parameters with the reference material in "2.2.1 DDname overview".

322 CPU time limit exceeded. Check the joblog for prior abend messages with a different abend code. If a prior abend occurred, solve this abend. Otherwise, increase the TIME parameter on the JOB card, code less functions together, or split the input (e.g. per volume or only one RACF dataset).

522 Check in the joblog that the job was not waiting for a tape mount or offline or inaccessible device.

722 Too many output lines. Make your selection more specific or increase the output limit for your job (for instance with a `/*JOBPARM L=`*nn* card , where *nn* is thousands of lines allowed).

80A
878 GETMAIN error. Try to increase the REGION parameter on the EXEC or JOB card. If you have reached your site's maximum, code less functions together, or split the input (e.g. per volume or per RACF dataset).

D37
B37 One of the output datasets was too small, or there was no space left on the volume to extend the dataset. Look at the message in your job log to determine the DDname.

Most abends (except some I/O related abends) are accompanied by a summary dump. For assistance on a problem by CONSUL Risk Management, you will generally have to provide at least this summary dump, the JCL used, and the listing of the input commands.

PART 2 Reference

The reference material describes the interactive component, the batch component JCL, and the batch component Command Language. Examples of the use of the commands are given in Part I.

2.1 Interactive component

The interactive component is an ISPF application. It is invoked under ISPF by means of the CLIST CNR:

```
%CNR
```

After this, the primary option menu is displayed:

```
------------------  CONSUL/RACF PRIMARY OPTION MENU ---------------------------
OPTION  ===> _

   1  DATABASES    - Range table and dataset name table display
   2  STC          - Started Procedure Table display
   3  AUTAB        - Authorized Caller Table display
   4  ROUTER       - SAF Router Table display
   5  CLASS        - Class Descriptor Table and class option display

Enter END command to terminate application.

(C) Copyright 1989, 1991, Hans Schoone and CONSUL Risk Management B.V.
```

The first option leads to a display of the database range table ICHRRNG currently in use by the system. Use the scroll keys to scroll forward and backward through the table.

```
CNRIDSNT 1.1.0 ------------ CONSUL/RACF RANGE TABLE DISPLAY ------  ROW 1 OF 1
Command input ===> _                                          Scroll ===> CSR

  Select a range to display the corresponding RACF datasetname table entry.

  Seq   Key range starts (EBCDIC / HEX)
_ 1
       00000000000000000000000000000000000000000000000000000000000000000000000000000000
******************************** BOTTOM OF DATA ********************************
```

The range table display shows the RACF dataset sequence number and the start key of the range that will be directed to this dataset. The key is shown both in characters and in hexadecimal representation (though truncated on most terminal types).

By selecting one of the ranges, the database name table ICHRDSNT will be displayed with the selected sequence number at the top of the display.

```
CNRIDSNT 1.1.0 --------- CONSUL/RACF DATASET NAME TABLE DISPLAY --  ROW 1 OF 1
Command input ===> _                                          Scroll ===> CSR

  Seq Dataset                          Volume #   Rec/  Status
  no  name                             serial buf Trk   flags
  1   SYS1.M9002.ICH.PRIMARY           SPRG19 40  31  Act Prim Mstr Databuf
      SYS1.M9002.ICH.DUPLEX            SPRG22 0   31  Act Mstr
******************************** BOTTOM OF DATA ********************************
```

For each dataset sequence number, two lines are displayed giving the primary and duplex dataset names. In addition, the number of incore buffers (of 1Kbyte for non-RDS and 4Kb for RDS) is displayed, and a number of flags. The meaning of the flags is:

Act	This dataset is currently active (RACF directs I/O towards this dataset).
Prim	This dataset is the primary (RACF directs statistics only to this dataset)
Mstr	This dataset is the master database.
Databuf	The buffers may contain both index and data blocks.
RDS	The dataset is in restructured database format

The volume containing the datasets is also displayed, but this is not part of the database name table. At IPL time, RACF locates the datasets mentioned in the name table through the master catalog and optionally a user catalog.

Option **2** will display the started procedure table ICHRIN03 that is currently in use (use the scroll keys to browse through the table):

```
CNRISTC 1.1.0 ---- CONSUL/RACF STARTED PROCEDURE TABLE DISPLAY --  ROW 1 OF 32
Command input ===> _                                          Scroll ===> CSR

   Procedure   RACF user  RACF group  Flags

  *            =
  CLRDUMP      RCBBMAN    SYSOPR
  CNMAPRC      SYSCNM     SYSSTC      Privileged
  CNMASSI      SYSCNM     SYSSTC      Privileged
  CNMPROC      SYSCNM     SYSSTC      Privileged
  CNMPSSI      SYSCNM     SYSSTC      Privileged
  DAGRCONT     RCBBDAG    SYSOPR
  DAGRPRLG     RCBBDAG    SYSOPR
  DEALLOC      RCBBMAN    SYSOPR
  DFHSM        SYSHSM     SYSSTOR
  DUMP         RCBBDMP    SYSOPR
  EREP         SYSEREP    SYSOPR
  IDMS         CCISIDMS   CCIS
  IDMSAPPL     CCISIDMS   CCIS
  IDP1         CCISIDMS   CCIS
  JEST         SYSJES2    SYSSTC      Privileged
  JES2         SYSJES2    SYSSTC      Privileged
  JES328X      JES328X    SYSAPPL
  LEEGDUMP     RCBBMAN    SYSOPR
```


Option **3** will display the authorized caller table ICHAUTAB. Generally, this should be empty.

Option **4** will display the SAF router table ICHRFR01 (use the scroll keys to browse through the table):

```
11:06   ---------- CONSUL/RACF ROUTER TABLE DISPLAY ------------  ROW 1 OF 60
Command input ===> _                                          Scroll ===> CSR

   Class        Subsystem  Requestor   Action if not RACF

  ACCTNUM
  ACICSPCT
  AIMS
  APPL
  BCICSPCT
  DASDVOL
  DATASET
  DATASET       OCEOV      CLOSE
  DATASET       OCEOV      TAPEEOV
  DATASET       OCEOV      TAPEOPEN
  DATASET       RESTART    TAPERST
  DCICSDCT
  DSNR
  ECICSDCT
  FACILITY
  FACILITY      ABDUMP     ABDUMP
  FCICSFCT
  FIELD
  GCICSTRN
```


Option **5** will display the class descriptor table ICHERCDE as well as the incore class activity settings. Each class can be selected to display a full overview of the class options currently in effect. Use the scroll keys to browse through the table.

```
18:05   ---------- CONSUL/RACF CLASS OPTION OVERVIEW -----------  ROW 12 OF 61
Command input ===> _                                            Scroll ===> CSR
Enter S or / before class to display all class options

  Class     Opt Related classes     Protect   Profile  Dflt   Oper Profiles Global
  name      Pos grouping member     status    type     UACC   OK   resident active
  FACILITY  8                       Noaudit             NONE                 Glob
  FCICSFCT  5   HCICSFCT            Noaudit             NONE        n/a      Glob
  FIELD     121                     Noaudit             NONE
  FIMS      101 HIMS                Inactive Discrete   NONE        n/a
  GCICSTRN  5             TCICSTRN  Noaudit             NONE        n/a      Glob
  GDASDVOL  0             DASDVOL   Noaudit             ACEE   OPER n/a      Glob
  GIMS      4             TIMS      Noaudit             NONE        n/a      Glob
  GLOBAL    6             GMBR      Noaudit             NONE        n/a      Glob
  GMBR      6   GLOBAL              Noaudit             NONE        n/a      Glob
  GTERMINL  2             TERMINAL  Noaudit             ACEE        n/a      Glob
  HCICSFCT  5             FCICSFCT  Noaudit             NONE        n/a      Glob
  HIMS      101           FIMS      Inactive Discrete   NONE        n/a
  JCICSJCT  5   KCICSJCT            Noaudit             NONE        n/a      Glob
  KCICSJCT  5             JCICSJCT  Noaudit             NONE        n/a      Glob
  MCICSPPT  5   NCICSPPT            Noaudit             NONE        n/a      Glob
  MGMTCLAS  123                     Inactive Discrete   NONE
  NCICSPPT  5             MCICSPPT  Noaudit             NONE        n/a      Glob
  OIMS      101 WIMS                Inactive Discrete   NONE        n/a
```

The columns are designed in such a way that the better the protection options, the less intensified fields are on the display. The columns have the following meaning:

Class name	Name of the class in the Class Descriptor Table	
Opt Pos	The POSIT number used for determining the options setting for the class. A RACF command changing class options for a specific class also changes these options for all classes with the same POSIT value.	
Related grouping class	This column is filled in for (member) classes that have a related grouping class.	
Related member class	This column is filled in for (grouping) classes that have a related member class.	
Protect status	This column summarizes a number of options. The most 'offensive' option from the security viewpoint is displayed. The priority order for the displayed values is:	
	Inactive	The class is not active
	Noaudit	Profile changes are not audited
Profile type	If generic profiles are not allowed, then **Discrete** is displayed in this column	
Dflt UACC	This column contains the default universal access for profiles created in this class. It can be ALTER, CONTROL, UPDATE, READ, NONE, or ACEE. The last indication means that RACF will use the default UACC of the user's ACEE.	
Oper OK	This column contains the text OPER if the systemwide OPERATIONS attribute applies to this class. This means that all users with this attribute have access unless specifically denied access.	
Profiles resident	This column summarizes the options available to regulate the incore residency of profiles. The options displayed (in order of priority) are:	
	Nowhere	The class cannot have any profile (PROFDEF=NO in the Class Definition Table)
	RaclReq	The profiles will be resident if the class is active, because RACLIST is required (RACLREQ=YES option in the CDT).
	Raclist	The profiles are resident (by means of the SETROPTS RACLIST option).
	Genlist	The generic profile are resident (by means of the SETROPTS GENLIST option).
	n/a	The profiles cannot be made resident by RACF (through the RACLIST=DISALLOWED option in the CDT). This typically is the case for classes meant to be accessed with RACLIST and FRACHECK by applications.
Global active	This column summarizes the activity of the global access table for the class:	
	Glob	Global access checking is active for the class. This does not mean that there are any profiles on the table.
	n/a	Global access checking is not used for this class, because a SETROPTS RACLIST has been issued.

When a class is selected, the class option panel is displayed. Depending on your RACF release, not all fields may be filled in: options not present in your RACF release are simply left blank.

```
18:06   ------ CONSUL/RACF class FACILITY option display ------- CNRICDT 1.1.0
Command input ===> _                                          Scroll ===> CSR

Profile syntax rules                    Class properties
                    1st     Rest
Alphabetic allowed  Yes     Yes         Class identifier            19
National allowed    Yes     Yes         POSIT (options set id)      8
Numeric allowed     Yes     Yes         Default UACC                NONE
Special allowed     Yes     Yes         Default not-found RC

Maximum length              39          Generic scan limit (quals)

Class activity options                  Profile residency options
Protection active           Yes         Profiles not allowed        No
GLOBAL (fast path) active   Yes         Profiles RACLISTed          No
Generics allowed            Yes         Profiles in dataspace
Generic commands allowed    Yes         Profiles GENLISTed          No
OPERATIONS honored          No          RACLIST required            No

Class audit options                     Manadatory access control properties
Command auditing active     No          SECLABEL required
Statistics collected        Yes         Reverse MAC checking
Logoptions
```

All class options available through CDT and SETROPTS are listed.

2.2 CNRACF Batch JCL

The load module to be executed is CNRACF. Execution is supported under both the VM/CMS and MVS operating systems. However, no support is included under VM/CMS to dynamically allocate the RACF database to be read, nor to read OS formatted minidisks. Hence the SYSRAC*nn* or SYSUT1 DDnames *must* be used under CMS. Under MVS, the load module may be called as a program as well as invoked as a TSO command processor.

Parameters are supported in all three cases (CMS, MVS program and TSO command), and are identical to the commands that can be given through the SYSIN file.

VTOC, VVDS, catalog, and PDS directory data is collected by the separate program CNFCOLL, the main component of CONSUL/COLLECT for MVS. CNFCOLL is supported only under MVS. The IOCONFIG file that is the result of CNFCOLL execution can however be transmitted to a VM/CMS system.

2.2.1 DDname overview

The following DDnames (FILEDEFs under CMS) may be specified:

STEPLIB You should include a STEPLIB to indicate in which library the program is installed (MVS only).

SYSPRINT Add a SYSPRINT DD statement to indicate where the listings and messages should go. Under CMS, the file FILE SYSPRINT will be created automatically on your A disk. The default LRECL will be set depending on other DCB parameters to enable a text line length of 132. To use longer lines, explicitly specify a higher LRECL. The maximum supported LRECL is 255. The RECFM values supported are F, V, FB, VB, FA, VA, FBA, and VBA. The default RECFM is VBA. For 3800-type laserprinters, OPTCD=J may be added to the DCB parameters in combination with CHARS=(*norm,bold*) where *norm* indicates the character set to be used for normal text and *bold* is the character set to be used for bold text. The default blocksize is the largest blocksize that will fit on the output device consistent with LRECL and RECFM.

SYSIN Add SYSIN to specify commands (described in the next chapter). The maximum LRECL supported is 255.

SYSTERM Optionally, the file SYSTERM may be allocated. This file will receive messages only (no reports). Messages echoed on SYSTERM include all messages resulting in a return code of 12 or higher, as well as a number of status messages. Under CMS you must issue a FILEDEF SYSTERM TERMINAL to receive the messages on your terminal. Under TSO you must issue ALLOC F(SYSTERM) DA(*) to receive the messages interactively.

CMDOUT Optionally the file CMDOUT may be allocated. This file will be filled with commands generated as result of REMOVE or VERIFY operations. Allocating the file implies that RACF commands must be generated where possible. DCB parameters set by default are LRECL=255 and RECFM=VB. Blocksize defaults to 32760 or smaller if on DASD with a smaller track size . The first 8 bytes of each line will be blank, since TSO EXEC processing expects line numbers in these positions, and ignores them. The file can be executed as a CLIST.

SYSUT2 Optionally the file SYSUT2 may be allocated. This file will receive one record per profile (or segment with a restructured database) as a result of the UNLOAD operation. Allocating the file implies the UNLOAD operation. Profiles too long for the LRECL will be truncated. Default for LRECL and RECFM is LRECL=X,RECFM=VBS (no limit, so no truncation). If the record format is not variable and spanned, then LRECL will default to the maximum allowed for the BLKSIZE.

Use this file once if you aim to run more analyses, and use SYSUT1 on all subsequent runs. It will result in faster operation and will generate less load on the RACF database.

SYSUT1 Optionally the file SYSUT1 may be allocated. If allocated, it must point to a file with previously unloaded RACF profiles, and no profiles will be read from the RACF database. See the comment on SYSUT2.

SYSRAC01 Optionally, this file may be allocated to indicate where the RACF master dataset or its backup copy may be found. If you omit SYSRAC01, the currently active primary RACF master dataset will be allocated dynamically (as well as other primary datasets to SYSRAC*nn*).

SYSRAC*nn* If SYSRAC01 is allocated too, then SYSRAC02 and up may be allocated to indicate additional non-master RACF datasets (or their backup or archive copies). If SYSRAC01 is omitted and the RACF database consists of more than one dataset, then the remaining currently active primaries (or whatever else has been implied by an ALLOC command) are automatically allocated.

IOCONFIG Optionally, this file may be allocated to indicated where VTOC, VVDS, catalogs, and PDS directories dumped by CNFCOLL may be found. The file is required to perform the VERIFY options DATASET, INDICATED, and PROGRAM.

ddname An arbitrary DDname may be indicated on the DDNAME parameter of the PRINT command for output of the LIST and SORTLIST commands. The same considerations apply as for SYSPRINT. For sample JCL, see "2.2.10 CNRCFSAS and CNRJSAS - Postprocess with SAS".

Sample JCL to unload the active database:

```
//CNRACF EXEC PGM=CNRACF
//STEPLIB  DD DISP=SHR,DSN=steplib
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DISP=(NEW,CATLG),DSN=prefix.RACF.UNLOAD,
//            UNIT=SYSDA,SPACE=(CYL,(10,10),RLSE)
```

Sample JCL to generate commands to remove permits to a user to be deleted:

```
//CNRACF EXEC PGM=CNRACF
//STEPLIB  DD DISP=SHR,DSN=steplib
//SYSPRINT DD SYSOUT=*
//CMDOUT   DD DISP=(NEW,CATLG),DSN=prefix.CMDOUT.CLIST,
//            UNIT=SYSDA,SPACE=(CYL,(1,1),RLSE)
//SYSUT1   DD DISP=SHR,DSN=prefix.RACF.UNLOAD
//SYSIN    DD *
REMOVE PERMIT=userid
```

Sample JCL to execute the generated commands by running TSO in the batch:

```
//CMDEXEC EXEC PGM=IKJEFT01
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD DISP=SHR,DSN=prefix.CMDOUT.CLIST
```

Sample TSO commands to run the generated command file interactively:

```
EXEC 'prefix.CMDOUT.CLIST'
```


2.2.2 CNRCOPYR and CNRJCPYR - Unload active primaries

The procedure CNRCOPYR and its sample call CNRJCPYR unloads the currently active primary RACF datasets. Note that you will have to change parameter DSTAT to OLD or SHR after the first run.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCOPYR PROC REGSIZE=4096K, Region for CONSUL/RACF (at least 2M)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library /RACF
//  DSTAT='NEW,CATLG',         Disposition of work datasets
//  DPREF='CRM.TEST',          Prefix for work datasets
//  DUNIT=SYSDA,               Esoteric unit name for work datasets
//  DVOLSER=,                  Optional volume serial for work datasets
//  OPT=                       Optional command like 'ALLOC BACKUP'
//*
//*********************************************************************
//* Name:    CNRJCPYR     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Unload RACF database to work dataset
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//CNRACF EXEC PGM=CNRACF,REGION=&REGSIZE,
//  PARM='SHOW CLASSES;&OPT'
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.UNLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(150,150),RLSE,,ROUND)
//       PEND
//*
//CNRCOPYR EXEC CNRCOPYR
```

Fig 36. JCL procedure CNRCOPYR in sample member CNRJCPYR.

The cataloged procedure can be found in the procedure library CRM.CNR113.CNRPROC and the sample JCL can be found in the CRM.CNR113.CNRSAMP dataset.

2.2.3 CNRCOPYV and CNRJCPYV - Unload VTOC, VVDS, and BCS

This procedure unloads the current contents of VTOC, VVDS, ICF catalogs, linklist library directories, and APF library directories using the CNFCOLL program (Consul/Collect). Note that you will have to change parameter DSTAT to OLD or SHR after the first run.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCOPYV PROC REGSIZE=4096K,       Region for CONSUL/COLLECT
//  IPREF='CRM.',                 Prefix. for CONSUL/COLLECT libraries
//  IPROJ=CNF203,                 Project for CONSUL/COLLECT libraries
//  ILOAD=CNFLOAD,                Last qualifier load library /COLLECT
//  DSTAT='NEW,CATLG',            Disposition of work datasets
//  DPREF='CRM.TEST',             Prefix for work datasets
//  DUNIT=SYSDA,                  Esoteric unit name for work datasets
//  DVOLSER=                      Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJCPYV     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Unload VTOC and VVDS for all volumes, and ICFcats if APF
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//IOCONFIG EXEC PGM=CNFCOLL,REGION=&REGSIZE,PARM='FOCUS=AUDIT'
//STEPLIB  DD DISP=SHR,DSN=&IPREF.&IPROJ..&ILOAD  !APF lib
//SYSPRINT DD SYSOUT=*
//IOCONFIG DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.IOCONFIG,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(30,30),RLSE,,ROUND)
//         PEND
//*
//CNRCOPYV EXEC CNRCOPYV
```

Fig 37. JCL procedure CNRCOPYV in sample member CNRJCPYV.

2.2.4 CNRCOPY and CNRJCPY - Unload VTOC, VVDS, BCS, and RACF

This procedure unloads the current VTOC and VVDS contents as well as the currently active primary RACF datasets. Note that you will have to change parameter DSTAT to OLD or SHR after the first run.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCOPY PROC REGSIZE=4096K,  Region CNRACF (at least 2048K)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library /RACF
//  IPREF='CRM.',              Prefix. for CONSUL/COLLECT libraries
//  IPROJ=CNF203,              Project for CONSUL/COLLECT libraries
//  ILOAD=CNFLOAD,             Last qualifier load library /Collect
//  DSTAT='NEW,CATLG',         Disposition of work datasets
//  DPREF='CRM.TEST',          Prefix for work datasets
//  DUNIT=SYSDA,               Esoteric unit name for work datasets
//  DVOLSER=                   Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJCPY    Level: SCR1103      Version: CONSUL/RACF 1.1.3
//* Purpose: Take system snapshot (unload RACF, VTOC and VVDS datasets)
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//CNRACF EXEC PGM=CNRACF,REGION=&REGSIZE
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.UNLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(150,150),RLSE,,ROUND)
//*
//CNFCOLL  EXEC PGM=CNFCOLL,REGION=&REGSIZE,PARM='FOCUS=AUDIT'
//STEPLIB  DD DISP=SHR,DSN=&IPREF.&IPROJ..&ILOAD !APF lib if separate
//SYSPRINT DD SYSOUT=*
//IOCONFIG DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.IOCONFIG,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(10,10),RLSE,,ROUND)
//         PEND
//*
//CNRCOPY  EXEC CNRCOPY
```

Fig 38. JCL procedure CNRCOPY in sample member CNRJCPY.

2.2.5 CNRCFLV and CNRJCFLV - Analyze RACF database versus resources

This procedure executes CONSUL/RACF command input on a system snapshot taken by CNRCOPY:

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//CNRCFLV  PROC REGSIZE=6000K, Region for CONSUL/RACF (at least 4M)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library /RACF
//  CSAMP=CNRSAMP,             Last qualifier sample library
//  DSTAT='NEW,CATLG',         Disposition of work datasets
//  DPREF='CRM.TEST',          Prefix for work datasets
//  DUNIT=SYSDA,               Esoteric unit name for work datasets
//  DVOLSER=                   Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJCFLV     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Perform function on previously unloaded database & config
//* Note:    You must change DSTAT to OLD after your first run
//*********************************************************************
//*
//CNRCFLV  EXEC PGM=CNRACF,REGION=&REGSIZE
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//IOCONFIG DD DISP=SHR,DSN=&DPREF..CNRACF.IOCONFIG
//SYSUT1   DD DISP=SHR,DSN=&DPREF..CNRACF.UNLOAD
//SYSPRINT DD SYSOUT=*
//CMDOUT   DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.CMDOUT,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,SPACE=(CYL,(1,1),RLSE)
//SYSIN    DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRV&MEMBER)
//         PEND
//*********************************************************************
//* Sample runs - remove asterisk to activate run
//*********************************************************************
//*
//*VERCONS EXEC CNRCFLV,MEMBER=CONS   Consistency checks
//*RPTWORM EXEC CNRCFLV,MEMBER=WORM   Report globally writeable
//*RPTTCB  EXEC CNRCFLV,MEMBER=TCB    Report TCB protection
//*       EXEC CNRCFLV,MEMBER=        Member CNRV is empty
//XSYSIN DD *                          .. Remove X in //XSYSIN to run
  /*****************************************************************/
  /* Sample function - include all VERIFY options                  */
  /*  -> may give a LOT of output the first time!                  */
  /*  -> Memory use may necessitate separate runs for each function! */
  /* Browse the CMDOUT file to check through the commands generated. */
  /* N.B. don't start your comment on pos 1 - that means end-of-file */
  /*****************************************************************/
  VERIFY ALL
```

Fig 39. JCL procedure CNRCFLV in sample member CNRJCFLV.

The sample JCL member lists several possible sample command input members (these are prefixed by CNRV in the CNRSAMP library). Select the one you want by removing the asterisk from the JCL line.

CONS Check database consistency.

WORM Report globably writable resources. Globably writable datasets are potential worm holes.

TCB List and check protection of the Trusted Computing Base.

In addition, the last unnamed step calls the program without parameters, unless you change //XSYSIN into //SYSIN - in that case you will execute the commands listed in the SYSIN file.

2.2.6 CNRCFL and CNRJCFL - Standard SORTLIST commands

This procedure executes CONSUL/RACF command input on an unloaded RACF database (for instance unloaded by CNRCOPYR):

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//CNRCFL   PROC REGSIZE=6000K, Region for CONSUL/RACF (at least 4M)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library
//  CSAMP=CNRSAMP,             Last qualifier sample library
//  DPREF='CRM.TEST',          Prefix for work datasets
//  MEMBER=,                   Pre-defined member to run
//  OPT=                       Options to use (e.g. 'PRINT PL=60')
//*
//*********************************************************************
//* Name:    CNRJCFL      Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Perform pre-defined function on previously unloaded data
//*********************************************************************
//*
//CNRCFL   EXEC PGM=CNRACF,REGION=&REGSIZE,
//   PARM='&OPT'
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//SYSUT1   DD DISP=SHR,DSN=&DPREF..CNRACF.UNLOAD
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&MEMBER)
//         PEND
//*********************************************************************
//* Sample invocation - remove asterisk from report you want
//* Note: It is faster to use CNRCFLS is you want more than one report
//*********************************************************************
//*LISTAUTH EXEC CNRCFL,MEMBER=AUTH    SPC/OPER/CLAUTH users
//*LISTCICS EXEC CNRCFL,MEMBER=CICS    CICS profiles
//*LISTGLOB EXEC CNRCFL,MEMBER=GLOB    Global profiles
//*LISTIMS  EXEC CNRCFL,MEMBER=IMS     IMS profiles
//*LISTPROG EXEC CNRCFL,MEMBER=PROG    Programs & PADS datasets
//*LISTREV  EXEC CNRCFL,MEMBER=REV     Revoked users & connects
//*LISTTAPE EXEC CNRCFL,MEMBER=TAPE    TAPEVOL & disc tape datasets
//*LISTUNAM EXEC CNRCFL,MEMBER=UNAM    User names / instdata
```

Fig 40. JCL procedure CNRCFL in sample CNRJCFL.

Several sample input members can be selected by removing the comment asterisk from the respective JCL lines. The list input members are present in the CNRSAMP library prefixed with CNRL. If you want to make more than one report, you should use CNRCFLS, not CNRCFL (see next section). The following list members are included:

APPL List all APPL profiles with their UACC and access lists.

AUTH List all userids with system wide authorities SPECIAL, OPERATIONS, or any class-authorization. The report includes programmer name and installation data.

CICS List all standard CICS profiles with their members.

GLOB List all profiles in the class GLOBAL.

IMS List all standard IMS profiles.

PROG List program profiles with their datasets, universal access, and access list, as well as all dataset profiles with a conditional access list.

REV List revoked users and connects with their revoke and resume dates.

TAPE List all TAPEVOL and DSTYPE=TAPE DATASET profiles.

UNAM List users with their name field.

All command members are constructed in such a way that they can be concatenated to be used in one run. The following section shows a procedure specifically designed to use this.

2.2.7 CNRCFLS and CNRJCFLS - Multiple SORTLIST commands

This procedure executes CONSUL/RACF command input on an unloaded RACF database (for instance unloaded by CNRCOPYR) to produce multiple reports in one pass:

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//CNRCFLS PROC REGSIZE=6000K,  Region for CONSUL/RACF (at least 4M)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library
//  CSAMP=CNRSAMP,             Last qualifier sample library
//  DPREF='CRM.TEST',          Prefix for work datasets
//  OPT=,                      Options to use (e.g. 'PRINT PL=60')
//  MEMBER=,                   Pre-defined member to run
//  L0=,                       xxxx for CNRLxxxx members, or blank
//  L1=,                       xxxx for CNRLxxxx members, or blank
//  L2=,                       xxxx for CNRLxxxx members, or blank
//  L3=,                       xxxx for CNRLxxxx members, or blank
//  L4=,                       xxxx for CNRLxxxx members, or blank
//  L5=,                       xxxx for CNRLxxxx members, or blank
//  L6=,                       xxxx for CNRLxxxx members, or blank
//  L7=,                       xxxx for CNRLxxxx members, or blank
//  L8=,                       xxxx for CNRLxxxx members, or blank
//  L9=                        xxxx for CNRLxxxx members, or blank
//*
//*********************************************************************
//* Name:    CNRJCFLS     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Perform pre-defined functions on previously unloaded data
//*********************************************************************
//*
//CNRACFL  EXEC PGM=CNRACF,REGION=&REGSIZE,
//   PARM='&OPT'
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//SYSUT1   DD DISP=SHR,DSN=&DPREF..CNRACF.UNLOAD
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(&MEMBER)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L0)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L1)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L2)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L3)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L4)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L5)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L6)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L7)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L8)
//         DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRL&L9)
//         PEND
//*********************************************************************
//*  Sample invocation
//*********************************************************************
//CNRL     EXEC CNRCFLS,MEMBER=CNRL$OPT,
//   L0=AUTH,              xxxx for CNRLxxxx members, or blank (//*)
//   L1=APPL,              xxxx for CNRLxxxx members, or blank
//   L2=CICS,              xxxx for CNRLxxxx members, or blank
//   L3=GLOB,              xxxx for CNRLxxxx members, or blank
//   L4=IMS,               xxxx for CNRLxxxx members, or blank
//   L5=PROG,              xxxx for CNRLxxxx members, or blank
//   L6=TAPE,              xxxx for CNRLxxxx members, or blank
//*  L7=UNAM,              xxxx for CNRLxxxx members, or blank
//   L9=                   xxxx for CNRLxxxx members, or blank
```

Fig 41. JCL procedure CNRACFLS in sample CNRJCFLS.

The empty member CNRL is especially included in the library to support this procedure. For a discussion of the remaining CNRL members, see the previous section.

2.2.8 CNRJSYNC - Synchronize non-VSAM

This procedure gives JCL to synchronize RACF dataset profiles with a volume that has been restored with full volume restore (without RACF processing).

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCSYNC PROC REGSIZE=6000K, Region for CONSUL/RACF (at least 4M)
//  CPREF='CRM.',              Prefix. for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library /RACF
//  IPREF='CRM.',              Prefix. for CONSUL/COLLECT libraries
//  IPROJ=CNF203,              Project for CONSUL/COLLECT libraries
//  ILOAD=CNFLOAD,             Last qualifier load library /COLLECT
//  DSTAT='NEW,CATLG',         Disposition of work datasets
//  DPREF='CRM.TEST',          Prefix for work datasets
//  DUNIT=SYSDA,               Esoteric unit name for work datasets
//  DVOLSER=,                  Optional volume serial for work datasets
//  VOL=                       Volume to be synchronized
//*
//*********************************************************************
//* Name:    CNRJSYNC     Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Synchronize RACF database with actual datasets on 1 volume
//* Usage:   Specify volume serial on VOL=XXXXXX parameter below
//* Note:    1. You must change DSTAT to OLD after the first run
//*          2. Warning: Only synchronizes non-VSAM datasets
//*********************************************************************
//*
//IOCONFIG EXEC PGM=CNFCOLL,REGION=&REGSIZE,
//    PARM='FOCUS=AUDIT,SELECT=VOL=&VOL,VVDS=NO'
//STEPLIB   DD DISP=SHR,DSN=&IPREF.&IPROJ..&ILOAD !APF lib if  separate
//SYSPRINT  DD SYSOUT=*
//IOCONFIG  DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.IOCONFIG,
//             UNIT=&DUNIT,VOL=SER=&DVOLSER,SPACE=(CYL,(10,10),RLSE)
//*
//CNRACF   EXEC PGM=CNRACF,REGION=&REGSIZE,
//  PARM=('VER ONVOL',
// 'PROT; SEL VOL=&VOL; SEL CLASS=DATASET,GENERIC; LIM MSG=10000')
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//IOCONFIG DD DISP=SHR,DSN=&DPREF..CNRACF.IOCONFIG
//SYSPRINT DD SYSOUT=*
//CMDOUT   DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.CMDOUT,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,SPACE=(CYL,(1,1),RLSE)
//         PEND
//XXXXXX EXEC CNRCSYNC,VOL=XXXXXX
```

Fig 42. JCL procedure CNRCSYNC in sample member CNRJSYNC.

2.2.9 CNRJCMD - Execute generated commands

This procedure gives JCL to execute the commands generated.

You should review the commands before executing them.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCMD   PROC REGSIZE=1024K, Region for TSO
//  DPREF='CRM.TEST'           Prefix for work datasets
//*
//*********************************************************************
//* Name:    CNRCMD    Level: SCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: Execute generated commands
//* Warning: 1. You should verify and/or edit the generated commands
//*             the commands before submitting this job.
//*          2. If many commands are present, elapsed time can be
//*             hours and CPU usage can be several minutes !
//*********************************************************************
//*
//IKJEFT01 EXEC PGM=IKJEFT01,PARM='PROF NOPREF',REGION=&REGSIZE
//SYSTSPRT DD SYSOUT=*
//SYSTSIN  DD DISP=SHR,DSN=&DPREF..CNRACF.CMDOUT
//         PEND
//*
//CNRCMD EXEC CNRCMD
```

Fig 43. JCL procedure CNRCMD in sample CNRJCMD.

The PROFILE NOPREFIX command is *not* required.

2.2.10 CNRCFSAS and CNRJSAS - Postprocess with SAS

This procedure gives JCL to perform a LIST function to a separate file and postprocess this 'flat file' with SAS®.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRCFSAS PROC REGSIZE=6000K, Region for CONSUL/RACF (at least 4M)
//  CPREF='CRM.',              Prefix for CONSUL/RACF libraries
//  CPROJ=CNR113,              Project for CONSUL/RACF libraries
//  CLOAD=CNRLOAD,             Last qualifier load library /RACF
//  CSAMP=CNRSAMP,             Last qualifier sample library
//  DPREF='CRM.TEST',          Prefix for work datasets
//  MEM=,                      Pre-defined member id to run (CNRX...Q/S)
//  OPT=,                      Options to use (e.g. 'PRINT PL=60')
//  DSTAT='NEW,CATLG',         Disposition of work datasets
//  DUNIT=SYSDA,               Esoteric unit name for work datasets
//  DVOLSER=                   Optional volume serial for work datasets
//*
//*********************************************************************
//* Name:    CNRJSAS      Level: SCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: Perform pre-defined function on previously unloaded data
//*          to generate input for postprocessing with SAS
//* Note:    You must change DSTAT to OLD after the first run
//*********************************************************************
//*
//CNRACF   EXEC PGM=CNRACF,REGION=&REGSIZE,
//   PARM='&OPT'
//STEPLIB  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CLOAD
//SYSUT1   DD DISP=SHR,DSN=&DPREF..CNRACF.UNLOAD
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRX&MEM.Q)
//SASYSIN  DD DISP=SHR,DSN=&CPREF.&CPROJ..&CSAMP.(CNRX&MEM.S)
//SASINPUT DD DISP=(&DSTAT),DSN=&DPREF..CNRACF.SASINPUT,
//            UNIT=&DUNIT,VOL=SER=&DVOLSER,
//            SPACE=(32760,(150,150),RLSE,,ROUND)
//         PEND
//*********************************************************************
//*  Sample invocation
//*********************************************************************
//CNR      EXEC CNRCFSAS,MEM=CFR                Invoke CNRACF
//SAS      EXEC SAS                             Invoke SAS
//SASINPUT DD DISP=SHR,DSN=*.CNR.CNRACF.SASINPUT
//SYSIN    DD DISP=SHR,DSN=*.CNR.CNRACF.SASYSIN
```

Fig 44. JCL procedure CNRCFSAS in sample CNRJSAS.

2.3 CNRACF Batch command language

In the batch, commands may be entered in the SYSIN input file as well as by passing them as parameters with the PARM keyword on the EXEC JCL statement. Commands consist of a command keyword followed by parameters. These parameters *must* be separated by commas. Parameters may be just a keyword or have a value, indicated by an equal sign and a value:

```
LIMIT IN=10000, OUT=10
```

Blanks are ignored behind (i.e. following) commas and on both sides of command keywords. A command is terminated either by a semicolon, or by a new line if the last non-blank symbol on the line was not a comma in a list of parameters.

```
SELECT CLASS=PROGRAM; LIST MEMLST
```

Some parameters can have a list in parentheses as value:

```
report redundant, by=(reason, key)
```

A few commands have no parameters, they are mainly meant for use through the program parameter string (like MARGINS and CAPS).

The commands may be entered in any case (lower and uppercase do not matter).

The order in which the commands are given is generally not significant. There is one exception to this: the NEWLIST command divides the input into separate list descriptions. In each list description behind a NEWLIST, the SELECT, EXCLUDE, and PRINT commands must precede any LIST or SORTLIST commands.

One of the main output commands must be entered for each execution of CNRACF with the exception of UNLOAD which is implied if a SYSUT2 file is allocated in the CNRACF execution JCL - see "2.2.1 DDname overview". The main output commands are:

UNLOAD To make a variable blocked copy of all selected profiles for later processing.

SHOW To report on templates or classes present in database.

LIST To list the key (name) and optional other fields from all selected profiles.

SORTLIST The same as LIST, but listed in ascending order of the fields to be listed.

VERIFY To perform consistency checks on the input profiles and generate RACF commands to repair the problems on the CMDOUT file (if allocated in CNRACF execution JCL).

REPORT To report profiles that satisfy criteria not covered by the SELECT command.

(RE)MOVE To generate RACF commands on the CMDOUT file to remove (or move) users, permits, or whatever is requested by the parameters.

The following commands may be used to indicate which profiles should be processed:

SELECT To indicate conditions that the profile must satisfy to be selected. Multiple parameters on a single SELECT imply an AND function between the parameters - that is, all criteria must be satisfied to cause selection.
Use of multiple SELECT statements implies an OR function - that is, any successful SELECT statement causes selection.

EXCLUDE To indicate conditions that must be satisfied to reject the profile. Multiple parameters on a single EXCLUDE imply an AND function between the parameters - that is, all criteria must be satisfied to cause rejection.
Use of multiple EXCLUDE statements implies an OR function - that is, any of the EXCLUDEs matching is sufficient to reject the profile.

The following commands also influence the information processed:

ALLOC To select a different set of RACF databases than the current primaries.

SUPPRESS To suppress error messages for certain volumes, catalogs, users, or groups.

LIMIT To limit input, output, or processing to maximum numbers, discretes, generics, etc.

The following commands influence SYSIN and SYSPRINT characteristics:

MARGINS To specify which columns are read from SYSIN.

CAPS To force all ouput to uppercase

PRINT To change print output file options (titles, pagelength, etc).

A few examples:

```
SELECT CLASS=GROUP, NOTERMUACC
LIST CLASS, KEY
```

This lists all RACF groups with the NOTERMUACC attribute.

```
SELECT OPERATIONS
SELECT SPECIAL
LIST CLASS, KEY
```

This lists all profiles with the OPERATIONS or SPECIAL attribute. Since no class is specified, both USER and CONNECT profiles will be listed.

```
VERIFY DATASET, PROTECTALL, INDICATED
SUPPRESS VOLSER=BCKUP1
SUPPRESS VOLSER=BCKUP2
```

This performs a cross-check of datasets with dataset profiles, but excludes two volumes from its analysis.

The remainder of this section (2.3) describes the syntax and parameters of all commands.

2.3.1 ALLOCATE

The allocate command can be used to select a different source of profiles than the active primary RACF datasets. It may be abbreviated to ALLOC.

The default if no SYSRAC*nn* files are allocated is

```
ALLOC PRIMARY, ACTIVE
```

The parameters that can be specified are:

DB=*n* **DB=(1,*n*,...)** **DATABASE=...**	To select database sequence numbers. Number 1 (the master dataset) must always be included. The DB parameter must give a number or a list of numbers. The highest number supported is 64. A maximum of 10 numbers may be combined in one run.
PRIMARY **PRIM**	To select primary datasets. This is the default.
BACKUP	To select backup datasets.
ACTIVE	To select active datasets. This is the default.
INACTIVE	To select inactive datasets.

The parameters may be combined to select the dataset you are interested in. The database sequence number are defined in the database name table. An example how to display this table is described in section 2.1. Examples of the use of the ALLOC command can be found in "1.1 Unloading and selecting RACF datasets".

Example - use backup datasets

To minimize impact on system operation, one might read the backup database instead of the primary. For most purposes except statistics, the information in the backup is identical to the primary. This is accomplished by the command

```
ALLOC BACKUP
```


2.3.2 CAPS

This command without parameters forces all subsequent output to uppercase, and may be used if your print chain does not print lowercase characters. It can be included as a program parameter (like all commands). You should however remember that multiple commands in the parameter string must be separated by semicolons. The output containing a listing of the input commands up to and including the line containing the CAPS parameter will not be converted to uppercase.

2.3.3 DEFAULT

This command can be used to change defaults used by CONSUL/RACF. They apply for the complete run.

OWNER=*id*	Select a different default ownership than SYS1 for non-dataset profiles that have an undefined owner or an owner that is to be removed.

2.3.4 LIMIT

The limit command can be used to set limits on execution or number of error messages. LIMIT may be abbreviated to LIM.

IN=*nn* **I=*nn***	To stop execution after the specified number of profiles has been read.
OUT=*nn* **O=*nnn***	To limit the number of profiles output on UNLOAD or LIST.
MSG=*nn* **M=*nnn***	To set a limit on the number of error messages for a specific volume.
GENERIC	To limit the output of the commands REPORT and REMOVE with the options NONREDUNDANT and REDUNDANT to generic profiles only.
DISCRETE	To limit the output of the commands REPORT and REMOVE with the options NONREDUNDANT and REDUNDANT to discrete profiles only.
ID=*id*	To limit messages and reports to lines concerning a specific user or group. This option will merely reduce output, not the processing time. Often SELECT QUAL= will be a better choice.
OLDTEMPLATE **OLD**	To limit the use of templates to the downward compatibility templates present in release 1.8 (compatible with pre-1.8 releases).
DELDSD	To limit the commands generated by REMOVE and MOVE in such a way as to exclude deletion of dataset profiles. For instance, on a REMOVE USER=*userid* command, the deletion of all profiles starting with *userid* can be prevented by this option. Note that this will probably result in failure of a DELUSER command.

Example 1 - limit discrete

To remove redundant discrete profiles but not redundant generic profiles, the scope of the REMOVE REDUNDANT command can be restricted to discretes by the following command sequence:

```
REMOVE REDUNDANT
LIMIT DISCRETE
```

Example 2 - limit msg

You may not see all detail error messages resulting from a VERIFY command on the first run, because of the default message limit of 50 messages per volume. If this is the case, then you will receive the message:

```
CNR091I volser message limit exceeded- nn messages suppressed
```

To see all detail messages, the message limit should be increased. This can be accomplished by setting it to a sufficently high value (at least the current limit plus the amount of messages suppressed according to the message). For instance:

```
LIMIT MSG=1000
```


2.3.5 LIST

The LIST and SORTLIST commands request formatted output of the profiles selected by SELECT and EXCLUDE to the SYSPRINT file (or a different output destination set by the PRINT command). For the LIST command, the output is *not sorted*. For sorted output, SORTLIST can be used. The two commands have a similar syntax. For each selected profile, the fields indicated as parameters to the LIST command are listed. For information on different output files or simultaneous multiple reports, see the NEWLIST command.

Field names

The parameters to the LIST command must be field names defined by the templates of the database or one of the keywords CLASS, KEY, RBA, DB, or SEGMENT. Alias field names may be specified as well. An error message will be issued if the name specified is not part of any template. You can use the SHOW TEMPLATES command to list the templates and their default output format. For a summary of the fields used most for reporting purposes, as well as the exact meaning of the keywords, see below. In addition, a full listing of RACF 1.9 template fields is included in appendix B.

Modifying output length

Optionally, the field name may be followed by a *length* in parentheses. This number indicates the number of output positions into which the field should be formatted. Most fields have default lengths predefined that are sufficient for standard RACF profiles. If the field is defined as a variable length field in the template, no default length is set by CONSUL/RACF (this is listed on the SHOW TEMPLATES output), and no explicit length is provided on the LIST command, then the actual length will be used for each profile. This results in disturbance of the column layout, because all subsequent fields will start at a varying position.

The output length can only be modified for text and hex format fields. Numbers, keywords, dates, times, and flags displayed in a non-text non-hex format always result in the same output length, that can be determined from the SHOW TEMPLATES output.

Modifying output format

Instead of the length, or in addition to it, a *format* may be defined. Generally this will be unnecessary, but it may for instance be helpful to debug problems by displaying the profile field contents in hex instead of in formatted form. This can be accomplished by specifying "hex" in parentheses. Probably, you will have to specify a different output length as well. Note that an inappropriate format may result in abends (e.g. trying to format text like a date). The formats that can be specified are described below.

Resulting display of repeat groups

If the field is part of a repeat group, then the listing will contain as many lines as necessary to list all member values in a column. If two field names are given that are part of the same repeat group, than the values for one repeat group will appear on the same line. If fields in other repeat groups are requested as well, than the line containing the profile key will contain the first value of each repeat group field.

Examples of the resulting output can be found in the sections "1.4.1 Listing profile fields" through "1.4.10 Finding all occurrences of a string".

Difference between RDS and non-RDS

The function of the LIST command differs depending on the database format. In the case of a non-restructured database format (non-RDS), each record processed by LIST is a complete profile including all its segments. In the case of a restructured database format (RDS - not available before RACF 1.9), each record processed by LIST is only one segment.

So if you want to list the key of a profile and a field from the BASE segment for all profiles of a certain class, then your selection statement should be modified to include SEGMENT=BASE in your selection, otherwise your listing will contain lines only containing the key for each non-base segment. This does not apply if your selection was testing a field in the base segment: in that case the selection will fail for all non-base segments anyway, and no modification is necessary.

Another consequence of moving to a restructured database format is that output fields belonging to different segments of a profile cannot be combined on one output line. For example, whilst it is possible with a non-RDS format to list (say) the default account from the TSO segment of a USER profile together with the PGMRNAME from the base segment of the profile on the same line, with an RDS input file, these would be listed on separate lines. The impact of this for reporting purposes can be minimal if you use SORTLIST to order the profiles by key; in this case the lines for the BASE and the TSO segment will immediately follow each other. However, if you are using postprocessing programs on LIST output, you have to be fully aware of this difference.

Special keyword field names

The meaning and format of the special keywords is:

CLASS	This will result in an 8 character display of the class name. To be more exact, this can be either the entity name present in the templates for the profile's entity type, or, in case of the entity name GENERAL, the class name present in the Class Descriptor Table, or, if the CDT is missing, the class name in the profile. In the case of a non-restructured database where the class is missing from the CDT, this will result in a four-character class name padded with blanks. There are some instances in which the class name listed may be in error. This can happen only for a non-BASE segment of a Restructured Database (RDS). Instances where this may happen can optionally be identified by including the DEBUG SEGMENT command. See also message CNR163I.
SEGMENT	The meaning and output of this keyword depends on the format of your RACF database. In the case of a Restructured Dataset (RDS), this will result in the 8 character display of the segment type of the selected profile segment, including BASE. The RDS format is only available with RACF 1.9 and higher releases. In the case of a non-Restructured Dataset (non-RDS), this will result in a display of all non-base segments present in the selected profile.
KEY	This keyword results in display of the profile name. In the case of a Restructured Dataset (RDS), this works for all segment types. The default length is 44. Note that in the case of a Restructured Dataset, general resource profiles may have a higher key length (up to 246). Generic profiles are displayed according to the current setting of the EGN option (Enhanced Generic Naming). To ease interfacing with postprocessing programs, generic profiles are not suffixed with the (G) found in standard RACF command output. For fully qualified dataset profiles, you cannot see the difference with a discrete profile. The ambiguity can be resolved by including the DSTYPE field in the output line (VOLSER would still leave ambiguity with model profiles). For general resource profiles, a generic character is always present if a profile is generic.
DB	This results in a 3 character display of the sequence number of the originating database (as defined by the database name table). This can be displayed even if the source is in the unloaded format (created by release 1.1 of CNRACF or higher).
RBA	This results in a 12 character display of the Relative Byte Address of the selected profile (segment) in the originating database. This can be displayed even if the source is in the unloaded format (created by release 1.1 of CNRACF or higher).

Format names

AUDAC	Audit access level. Indicates the access level at which auditing starts in a 7 character field that can be blank, READ, UPDATE, CONTROL or ALTER.
AUDIT	Audit type. Indicates which access are logged in an output field 7 characters wide. Can be ALL, SUCCESS, FAILURE, or NONE.
AUTHORITY	Connect authority in an output field 7 characters wide. Can be USE, CREATE, CONNECT, or JOIN.
CHAR	Character string. The field is copied without modification to the output file.
DATE	Date displayed as DD MMM YYYY. If used inappropriately, it will result in abends.
DSTYP	Dataset type in a field 7 characters wide. This can be blank, NONVSAM, VSAM, MODEL, or TAPE.
FLAG	This will display the setting of a high order bit flag byte. If the high order bit of the field is on and the rest off, YES will be displayed. Blanks will be displayed if the field contains all zeros or all ones in the first byte, and "? " otherwise.
HEX	This will display the field in hexadecimal digits. Remember that the length of a hex number display should be at least twice the length in bytes of the field to be displayed, so you will need to include an explicit length. The hex format will pad the field with blanks if the source field is shorter than would fit in the reserved output space. In this way, the actual field length is clear.
LOGDAYS	This will format a byte according to the setting for logon-allowed days, in the form SMTWTFS, where the day-of-week letter indicates that logon is allowed, and is left blank otherwise.
LOGTIME	This will format a logon-allowed time frame as HHMM:HHMM
NUM	This will format a 1, 2, 3, or 4 byte binary integer as a decimal number with length 3, 5, 8, or 12, respectively.

Some common field names

The most common template fields are listed below in alphabetical order with an indication for which class they may be used. The word GENERAL indicates all classes defined via the class descriptor table. If a field can have more values for one profile, then this is indicated by the text "repeated".

Fieldname	Description	Profile classes
CONGRPNM	Connect group (repeated)	USER
DFLTGRP	Default group	USER
INSTDATA	Installation data	GROUP, USER, DATASET, GENERAL
LJDATE	Last logon date	USER
MEMLST	Member resource (repeated)	GENERAL
MODELNAM	Model dataset	GROUP, USER
NOTIFY	Userid to notify	DATASET, GENERAL
OWNER	User or group	GROUP, USER, CONNECT, DATASET, GENERAL
PASSDATE	Last change of password	USER
PGMRNAME	User name	USER
PROGACS	Access in conditional access list (repeated)	DATASET
PROGRAM	Program in conditional access list (repeated)	DATASET
RESUMEDT	Resume date	USER, CONNECT
REVOKEDT	Revoke by date	USER, CONNECT
SUBGRPNM	Subgroup (repeated)	GROUP
SUPGROUP	Superior group	GROUP
TVTOCDSN	Dataset in TVTOC	GENERAL
UACC	Universal access	GROUP, USER, CONNECT, DATASET, GENERAL
USER2ACS	User/group in conditional access list (repeated)	DATASET
USERACS	Access level in standard access list (repeated)	GROUP, DATASET, GENERAL
USERID	User/group in standard access list (repeated)	GROUP, DATASET, GENERAL
VOLSER	Volume serial (repeated)	DATASET, GENERAL

A more complete list is included in appendix B.

Example 1 - profile key

One of the simplest requests is to list the class and name of selected profiles:

```
LIST CLASS, KEY
```

Example 2 - access list

To list the standard access list of resource profiles as well as the owner field and universal access, one might specify:

```
list class, key, owner, userid, useracs, uacc
```

Example 3 - changing length

If you know the profile type you selected (e.g. CLASS=USER), then you may want to modify the output length of the KEY field to 8 in order to get a concise listing of the users in your database:

```
SELECT CLASS=USER
SORTLIST KEY(8), PGMRNAME, INSTDATA
```


By using SORTLIST instead of LIST, the profiles will be sorted on the field values (in the order of the field names). This will in this example result in a report sorted by RACF userid.

Example 4 - changing format

To change the format of the display, you can include the format you wish. Let's assume you use the first 3 byte of the installation data for bit flags for your own purpose. The default textual format will result in non-displayable characters that may disturb the printer. In this case, you can display them in hex. You may also want to specify the maximum length to be displayed. In this case, that would be 6 bytes:

```
LIST KEY(8), INSTDATA(6,HEX)
```


2.3.6 MARGINS

This command sets the margins for reading the SYSIN file. The command must be followed by two decimal numbers separated by a comma and enclosed in parentheses:

```
MARGINS(nn,mm)
```

The first number gives the starting column for the text to consider (the first column is column number 1). The second number gives the last column to be read. Both numbers must be in the range 1 to 255, and the first number cannot be greater than the second number.

The default is

```
MARGINS(1,72)
```

this default will ignore line numbers if the input is a fixed block file with a record length of 80.

2.3.7 NEWLIST

Use this command if you want to generate more than one report with LIST or SORTLIST in one pass. The NEWLIST command marks the beginning of a new report description containing the commands PRINT, SELECT, EXCLUDE, LIST, or SORTLIST. If these commands were also specified in the input before the first NEWLIST, then these commands are processed before any of the options present behind the NEWLIST, and with each NEWLIST the options are reset to this default setting.

Within a NEWLIST block, print and selection options (if any) must be specified *before* the LIST or SORTLIST command.

The NEWLIST command is also the only way to direct LIST or SORTLIST output to a separate output file (for an example, see the PRINT command).

If no profiles are selected within the NEWLIST block, the list will be skipped completely in the output (no page header will be generated even if PRINT TITLE was specified in the NEWLIST block). This means that you can include reports that 'pop up' only if one of its SELECT statements hits.

NEWLIST by itself does not change the fact that LIST output lines are not stored in main memory. This works fine if you direct the output of multiple NEWLIST/LIST sequences to a different DDname for eacg sequence, but will cause the output lines to come from mixed LIST commands if you use the same DDname in the NEWLIST sequences. If you use SORTLIST the reports will be properly separated even if you use the same DDname.

For additional examples, see the PRINT, LIST, and SELECT commands.

Example

This example shows a selection applying to all reports, and two NEWLISTs with a further subselection. In addition, a title is generated common to all reports, and a subtitle that is different for each report.

```
PRINT TITLE='ABC Computer services Inc, phone 234-17829'
SELECT CLASS=USER
NEWLIST
   PRINT SUBTITLE='Users with system-wide SPECIAL attribute'
   SELECT SPECIAL
   SORTLIST KEY(8), PGMRNAME, DFLTGRP, INSTDATA
NEWLIST
   PRINT SUBTITLE='Users with system-wide OPERATIONS attribute'
   SELECT OPERATIONS
   SORTLIST KEY(8), PGMRNAME, DFLTGRP, INSTDATA
```


2.3.8 PRINT

Use this command to modify print options. The command can occur in two kinds of context: in the domain of a NEWLIST command, or outside the domain of a NEWLIST command (i.e. before the first NEWLIST). If it occurs in the domain, then it applies only to that NEWLIST, otherwise it sets a default for all subsequent NEWLIST domains.

PAGELENGTH=*number* **PAGELEN=*number*** **PL=*number***	Number of lines on each page to be used for printing. The default is 0 for LIST commands in the domain of a NEWLIST, and 50 for SORTLIST commands, and the file SYSPRINT.
DDNAME=*name* **FILE=*name***	Sets the output file name for a NEWLIST domain.
CAPS	Output must be in uppercase only.
TITLE='*string*'	Title to appear on subsequent output pages. It is printed below the page header line instead of the copyright message. Note that the quotes are required.
SUBTITLE='*string*'	Title to appear on subsequent output pages. It is printed below the TITLE or copyright line. Note that the quotes are required.
OVERPRINT=*n* **OVP=*n***	Number of overprints to get bold text on impact printers. Values must be in the range 0 to 9. Specifying 0 disables overprinting, and is the default. It is only used if RECFM=A is included and OPTCD=J is *not* included in the DCB parameters of the SYSPRINT file.
EGN	Format profile names as if enhanced generic naming were active.
NOEGN	Format profile names as if enhanced generic naming were *not* active. Profiles that would be supported by EGN but not by NOEGN can be found by searching for a colon (":").

Example 1 - pagelength

Changing the number of lines on each page to 60 can be done by the following command:

```
PRINT PL=60
```

Example 2 - different output file

If you want to create a file with profile fields to be used in postprocessing with e.g. SAS, you can direct the result of the LIST command to a seperate file name. This requires use of the NEWLIST command. The following example shows how to create such a file for user profiles.

```
NEWLIST
  PRINT DDNAME=LISTOUT
  SELECT CLASS=USER
  LIST KEY(8), PGMRNAME, INSTDATA
```


Example 3 - titles

If you want to create a number of reports, it is better to include a title with each report. The following example shows the use of TITLE and SUBTITLE in combination with two NEWLIST commands. The print options set outside the scope of a NEWLIST serve as a default for each new list.

```
PRINT TITLE='ABC Computer services Inc, phone 234-17829'
NEWLIST
   PRINT SUBTITLE='Users with system-wide SPECIAL attribute'
   SELECT CLASS=USER, SPECIAL
   SORTLIST KEY(8), PGMRNAME, DFLTGRP, INSTDATA
NEWLIST
   PRINT SUBTITLE='Users with system-wide OPERATIONS attribute'
   SELECT CLASS=USER, OPERATIONS
   SORTLIST KEY(8), PGMRNAME, DFLTGRP, INSTDATA
```


2.3.9 (RE)MOVE

Use this command if you want to generate RACF commands on file CMDOUT to move or remove users, groups, permits or notify fields[4]. There are three kinds of parameters: those to specify users/groups to be processed, modifiers for these functions, and independent functions.

The following group of user/group processing options are mutually exclusive on one REMOVE command (but there can be more than one REMOVE command). In addition, these options are mutually exclusive with VERIFY PERMIT. In fact they simulate the situation that the *id* definition is missing from the database. The resulting messages have the same number, but say 'Removing' instead of 'Undefined'.

[4] Commands on file CMDOUT may also be generated as the result of a VERIFY command.

PERMIT=*id*	To prepare (re)moving the userid or group, generate commands that would remove references to *id* from access lists, OWNER fields, NOTIFY fields, RESOWNER fields, NODES members, and functional positions in profile keys of the classes DATASET, DLFCLASS, VMMDISK, VMRDR, VMBATCH, VMCONECT, VMEVENT, VMXEVENT, SURROGAT, PROPCNTL, JESJOBS, JESSPOOL, and NODES. The scope of the command can be limited by the FROMGROUP, TOGROUP, and ALLPERMITS parameters (see below), as well as by the SUPPRESS DELDSD command. If FROMGROUP is specified, only references in dataset profiles of the indicated groups will be removed, and no references general resource profiles or other user or group profiles. If TOGROUP is specified, all references are removed except from the indicated group, the subject user, and general resource profiles. Commands are generated to delete profiles, access list entries, and RESOWNER fields, and to change profile ownership and NOTIFY fields as appropriate. Profile ownership is changed to the first qualifier (as changed by ICHCNX00) for group dataset profiles, to the group for connect profiles, and to the owner set by the command DEFAULT OWNER= (default is SYS1) otherwise. NOTIFY fields are removed unless the NEWNOTIFY parameter is used to indicate which user should replace the one to be removed.
NOTIFY=*id*	This command performs a subset of REMOVE PERMIT= processing, namely limited to NOTIFY fields.
USER=*id*	This command removes a user from the database or from a number of groups (as specified in the FROMGROUP parameter), including all references and the profiles solely used for that user. It does all processing of REMOVE PERMIT=, and in addition it generates RACF commands to remove the user from its connect groups, delete the user profile if requested, and modify the user's default group as needed.
GROUP=*id*	This command removes a group from the database, including all references and the profiles solely used for the group. It does all processing of REMOVE PERMIT=, and in addition it generates RACF commands to remove users from the group, delete the group profile as requested, and modify default groups of users as needed. However, it does *not* automatically delete user profiles if the group is the last remaining group of a user. The remove commands generated will simply fail.

The following options can be used to adjust the processing performed by the command options given above:

FROMGROUP=*idlist*	This option limits the scope of the removal to the groups specified in *idlist*, wich can be a single group name, or a list of group names enclosed in parentheses and separated by commas. This limitation of scope also extends to the dataset profiles of the group: removal of access lists, owner fields, etc. is only done for profiles belonging to one of the groups in the list (either by their first qualifier or the qualifier returned by ICHCNX00).
TOGROUP=*idlist*	This option is only valid with the MOVE USER= command, and specifies a group (or list of groups enclosed in parentheses and separated by commas) to which the user is to be connected instead of all its current groups, or the groups indicated in the FROMGROUP parameter (if present). In addition, it reduces the scope of removal so as to exclude the user's personal dataset profiles, the group-dataset profiles of groups in the TOGROUP list, and all general resource profiles. If the ALLPERMITS parameter is added, then even the user's profiles and references in general resource profiles will be removed. The specified group may be a current connect group; in this case no connect command is generated.
ALLPERMITS	This option is valid only behind TOGROUP=. It causes references in all profiles to be removed, except by profiles of the group(s) specified on TOGROUP. This includes the user's profiles and general resource profiles that would be omitted by TOGROUP. Its main use is to move a user to a 'holding group' prior to deleting it anyway. See also REVOKE.
NEWNOTIFY=*id*	This option indicates a replacement userid to be used for NOTIFY fields processed by the user/group removal commands.
REVOKE	This option can be used to revoke the userid as well as performing the other actions on the REMOVE command. Its primary use is to cause a userid moved to a holding group prior to deletion to be revoked, i.e. in a command like MOVE USER=*id*, TOGROUP=*hold*, REVOKE.

The following REMOVE options are independent functions:

REDUNDANT	Generate commands to remove discrete dataset profiles that are covered by a generic profile whose OWNER, WARNING, ERASE (where active globally), and AUDIT settings are identical or can be considered *similar*. Access is considered *similar* if the access list contains the group to which the dataset belongs with a lower level than present in GLOBAL DATASET member "&RACGPID.*". Access is also considered *similar* if a userid is present on the discrete profile access list with an access that is also granted via one of the groups to which the userid is connected. You can use the LIMIT DISCRETE and LIMIT GENERIC commands to process only discrete or generic profiles.

Example 1 - remove references

This example generates commands to remove all meaningful references to a specific userid or group JONES from the RACF database.

```
REMOVE PERMIT=JONES
```

Example 2 - remove a user

This example generates commands to altogether remove a userid JONES from the database, including all his personal profiles.

```
REMOVE USER=JONES
```

Example 3 - move user to staging group

This example generates commands to remove all references to a userid JONES from the database, including all his profiles, and move the userid to a holding group temporarily to be deleted later on (e.g. after it has been removed from non-RACF user definitions). In addition, the userid is revoked. The user's default group will be changed as necessary.

```
MOVE USER=JONES, TOGROUP=HOLDDEL, ALLPERMITS, REVOKE
```

Example 4 - move user to another department

This example generates commands to move a user to a new department. It removes all authorities of the user on dataset profiles of the group, and connects him to the new group. All authorities on general resource profiles and his personal profiles remain intact.

```
MOVE USER=JONES, FROMGROUP=DEPT1, TOGROUP=DEPT2
```

Example 5 - transfer NOTIFY to a different user

This example shows how to generate commands to transfer all NOTIFYs on the dataset profiles of a group to another user (e.g. because he is going on holiday).

```
MOVE NOTIFY=JONES, FROMGROUP=DEPT1, NEWNOTIFY=PARKER
```

Example 6 - remove redundant profiles

This example shows how to generate commands to remove redundant discrete dataset profiles

```
REMOVE REDUNDANT
LIMIT DISCRETE
```


2.3.10 REPORT

This command requests reporting functions based on analysis after completing a scan of the database (contrary to the LIST command that is performed *during* this scan). The REPORT command will use much more memory than the LIST function.

PERMIT=*id*	Report resources to which the specified user or group has access by explicit reference in the access list (or OWNER field). This might be called 'direct' access. The minimum level of access is defined by the ACCESS parameter. More than one PERMIT command may be given. It is mutually exclusive with SCOPE. For an example, see "1.4.7 Reporting user or group scope".
SCOPE=*id*	Report resources to which the specified user or group has direct or indirect access. The minimum level of access is defined by the ACCESS parameter. If *id* is a user, then access through connects is also listed. Access through universal access and dataset global access table is also included with the proper exceptions. If a profile has the warning attribute, then this is considered equivalent to ALTER access to the profile. Group-special and group-operations are propagated down the group tree to resources owned by owned groups and to resources owned by users owned by owned groups. However, group-operation is in this release treated as group-special (exceptions through explicit access of connect groups are not taken into account, and the access listed is the text "OWNER"). This parameter may use much more memory than PERMIT=. By specifying * for *id*, a report is created showing the scope of jobs without a RACF user. The number of profiles listed may be reduced by including the SUPPRESS REASON= command (see "2.3.14 SUPPRESS").
ACCESS=*access*	Minimum level of access to be used in conjunction with the PERMIT or SCOPE parameter. Must be EXECUTE, READ, UPDATE, CONTROL or ALTER.

Note that for general resource classes where both the member class and the grouping class contain the same (member) resource, the SCOPE command will consider the profiles separately (as RACF commands would do), which may yield different access to the member resource than a FRACHECK.

The following report requests are mutually exclusive:

NONREDUNDANT	Report the access list of all dataset profiles that are either not covered by a less specific generic profile or are covered by a less specific generic profile but not one whose WARNING, AUDIT, and ERASE (where active globally) settings are identical and whose access list is identical or can be considered *similar*. Access is considered similar if the access list contains the group to which the dataset belongs with a lower level than present in GLOBAL DATASET member "&RACGPID.*". Access is also considered *similar* if a userid is present on the discrete profile access list with an access that is also granted via one of the groups to which the userid is connected. Erase-on-scratch is not considered if ERASE(ALL) or NOERASE global options are active. The report indicates a reason for inclusion. Reasons enclosed in hyphens are not reasons for non-redundancy at all, they are included only to help understand the report. This applies for instance to generic or global profiles that were considered to take over from a more specific, potentially redundant, profile (called 'candidate profiles'). For an example, see "1.4.6 Reporting non-redundancy reasons for profiles".
REDUNDANT	In addition to everything reported by NONREDUNDANT, also list the redundant profiles and the candidate profiles making them redundant.
NONDEFAULT	To report all profiles with non-default access control. Generally, each installation will have their own default (this is part of the security policy), but CONSUL/RACF tries to report in a general way profiles that deserve consideration. For *user datasets* "default" means that the owner field equals the first qualifier and that there is no access list and a UACC of NONE. For *group datasets with a user as owner*, "default" means that the userid is in the access list with ALTER and that the group dataset qualifier (the first qualifier unless altered by ICHCNX00) is in the access list with UPDATE. For *group datasets with a group as owner*, "default" means that this owning group is in the access list with ALTER. An example is given is "1.4.5 Reporting non-standard dataset access lists".
OUTOFGROUP	To report all outstanding permits on group datasets to users or groups outside that group (identified by the first qualifier unless changed by ICHCNX00). An example is given in "1.4.4 Reporting dataset access outside group".

If you want to restrict the options above to a subset of the dataset profiles, then you must be aware that you do not exclude profiles in the classes USER, CONNECT, GROUP, and GLOBAL. In addition, you must include *all* datasets profiles starting with the same qualifier if you want to process *any* dataset with that qualifier. See "1.4.6 Reporting non-redundancy reasons for profiles" for such examples of SELECT and EXCLUDE.

The following options are mutually exclusive:

DATASETS **DATASET** **DSN**	Report which datasets are covered by each profile reported by SCOPE, NONDEFAULT, NONREDUNDANT, OUTOFGROUP, or SENSITIVE. In the case of SENSITIVE, it adds *all* datasets covered by generic profiles, while without this keyword only the sensitive datasets are displayed. An IOCONFIG file is required.
AC1	Report on the protection of load modules that have the potential to circumvent RACF. This includes the AC(1) members of all APF libraries with all their entry points, and modules in APF libraries that are present in the Program Property Table (PPT) with the BYPASS option or a system key (0-7). The report includes the highest access to arbitrary users as given by the combined action of PROGRAM profile UACC, DATASET profile UACC, linklist residency, LPA residency, global access table, and profile warning mode. In addition it shows all module occurrences with linklist and LPA list concatenation numbers, incore MLPA residency, base member names for alias entries, the authorizing attributes, and the RACF profile member names. An IOCONFIG file is required. This report extends beyond column 132, and will be truncated if using the default line length. However, the last field is the dataset profile name, and depending on the length of your profile names, most profile names may well fit on the default line length. See "1.4.9 Verifying the protection of AC=1 APF modules" for a sample and explanation.
SENSITIVE **SENS**	Report on all dataset profiles and GLOBAL DATASET members that cover sensitive datasets that require protection. This currently includes APF datasets, page- and swap-datasets, SMF datasets, JES2 checkpoint datasets, SYS1.NUCLEUS, and RACF datasets. An IOCONFIG file is required.

The options above will not produce the correct results if you have excluded:

- Any profiles in the class DATASET starting with the same qualifier
- The GLOBAL DATASET general resource profile
- Any PROGRAM profiles (for AC1)

The output sort order can be given with the BY keyword.

BY=*list*	Keywords indicating report sort order. *list* is a list enclosed in parentheses with the keywords separated by commas indicating one or more of the following sort fields (the order given is the default sort order):
ID	User or group to which dataset belongs (but module name with REPORT AC1).
KEY **DSN**	Datasetname (resource name)
REASON	Reason (usable with NONDEFAULT and NONREDUNDANT)

Page headers may be inserted at appropriate points by specifying the PAGEBY keyword. The BY keyword must precede the PAGEBY keyword.

PAGEBY=*key*	Keywords indicating report page separation. The *key* specified must be the first one in the BY= list. *key* is one of the following fields:
ID	User or group to which dataset belongs
KEY **DSN**	Datasetname (resource name)
REASON	Reason (usable with NONDEFAULT and NONREDUNDANT)

Example 1 - report scope

This example shows how to request a report of all profiles that can be updated by anybody, and also requests a list of the datasets covered by each dataset profile.

```
REPORT SCOPE=*, ACCESS=UPDATE, DATASETS
```

Example 2 - report sensitive

This example requests a report on the protection of sensitive datasets.

```
REPORT SENSITIVE
```

Example 3 - report nondefault

This example shows how to report non-default profiles in a way suitable for distribution to RACF group owners. This accomplished by requesting page separators for each identity (first qualifier). The page header will change when the PAGEBY value changes.

```
REPORT NONDEFAULT, PAGEBY=ID
```


2.3.11 SELECT and EXCLUDE

The two commands have the same parameters. Different parameters on one SELECT or EXCLUDE command imply an AND function. More than one SELECT (or EXCLUDE) command implies an OR function between the commands. SELECT may be abbreviated to SEL. EXCLUDE may be abbreviated to EXCL. Repeating the same parameter in one SELECT or EXCLUDE is allowed *only* for the SCAN operand.

The selection is done prior to processing the LIST, VERIFY, REPORT, REMOVE, and UNLOAD commands. If you use SELECT/EXCLUDE in combination with VERIFY, REMOVE, and REPORT, then you *must* be aware of all profile types that these commands require. See the appropriate commands for additional detail.

The commands can occur in the context of a NEWLIST command. In this context, they specify a further subset of the profiles selected by selections outside the NEWLIST context (i.e. before the first NEWLIST).

The following parameters can be used to specify values for properties of profiles:

CLASS=*class*	The RACF class (e.g. USER, GROUP, DATASET, CONNECT, PROGRAM, ACCTNUM, GCICSTRN, ...). The special value GENERAL indicates all kinds of general resource classes.
SEGMENT=*segment*	This parameter selects profiles including the indicated segment types. In Restructured databases (not available prior to RACF 1.9), only the segment will be present in the selected record; in Non-restructured RACF databases, a complete profile will be selected including all its segments. SEGMENT=BASE will always be true for a non-RDS database.
DB=*number*	This selects profiles based on the sequence number of the RACF database that originally contained them (as defined in the RACF database name table).
RBA=*hex*	This selects profiles based on their Relative Byte Address in the originating database. Together with the DB= parameter, this would describe exactly one profile. Its main use is to keep operating in situations where BAM conflicts are reported by ICHUT200, for instance by excluding profiles that are in use, but not present in the index. See "1.5.4 Handling database layout problems" for an example.
VOLUME=*volser* **VOL=*volser*** **VOLSER=*volser***	Selects the presence of a specific volume serial in the list of volumes in a profile.
SCAN=*name* **SCAN='*string*'**	A string to scan for. All of the profile is scanned. This parameter is a fast way to restrict operations to certain profiles. It is also the only way to scan for substrings in arbitrary fields. If more SCAN parameters are coded on one SELECT or EXCLUDE then the scan matches if any of the SCAN values matches. Use quotes around the value if it contains blanks, commas, lowercase, parentheses, or the comment delimiter '/*'. The unquoted SCAN parameter is converted to uppercase before comparison.

The following parameters can be used to search for values in a specific field that you can name:

fieldname=val ***fieldname<val*** ***fieldname>val*** ***fieldname<=val*** ***fieldname>=val*** ***fieldname<>val***	Specifies a field name to be searched for in the profile. The *fieldname* must be one defined in a template or the keyword FIELDVALUE to indicate that the field name is given by the FIELD parameter (see appendix B for RACF 1.9 field names or "2.3.5 LIST" for a short summary). The *val* specifies a value to search for in the field specified. It can be enclosed in quotes, but this is optional unless it contains a blank, comma, or semicolon. All normal comparison operators are supported. Conversion to internal format is supported for the following types: binary, hex, date, access, authority, audit access, audit, and text. The type that CONSUL/RACF assumes for a field can be displayed with the SHOW TEMPLATES command. The formats accepted are like the display format detailed in the LIST reference section, except for the date format. The following date formats are supported on input: the julian format YYDDD, the keyword TODAY, and the keyword DUMPDATE (indicating the date the RACF database was unloaded).
FIELD=*fieldname*	Specifies a field name to be searched for in the profile. Must be used in conjunction with the keyword fieldname FIELDVALUE=, as explained above. The field name must be one defined in a template. This parameter is meant for the situation that the fieldname is the same as one of the other SELECT parameters.

The following parameters can be used for selections based on the profile name. They are mutually exclusive on a single command.

PROFILE=*name*	Selects a specific profile name
MASK=*mask* **FILTER=*mask***	A mask for the profile key. Use %, * and ** (enhanced generic naming). The filter is *always* interpreted as enhanced generic, independent of the RACF database setting for EGN.
QUAL=*id*	Matches dataset profiles with *id* as the first qualifier (as changed by ICHCNX00). In addition, it matches non-dataset profiles (as would be required for a SELECT command with REPORT/REMOVE REDUNDANT and other functions needing the user/group structure and some general resource profiles - however be aware that in the case of an EXCLUDE command you should add CLASS=DATASET).

The SELECT and EXCLUDE commands have many keyword parameters. Most of these are only active for specific resource classes. In the tables below these parameters are grouped by resource class.

The following parameters are valid for any resource class:

GENERIC	Selects generic profiles.
DISCRETE	Selects discrete profiles.
WARNING **WARN**	Selects profiles with the warning indicator on.
NOWARNING **NOWARN**	Selects profiles with the warning indicator off.

For the dataset class valid attributes are:

PADS	Selects (dataset) profiles with a non-empty conditional access list as well as profiles in the class PROGRAM.
VSAM	Select DATASET profiles with the VSAM indicator.
NONVSAM	Select non-VSAM dataset profiles.
MODEL	Select DATASET profiles with the MODEL indicator on.
NOMODEL	Select DATASET profiles with the MODEL indicator off.
TAPEDSN	Select DATASET profiles with DSTYPE=TAPE.
NOTAPEDSN **NOTAPE**	Select DATASET profiles without DSTYPE=TAPE.
GROUPDS **GROUPDSN**	Select DATASET profiles with group-dataset indicator on.
USERDS **USERDSN**	Select DATASET profiles with the user-dataset indicator on.
ERASE	Select (dataset) profiles with the erase-on-scratch flag on. This is independent of the global ERASE setting.
NOERASE	Select (dataset) profiles with the erase-on-scratch flag off. This is independent of the global ERASE setting.

For the TAPEVOL class valid attributes are:

SINGLEDS	Selects (tape volume) profiles with the single-dataset indicator on.
NOSINGLEDS	Selects (tape volume) profiles with the single-dataset indicator off.
AUTOTAPE **AUTO**	Select automatic TAPEVOL profiles.
NOAUTOTAPE **NONAUTOTAPE** **NOTAUTOTAPE** **NONAUTO** **NOTAUTO** **NOAUTO**	Select non-automatic TAPEVOL profiles.
TVTOC	Select TAPEVOL profiles with a TVTOC.
NOTVTOC	Select TAPEVOL profiles without a TVTOC.

Valid attributes for both USER and CONNECT classes are:

SPECIAL **SPEC**	Select profiles with the SPECIAL attribute (CONNECT or USER).
NOSPECIAL **NOSPEC**	Select profiles without the SPECIAL attribute (CONNECT or USER).
OPERATIONS **OPER**	Select profiles with the OPERATIONS attribute (CONNECT or USER).
NOOPERATIONS **NOOPER**	Select profiles without the OPERATIONS attribute (CONNECT or USER).
AUDITOR	Select profiles with the AUDITOR attribute (CONNECT or USER).
NOAUDITOR	Select profiles without the AUDITOR attribute (CONNECT or USER).
REVOKE	Select profiles with REVOKE attribute (CONNECT or USER). In addition, the date of the RACF dataset unload is used to select profiles that are revoked by date based on the revoke date and resume date fields.
NOREVOKE	Select profiles without the REVOKE attribute (CONNECT or USER). In addition, the date of the RACF dataset unload is used to verify that profiles are not revoked by date based on the revoke date and resume date fields.
ADSP	Select profiles with the ADSP (automatic dataset protection) attribute (CONNECT or USER).
NOADSP	Select profiles with the NOADSP attribute (CONNECT or USER).
GRPACC	Select profiles with the GRPACC (group access) attribute (CONNECT or USER).
NOGRPACC	Select profiles without the GRPACC attribute (CONNECT or USER).

Valid attributes for both GROUP and CONNECT classes are:

NOTERMUACC **NOTERM**	Select profiles with the NOTERMUACC (no terminals access based on UACC) attribute.
TERMUACC	Select profiles without the NOTERMUACC attribute.

Valid attributes for the USER class only are:

UAUDIT	Select USER profiles that have user-level auditing active.
NOUAUDIT	Select USER profiles that have no user-level auditing active.
PASSWORD	Select USER profiles that have a password.
NOPASSWORD	Select USER profiles that do not have a password.
OIDCARD **OID**	Select USER profiles that have an OID card key.
NOOIDCARD **NOOID**	Select USER profiles that do not have an OID card key.

Example 1 - the AND function

This example selects dataset profiles with a first qualifier of SYS1. It illustrates the AND function between the parameters on one SELECT statement.

```
SELECT CLASS=DATASET, QUAL=SYS1
```

Example 2 - the OR function

This example selects DATASET profiles as well as DLFDATA profiles that match a filter SYS1.** (i.e. everything with SYS1 as the first qualifier). It illustrates the OR function between multiple SELECT statements.

```
SELECT CLASS=DATASET, FILTER=SYS1.**
SELECT CLASS=DLFDATA, FILTER=SYS1.**
```

Example 3 - combining select and exclude

This example selects dataset profiles with UACC greater than NONE, but excludes all SYS2 profiles. This illustrates the exception function of the EXCLUDE command.

```
SELECT CLASS=DATASET, UACC>NONE
EXCLUDE FILTER=SYS2.**
```

Example 4 - muliple keywords

This example illustrates the multiple scan facility. It is much more effcient than repeated SELECT commands that would produce the same result. Note that the SCAN= operand value is case-sensitive if it is enclosed in quotes.

```
select scan='JONES', scan=parker, scan='SMITH', scan=perry
```


2.3.12 SHOW

This command can be used to request fixed-format reports where the sort order cannot be changed, unlike reports requested with the REPORT option.

TEMPLATES **TEMPLATE**	List all fields present in the templates, together with the default input and output format assigned by CONSUL/RACF. The sort order is by field name. Sample output is present in appendix B.
CLASSES **CLASS**	List the number of generic and discrete profiles for each general resource class.

Example - show templates

The following command request a listing of all template fields.

```
SHOW TEMPLATES
```


2.3.13 SORTLIST

This command has the same syntax as the LIST command. It is followed by a list of field names. However, unlike the LIST command, the profiles are listed in ascending sequence of the fields to be listed. For instance , if you specify:

```
SORTLIST CLASS, KEY
```

then the class and profile key of the selected profiles will be output sorted in ascending sequence of class and profile key.

Another difference is that in the scope of a NEWLIST command, the SORTLIST will generate page headers by default, while the LIST command will *not* generate any pageheaders by default.

For a further discussion of the syntax as well as examples, see the reference section on LIST.

2.3.14 SUPPRESS

This command can be used to suppress verification failure messages for specific volumes or catalogs. This is most useful if your installation has full-volume copies of critical volumes stand-by with a different volume label, or emergency copies of the master catalog.

VOL=*volser* **VOLUME=*volser***	Suppress error messages relating to this volume.
CAT=*catnames* **CATALOG=*catname***	Suppress error messages related to this catalog.
ICHCNX00	Suppress invocation of this RACF exit to determine the first qualifier. This command may be necessary if the exit depends on key 0 operation.
REASON=*list*	Change REPORT SCOPE= or REPORT PERMIT= reports to exclude profiles that would be included only for the reasons indicated. The reasons in the list can be UACC, GLOBAL, WARN, or SELFCONNECT. The latter reason suppresses profile accesses to owned groups that would be included in the scope report for group-special users, because the user is allowed to CONNECT himself to owned groups.
MSG=172	Suppress error message 172.
ID=*id*	Suppress error messages and report lines concerning this user or group.

Example - suppress by volume

The following example shows how to suppress eror messages for a non-existing volume used as a dummy for archived disk datasets.

```
SUPPRESS VOLUME=MIGRAT
```


2.3.15 UNLOAD

Unload of the currently active and primary RACF datasets is the default action if file SYSUT2 is present. In addition, it may be requested explicitly by the command

```
UNLOAD
```

without any parameters. The command ALLOC or allocation of files SYSRAC00 through SYSRAC*nn* can be used to change the source of RACF profiles. For details, see "1.1 Unloading and selecting RACF datasets", "2.3.1 ALLOCATE", and "2.2.1 DDname overview".

While allocating SYSUT2 is generally equivalent to specifying UNLOAD, the UNLOAD command has the following benefits:

- In a MVS system an error message CNR002I will be issued if no SYSUT2 file was found allocated.
- In a CMS system the file FILE SYSUT2 A will be created automatically if no GLOBAL SYSUT2 was defined.

2.3.16 VERIFY

The VERIFY command can be used to check the consistency of the database internally as well as in relation to resource data collected by CONSUL/COLLECT. Commands to remedy the shortcomings found are generated if the CMDOUT file has been allocated.

This command has the following parameters to analyze a RACF database without an IOCONFIG file prepared by CONSUL/COLLECT:

PADS	Verify that each program mentioned in a conditional access list indeed exists as a program profile. If the CMDOUT file has been allocated, commands will be generated to delete conditional access list entries without a program profile. For discussion and example see "1.3.4 Checking for obsolete conditional access lists".
PERMIT **PERM**	Verify that each id entry in a standard or conditional access list, each owner, superior group, or notify field is defined as a user or group. If the CMDOUT file has been allocated, then commands will be generated to delete the permits and to change OWNER fields. For an example and additional discussion see "1.3.2 Finding and removing orphan permits".
CONNECT **CON**	Verify that the user, group, and connect profiles all provide the same connect information. No commands can be generated to remedy the situation. For an example see "1.3.10 Finding user/group/connect inconsistencies".

To analyze the database in combination with the resources defined in the system, an IOCONFIG file is necessary. If present, the following parameters are supported to verify that profiles are present for the resources to be protected:

PROTECTALL **PROT**	Verify that all datasets in the VTOC and VVDS are protected by a (discrete or generic) profile. In a PROTECTALL(FAIL) environment the datasets will be *inaccessible*, in a NOPROTECTALL environment the datasets will be *unprotected*. No commands are generated to remedy the situation. The IOCONFIG file is required. For an example see "1.3.6 Finding and protecting unprotected datasets".
INDICATED **IND**	Verify that datasets with the RACF indicated bit on in the VTOC (non-VSAM) or catalog (VSAM) have a corresponding discrete profile, even if covered by a generic profile. If the CMDOUT file has been allocated, then commands are generated to add discrete profiles with an access list that is a copy of the generic profile currently used (through use of the FROM parameter). An IOCONFIG file is required. For an example see "1.3.9 Finding and resetting unnecessary RACF indicated bits".

In addition, the following parameters may be specified to verify that the profiles cover resources:

DATASET **ONVOLUME** **ONVOL**	Verify that each discrete profile has a corresponding RACF indicated dataset. If the CMDOUT file has been allocated, then commands are generated to delete the misleading (because unused) discrete profiles. An IOCONFIG file is required. For an example, see "1.3.7 Removing unused discrete profiles".
NOTEMPTY **NONEMPTY** **GENERIC** **GEN**	Verify that generic dataset profiles that are also covered by a less specific generic profile are non-empty, i.e. datasets exist that are covered by the generic profile. This is meant to remove obsolete generic profiles. You should be aware that profiles intended to prevent or allow *allocation* in a PROTECTALL environment (in a way that differs from the less specific profile) will also be flagged non-empty, while it may not be your intention to remove these profiles (for instance because the datasets have been archived or because you have ADSP active, or because they are used for tape datasets). If the CMDOUT file has been allocated, then commands are generated to delete the empty profiles. An IOCONFIG file is required. For an example and additional discussion, see "1.3.8 Removing unused generic profiles".
ALLNOTEMPTY **ALLNONEMPTY**	Verify that all generic dataset profiles cover real datasets. In addition to the NOTEMPTY option above, profiles for which *no* less specific profile exists will also be flagged. If the CMDOUT file has been allocated, then commands are generated to delete the empty profiles. An IOCONFIG file is required.
PROGRAM **PGM**	Verify that each dataset added as a member to a PROGRAM profile indeed exists. If the CMDOUT file has been allocated, then commands are generated to deleted the non-existing dataset from the PROGRAM profile. An IOCONFIG file is required. For an example, see "1.3.5 Checking for program existence".

There is one option to invoke all VERIFY options at the same time. This may cause excessive memory and CPU usage on very large systems.

ALL	Select all analysis parameters at the same time

In addition, the sort order of the error messages and generated commands may be changed. This is often a good way to detect related events.

BY=*list*	Keywords indicating error message sort order. *list* is a list enclosed in parentheses with the keywords separated by commas indicating one or more of the following sort fields (the order given is the default sort order):
MSG	Message type. This is an internal value (not the message number) that will cause messages to be sorted by issuing function (like VERIFY PROGRAM or REMOVE PERMIT). In addition, similar message will appear in groups. If MSG is the first in the sort order (default), the messages for each issuing function will start on a new page with a header indicating the function. Some commands invoke more than one function, causing the output to be split across the functions invoked (like VERIFY ALL or REMOVE USER).
VOL	Volume serial.
DSN	Datasetname (resource name).
PGM	Program name.
ID **PERM**	User or group (permit).

Example - combining verifications

This example shows a combination of verification options that you could use for a periodical "health-check" on your database.

```
VERIFY PROGRAM, PADS, PROTECTALL, ONVOLUME, PERMIT, CONNECT
```


PART 3 Messages

The messages issued by CONSUL/RACF have a message identifier of the form CNR*nnn*I where *nnn* is the message number. Behind the message identifier, a severity code is indicated. The program returns as completion code the highest severity code encountered. The general meaning of the severity codes and hence of the completion code is as follows:

00 Normal message, giving status or summary information.

04 Error condition found as result of VERIFY or REPORT processing. Removal of the error condition can be attempted by means of a command generated by CONSUL/RACF (if CMDOUT was allocated).

08 Error condition found as result of VERIFY or REPORT processing. No commands can be generated by CONSUL/RACF to remove the error.

12 Syntax error in command input.

16 Invalid or unsupported files connected to CONSUL/RACF.

20 Unsupported condition found in RACF database, VTOC or VVDS.

24 Internal error or other unexpected and unsupported condition in CONSUL/RACF detected.

In the rest of this section, all error messages are listed with an explanation and possible actions to take.

3.1 CNRACF messages

CNR000I 12 CNRACF terminated due to input errors
Previous messages indicate an error in the program parameters or command input file. CONSUL/RACF does not perform any commands if the command input is not syntactically correct. Correct the errors and submit the job again.

CNR001I 12 No UNLOAD, LIST, SHOW, DISPLAY, REPORT or VERIFY specified
No commands were given or implied that would result in any output. Specify one of the commands indicated in the message.

CNR002I 16 Output dataset OPEN failed - file *ddname*
The OPEN for the indicated file (SYSUT2 or CMDOUT) failed. If you are running a batch job, refer to the job log for an abend code and reason code (the abend code is probably 013). If you are running TSO interactively and no abend code is listed on your terminal, try specifying PROFILE WTPMSG and try it again. The *ddname* field in the CNR002I message probably contains garbage. The meaning of the abend code and reason code can be found in the MVS system messages and codes manuals.

CNR003I 16 Open for input failed on file SYSUT1
Check the DD statement for SYSUT1, correct the error and submit the job again.

CNR004I 00 Processing started for SYSUT1
unloaded by program CNRACF *v.l.m date time* **job** *name* **at** *date time*
source dataset *i* **was** *volume datasetname*
This message indicates the version of CONSUL/RACF that created the unloaded RACF database, as well as the date and time the database was unloaded. For each unloaded RACF dataset contained in the file, the original volume and datasetname are listed.

CNR005I 00 *nnnnnn* **profiles read,** *yyyy* **profiles selected (***pp***%)**
This message is written at the end of profile input phase. During this phase SELECT, EXCLUDE, LIST and UNLOAD commands are processed and information is stored for the other commands. The total number of profiles in use in the database read is listed, as well as the number of profiles selected by the SELECT and EXCLUDE commands. This does not apply to SELECT and EXCLUDE in the scope of a NEWLIST command.

CNR006I 04 *nn* **profiles truncated on SYSUT2**
Due to the insufficient record length of file SYSUT2, profiles were truncated. This may result in erroneous error messages with respect to the truncated profiles if subsequent processing is done on the unloaded file, but this is not necessarily the case. For instance, truncated group profiles will cause spurious error messages if you try or imply the VERIFY CONNECT command, but in general it will not cause any other trouble, due to the redundancy in the database.

CNR007I 16 SYSUT1 file is empty
The SYSUT1 file was allocated, but contained no records.

CNR008I 16 SYSUT1 end-of-file reached before *type* **record**
SYSUT1 contained some status records, but the indicated record type was not present. The indicated *type* can be ICB for the first RACF database record, or CRDB for an origin database record. Probably the unload failed, or the system catalog points to a previous version of your unloaded dataset (see CNR014I for a possible cause for this problem).

CNR010I 16 OPEN abend *hhh* on file *ddname*
The OPEN for the indicated file (SYSRAC*nn*) failed. If you are running a batch job, refer to the job log for an abend code and reason code (the abend code is probably 013). If you are running TSO interactively and no abend code is listed on your terminal, try specifying PROFILE WTPMSG and try it again. The *ddname* field in the CNR010I message probably contains garbage.

CNR011I 16 I/O error: *synadaf message*
An I/O error occurred on one of the SYSRACNN files. Check that the file allocated is indeed a RACF database with RECFM=F and LRECL=1024 for a non-restructured database and LRECL=4096 for a restructured database. On a VM system, it may also occur for a database on an OS formatted minidisk; in this case you can process the database by copying it to a temporary CMS formatted minidisk and process this copy.

CNR012I 12 More than 10 RACF datasets parallel not supported - use separate runs
This version of CONSUL/RACF does not support processing more than 10 databases at the same time. Use the ALLOC DB= command to select 10 or less databases for processing. If your site requires operation on more than 10 datasets, please direct a request to CONSUL Risk Management B.V. or your local sales representative.

CNR013I 16 No file SYSUT1 or SYSRAC01 pre-allocated
No source of RACF profiles was found. Normally the current RACF database would be allocated dynamically, but you are running on a CMS system, or on an MVS system without RACF active. Allocate the database you want to process explicitly to the SYSRAC01 file (or if the database is split, to the SYSRAC*nn* files).

CNR014I 16 SYSUT1 file does not start with CRCF record
Either the dataset connected to file SYSUT1 is not an unloaded RACF database, or it was unloaded by an old version of the program that is not supported anymore. Check your JCL. If no errors can be found there, check the job that created the dataset. Possibly you specified DISP=(NEW,CATLG) with a non-specific volume request (e.g. just UNIT=SYSDA), while a dataset with the same name was already cataloged. In that case, the IEF messages will contain NOT CATLGD 2 as final disposition.

CNR015I 16 OPEN of master failed: file SYSRAC01 dataset *datasetname* on *volume*
Refer to CNR002I and CNR010I for a discussion.

CNR016I 16 OPEN for input failed: file SYSRAC*nn* dataset *datasetname* on *volume*
Refer to CNR002I and CNR010I for a discussion.

CNR017I 00 Processing started for SYSRAC*nn* *volume datasetname*
The dataset open was successful for the file indicated, and input of the database was started.

CNR018I 16 No extents present for SYSRAC*nn* *volume datasetname*
The file indicated was opened successfully, but no extents were present (the dataset is empty).

CNR019I 16 ALLOC invalid if SYSRAC01 pre-allocated
An ALLOCATE command was present in the commands as well as a pre-allocated database. You must either remove the ALLOCATE command or remove the SYSRAC01 file.

CNR020I 00 Profile input terminated, LIMIT *lim* reached
The OUT or IN limit you specified on a LIMIT command has been reached, no more profiles will be read.

CNR021I 20 Unsupported BAM format: 1st block on odd nibble, block number *nnnn*, database *i*

During input of the Block Availability Map (BAM) an unsupported format was detected (a nibble is four bit and describes the segments of one block in non-RDS format). If no other errors are found and the error is reproducible, please send a problem report.

CNR022I 20 Unsupported BAM format: odd # blks in other than last BAM block - block number *nnnn* database *i*

During input of the Block Availability Map (BAM) an unsupported format was detected. If no other errors are found and the error is reproducible, please send a problem report.

CNR023I 16 Open for input with QSAM failed for file *ddname* dataset *dsn* on *vol*

While using BDAMQSAM processing (currently this is the default mode), after conclusion of BDAM processing the dataset could not be opened again with QSAM processing. Possibly other error messages were issued to indicate what went wrong.

CNR024I 20 Index marker not on block boundary: *ddname* block *nnnn* segment offset *i*

The RACF database was found to start an index block at an other segment than the first in a block. This format is not supported. If the problem is reproducible, run ICHUT200. If no errors are revealed, please send a problem report.

CNR025I 20 Index block with invalid length: *ddname* block *nnnn* length *len*

The RACF database was found to contain an index block with a length unequal to 1024 for non-RDS and 4096 for RDS. This format is not supported. Please send a problem report if the error is reproducible.

CNR026I 20 End of file in 2nd segment of profile: *ddname* block *nnnn* segment offset *i*

At the specified position in the RACF database a profile was being read and not complete at the end of the dataset. Please send a problem report if the error is reproducible.

CNR027I 20 Unused segment instead of profile continuation: *ddname* block *nnnn* segment offset *i*

At the specified position in the RACF database a profile was being read and not complete according to the physical profile length field, but the block availability map indicates that next segment is not occupied. This *may* happen because of update activity on the database while performing the read.

If the problem and the place where it occurs is reproducible, run ICHUT200 to analyze the database. If still no errors are revealed, please send a problem report.

If the problem is intermittent and annoying, please send a problem report, too. The developers never experienced trouble serious enough to justify an enqueue on the database against updates.

CNR028I 20 End of file in middle of profile: *ddname* block *nnnn* segment offset *i*

At the specified position in the RACF database a profile was being read and not complete at the end of the dataset. Please send a problem report if the error is reproducible.

CNR029I 20 Segment type X'*hh*' not supported - *ddname* block *nnnn* segment offset *i*

An unknown database segment type was encountered. If the problem is reproducible at the same place, run ICHUT200. If this does not reveal structural errors, please send a problem report.

CNR030I 20 Unsupported template *addr. hexvalue* len *ll* searching *fldname* in entity type *n* ICB at *addr*

While using the templates to scan a profile, an unsupported kind of template was encountered. If the error is reproducible, please submit a problem report.

CNR031I 20 Template not found for *profile key*
The profile indicated in the message had an entity type for which no template was found in the database. If the error is reproducible, please submit a problem report.

CNR032I 12 File *ddname* not allocated
The filename requested on a PRINT command was not found allocated. Review your JCL.

CNR033I 00 *datasetname* has *number* segments in use, *number* segments free (*pp*% used)
Index uses *pp*%. Space beyond *pp*% never used
Free space completely fragmented
This message reports on the contents of a RACF dataset. Each segment is 256 byte. Free space can be present at the end of the database (never used), or fragmented through the database. If all space is fragmented, then the "completely fragmented" message text is used, otherwise the "space beyond *pp*% never used" message text.

CNR034I 00 No reference found to *identity*
A function was requested specifically for a user or group identity, but no references were found.

CNR035I 00 at SYSUT1 record *nnnnn*, originally DB seq *i* RBA *hexnum*
This message gives the location where a previous error message occurred.

CNR036I 00 at SYSRAC*nn* block *nnnn* segment offset *i* RBA *hexnum*
This message gives the location where a previous error message occurred.

CNR037I 16 Multisegment record truncated
CNR037I 16 Last record truncated at end-of-file
This message indicates that the SYSUT1 input file contained records longer than the 256K supported by CONSUL/RACF, or that end-of-file was reached for a RECFM=VBS input file in the middle of a multi-segment record.

CNR038I 12 License period has expired
The license period for CONSUL/RACF has expired. You can obtain a new authorization code from your software vendor. This authorization code has to be applied to the CNRACF load module as described in the installation section.

CNR039I 00 CNRACF used *ss.t* CPU seconds and took *nnn* wall clock seconds.
This message is given during successful termination. It indicates the resource usage as well as the elapsed time for this run.

CNR040I 04 RACF indicator set but no discrete profile found for *volser datasetname*
This message is issued due to a VERIFY INDICATED command.

CNR041I 04 Discrete profile found but RACF indicator not set *volser datasetname*
This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a DELDSD NOSET command will be generated.

CNR042I 04 Discrete profile present but no dataset on volume *volser datasetname*
This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a DELDSD NOSET command will be generated.

CNR043I 04 Discrete profile present but volume not mounted *volser datasetname*
This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a DELDSD NOSET command will be generated.

CNR044I 04 Dataset not found for program *progname* - *volser datasetname*
This message is issued due to a VERIFY PROGRAM command. To solve the error condition an RALTER DELMEM command will be generated.

CNR045I 04 **Obsolete permit** *identity* **unknown program** *progname - volser datasetname*
This message is issued due to a VERIFY PADS command. To solve the error condition a PERMIT DELETE WHEN(PROGRAM()) command will be generated.

CNR046I 04 *event* **permit** *identity* **in access list of non-VSAM** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() command will be generated.

CNR047I 04 *Event* **permit** *identity* **on condition** *progname - volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() WHEN(PROGRAM()) command will be generated.

CNR048I 04 *event* **permit** *identity* **in access list VSAM profil** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() command will be generated.

CNR049I 04 *event* **permit** *identity* **on VSAM condition** *progname - volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() WHEN(PROGRAM()) command will be generated.

CNR050I 04 *event* **permit** *identity* **in access list generic profile** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT GENERIC DELETE command will be generated.

CNR051I 04 *event* **permit** *identity* **on condition** *progname* **dataset** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT GENERIC DELETE WHEN(PROGRAM()) command will be generated.

CNR052I 04 *event* **permit** *identity* **in access list model dataset** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE command will be generated.

CNR053I 04 *event* **permit** *identity* **on condition** *progname* **model** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE WHEN(PROGRAM()) command will be generated.

CNR054I 04 *event* **permit** *identity* **on program prof** *progname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE command will be generated.

CNR055I 04 *event* **owner** *identity* **of non-VSAM dataset profil** *volser datasetname* **- make** *newowner*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a ALTDSD VOL() OWNER() command will be generated with the default owner selected with DEFAULT OWNER=, or, if defined, the first qualifier as owner id. The new owner selected is shown in the message.

CNR056I 04 *event* **owner** *identity* **of VSAM dataset profile** *volser datasetname* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a ALTDSD VOL() OWNER() command will be generated with the default owner selected with DEFAULT OWNER=, or, if defined, the first qualifier as owner id. The new owner selected is shown in the message.

CNR057I 04 *event* **owner** *identity* **of generic dataset profile** *datasetname* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a ALTDSD GENERIC OWNER() command will be generated with the default owner selected with DEFAULT OWNER=, or, if defined, the first qualifier as owner id. The new owner selected is shown in the message.

CNR058I 04 *event* **owner** *identity* **of model dataset profile** *datasetname* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a ALTDSD OWNER() command will be generated with the default owner selected with DEFAULT OWNER=, or, if defined, the first qualifier as owner id. The new owner selected is shown in the message.

CNR059I 04 *event* **owner** *identity* **on program prof** *progname* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a RALTER OWNER() command will be generated with the default owner selected with DEFAULT OWNER=. The new owner selected is shown in the message.

CNR060I 04 *event* **owner** *identity* **on user** *userid* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition an ALTUSER OWNER() command will be generated with the default owner selected with DEFAULT OWNER= as the new owner. The new owner selected is shown in the message.

CNR061I 04 *event* **owner** *identity* **on group** *group* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition an ALTGROUP OWNER() command will be generated with the default owner selected as the new owner. The new owner selected is shown in the message.

CNR062I 04 *event* **owner** *identity* **connect** *userid/group* **- make** *newowner*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a CONNECT OWNER() command will be generated with the connect group as the new owner. The new owner selected is shown in the message.

CNR063I 04 *event* **owner** *identity* **general resource profile** *class key*

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a RALTER OWNER() command will be generated with the default owner selected.

CNR064I 04 *event* **permit** *identity* **general resource profile** *class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE command will be generated.

CNR065I 08 Missing userid *userid* **on group** *group*
This message is issued due to a VERIFY CONNECT command. No support is present to remove this condition.

CNR066I 08 Missing group *group* **on user** *userid*
This message is issued due to a VERIFY CONNECT command. No support is present to remove this condition.

CNR067I 08 Missing connect *userid* **to group** *group*
This message is issued due to a VERIFY CONNECT command. No support is present to remove this condition.

CNR068I 00 *event* **id -** *identity* **referenced** *number* **time as owner or permit**
This message summarizes the error conditions found by the VERIFY PERMIT or MOVE/REMOVE PERMIT/USER/GROUP/NOTIFY commands for each undefined or removed *identity*.

CNR069I 08 Catalog found on *volser* **also found on volume** *volser datasetname*
This message is issued if the IOCONFIG file contained two volumes with the same catalog on it, and no information was present to decide which one to use. See "1.5.1 Handling hot-standby volumes" for a discussion.

CNR070I 08 Component name found twice in VTOC - *volser datasetname*
Two identical format 1 DSCB keys in the VTOC are not supported. If the error is reproducible (run CNFCOLL again first), the condition may be resolved by letting the VTOC index (if present) decide which one is in use, and modifying the DSCB of the other one to another name (if you want to keep the data) or to a format 0 DSCB. If you modify the DSCB, you will have to rebuild the VTOC index.

CNR071I 08 Component name found in VVDS but not in VTOC - *volser datasetname*
Incidental cases may be the result of actions performed by the system between reading of the VTOC and the VVDS by IOCONFIG (opening the VVDS takes a considerable amount of time). If this message is reproducible for the same component (run IOCONFIG again first), then you have a problem. Perform the IDCAMS DIAGNOSE function on the VVDS: maybe a DELETE CLUSTER or DELETE VVR command will help.

CNR072I 08 Catalog not found on any volume for cluster name *datasetname*
This message is issued together with CNR073I to indicate that the VVDS points to a catalog that was not found in the IOCONFIG file. This message lists the cluster name that was cataloged in the now-unavailable catalog.

CNR073I 08 Catalog not found on any volume *datasetname*
This message is issued to indicate that references were found from the VVDS to the catalog indicated. The cluster names that were cataloged in the now-unavailable catalog are listed by separate CNR072I messages.

CNR074I 04 Discrete profile for VVDS present (not used by DFP) *volser datasetname*
DFP does not consult RACF for operations on the VVDS. Instead, APF authorization is required to open it (DFP 3.1) or to open it for UPDATE (pre-DFP 3.1). Therefore, the VVDS profile gives a *false* picture of the access requirements of the VVDS. For a pure RACF/DFP combination it should be deleted to avoid misleading data. However, you might want to verify that your non-IBM storage management products are properly using DASDVOL class and not using a VVDS dataset profile.

CNR075I 04 Inaccessible dataset (RACF indicated and no profile) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL or VERIFY INDICATED command. To solve the error condition an ADDSD NOSET command will be generated, but only if VERIFY INDICATED was *not* specified. Note that adding the profile may not be enough, you might want to enhance the access list, or use a generic profile instead.

CNR076I 08 Unprotected dataset (not RACF-indicated, no generic) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL command in NOPROTECTALL or PROTECTALL(WARN) environment. No command is generated.

CNR077I 04 Generic profile without matching datasets *datasetname*
This message is issued due to a VERIFY GENERIC command. To remove the condition, a DELDSD GENERIC command is generated. Note that in general you might not want this to happen, a better alternative is offered by the REMOVE REDUNDANT; LIMIT GENERIC command sequence. Or you might go into the editor and delete unwanted DELDSD commands.

CNR078I 04 Redundant non-VSAM dataset profile *volser datasetname*
This message is issued due to the REMOVE REDUNDANT command. The command generated is DELDSD VOL().

CNR079I 04 Redundant VSAM dataset profile *volser datasetname*
This message is issued due to the REMOVE REDUNDANT command. The command generated is DELDSD VOL().

CNR080I 04 Redundant TAPE dataset profile *volser datasetname*
This message is issued due to the REMOVE REDUNDANT command. The command generated is DELDSD VOL().

CNR081I 04 Redundant MODEL dataset profile *datasetname*
This message is issued due to the REMOVE REDUNDANT command. The command generated is DELDSD.

CNR082I 08 Inaccessible dataset (not indicated and no generic) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL command in a PROTECTALL(FAIL) environment. No command is generated.

CNR083I 04 Redundant generic dataset profile *datasetname*
This message is issued due to the REMOVE REDUNDANT command. The command generated is DELDSD.

CNR084I 08 Catalog dump of *volser* conflicts with same catalog *vol2 catalogname*
This message indicates that the configuration input file IOCONFIG contains catalog dumps for two catalogs with the same name but residing on different volumes. CONSUL/RACF cannot handle this situation.

CNR085I 08 Duplicate cluster entry found in 1 catalog on volume *volser datasetname*
This message indicates that the configuration input file IOCONFIG contains a catalog dump for a catalog on volume *volser* with the same cluster entry *datasetname* appearing twice. This might happen if you concatenate two IOCONFIG files containing dumps of the same catalog.

CNR086I 08 Ownership cell not found for cluster cataloged on *volser datasetname*
This message indicates that the configuration input file IOCONFIG contains a catalog dump from a catalog on volume *volser* with a cluster entry *datasetname* for which the ownership cell was not found. Check whether the record length of the IOCONFIG file is sufficient for your catalog records.

CNR087I 00 Number of detail error messages is *nnn*
This message summarizes the total number of detail error messages that will subsequently be issued.

CNR088I 00 ID based suppress or limit request(s) - *nnn* detail message(s) suppressed
This message summarizes the number of suppressed messages due to the SUPPRESS ID= and LIMIT ID= commands. Note that these two commands will only limit the number of messages issued, not the work performed by the VERIFY and REMOVE commands (use SELECT QUAL= for this if applicable).

CNR089I 08 Cluster found in VVDS not found in catalog on *volser datasetname*
The indicated cluster (*datasetname*) was found in a VVDS pointing to a certain catalog name. A catalog with this catalog name was found on volume *volser*, but it did not contain the cluster name. This may be caused by either a synchronization problem (VVDS dump not exactly the same situation as the catalog dump), a problem caused by multiple catalogs with the same name on different volumes, or a broken catalog. The last two cases should show up if you run the IDCAMS DIAGNOSE command on the VVDS. The first case is easily decided by comparing two separate runs - error messages due to synchronization problems will not be reproducible.

CNR090I 00 *volser* suppress request - *nnn* detail messages suppressed
This message summarizes the result of the SUPPRESS VOL= command per volume.

CNR091I 08 *volser* message limit exceeded - *nnn* detail messages suppressed
This message summarizes the result of the LIMIT MSG= command per volume.

CNR092I 00 *volser* has *nnn* RACF indicated dataset(s) without profile
This message summarizes the result of the VERIFY INDICATED command per volume.

CNR093I 00 *volser* has *nnn* discrete profile(s) for non-RACF indicated datasets
This message (with CNR094I and CNR095I) summarizes the result of the VERIFY ONVOLUME command per volume.

CNR094I 00 *volser* has *nnn* discrete profile(s) without dataset on the volume
This message (with CNR093I and CNR095I) summarizes the result of the VERIFY ONVOLUME command per volume.

CNR095I 00 *volser* has *nnn* discrete profile(s) but volume not mounted
This message (with CNR093I and CNR094I) summarizes the result of the VERIFY ONVOLUME command per volume.

CNR096I 00 *volser* has *nnn* inaccessible dataset(s) (RACF indicated, no profile)
This message (with CNR097I and CNR098I) summarizes the result of the VERIFY PROTECTALL command per volume.

CNR097I 00 *volser* **has** *nnn* **inaccessible dataset(s) (not indicated, no profile)**
This message (with CNR096I and CNR098I) summarizes the result of the VERIFY PROTECTALL command per volume in a PROTECTALL(FAIL) environment.

CNR098I 00 *volser* **has** *nnn* **unprotected dataset(s) (not indicated, no profile)**
This message (with CNR097I and CNR098I) summarizes the result of the VERIFY PROTECTALL command per volume in a NOPROTECTALL or PROTECTALL(WARN) environment.

CNR099I 00 *nnn* **messages suppressed for catalog** *catalog name*
This message summarizes the result of the SUPPRESS CAT= command.

CNR100I 12 Duplicate request for ID=*name*
More than one specific and incompatible request was made for one identity. Remove duplicates and use separate runs for conflicting requests.

CNR101I 12 Duplicate REPORT PERMIT/SCOPE=*id*
An identity occurred twice in the indicated commands. Remove duplicates.

CNR102I 12 The parameters OUTOFGROUP, NONDEFAULT and (NON)REDUNDANT are mutually exclusive
You must use separate runs for each of these REPORT options.

CNR103I 12 Field "*fldname***" to be processed not found in any profile**
The field you requested on the LIST command was not found in the templates for any type of entity. Verify the spelling (use the SPL: RACF manual or run the SHOW TEMPLATES command).

CNR104I 12 FIELD and FIELDVALUE must be specified both
Both the field to be used as selection criterion and the value for it must be specified.

CNR105I 12 Volume "*volser***" specified more than once**
The same volume was mentioned more than once for the same function. Possibly you used the repeat command of the editor and intended to change it to another volume.

CNR106I 12 Catalog "*catname***" specified more than once**
The same catalog was mentioned more than once for the same function. Possibly you used the repeat command of the editor and intended to change it to another name.

CNR107I 12 The parameters PROFILE, MASK/FILTER and QUAL are mutually exclusive
On one SELECT or EXCLUDE command only one selection option based on the profile key can be given.

CNR108I 12 Left margin cannot exceed right margin
In the MARGINS(*x*,*y*) command, *x* (the left margin) cannot exceed *y* (the right margin).

CNR109I 12 BY= must precede PAGEBY=
The PAGEBY value must be the first in the BY list and the BY list must be in front of the PAGEBY option.

CNR110I 12 PAGEBY and BY combination implies page per profile
The combination of BY and PAGEBY parameter as specified or implied would result in a new page for each profile. This is probably not what you meant.

CNR111I 12 DB=1 must be included because it is the master database.
The master database must always be included in the databases selected because it contains the RACF options to be used.

CNR112I 12 DB only supported in range 1..64

Selection by sequence number is only supported for sequence number 1 through 64. To use higher sequence numbers, you must pre-allocate SYSRAC*nn* files.

CNR113I 12 LIST must be followed by CLASS, KEY, DB, RBA, SEGMENT, or a field name defined in a template
The LIST command may not be specified without any operands (this would result in an empty line for each selected profile).

CNR114I 12 Value selection for field *field* not supported
The specified field has internally coded field values. This type is not supported. See "2.3.11 SELECT and EXCLUDE" for a list of supported types.

CNR115I 12 *option* only valid behind USER/PERMIT=
The option indicated is only valid behind COPY, MOVE or REMOVE options USER= or PERMIT=. Possibly you only need to change the order of the parameters.

CNR116I 12 *option* only valid behind USER/GROUP=
The option indicated is only valid behind COPY, MOVE or REMOVE options USER= or GROUP=. Possibly you only need to change the order of the parameters.

CNR117I 12 *option* only valid behind TOGROUP=
The option indicated is only valid behind COPY, MOVE or REMOVE option TOGROUP=. Possibly you only need to change the order of the parameters.

CNR118I 12 *option* only valid behind USER/GROUP/NOTIFY/PERMIT=
The option indicated is only valid behind COPY, MOVE or REMOVE options USER= or GROUP=. Possibly you only need to change the order of the parameters.

CNR119I 12 *option* only valid behind USER=
The option indicated is only valid behind COPY, MOVE or REMOVE options USER=. Possibly you only need to change the order of the parameters.

CNR120I 12 *option* not valid with COPY
The option indicated is only valid behind MOVE or REMOVE commands, not behind COPY.

CNR121I 12 Print options behind NEWLIST must be specified before the (SORT)LIST
In the scope of a NEWLIST command, the print and selection options must be specified before the LIST or SORTLIST command(s). Change the order of your commands, and run the job again.

CNR122I 12 Selection behind NEWLIST must be specified before the (SORT)LIST
In the scope of a NEWLIST command, the print and selection options must be specified before the LIST or SORTLIST command(s). Change the order of your commands, and run the job again.

CNR123I 12 Segment or entity name is not a field - *name*
A field name is expected, but a segment or entity name was specified instead. Possibly, you meant to list the key of a profile of a specific class and specified the entity name (e,g, USER). In this case, replace it with KEY and run the job again.

CNR124I 12 Invalid date - valid are YYDDD, TODAY, and DUMPDATE - "*value*"
A date is expected but the format is not recognized. CONSUL/RACF supports a julian date (without dot), and the two keywords TODAY and DUMPDATE.

CNR130I 16 OPEN failed for IOCONFIG
Refer to CNR002I and CNR010I for a discussion.

CNR131I 16 IOCONFIG file empty
Refer to CNR002I and CNR010I for a discussion.

CNR132I 00 Configuration for system *name* running MVS/SPv.*l.m* (*type*) with DFP *v.l.m*
Created by program *progname* job *jobname* *dd mmm yyyy hh'mm*
This message indicates when, where and how the IOCONFIG file was created.

CNR133I 12 VERIFY PERMIT and REMOVE PERMIT are mutually exclusive
The REMOVE USER/NOTIFY/PERMIT= and VERIFY PERMIT commands cannot both be specified (since both commands use the same method internally).

CNR140I 00 Number of profiles referring outside group is *number*
This message summarizes the number of profiles listed by a REPORT OUTOFGROUP command.

CNR141I 00 Number of non-default profiles found is *number*
This message summarizes the number of profiles listed by a REPORT NONDEFAULT command.

CNR142I 00 Of the *xxxx* profiles tested *yyyy* are redundant (*pp*%)
This message gives the number of profiles considered redundant by a REPORT NONREDUNDANT or REPORT REDUNDANT command. In addition, it compares this to the total number of profiles tested for redundancy.

CNR143I 00 Number of profiles in selected scope is *number*
This message summarizes the number of profiles listed by a REPORT SCOPE= or REPORT PERMIT= command.

CNR160I 16 Unsupported RACF database blksize *nnnnn* (must be 1024 or 4096)
The database to be read had an unsupported blocksize. This may happen if you transmit a database to another system and receive it there without explicitly requesting the proper blocksize; the system will select another blocksize in this case.

CNR161I 16 Segment name not in templates - *name* for entity type *xx*
A profile in a restructured database was read with a segment name that could not be found in the template for the indicated entity type. The message is followed by the exact source location of the profile to assist in further analysis.

CNR162I 16 Entity type not found in BASE segment of *key*
The entity type of the base segment of a profile in a restructured database was not found in the expected place in the profile. Please report this message if the profile can be displayed normally by RACF commands.

CNR163I 00 Entity type user assumed - segment *segname* of *key*
This message indicates that a non-base segment was encountered for which the entity type could not be determined. The message is issued only if DEBUG SEGMENT has been issued. For RACF 1.9, this can only occur for a DFP segment of a USER or GROUP profile. For most purposes, this does not really matter, since they are treated the same most of the time (i.e. as accessor ids). However, if you request a LIST with CLASS, then the class may erroneously show USER.

CNR164I 16 Segment name *segname* not in segment table for entity type *xx*
A profile segment in a restructured database was read with a segment name that could not be found in the segment table for the indicated entity type. The message is followed by the exact source location of the profile to assist in further analysis.

CNR165I 16 Template not found for entity type *xx*
A profile in a restructured database was read with the indicated entity type. The ICB did not contain a template pointer for the indicated entity type. The message is followed by the exact source location of the profile to assist in further analysis.

CNR166I 08 Conditional access list refers to unknown class *class*
A general resource profile in a restructured database contained a conditional access list containing a reference to a class not found in the class descriptor table.

CNR167I 16 Grouping resource in conditional access list not supported - *class key*
A general resource profile in a restructured database contained a conditional access list with a reference to a grouping class. CONSUL/RACF supports only non-grouping classes in the conditional access list.

CNR168I 00 Maximum profile length is *nnnnn* **bytes for** *class key*
This informational message details the maximum profile length found in your database. It can be used to determine how near you are to problems. For non-restructured databases, the maximum length is 64Kb.

CNR169I 08 Cluster protection cannot be determined (not found in any catalog or VVDS) - *clustername*
The indicated cluster cannot be represented properly in the reports, because the VVDS or catalog information is missing.

CNR170I 20 Duplicate vol/dsn combination in program profile *program - volser dsname*
The indicated combination of volume and datasetname was encountered more than once as a member on a program profile. This is not supported by CONSUL/RACF. The first one will be the one used.

CNR171I 16 Class not in descriptor table, default properties assumed - *class*
The indicated class (or its 4 character prefix in non-RDS databases) was present in the database, but not in the class descriptor table. Hence, CONSUL/RACF cannot know which properties the class has and may use it incorrectly. This may for instance happen if you process a RACF database from a different system, or if classes were deleted from the class descriptor table without first removing all profiles in these classes. The message is followed by an indication which profile was first encountered with the offending class. To find all profiles you can use the SELECT CLASS= command.

CNR172I 16 ICHCNX00 returns qualifier "*qual1***" for internal but "***qual2***" for external format of** *dsname*
The installation exit returns different qualifiers for the internal and external formats of the datasetname, both of which are unequal to the first qualifier of the datasetname. CONSUL/RACF will choose the external one. The message can be suppressed by the command SUPPRESS MSG=172.

CNR173I 00 ICHCNX00 returns qualifier "*qual1***" for internal but "***qual2***" for external format of** *dsname*
The installation exit returns different qualifiers for the internal and external formats of the datasetname. CONSUL/RACF will choose the external one. This message is issued only if the DEBUG QUAL command was issued.

CNR200I 20 Duplicate NONVSAM profile volume *volser* **dataset** *datasetname*
Two identical profile keys were found for the same volume. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR201I 20 Duplicate TAPEDSN profile volume *volser* **dataset** *datasetname*
Two identical profile keys were found for the same volume, and both with DSTYPE=TAPE. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR202I 20 Duplicate VSAM profile volume *volser* **cluster** *datasetname*

Two identical profile keys were found for the same volume and both with DSTYPE=VSAM. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR203I 20 Duplicate MODEL profile *datasetname*
Two identical profile keys were found, both for a model dataset. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR204I 20 Duplicate generic dataset profile *datasetname*
Two identical profile keys were found, both for a generic dataset profile. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR205I 20 *field* **not found in profile** *datasetname*
Here *field* can be DSTYPE or MODELNAM. While searching the dataset profile indicated for the specified field, end-of-profile was reached or the template did not contain the field. If the error is reproducible, please submit a problem report.

CNR206I 20 Duplicate GLOBAL profile *datasetname*
Two identical group profile members were found in the MEMLST (member list) of the profile DATASET in the class GLOBAL. This is an anomaly in the RACF database. Only the first profile will be used in CONSUL/RACF, and no support is present to remove the condition.

CNR207I 20 Model name too long on profile *identity*
The model profile name on a user or group profile contained more than 44 characters. CONSUL/RACF provides no support for this condition.

CNR208I 20 *field* **not found in** *type* **profile**
Here *type* can be "dataset" or "general", and *field* can be UNIVACS, UACC, FLAG1, AUDIT, AUDITQS, AUDITQF, GAUDITQS or GAUDITQF.

CNR209I 20 *identity* **defined as both USER and GROUP**
The indicated identity was found as a profile in the class USER as well as the class GROUP. No support exists to handle this condition.

CNR210I 20 USER *identity* **doubly defined**
Two user profiles were encountered with identical keys. Possibly you combined two copies of the same database in one run.

CNR211I 20 GROUP *identity* **doubly defined**
Two group profiles were encountered with identical keys. Possibly you combined two copies of the same database in one run.

CNR212I 20 Numeric or flag field *fldname* **exceeds supported length (4 byte) for** *key*
This message indicates that during SELECT or EXCLUDE processing a profile was encountered with the field length for the indicated field exceeding 4 bytes. CONSUL/RACF assumes that all numeric fields are 4 bytes or less in length.

CNR220I 20 Unsupported date length for *fieldname* **profile** *key*
This message is issued when trying to format a variable length date field with an unsupported length. The field name from the template is indicated in the message, as well as the profile key.

CNR235I 04 Replacing notify *identity* **on non-VSAM dataset profil** *volume datasetname* **-**
with *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an ALTDSD NOTIFY() command will be generated to change the notify field.

CNR236I 04 Replacing notify *identity* **on VSAM dataset profile** *volume datasetname* **- with** *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an ALTDSD NOTIFY() command will be generated to change the notify field.

CNR237I 04 Replacing notify *identity* **on generic dataset profile** *datasetname* **- with** *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an ALTDSD NOTIFY() command will be generated to change the notify field.

CNR238I 04 Replacing notify *identity* **on model dataset profile** *datasetname* **- with** *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an ALTDSD NOTIFY() command will be generated to change the notify field.

CNR239I 04 Changing notify *identity* **on program prof** *program* **with** *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an RALTER NOTIFY() command will be generated to change the notify field.

CNR240I 04 BCS RACF indicator set but no discrete VSAM profile *volser clustername*

This message is issued due to a VERIFY INDICATED command.

CNR241I 04 Discrete VSAM profile found but BCS RACF indicator not set *volser clustername*

This message is issued due to a VERIFY ONVOLUME command. The volume indicated is the volume of the catalog (BCS) that contained an ownership cell with the RACF indicator bit off for the indicated cluster. To solve the error condition a DELDSD NOSET command will be generated.

CNR242I 04 Discrete VSAM profile present but no cluster found *volser clustername*

This message is issued due to a VERIFY ONVOLUME command. The volume indicated is the volume of the catalog that did not contain the cluster. To solve the error condition a DELDSD NOSET command will be generated.

CNR243I 04 Discrete VSAM profile present but BCS volume not mounted *volser clustername*

This message is issued due to a VERIFY ONVOLUME command. The profile indicates a volume that is not mounted. To solve the error condition a DELDSD NOSET command will be generated.

CNR244I 04 Replacing notify *identity* **on tape dataset profile** *datasetname* **- with** *newnotify*

This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an ALTDSD NOTIFY() command will be generated to change the notify field.

CNR245I 04 *event* **qualif** *identity* **on tape dataset profile** *volser datasetname* **- delete profile**

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the dataset profile has a first qualifier that is undefined or to be removed. To solve the condition, a DELDSD VOL() command will be generated. The message and action may be suppressed by means of a SUPPRESS command.

CNR246I 04 *event* **qualif** *identity* **on non-VSAM dataset profil** *volser datasetname* **- delete profile**

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the dataset profile has a first qualifier that is undefined or to be removed. To solve the condition, a DELDSD VOL() command will be generated. The message and action may be suppressed by means of a SUPPRESS command.

CNR247I 04 *event* **qualif** *identity* **on VSAM dataset profile** *volser datasetname* **- delete profile**

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the dataset profile has a first qualifier that is undefined or to be removed. To solve the condition, a DELDSD VOL() command will be generated. The message and action may be suppressed by means of a SUPPRESS command.

CNR248I 04 *event* **qualif** *identity* **on generic dataset profile** *datasetname* **- delete profile**

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the dataset profile has a first qualifier that is undefined or to be removed. To solve the condition, a DELDSD command will be generated. The message and action may be suppressed by means of a SUPPRESS command.

CNR249I 04 *event* **qualif** *identity* **on model dataset profile** *datasetname* **- delete profile**

This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the dataset profile has a first qualifier that is undefined or to be removed. To solve the condition, a DELDSD command will be generated. The message and action may be suppressed by means of a SUPPRESS command.

CNR250I 04 Multivolume discrete profile but no RACF indicator

This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a ALTDSD DELVOL command will be generated.

CNR251I 04 Multivolume discrete profile but no dataset found

This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a ALTDSD DELVOL command will be generated.

CNR252I 04 Multivolume discrete profile but volume not mounted

This message is issued due to a VERIFY ONVOLUME command. To solve the error condition a ALTDSD DELVOL command will be generated.

CNR253I 08 Cluster found in VVDS points to undumped catalog on *volume clustername*

CONSUL/RACF was unable to determine whether the indicated cluster was RACF indicated or not, since the catalog in which it was cataloged according to the VVDS, was not present in the catalog dump.

CNR254I 04 Discrete profile not used because GDG model present
This message is issued due to a VERIFY ONVOLUME command. It indicates that a discrete profile is present for a GDG generation, while at the same time a discrete non-VSAM or model profile exists for the GDG base name, and GDG modelling is active. To solve the error condition a DELDSD NOSET command will be generated. However, if the generation has already been rolled off the GDG, the command may be rejected with the error message "NOT FOUND IN CATALOG". In this case, the profile can only be removed by deactivating the system-wide MODEL(GDG) option before issuing the command.

CNR255I 04 *event* **notify** *identity* **of non-VSAM dataset profil** *volser datasetname* **- make** *newowner*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. To solve the error condition an ALTDSD VOL() NONOTIFY command will be generated.

CNR256I 04 *event* **notify** *identity* **of VSAM dataset profile** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. To solve the error condition an ALTDSD VOL() NONOTIFY command will be generated.

CNR257I 04 *event* **notify** *identity* **of generic dataset profile** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. To solve the error condition an ALTDSD NONOTIFY command will be generated.

CNR258I 04 *event* **notify** *identity* **of model dataset profile** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. To solve the error condition an ALTDSD NONOTIFY command will be generated.

CNR259I 04 *event* **notify** *identity* **on program prof** *progname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a RALT NONOTIFY command will be generated.

CNR260I 04 *event* **member** *identity* **general resource profile** *class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a (RE)MOVE PERMIT/USER command. This message is only issued for the NODES resource class. To solve the condition an RDEL command will be generated to remove the entire profile.

CNR261I 04 Key with unkown *identity* **general resource profile** *class key*
This message is issued due to a VERIFY PERMIT or (RE)MOVE PERMIT/USER command. This message is only issued for resource classes where some qualifier can be a userid of group, like VMMDISK, VMBATCH, DLFDATA, JESJOBS, NODES, JESSPOOL, PROPCNTL, VMEVENT, and VMXEVENT. To solve the condition an RDEL command will be generated to remove the profile.

CNR263I 04 *event* **notify** *identity* **general resource profile** *class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a (RE)MOVE NOTIFY/PERMIT/USER command. To solve the condition a RALTER NONOTIFY command will be generated to remove the notify field.

CNR264I 04 *event* **R-ownr** *identity* **on non-VSAM dataset profil** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. It means that the identity to be removed was found in the RESOWNER field of the DFP segment. To solve the error condition an ALTDSD VOL() NODFP command will be generated.

CNR265I 04 *event* **R-ownr** *identity* **on VSAM dataset profile** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. It means that the identity to be removed was found in the RESOWNER field of the DFP segment. To solve the error condition an ALTDSD VOL() NODFP command will be generated.

CNR266I 04 *event* **R-ownr** *identity* **on generic dataset profile** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. It means that the identity to be removed was found in the RESOWNER field of the DFP segment. To solve the error condition an ALTDSD VOL() NODFP command will be generated.

CNR267I 04 *event* **R-ownr** *identity* **on model dataset profile** *datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. It means that the identity to be removed was found in the RESOWNER field of the DFP segment. To solve the error condition an ALTDSD VOL() NODFP command will be generated.

CNR268I 04 *event* **permit** *identity whenclass whenprofile(1-12) class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the identity was found in the conditional access list of a discrete general resource profile of a member resource class. Only the first 12 characters of the key in the conditional permit are shown. To solve the condition a PERMIT DELETE WHEN(...()) command will be generated.

CNR269I 04 *event* **permit** *identity whenclass whenprofile(1-12) class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the identity was found in the conditional access list of a discrete grouping class general resource profile. Only the first 12 characters of the key in the conditional permit are shown. To solve the condition a PERMIT DELETE WHEN(...()) command will be generated.

CNR270I 04 *event* **permit** *identity whenclass whenprofile(1-12) class key*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. It indicates that the identity was found in the conditional access list of a generic general resource profile. Only the first 12 characters of the key in the conditional permit are shown. To solve the condition a PERMIT GENERIC DELETE WHEN(...()) command will be generated.

CNR271I 04 *event* **permit** *identity* **in access list of tape dsn** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() command will be generated.

CNR272I 04 *event* **permit** *identity* **on tape condition** *progname* **-** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a PERMIT DELETE VOL() WHEN(PROGRAM()) command will be generated.

CNR273I 04 *event* **owner** *identity* **of tape dataset profile** *volser datasetname* **- make** *newowner*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE PERMIT/USER command. To solve the error condition a ALTDSD VOL() OWNER() command will be generated with the default owner selected with DEFAULT OWNER=, or, if defined, the first qualifier as owner id. The new owner selected is shown in the message.

CNR274I 04 *event* **notify** *identity* **of tape dataset profile** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. To solve the error condition a ALTDSD VOL() NONOTIFY command will be generated.

CNR275I 04 Inaccessible cluster (RACF indicated and no profile) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL or VERIFY INDICATED command. To solve the error condition an ADDSD NOSET command will be generated, but only if VERIFY INDICATED was *not* specified. Note that adding the profile may not be enough, you might want to enhance the access list, or use a generic profile instead.

CNR276I 08 Unprotected cluster (not RACF-indicated, no generic) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL command in NOPROTECTALL or PROTECTALL(WARN) environment. No command is generated.

CNR277I 04 *event* **R-ownr** *identity* **on tape dataset profile** *volser datasetname*
This message is issued with *event* equal to **Undefined** due to a VERIFY PERMIT command, and with *event* equal to **Removing** due to a REMOVE NOTIFY/PERMIT/USER command. It means that the identity to be removed was found in the RESOWNER field of the DFP segment. To solve the error condition an ALTDSD VOL() NODFP command will be generated.

CNR278I 00 Revoke for user *identity* **requested**
This message is issued due to a (RE)MOVE USER=, REVOKE command. In response, an ALTUSER REVOKE command will be generated.

CNR279I 00 Connecting user *identity* **to group** *identity* **as requested**
This message is issued due to a MOVE USER=, TOGROUP= command. In response, an CONNECT command will be generated.

CNR280I 00 Default group of *identity* **becomes** *identity* **because removing former default**
This message is issued due to a MOVE USER=, TOGROUP= command. In response, an ALTUSER DFLTGRP() command will be generated.

CNR281I 00 Removing user *identity* **from** *identity* **as requested**
This message is issued due to a MOVE USER=, TOGROUP= command. In response, an REMOVE command will be generated.

CNR282I 08 Inaccessible cluster (not indicated and no generic) *volser datasetname*
This message is issued due to a VERIFY PROTECTALL command in a PROTECTALL(FAIL) environment. No command is generated.

CNR283I 00 Deleting userid *identity* **group** *identity* **as requested**
This message is issued due to a REMOVE USER= command. In response, an DELUSER command will be generated.

CNR285I 04 Replacing notify *identity* **general resource profile** *class key* **with** *newnotify*
This message is issued due to a (RE)MOVE NOTIFY/PERMIT/USER, NEWNOTIFY= command. In response, an RALTER NOTIFY() command will be generated to change the notify field.

CNR900I 24 Internal error: nonzero RDJFCB return code
The RDJFCB SVC returned a nonzero return code for one of the SYSRAC*nn* files. Please submit an error report.

CNR901I 24 CNREPNDF: PERM$OWN PERMXREF not TRID
Please submit an error report.

CNR902I 24 CNREPNDF: PERMXREF invalid with NONDEFAULT
Please submit an error report.

CNR903I 24 CNROUGRP: PERMXREF invalid with OUTOFGROUP
Please submit an error report.

CNR904I 24 CNRPRTFL: internal error
Please submit an error report.

CNR905I 24 CNRPRRPT: PERMXREF points to PERM
Please submit an error report.

CNR906I 24 ADDPERM called with nil treeptr
Please submit an error report.

CNR907I 24 ADDNVD: Secondary volume finds empty treeptr
Please submit an error report.

CNR908I 24 ADDTPDA: Secondary volume finds empty treeptr
Please submit an error report.

CNR909I 24 ADDPERM called with nil treeptr
Please submit an error report.

CNR911I 24 Undefined ID *identity* **without PERM**
Please submit an error report.

CNR912I 24 Undefined ID *identity* **PERM w/o XREF**
Please submit an error report.

CNR913I 24 CNRVERIF: no PERMXREF handling for *type*
Please submit an error report.

CNR914I 24 Unknown error message *type* **for** *volser datasetname*
Please submit an error report.

CNR915I 24 TRID group not found for TRCO *user/group*
Please submit an error report.

CNR916I 24 CNRPRTL internal error: OUTC element invalid
Please submit an error report.

CNR917I 24 Unsupported comparand type *bbbb*

Please submit an error report.

CNR918I 24 Premature end-of-file on *ddname*
Please submit an error report.

CNR919I 24 Internal error: *type*__QUAL=0__ **for** *type*__NAME__
Please submit an error report.

CNR920I 24 CNRFLD internal error searchiong *field*
Please submit an error report.

CNR921I 24 CNRSTFL internal error
Please submit an error report.

CNR922I 24 ADDPROF called invalidly
Please submit an error report.

CNR923I 24 ADDPERM1 unsupported - *field*
Please submit an error report.

CNR924I 24 ADDPERM2 unsupported - *field*
Please submit an error report.

CNR925I 24 ADDCLU of *name* **returns** *cluster*
Please submit an error report.

CNR926I 24 ADDPMB with invalid treeptr
Please submit an error report.

CNR927I 24 TNVR for TPMB not TNVD or TGDA but *xxxx* **-** *dsname vol*
Please submit an error report.

CNR928I 24 TNVR not TNVD or TGDA but *xxxx* **-** *dsname vol*
Please submit an error report.

CNR929I 24 Not supported in this release - *function*
The function has not been activated.

CNR930I 24 CNRPRRPT missing PERMWHEN on *key*
Please submit an error report.

CNR931I 24 CNRPRRPT no PERMWHEN support for *type key*
Please submit an error report.

CNR932I 24 CNRPRRPT PERMWHEN expected *type1* **found** *type2*
Please submit an error report.

CNR981I 12 Invalid *type* **"***value***"**
This message indicates that the text *value* is not a valid value in the context *type*.

CNR982I 12 Internal error: unknown error code at *ddname* **line** *number*
The input parser error routine encountered an invalid error code. Please report this error to your service representative.

CNR983I 12 Expecting list separator or terminator instead of *type* **"***value***" at** *ddname* **line** *number*
This message indicates that the input parser expected a list separator or terminator for the current list (this can for instance be a comma, blank, or end-of-line, depending on the context). Instead, it encountered the indicated token type *type* (and text *value*, if available). The input parser skips all input until it encounters a valid list separator or terminator for the current list.

CNR984I 12 Invalid list element type *type* **"***value***" at** *ddname* **line** *number*

This message indicates that the input parser expected a list element, but found a token of a type not supported as a list element in this context. If available, the offending text *value* is also listed in the message. The input parser skips all input until it encounters a valid list separator or terminator for the current list.

CNR985I 12 Required list element/parameter "*value*" missing at *ddname* line *number*
This message indicates that the input parser detected a missing required parameter or element in the list at the indicated line.

CNR986I 12 Duplicate parameter *value* at *ddname* line *number*
This message indicates that the input parser detected a duplicate occurrence of the parameter or list element *value* at the indicated line.

CNR987I 12 Syntax error: *type1* expected instead of *type2* at "*value*" on *ddname* line *number*
This message indicates that the input parser expected a specific token type *type1* in the current context. Instead of this, it found the token type *type2* (at the text *value*, if available) on the indicated input line.

CNR988I 12 Syntax error: "*c*" expected instead of *type* at "*value*" on *ddname* line *number*
This message indicates that the input parser expected a specific character "*c*" (presumably a delimiter) in the current context. Instead of this, it found the token type *type* (at the text *value*, if available) on the indicated input line.

CNR989I 12 Unexpected *type* "*value*" at *ddname* line *number*
This message indicates that the input parser expected any one of a number of specific token types, but found a different token type instead. If available, the offending text *value* is also listed in the message.

CNR991I 04 ESTAE return code *rc*
This message indicates that the program failed to establish an abend exit linkage

CNR993I DIAGNOSTIC DUMP SUPPRESSED FOR ABEND *xxx*
This message indicates that the program abend exit did not attempt to make a diagnostic summary dump. This is done to prevent recursive abend conditions involving the print file.

CNR995I LRECL INVALID; NOT OVERRULED BECAUSE PARTITIONED
This message indicates that the print file open routine detected an invalid record length for the output file. This would have been overruled with a correct length for a Physical Sequential dataset, but this is not done for Partitioned datasets to prevent making any existing PDS members inaccessible. Subsequent 013 or 002 abends may be caused by the invalid record length.

CNR999I 16 GETMAIN FAILED FOR HEAP *name* - INCREASE REGION
This message indicates that the program needs more storage. If the heap name is **LOWHEAP**, then the request is for storage below the 16MB line.

A. Installation

The CONSUL/RACF package is supplied on a standard labelled cartridge tape suitable for either direct load or installation with SMP/E release 5 or higher. The volume serial is CR*vrmm*, for this level of the manual it is CR1103 (please check the separately supplied cover letter for the exact modification level and hence volume serial of your tape). The tape consists of three parts: the first part contains the datasets as required by SMP/E installation, next comes an unloaded library with sample installation JCL for both SMP and non-SMP installation, and the last part contains datasets required for non-SMP installation.

The object of the installation process is to make available to users of the package the datasets shown below with their default disk datasetnames:

- A sample library with sample JCL (inline procedures) and sample command input members (CRM.CNR113.CNRSAMP).
- A JCL procedure library for use as cataloged procedure or JCLLIB (CRM.CNR113.CNRPROC).
- An ISPF panel library (CRM.CNR113.CNRPLIB).
- The CONSUL/RACF load library, with the RACF database analysis program as well as the interactive component (CRM.CNR113.CNRLOAD).
- A CLIST library containing a sample CLIST to invoke the interactive component (CRM.CNR113.CNRCLIB).
- The CONSUL/COLLECT load library, with the VTOC, VVDS, ICF catalog, and PDS directory collector program (CRM.CNF203.CNFLOAD).

Besides these datasets that must be available to users, the installation process uses an installation JCL library, and optionally a number of SMP/E support datasets.

A.1. Installation preparation

1. Decide on a *naming scheme* for the datasets. The default (and recommended) naming scheme is to use 3 qualifiers in the datasetname (e.g. CRM.CNR113.CNRLOAD):

   CRM Prefix common to all CONSUL products and installation support datasets. You may replace this by one or more qualifiers.

   nnnvrm One qualifier containing product prefix *nnn* (e.g. CNR) and the version, release, and modification level (e.g. 113). Keep in mind that the CONSUL/RACF package contains the three product prefixes CNR (CONSUL/RACF), CNF (CONSUL/COLLECT), and HLL (run time system support - only if you use SMP installation), each with its own release number.

   dddef The last qualifier equal to the SMP/E DDDEF. Each DDDEF starts either with the product prefix (for target datasets), or with A followed by the product prefix (for distribution libraries). If you use non-SMP installation, you will only get target datasets.

   If you decide to deviate from this default, then you will have to tailor the JCL both before and after the installation, *and after each fix applied to a sample JCL member.* A sample for this tailoring process is present in the form of IPOUPDTE input and JCL. The tailoring process is designed so that it is possible to select the following alternative naming conventions:

   a. Change the prefix CRM to a different qualifier or more than one qualifier (e.g. SYS1.CNR113.CNRLOAD or SYS1.CRM.CNR113.CNRLOAD).

   b. Eliminate the common prefix CRM and use qualifiers V*v*R*r*M*m* to indicate the release (e.g. CNR.V1R1M3.CNRLOAD). This looks more like the way ISPF and PDF are shipped.

   c. Eliminate release dependency and optionally the product prefixes (e.g. CRM.CNRLOAD). Note that all DDDEF names contain the product prefix anyway. This looks more like the way MVS is generated.

   d. Insert a qualifier applying to the SMP/E installation process, relating the datasetnames to the SMP/E zones (e.g. CRM.CRZONE.CNRLOAD for DDDEF CNRLOAD in zone CRZONET and CRZONED). This will allow you to install a new release in a new zone without need to disturb existing allocations or SMP/E definitions. You can accomplish this by using a new zone name (change CRZONE to CRTEST). If you know beforehand that you want to do this for each new distribution tape, then you might rename CRZONE into an installation-tape dependent name (e.g. CR1103). It also possible to use two zones, CRTEST and CRPROD.

   If you are going to install with SMP/E, you may wish to read the discussion on zones in the SMP/E installation section below, before deciding on the datasetnames.

2. The second step in installation is to load the installation JCL library from tape. Reading the JCL dataset requires a job like the one shown in the next figure (the expiration date indicates that the tape is external to a tape management system). You can adjust your disk datasetname to the naming convention you have chosen.


```
// EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSUT1   DD DISP=(OLD,KEEP),DSN=userid.CONSUL.INSTLIB.CNTL,
//         UNIT=3480,VOL=(,RETAIN,SER=CRvrmm),
//         LABEL=(9,EXPDT=98000)
//SYSUT2   DD DISP=(NEW,CATLG),DSN=prefix.CNRvrm.CNRINST,
//         UNIT=SYSDA,SPACE=(TRK,(5,5,35))
```

Figure 45. *Sample job to read installation JCL library.*

3. Now the JCL customization parameters must be set in member CNRZUPD. If you have the IPOUPDTE program available, then it is recommended that you edit the member CNRZUPD containing predefined IPOUPDTE input statements for various tailoring tasks. These statements are in a member separate from the JCL to ensure that you make the same customizations before and after the installation process (JCL member CNRZUPDZ and CNRZUPDT, respectively). You should check at least datasetname prefix, esoteric unit name for disk dataset allocation, disk volume serial, and jobcard information.

   The datasetnames are determined by parameters CPREF and CPROJ. You must be aware that these parameters are used in the datasetname *without* intermediary dot. This is done to ensure that you can delete one of the qualifiers. This means that you may have to end CPREF with a dot to create a real qualifier. The product is shipped with CPREF set to 'CRM.'.

   Note that the member's comment lines contain detailed information on how to change the common prefix to a different value.


```
</***** CUSTOMIZATION PARAMETERS ***************************
</*****
</***** These parameters must be applied twice if installing with SMP:
</*****   - before installation: job CNRZUPDZ
</*****   - after  installation: job CNRZUPDJ
</***** YOU SHOULD NEVER CHANGE ANYTHING ON LINES MARKED .NU..RU.
</*****
</***** NOTICE:
</*****    (1)  If you use SMP/E installation, you should be aware
</*****         that changing any of the parameters below at any time
</*****         will force you to make the same change for each
</*****         future update (in SMP format) to a JCL sample member
</*****         or procedure. The exceptions to this rule are SMP/E
</*****         zone-related dataset names (GPREF/ZPREF/ZPROJ etc.)
</*****         The recommended SMP scenario, however, does change
</*****         the names (see last sample scenario below).
</*****    (2)  The xPREF parameters and xPROJ parameters are used
</*****         in the JCL 'without' dot in between.  The default
</*****         datasetname CRM.CNR113.<dddef> is constructed  .NU..RU.
</*****         from CPREF='CRM.' and CPROJ=CNR113 to be able  .NU..RU.
</*****         to use simple 'change all' algorithms (like this job)
</*****         to construct new datasetnames without losing
</*****         reversability. Note that with CPREF=CRM        .NU..RU.
</*****         it would not have been possible to change to the
</*****         ISPF IPO style names CNR.... and CNF....       .NU..RU.
</*****         (they would only be changable to '.CNR...')
</*****
</*****  Sample scenarios:
</*****    (1)   No release dependency: DSN=CRM.<dddef>        .NU..RU.
</*****          Change CNR113, CNF203, and HLL120 to empty.   .NU..RU.
</*****          You can install a new release in the same zone.
</*****          Additional measures are needed for simultaneous test
</*****          and production (e.g. test system or copy).
</*****    (2)   CBIPO style (ISPF): DSN=<pfx>.VxRxMx.<dddef>  .NU..RU.
</*****          Change CNR113 to CNR.V1R1M3, CNF203 to        .NU..RU.
</*****          CNF.V2R0M3, and HLL120 to HLL.V1R2M0;         .NU..RU.
</*****          Change CRM. to empty.                         .NU..RU.
</*****          You would install a new release in a new zone, but
</*****          you risk problems if one of the releases
</*****          remains the same and you use the same catalog.
</*****          Typically this approach is valid
</*****          if the new release is on a different pack and the
</*****          datasets are cataloged in the master catalog.
</*****    (3)   Consul default:  CRM.<product>.<dddef>        .NU..RU.
</*****          where product is CNR113, CNF203, or HLL120.   .NU..RU.
</*****          You would install a new release in a new zone.
</*****          You risk problems on installing the next distribution
</*****          tape if one of the releases remains the same.
</*****          This is the way the product is shipped, do not need
</*****          to change any datasetname.
</*****    (4)   Fail safe:  CRM.<inst>.<product>.<dddef>      .NU..RU.
</*****          where <installation> reflects the installation attempt
</*****          at the SMP/E target/distribution zone pair level. Thus,
</*****          it might be PROD, CNR113, CR9108, M9108.      .NU..RU.
</*****          You might install a new release or a different product
</*****          in a new zone, you do not need to.
</*****          Change CRM. into CRM.<inst>. globally         .NU..RU.
</*****          (mind the trailing dot!).
</*****          Change CRZONE into <inst> globally.           .NU..RU.
</*****    (5)   SMP mixture:  CRM.<installation>.<dddef>      .NU..RU.
</*****          Probably the least-effort method in the long term.
</*****          The <installation> simply gives a new start (much
</*****          like a CBIPO). The common prefix qualifier minimizes
</*****          RACF group/alias work to be done on next releases.
</*****          Depending on whether you intend to install other
</*****          packages of CRM into the same zone, you might name it
</*****          CR9108 or CRTEST/CRPROD, with the SMP zone    .NU..RU.
</*****          names the same but suffixed with T or D.
</*****          Change CRZONE into <inst> globally.           .NU..RU.
</*****          Change CRM. into CRM.<inst> (no trailing dot) .NU..RU.
</*****          Change CNR113, HLL120, and CNF203 into empty. .NU..RU.
</*****
</*****  NOTE:
</*****     *    If you want to change the datasetname prefix globally
</*****          you can use PDF edit commands in this member:
</*****              EXCLUDE '.NU.' ALL                        .NU..RU.
</*****              CHANGE  'CRM.' 'prefix.' ALL NX           .NU..RU.
</*****************************************************************
```

Figure 46. *Comment on IPOUPDTE customization*

```
</* ---- JOBCARD 1 ------------------------------------------------
//*JOB1<+       to be replaced by (prefix blank, suffix <<):   .NU..RU.
 //JOBNAME  JOB ACCT,CONSUL,MSGCLASS=A,TIME=1,USER=<<
</* ---- JOBCARD 2 ------------------------------------------------
//*JOB2<+       to be replaced by (prefix blank, suffix <<):   .NU..RU.
 //*JOB2<<
</* ---- JOBCARD 3 ------------------------------------------------
//*JOB3<+       to be replaced by (prefix blank, suffix <<):   .NU..RU.
 //*JOB3<<
</* ---- JOBCARD 4 ------------------------------------------------
//*JOB4<+       to be replaced by (prefix blank, suffix <<):   .NU..RU.
 //*JOB4<<
</*****************************************************************
</* ---- Generic unit name work, installation & test datasets ---------
SYSDA<+       to be replaced by (prefix blank, suffix <<):   .NU..RU.
 SYSDA<<
</* ---- Volume serial for SMP VSAM datasets --------------------------
SMS001<+      to be replaced by (prefix blank, suffix <<):   .NU..RU.
 SMS001<<
</*----- DPREF ---- Two qualifier(s) test datasets --------------------
CRM.TEST<+                                                   .NU..RU.
 CRM.TEST<<                      <==
</*----- INSTLIB in installation JCL ----------------------------------
CRM.CNR113<+                                                 .NU..RU.
 CRM.CNR113<+                    <==
  INSTLIB=<
</*****************************************************************
</*----- CPREF ---- First qualifier(s) CONSUL/RACF (JCL) (trailing dot)
CRM.<+                                                       .NU..RU.
 CRM.<+                          <==
  CPREF=<
</*----- CPREF ---- First qualifier(s) CONSUL/RACF (CLIST) -----------
CRM.<+                                                       .NU..RU.
 CRM.<+                          <==
  CPREF(<
</*----- CPROJ --- Last-but-one qualifier CONSUL/RACF (JCL) ----------
CNR113<+                                                     .NU..RU.
 CNR113<+                       <==
  CPROJ=<
</*----- CPROJ --- Last-but-one qualifier CONSUL/RACF (CLIST) --------
CNR113<+                                                     .NU..RU.
 CNR113<+                       <==
  CPROJ(<
</*----- CPREF/CPROJ Combinations in DDDEF ----------------------------
CRM.CNR113<+                                                 .NU..RU.
 CRM.CNR113<+                    <==
  DA(<
</*----- CPREF/CPROJ Combinations in deletes --------------------------
CRM.CNR113<+                                                 .NU..RU.
 CRM.CNR113<+                    <==
  DELETE<
</*****************************************************************
</*----- IPREF ---- First qualifier(s) CONSUL/COLLECT if trailing dot-
CRM.<+                                                       .NU..RU.
 CRM.<+                          <==
  IPREF=<
</*----- IPROJ ----Last-but-one qualifier CONSUL/COLLECT (JCL) -------
CNF203<+                                                     .NU..RU.
 CNF203<+                       <==
  IPROJ=<
</*----- IPREF/IPROJ Combinations in DDDEF ----------------------------
CRM.CNF203<+                                                 .NU..RU.
 CRM.CNF203<+                    <==
  DA(<
</*----- IPREF/IPROJ Combinations in deletes --------------------------
CRM.CNF203<+                                                 .NU..RU.
 CRM.CNF203<+                    <==
  DELETE<
</*****************************************************************
</*****  Remainder is for SMP/E installation only *******************
</*****************************************************************
</*----- HPREF ---- First qualifier(s) CONSUL/HLL if trailing dot -
CRM.<+                                                       .NU..RU.
 CRM.<+                          <==
  HPREF=<
</*----- HPROJ ----Last-but-one qualifier CONSUL/HLL (JCL)------------
HLL120<+                                                     .NU..RU.
 HLL120<+                       <==
  HPROJ=<
</*----- HPREF/HPROJ Combinations in DDDEF ----------------------------
CRM.HLL120<+                                                 .NU..RU.
 CRM.HLL120<+                    <==
  DA(<
</*----- HPREF/HPROJ Combinations in deletes --------------------------
CRM.HLL120<+                                                 .NU..RU.
 CRM.HLL120<+                    <==
  DELETE<
</*****************************************************************
</*----- GPREF ---  First qualifier(s) global SMP if trailing dot -
CRM.<+                                                       .NU..RU.
 CRM.<+                       <==
  GPREF=<
</*----- GLOBAL --- Two qualifier(s) GLOBAL SMP datasets -------------
CRM.GLOBAL<+                                                 .NU..RU.
 CRM.GLOBAL<<                     <==
</*****************************************************************
</*----- ZPREF ---- First qualifier(s) T/D zone SMP if trailing dot -
CRM.<+                                                       .NU..RU.
 CRM.<+                       <==
  ZPREF=<
</*----- ZPROJ ----Last-but-one qualifier T/D zone SMP datasets ------
CRZONE<+                                                     .NU..RU.
 CRZONE<+                        <==
  ZPROJ=<
</*----- ZPREF/ZPROJ qualifier(s) SMP T/D zone datasets ----------
CRM.CRZONE<+                                                 .NU..RU.
 CRM.CRZONE<<                     <==
</*----- CRZONET -- Target zone SMP -----------------------------------
CRZONET<+                                                    .NU..RU.
 CRZONET<<                      <==
</*----- CRZONED -- Distribution zone SMP ----------------------------
CRZONED<+                                                    .NU..RU.
 CRZONED<<                       <==
```

Figure 47. *Sample member CNRZUPD containing input to IPOUPDTE in order to customize installation JCL library.*

4. Now you must run member CNRZUPDZ after including appropriate job information, checking the datasetname of the installation JCL library, and checking the library where IPOUPDTE resides (note that the STEPLIB card is shipped as a comment). Check the output for correctness of your customization.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRZUPDZ PROC REGSIZE=2048K, Region for IPOUPDTE
//  INSTLIB='CRM.CNR113.CNRINST',          Installation library
//  UPDTE=CHECK  UPDATE        Choose CHECK or UPDATE
//*
//*********************************************************************
//* Name:    CNRZUPDZ  Level: SCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: Customize JCL before installation with SMP/E
//* Usage:   1. Customize member CNRZUPD before running this job
//*          3. Run with UPDTE=CHECK, check replacements
//*          4. Run with UPDTE=UPDATE
//*********************************************************************
//*
//IPOUPDTE EXEC PGM=IPOUPDTE,REGION=&REGSIZE,
//            PARM=&UPDTE
//*STEPLIB DD DISP=SHR,DSN=IPO1.LINKLIB
//SYSPRINT DD SYSOUT=*
//@CNRINST DD DISP=SHR,DSN=&INSTLIB
//SYSIN    DD DISP=SHR,DSN=&INSTLIB.(CNRZUPD)
//         PEND
//*
//CNRZUPDZ EXEC CNRZUPDZ
```

Figure 48. *Sample job CNRZUPDZ to customize all members in the installation JCL library.*

5. In member CNRZUPDZ, change UPDTE=CHECK to UPDTE=UPDATE and run the job again. This will result in all installation JCL members being customized.

6. Decide whether you will use SMP/E installation or load the package without SMP. Using SMP has advantages for change management and will ease future maintenance, while using the non-SMP installation costs less effort to get the software loaded the first time. Depending on your decision, select one of the subsequent installation sections in this manual.

A.2. Non-SMP/E installation

You should have completed the installation preparation described in the previous section before continuing here.

1. Update the authorization code in member CNRZLOAD with the 8 byte zap that is supplied in the cover letter by your software vendor.

2. To prevent conflict with security procedures in your installation that would prevent you from loading datasets starting with *CONSUL*, you can set the parameter TPREF in member CNRZLOAD to your userid, causing the datasetnames to be prefixed with your userid. RACF and ACF2 will generally allow you to read datasets on tape starting with your userid, and only the last 17 characters of the datasetname are physically present on tape.

3. Load the target datasets, using the JCL in member CNRZLOAD. The dataset names and characteristics of the datasets are shown in the figure below.

Dataset	Label	Record length	Block size	record format	size	directory blocks
CONSUL.CNR113.CNRSAMP	10	80	23440	FB	5 TRK	35
CONSUL.CNR113.CNRPLIB	11	80	23440	FB	5 TRK	35
CONSUL.CNR113.CNRLOAD	12	0	19069	U	1 CYL	35
CONSUL.CNR113.CNRCLIB	13	80	23440	FB	1 TRK	10
CONSUL.CNR113.CNRPROC	14	80	23440	FB	1 TRK	10
CONSUL.CNR113.CNFLOAD	15	0	19069	U	1 CYL	35

Figure 49. *Names and characteristics for the datasets on the distribution tape for non-SMP installation only.*


```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//CNRZLOAD PROC REGSIZE=4096K,  Available region
//  CPREF='CRM.',               Prefix. for disk libraries (note 3)
//  CPROJ=CNR113,               Project for disk libraries (note 4)
//  IPREF='CRM.',               Prefix. for disk libraries (note 3)
//  IPROJ=CNF203,               Project for disk libraries (note 4)
//  DVOLSER=,                   Disk volume for libraries (note 5)
//  DUNIT=SYSALLDA,             Unit name for libraries (note 5)
//  DSTAT='NEW,CATLG',          Status of disk libraries
//  TPREF='CRM',                Prefix for tape datasets (see note 1)
//  TEXPDT=98000,               Expiry date indicating external tape
//  TUNIT=3480,                 UNIT name for medium (note 2)
//  TVOLSER=CR1103,             Volume serial (name) for SL tape
//  TPROJ=CNR113                Release qualifier contained on tape
//*
//*****************************************************************
//* Name:    CNRZLOAD                      Version: CONSUL/RACF 1.1.3
//* Purpose: Load CONSUL/RACF from distribution tape to disk
//*          ===> non-SMP installation only
//*          ===> for SMP installation refer to CNRZSMP
//* Note:    1. You may want to change TPREF to 'userid' to be able
//*             to use it on systems with tape dataset protection.
//*             It is not present on tape (i.e. not in last 17 chars)
//*          2. You may have to change TUNIT to an esoteric unit name
//*             giving access to the tape or cartridge type delivered
//*     .C   3. Change CPREF and IPREF to the high level qualifier(s)
//*     .N      that you want to use for CONSUL/RACF and CONSUL/COLLECT,
//*     .R      reapectively.
//*     .Z      Note: step 3..5 may be done by running CNRZUPDZ
//*     .U   4. You may want to change CPROJ and IPROJ if you do not
//*     .P      want VRM in name to avoid conflict with older release
//*     .D   5. Change DUNIT and optionally DVOLSER to indicate
//*     .Z      where the libraries should be loaded.
//*          6. Change the expiration date zap at the end of this member
//*             to the value supplied by your vendor
//*          7. Run the job, afterwards you may want to run CNRZUPDT
//*             to update values in sample JCL and procedures.
//*****************************************************************
//*
//CNRSAMP EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(10,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNRSAMP
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&CPREF.&CPROJ..CNRSAMP,
//             SPACE=(TRK,(5,5,35),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//CNRPLIB EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(11,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNRPLIB
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&CPREF.&CPROJ..CNRPLIB,
//             SPACE=(TRK,(5,5,35),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//CNRLOAD EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(12,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNRLOAD
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&CPREF.&CPROJ..CNRLOAD,
//             SPACE=(CYL,(1,1,35),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//CNRCLIB  EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(13,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNRCLIB
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&CPREF.&CPROJ..CNRCLIB,
//             SPACE=(TRK,(1,1,10),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//CNRPROC  EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(14,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNRPROC
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&CPREF.&CPROJ..CNRPROC,
//             SPACE=(TRK,(1,1,10),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//CNFLOAD EXEC PGM=IEBCOPY,PARM='SIZE=999K',REGION=&REGSIZE
//SYSPRINT  DD SYSOUT=*
//SYSIN     DD DUMMY
//SYSUT1    DD DISP=(OLD,PASS),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//             LABEL=(15,SL,EXPDT=&TEXPDT),
//             DSN=&TPREF..CONSUL.&TPROJ..CNFLOAD
//SYSUT2    DD DISP=(&DSTAT),
//             DSN=&IPREF.&IPROJ..CNFLOAD,
//             SPACE=(CYL,(1,1,35),RLSE),UNIT=&DUNIT,VOL=SER=&DVOLSER
//*
//ZAP     EXEC PGM=AMASPZAP
//SYSLIB   DD DISP=SHR,DSN=*.CNRLOAD.SYSUT2
//SYSPRINT DD SYSOUT=*
//         PEND
//*
//CNRZLOAD EXEC CNRZLOAD
//ZAP.SYSIN   DD *
 NAME CNRACF EXPIRES                          8 BYTE CSECT FOR AUTHCODE
 REP     00    01020304,05060708              REPLACE BY YOUR AUTH CODE
```

Figure 50. *Sample job CNRZLOAD with non-SMP installation JCL.*

4. You can now proceed to "A.4. Installation completion".

A.3. SMP/E installation

You should have completed the installation preparation described in a previous section before continuing here.

1. First you will have to think about zones. The CONSUL/RACF installation JCL supports the following, alternative, scenarios ('installation process offerings'), ordered by installation JCL applicability:

   a. Automatically create a new global zone as well as a new target and distribution zone in it. This will work with a minimum customization effort. Start SMP/E-specific installation with member CNRZONE0.

   b. Use your site's global zone and automatically create a new target and distribution zone in it. You will have to define an OPTIONS member CRMOPTS yourself or modify the installation JCL to define another default for the new zones. Start SMP/E-specific installation with member CNRZONE1.

      You should check for conflicts in SYSMOD naming convention that may exist with non-CONSUL non-IBM software in the GLOBAL zone before attempting to receive software into a shared global zone. The SYSMOD naming convention used by CONSUL is in the not-reserved-by-IBM range and uses the format illustrated in the figure below.

R*ppvrmm*	Base function product *pp* version *v* release *r* modification level *mm*.
S*ppvrmm*	Dependent function to make product *pp* version *v* release *r* modification level *mm*.
P*pymmnn*	PTF for product line *p*, with a number that loosely reflects a date missing the decennia. The day may be simply a number. The problem number may have been the same with P replaced by Q.
Q*pymmnn*	APAR fix for product line *p*, with a number that loosely reflects a date missing the decennia. The day may be simply a number. This is not an IBM APAR number, but an SMP/E APAR. These are used for changes to the software that are not recommended generally, but may enable one function at the cost of other functions, usually at some customer's request.

Figure 51. *SYSMOD naming convention for CONSUL products.*

      You still have to decide how many zones you will use when a future release or another CONSUL product arrives. A new release replaces the old release if installed in the same zone. Another CONSUL product may have common components at a different release level (e.g. CONSUL/COLLECT !), thus affecting the CONSUL/RACF installation. Depending on how you intend to use SMP/E zones, you can customize the zone name prefix (e.g. CR1103 if release-dependent; CRZONE if you install everything in the same zone, CRTEST and CRPROD if you use a test- and a production zone).

   c. Use an existing global, target, and distribution zone. We advise against sharing a target or distribution zone with other than CONSUL products. While CONSUL/RACF uses a limited number of module prefixes (currently CNR, CNF, and FMT), this may not be so for all CONSUL products you would like to install in the future. In addition, the prefixes are

inside the 'reserved for IBM' range for historic reasons (CONSUL starts with a 'C'). However, if your installation's policy prescribes one zone for all packages, then it *is* possible. Start SMP/E-specific installation with member CNRZONE2.

2. You have the option to run the installation procedures (as customized by CNRZUPDZ) either one at a time, or concatenated into one job. If you choose the latter option, you must use the member CNRZSMP, and delete the steps CNRZONE0 or CNRZONE1 if you do not need them (as decided in step 1 above). The remainder of this section lists the members and discusses their function separately; if you run CNRZSMP they will all be run in one job (note: if your CPU is slower than a 3081 you will have to increase the TIME parameter on the job card of the CNRZSMP run).

3. Member CNRZONE0 allocates and initializes a new GLOBAL zone with its required datasets (like the SMPPTS). It adds the DDDEFs that are needed, defines the OPTION member CRMOPTS and a number of UTILITY entries, and defines global options like the SMPTLIB prefix.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*********************************************************************
//* Member:  CNRZONE0      Level: RCR1103    Version: CONSUL/RACF 1.1.3
//* Purpose: to allocate a global zone for use by CONSUL products
//*          This step is optional, you may use your own global CSI.
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure (see also 2. and 3.)
//*            2. Disk volser for SMPTLIB and for CSI
//*               e.g. CHANGE SMS001 volser ALL
//*            3. The statements in the SMP/E input marked with <======
//*               e.g. CHANGE 'CRM.' 'commonprefix.' ALL
//*               (Mind potential trailing dots)
//*********************************************************************
//*
//CNRZONE0 PROC REGSIZE=4096K, Region
//  GPREF='CRM.',              Prefix. for common CONSUL libraries
//  DUNIT=SYSDA,               Required unit name for libraries
//  DVOL=,                     Optional volume for libraries
//  DBLK80=23440,              Blksize for FB 80 (23440/27920)
//  DBLKV=23476                Blksize for VB (23476/27998)
//*
//ZONE0CSI EXEC PGM=IDCAMS,REGION=&REGSIZE,COND=(0,LT)
//SYSPRINT DD SYSOUT=*
//ZPOOL    DD DISP=SHR,DSN=SYS1.MACLIB(GIMZPOOL)
//*
//ZONE0DA EXEC PGM=IEFBR14,COND=(0,LT)
//SMPLOG   DD DISP=(NEW,CATLG),DSN=&GPREF.GLOBAL.SMPLOG,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(5,5)),
//            DCB=(RECFM=VB,LRECL=264,BLKSIZE=&DBLKV)
//SMPPTS   DD DISP=(NEW,CATLG),DSN=&GPREF.GLOBAL.SMPPTS,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(15,15,45)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//*
//ZONE0SMP EXEC PGM=GIMSMP,COND=(0,LT),REGION=&REGSIZE
//SMPCSI   DD DSN=&GPREF.GLOBAL.CSI,DISP=SHR
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//SMPLOG   DD DISP=MOD,DSN=&GPREF.GLOBAL.SMPLOG
//SMPPTS   DD DISP=SHR,DSN=&GPREF.GLOBAL.SMPPTS
//         PEND
//*
//CNRZONE0 EXEC CNRZONE0
//ZONE0CSI.SYSIN  DD *
 DEFINE CLUSTER(NAME(CRM.GLOBAL.CSI)          -
                VOLUME(SMS001)                -
                TRK(5 5)                      -
                RECSZ(24 143) KEYS(24 0) FSPC(10,5) SHR(2) UNIQUE)
 REPRO INFILE(ZPOOL) ODS(CRM.GLOBAL.CSI)
//ZONE0SMP.SMPCNTL DD *
SET BDY(GLOBAL).
UCLIN.
 ADD GZONE SREL(Z038) OPTIONS(CRMOPTS)
     ZDESC(GLOBAL ZONE FOR PRODUCTS OF CONSUL RISK MANAGEMENT B.V.).

 ADD DDDEF(SMPPTS)    SHR DA(CRM.GLOBAL.SMPPTS).
 ADD DDDEF(SMPLOG)    MOD DA(CRM.GLOBAL.SMPLOG).
 ADD DDDEF(SYSUT1)    NEW DELETE CYL SPACE(1,1)
                          UNIT(SYSDA).
 ADD DDDEF(SYSUT2)    NEW DELETE CYL SPACE(1,1)
                          UNIT(SYSDA).
 ADD DDDEF(SYSUT3)    NEW DELETE CYL SPACE(1,1)
                          UNIT(SYSDA).
 ADD DDDEF(SYSUT4)    NEW DELETE TRK SPACE(1,1)
                          UNIT(SYSDA).
 ADD DDDEF(SMPTLIB)   UNIT(SYSDA)
                      VOLUME(SMS001).
 ADD DDDEF(SMPOUT)    SYSOUT(*).
 ADD DDDEF(SMPRPT)    SYSOUT(*).
 ADD DDDEF(SMPLIST)   SYSOUT(*).
 ADD DDDEF(SYSPRINT)  SYSOUT(*).

 ADD UTILITY(IEV90)     RC(4) PRINT(ASMPRINT)
     PARM(XREF(SHORT),NOOBJECT,DECK,ESD,NORLD,NORENT,ALIGN,
          LINECOUNT(60))                              /* <====PL==== */.
 ADD UTILITY(HEWL)      RC(8)
     PARM(NCAL,LIST,LET,XREF,SIZE=(2048K,128K)).
 ADD UTILITY(IEBCOPY)   RC(4).
 ADD UTILITY(AMASPZAP)  RC(4).

 ADD OPTIONS(CRMOPTS) ASM(IEV90) LKED(HEWL) ZAP(AMASPZAP)
     DSSPACE(15,15,45) NOPURGE NOREJECT NUCID(8) RETRYDDN(ALL)
     PAGELEN(60)      /* file SMPLIST/SMPRPT */     /* <=====PL==== */
     DSPREFIX(CRM.GLOBAL.SMPTLIB).
ENDUCL.
```

Figure 52. *Sample job CNRZONE0 to create global zone for CONSUL products*

4. Member CNRZONE1 allocates and initializes new target and distribution zones, and defines them in the global zone. The default names are CRZONET and CRZONED, they may be changed by CNRZUPDZ.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*****************************************************************
//* Member:  CNRZONE1     Level: RCR1103     Version: CONSUL/RACF 1.1.3
//* Purpose: to allocate new target/distribution zones for installing
//*          a new release of CONSUL/RACF without deleting old releases
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*            2. The statements in the SMP/E input marked with <==
//*            3. Datasetnames
//*                 e.g. CHANGE 'CRM.' 'commonprefix.' ALL
//*                 (mind potential trailing dots)
//*****************************************************************
//*
//CNRZONE1 PROC REGSIZE=4096K,          Region
//  ZPREF='CRM.',                 Prefix. for libraries of new zone
//  ZPROJ=CRZONE,                 Project for libraries of new zone
//  GLOBAL='CRM.GLOBAL.CSI',      Dsname for global CSI
//  DUNIT=SYSDA,                  Required unit name for libraries
//  DVOL=,                        Optional volume for libraries
//  DBLK80=23440,                 Blksize for FB 80 (23440/27920/0)
//  DBLKV=23476                   Blksize for VB (23476/27998/0)
//*
//ZONE1CSI EXEC PGM=IDCAMS,REGION=&REGSIZE,COND=(0,LT)
//SYSPRINT DD SYSOUT=*
//ZPOOL    DD DISP=SHR,DSN=SYS1.MACLIB(GIMZPOOL)
//*
//ZONE1DA EXEC PGM=IEFBR14,COND=(0,LT)
//SMPLOG   DD DISP=(NEW,CATLG),DSN=&ZPREF.&ZPROJ..SMPLOG,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(5,5)),
//            DCB=(RECFM=VB,LRECL=264,BLKSIZE=&DBLKV)
//SMPMTS   DD DISP=(NEW,CATLG),DSN=&ZPREF.&ZPROJ..SMPMTS,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,20)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//SMPSTS   DD DISP=(NEW,CATLG),DSN=&ZPREF.&ZPROJ..SMPSTS,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,20)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//SMPSCDS  DD DISP=(NEW,CATLG),DSN=&ZPREF.&ZPROJ..SMPSCDS,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,20)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//*
//ZONE1SMP EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
// PARM='CSI=&GLOBAL'
//SMPLOG   DD DISP=SHR,DSN=&ZPREF.&ZPROJ..SMPLOG
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//         PEND
//*
//CNRZONE1 EXEC CNRZONE1
//ZONE1CSI.SYSIN  DD *
 DEFINE CLUSTER(NAME(CRM.CRZONE.CSI)                    /* <========= */ -
                VOLUME(SMS001)                -
                TRK(5 5)                      -
                RECSZ(24 143) KEYS(24 0) FSPC(10,5) SHR(2) UNIQUE)
 REPRO INFILE(ZPOOL) ODS(CRM.CRZONE.CSI)                /* <========= */
//ZONE1SMP.SMPCNTL DD *
```

Figure 53. *Sample job CNRZONE1 to create target and distribution zone for CONSUL product(s) (JCL)*


```
SET BDY(GLOBAL).
UCLIN.
 ADD GZONE ZONEINDEX(
           (CRZONET,
            CRM.CRZONE.CSI,TARGET),                    /* <========== */
           (CRZONED,
            CRM.CRZONE.CSI,DLIB))                      /* <========== */.
ENDUCL.
SET BDY(CRZONET).
UCLIN.
 ADD TZONE(CRZONET) RELATED(CRZONED) SREL(Z038)
     ZDESC(TARGET ZONE FOR CONSUL PRODUCTS)
     OPTIONS(CRMOPTS)                                  /* <========== */.
 ADD DDDEF(SMPPTS)   SHR DA(CRM.GLOBAL.SMPPTS).
 ADD DDDEF(SMPLOG)   MOD DA(CRM.CRZONE.SMPLOG).
 ADD DDDEF(SMPMTS)   SHR DA(CRM.CRZONE.SMPMTS).
 ADD DDDEF(SMPSTS)   SHR DA(CRM.CRZONE.SMPSTS).
 ADD DDDEF(SMPSCDS)  SHR DA(CRM.CRZONE.SMPSCDS).
 ADD DDDEF(SYSLIB)   CONCAT(SMPMTS).
 ADD DDDEF(SYSUT1)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT2)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT3)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT4)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK1)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK2)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK3)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK4)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK5)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK6)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPOUT)     SYSOUT(*).
 ADD DDDEF(SMPRPT)     SYSOUT(*).
 ADD DDDEF(SMPLIST)    SYSOUT(*).
 ADD DDDEF(SYSPRINT)   SYSOUT(*).
ENDUCL.
SET BDY(CRZONED).
UCLIN.
 ADD DZONE(CRZONED) RELATED(CRZONET) SREL(Z038)
     ZDESC(DISTRIBUTION ZONE FOR CONSUL PRODUCTS)
     OPTIONS(CRMOPTS)                                  /* <========== */.
 ADD DDDEF(SMPPTS)   SHR DA(CRM.GLOBAL.SMPPTS).
 ADD DDDEF(SMPLOG)   MOD DA(CRM.CRZONE.SMPLOG).
 ADD DDDEF(SMPMTS)   SHR DA(CRM.CRZONE.SMPMTS).
 ADD DDDEF(SMPSTS)   SHR DA(CRM.CRZONE.SMPSTS).
 ADD DDDEF(SMPSCDS)  SHR DA(CRM.CRZONE.SMPSCDS).
 ADD DDDEF(SYSLIB)   CONCAT(SMPMTS).
 ADD DDDEF(SYSUT1)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT2)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT3)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SYSUT4)   NEW DELETE SPACE(1,15) TRK
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK1)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK2)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK3)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK4)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK5)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPWRK6)  NEW DELETE SPACE(1,15) TRK DIR(20)
                         UNIT(SYSDA).
 ADD DDDEF(SMPOUT)     SYSOUT(*).
 ADD DDDEF(SMPRPT)     SYSOUT(*).
 ADD DDDEF(SMPLIST)    SYSOUT(*).
 ADD DDDEF(SYSPRINT)   SYSOUT(*).
ENDUCL.
```

Figure 54. *Sample job CNRZONE1 input to create target and distribution zone for CONSUL product(s)*

5. Member CNRZONE2 allocates target and distribution datasets for the CONSUL/RACF function, and defines DDDEFs in the appropriate zones.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*****************************************************************
//* Member:  CNRZONE2   Level: RCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To allocate new target/distribution libraries for
//*          CONSUL/RACF and to define the DDDEF entries in target
//*          and distribution zones
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*            2. The statements in the SMP/E input marked with <==
//*            3. Datasets
//*                 e.g. CHANGE 'CRM.' 'productprefix.' ALL
//*                 (Mind potential trailing dots)
//*****************************************************************
//*
//CNRZONE2 PROC REGSIZE=4096K,         Region available
//  CPREF='CRM.',                      Prefix. for libraries of new zone
//  CPROJ=CNR113,                      Project for libraries of new zone
//  GLOBAL='CRM.GLOBAL.CSI',           DSname for global CSI
//  DUNIT=SYSDA,                       Required unit name for libraries
//  DVOL=,                             Optional volume for libraries
//  DBLK80=23440,                      Blksize FB 80 (23440/27920/0)
//  DBLKU=32760                        Blksize loadlib (32760)
//*
//ZONE2DA EXEC PGM=IEFBR14,COND=(0,LT)
//CNRSAMP  DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..CNRSAMP,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(2,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//ACNRSAMP DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..ACNRSAMP,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(2,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//CNRPLIB  DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..CNRPLIB,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(2,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//ACNRPLIB DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..ACNRPLIB,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(2,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//CNRCLIB  DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..CNRCLIB,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//ACNRCLIB DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..ACNRCLIB,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//CNRPROC  DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..CNRPROC,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//ACNRPROC DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..ACNRPROC,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=&DBLK80)
//CNRLOAD  DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..CNRLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOL.SPACE=(TRK,(13,1,10)),
//            DCB=(RECFM=U,LRECL=0,BLKSIZE=&DBLKU)
//ACNRLOAD DD DISP=(NEW,CATLG),DSN=&CPREF.&CPROJ..ACNRLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(18,1,20)),
//            DCB=(RECFM=U,LRECL=0,BLKSIZE=&DBLKU)
//*
//ZONE2SMP EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
//   PARM='CSI=&GLOBAL'
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//         PEND
//*
//CNRZONE2 EXEC CNRZONE2
//ZONE2SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
UCLIN.
 ADD DDDEF(CNRSAMP)    SHR DA(CRM.CNR113.CNRSAMP).
 ADD DDDEF(CNRPROC)    SHR DA(CRM.CNR113.CNRPROC).
 ADD DDDEF(CNRPLIB)    SHR DA(CRM.CNR113.CNRPLIB).
 ADD DDDEF(CNRLOAD)    SHR DA(CRM.CNR113.CNRLOAD).
 ADD DDDEF(CNRCLIB)    SHR DA(CRM.CNR113.CNRCLIB).
 ADD DDDEF(ACNRSAMP)   SHR DA(CRM.CNR113.ACNRSAMP).
 ADD DDDEF(ACNRPLIB)   SHR DA(CRM.CNR113.ACNRPLIB).
 ADD DDDEF(ACNRLOAD)   SHR DA(CRM.CNR113.ACNRLOAD).
 ADD DDDEF(ACNRCLIB)   SHR DA(CRM.CNR113.ACNRCLIB).
ENDUCL.
SET BDY(CRZONED)                                    /* <========== */.
UCLIN.
 ADD DDDEF(ACNRSAMP)   SHR DA(CRM.CNR113.ACNRSAMP).
 ADD DDDEF(ACNRPROC)   SHR DA(CRM.CNR113.ACNRPROC).
 ADD DDDEF(ACNRPLIB)   SHR DA(CRM.CNR113.ACNRPLIB).
 ADD DDDEF(ACNRLOAD)   SHR DA(CRM.CNR113.ACNRLOAD).
 ADD DDDEF(ACNRCLIB)   SHR DA(CRM.CNR113.ACNRCLIB).
ENDUCL.
```

Figure 55. *Sample job CNRZONE2 to create target and distribution libraries for CONSUL/RACF*

6. Member CNRZONE3 allocates target and distribution datasets for the CONSUL/COLLECT function, and defines DDDEFs in the appropriate zones.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*********************************************************************
//* Member:  CNRZONE3   Level: RCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To allocate new target/distribution libraries and
//*          define the DDDEF entries in target and distribution zones
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*            2. The statements in the SMP/E input marked with <======
//*            3. Datasets
//*                 e.g. CHANGE 'CRM.' 'productprefix.' ALL
//*                 (Mind potential trailing dots)
//*********************************************************************
//*
//CNRZONE3 PROC REGSIZE=4096K, Region available
//  IPREF='CRM.',              Prefix. for libraries of new zone
//  IPROJ=CNF203,              Project for libraries of new zone
//  GLOBAL='CRM.GLOBAL.CSI',   Dsname for global CSI
//  DUNIT=SYSDA,               Required unit name for libraries
//  DVOL=,                     Optional volume for libraries
//  DBLKU=32760                Blksize loadlib (32760)
//*
//ZONE3DA EXEC PGM=IEFBR14,COND=(0,LT)
//CNFLOAD  DD DISP=(NEW,CATLG),DSN=&IPREF.&IPROJ..CNFLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(4,1,1)),
//            DCB=(RECFM=U,LRECL=0,BLKSIZE=&DBLKU)
//ACNFLOAD DD DISP=(NEW,CATLG),DSN=&IPREF.&IPROJ..ACNFLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(7,1,30)),
//            DCB=(RECFM=U,LRECL=0,BLKSIZE=&DBLKU)
//*
//ZONE3SMP EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
//  PARM='CSI=&GLOBAL'
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//         PEND
//*
//CNRZONE3 EXEC CNRZONE3
//ZONE3SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
UCLIN.
 ADD DDDEF(CNFLOAD)   SHR DA(CRM.CNF203.CNFLOAD).
 ADD DDDEF(ACNFLOAD)  SHR DA(CRM.CNF203.ACNFLOAD).
ENDUCL.
SET BDY(CRZONED)                                    /* <========== */.
UCLIN.
 ADD DDDEF(ACNFLOAD)  SHR DA(CRM.CNF203.ACNFLOAD).
ENDUCL.
```

Figure 56. *Sample job CNRZONE3 to create target and distribution libraries for CONSUL/COLLECT*

7. Member CNRZONE4 allocates a distribution dataset for the CONSUL/HLL run time system function, and defines DDDEFs in the appropriate zones.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*********************************************************************
//* Member:  CNRZONE4   Level: RCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To allocate new target/distribution libraries and
//*          define the DDDEF entries in target and distribution zones
//*          for CONSUL/HLL procedure library.
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*            2. The statements in the SMP/E input marked with <======
//*            3. Datasets
//*                 e.g. CHANGE 'CRM.' 'productprefix.' ALL
//*                 (Mind potential trailing dots)
//*********************************************************************
//*
//CNRZONE4 PROC REGSIZE=4096K, Region available
//  HPREF='CRM.',              Prefix. for libraries of new zone
//  HPROJ=HLL120,              Project for libraries of new zone
//  GLOBAL='CRM.GLOBAL.CSI',   Dsname for global CSI
//  DUNIT=SYSDA,               Required unit name for libraries
//  DVOL=,                     Optional volume for libraries
//  DBLKU=32760                Blksize loadlib (32760)
//*
//ZONE4DA EXEC PGM=IEFBR14,COND=(0,LT)
//AHLLLOAD DD DISP=(NEW,CATLG),DSN=&HPREF.&HPROJ..AHLLLOAD,
//            UNIT=&DUNIT,VOL=SER=&DVOL,SPACE=(TRK,(1,1,10)),
//            DCB=(RECFM=U,LRECL=0,BLKSIZE=&DBLKU)
//*
//ZONE4SMP EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
//  PARM='CSI=&GLOBAL'
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//         PEND
//*
//CNRZONE4 EXEC CNRZONE4
//ZONE4SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
UCLIN.
 ADD DDDEF(AHLLLOAD)  SHR DA(CRM.HLL120.AHLLLOAD).
ENDUCL.
SET BDY(CRZONED)                                    /* <========== */.
UCLIN.
 ADD DDDEF(AHLLLOAD)  SHR DA(CRM.HLL120.AHLLLOAD).
ENDUCL.
```

Figure 57. *Sample job CNRZONE4 to create distribution library for CONSUL/HLL run time support routines*

8. Member CNRZSMP0 receives SYSMODs from the distribution tape. Note that the BYPASS(FMID) parameter is required due to the order of SYSMODs in the SMPMCS dataset on tape.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*
//*********************************************************************
//* Member:  CNRZSMP0   Level: SCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To receive SMP data from tape.
//*          BYPASS(FMID) is required because of order of FMIDs on tape
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*                 e.g. CHANGE 'CRM.'          'globalprefix.' ALL
//*********************************************************************
//*
//CNRZSMP0 PROC REGSIZE=4096K, Region available
//  GLOBAL='CRM.GLOBAL.CSI',   Datasetname for global CSI
//  TEXPDT=98000,              Expiry date indicating external tape
//  TUNIT=3480,                UNIT name for medium (note 2)
//  TVOLSER=CR1103             Volume serial (name) for SL tape
//*
//SMP0RECV EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
//  PARM='CSI=&GLOBAL'
//SMPPTFIN DD DISP=(OLD,KEEP),UNIT=&TUNIT,VOL=(,RETAIN,SER=&TVOLSER),
//            LABEL=(1,SL,EXPDT=&TEXPDT),
//            DSN=SMPMCS
//         PEND
//*
//CNRZSMP0 EXEC CNRZSMP0
//SMP0RECV.SMPCNTL DD *
SET BDY(GLOBAL).
  RECEIVE SYSMODS BYPASS(FMID) LIST SOURCEID(CR1103).
```

Figure 58. *Sample job CNRZSMP0 to receive sysmods from distribution tape*

9. Member CNRZSMP1 contains JCL to run SMP/E and contains the following steps (you might want to run these one at a time):

   a. APPLY CHECK for the functions.

   b. APPLY for the functions.

   c. APPLY CHECK for the PTFs.

   d. APPLY for the PTFs.

   e. ACCEPT CHECK for the functions.

   f. ACCEPT for the functions.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*********************************************************************
//* Member:  CNRZSMP1   Level: SCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To apply functions
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*                 e.g. CHANGE 'CRM.'          'globalprefix.' ALL
//*            2. The SMP/E zonename
//*********************************************************************
//*
//CNRZSMP1 PROC REGSIZE=4096K, Region available
//  GLOBAL='CRM.GLOBAL.CSI' Datasetname for global CSI
//*
//SMP1SMP EXEC PGM=GIMSMP,REGION=&REGSIZE,COND=(0,LT),
//  PARM='CSI=&GLOBAL'
//         PEND
//*
//APPLYCK  EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
  APPLY S(RHL1200,SCF2003,SCR1103) CHECK.
//APPLY    EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
  APPLY S(RHL1200,SCF2003,SCR1103).
//PTFCK    EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
  APPLY FORFMID(RHL1200,SCF2003,SCR1103) CHECK.
//PTF      EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONET)                                    /* <========== */.
  APPLY FORFMID(RHL1200,SCF2003,SCR1103).
//ACCEPTCK EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONED)                                    /* <========== */.
  ACCEPT S(RHL1200,SCF2003,SCR1103) CHECK.
//ACCEPT   EXEC CNRZSMP1
//SMP1SMP.SMPCNTL DD *
SET BDY(CRZONED)                                    /* <========== */.
  ACCEPT S(RHL1200,SCF2003,SCR1103).
```

Figure 59. *Sample job CNRZSMP1 to run SMP/E for apply and accept.*

10. Member CNRZSMP9 contains JCL to apply a USERMOD to set the authorization code (expiration date). Fill in the authorization code supplied to you on the cover letter before running this step.

```
//*JOB1
//*JOB2
//*JOB3
//*JOB4
//*********************************************************************
//* Member:  CNRZSMP9   Level: SCR1103       Version: CONSUL/RACF 1.1.3
//* Purpose: To apply expiration date zap as a USERMOD
//* Note:    You must assure correctness of:
//*            1. The parameters of the procedure
//*                 e.g. CHANGE 'CRM.'          'globalprefix.' ALL
//*            2. The SMP/E zonename
//*            3. The authorization code / expiry date zap
//*********************************************************************
//*
//CNRZSMP9 PROC REG=4096K,     Region available
//  GLOBAL='CRM.GLOBAL.CSI'    Datasetname for global CSI
//*
//SMP9SMP EXEC PGM=GIMSMP,REGION=&REG,COND=(0,LT),
//   PARM='CSI=&GLOBAL'
//SMPOUT   DD SYSOUT=*
//SMPRPT   DD SYSOUT=*
//SMPLIST  DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SMPLOGA  DD SYSOUT=*
//         PEND
//*
//CNRZSMP9 EXEC CNRZSMP9
//*
//SMP9SMP.SMPCNTL DD *
SET BDY(GLOBAL).
  RECEIVE SYSMOD S(TRZAP00).
SET BDY(CRZONET)                                    /* <========== */.
  APPLY S(TRZAP00) CHECK /*REDO*/.
  APPLY S(TRZAP00) /*REDO*/.
//*
//SMP9SMP.SMPPTFIN DD *
++USERMOD(TRZAP00) /* REWORK(yyyyddd) */
 /*
    Purpose: Apply zap to extend license period
 */.
++VER(Z038) FMID(SCR1103).
++ZAP(CNRACF) /* Next line must be " NAME CNRACF EXPIRES" */.
 NAME CNRACF EXPIRES                        8 BYTE CSECT FOR AUTH-CODE
 REP     00    01020304,05060708            8 BYTES SUPPLIED BY VENDOR
```

Figure 60. *Sample job CNRZSMP9 to run SMP/E to receive and apply required USERMOD to set the authorization code*

11. Next proceed with the next section "A.4. Installation completion".

A.4. Installation completion

1. Run member CNRZUPDT of the installation JCL library to complete the JCL customization on the target datasets. It applies the changes you defined in the CNRZUPD member to the CNRCLIB, CNRSAMP, and CNRPROC datasets.

At this stage, the CONSUL/RACF program is ready for use: no system modifications nor authorizations are *required.* The program can be used in SP4, ESA, XA and pre-XA systems without modifications. You should, however, install the CNFCOLL program for authorized use to obtain optimal operation (see below).

2. To test the operation of the CNRACF program, run the member CNRJCPYR from the CRM.CNR113.CNRSAMP dataset. Help on solving common abends is given in section "1.5.5 Abends and other problems" for CNRACF.
3. Run the member CNRJCFLS and see if it yields the expected output.
4. If you have SAS, run the member CNRJSAS and see if it yields the expected output.
5. Invoke the CNR CLIST under ISPF and see if it yields the expected displays.
6. To test the (non-authorized) operation of the CNFCOLL program, run the member CNRJCPYV from the CRM.CNR113.CNRSAMP dataset. Help on solving common abends is given in the appropriate section in the CONSUL/COLLECT manual.
7. Edit member CNRJCFLV to run one or more verification jobs (remove comment asterisk at the appropriate procedure invocation). Note that Consul/Collect was run unauthorized and will miss VSAM information.
8. Obtain APF authorization for the CRM.CNF203.CNFLOAD dataset and permits on the FACILITY profile $CNF.AUDIT as described in the installation section of the CONSUL/COLLECT manual. Next rerun CNRJCPY (or use CNRJCPYV) - remember to change DSTAT to OLD. Check that catalogs and PDS directories are now included in the report. Next run the CNRJCFLV steps again.

A.5. Miscellaneous sample JCL

The library CRM.CNR113.CNRSAMP contains JCL samples, to run CNRACF and CNFCOLL, as well as sample input members for CONSUL/RACF. The members starting with CNRJ contain sample procedures that are intended for regular use of CNFCOLL and CNRACF (the regular procedures are also available in CRM.CNR113.CNRPROC). The members starting with CNRL are meant for use with sample member CNRJCFL (procedure CNRCFL) or CNRJCFLS (procedure CNRCFLS). The members starting with CNRV are meant for use with sample member CNRJCFLV (procedure CNRCFLV). The members starting with CNRX are meant for use with sample member CNRJSAS. You should at least read the usage guide starting at "1.1 Unloading and selecting RACF datasets".

The CNRJT200 member contains sample JCL to check a RACF database for structural conflicts with the RACF utility ICHUT200. The JCL indicates a much faster way to do so than presented in the RACF manual, provided that your site allows use of VIO for large datasets.

The CNRZIDR member in CRM.CNR113.CNRINST can be used to create more room for IDRDATA in a load module, as may be needed when many ZAPs have been applied to a module. It is meant for non-SMP installations only.

The CNRZAPEX member can be used to extend the license period for non-SMP installations. You will need an authorization code supplied by your software distributor.

B. Sample templates

This appendix contains sample output of the SHOW TEMPLATES command for a RACF 1.9 system

```
CNRACF 1.1.b 02/03/91 15.51   C O N S U L / R A C F   D A T A B A S E   U T I L I T Y  12 Feb 1991
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

SYSIN: show templates

CNR017I 00 Processing started for SYSRAC01 DISK45 CPOACH.RACF19.RDS2
          at 12 Feb 1991 16:26 running RACF 1.8.1
          Restructured database format RACF release 1.9.0
```

```
CNRPRTMP CNRACF 1.1.a 01/11/91 09.35 T E M P L A T E   F I E L D   D E F I N I T I O N S  12 Feb 1991
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,
```

Field	Entity	Segment	Id	Alias-of	Group	Bytes	Dflt	Format	Outlen	Flags
ACLCNT	GROUP	BASE	18			4	00	Decimal	5	
ACLCNT	DATASET	BASE	44			4	00	Decimal	5	
ACLCNT	GENERAL	BASE	47			4	00	Decimal	5	
ACL2ACC	GENERAL	BASE	59		ACL2CNT	1	00	Access	7	
ACL2ACNT	GENERAL	BASE	60		ACL2CNT	2	00	Decimal	5	
ACL2CNT	DATASET	BASE	32			4	00	Decimal	5	
ACL2CNT	GENERAL	BASE	56			4	00	Decimal	5	
ACL2NAME	GENERAL	BASE	57		ACL2CNT	8	00		8	
ACL2RSVD	GENERAL	BASE	61		ACL2CNT	varies	00	When	23	
ACL2UID	GENERAL	BASE	58		ACL2CNT	8	00		8	
ACL2VAR	DATASET	BASE	37		ACL2CNT	varies	00		varies	
ACSALTR	DATASET	BASE	8			2	FF	Decimal	5	Stat
ACSALTR	GENERAL	BASE	9			2	FF	Decimal	5	Stat
ACSCNT	GROUP	BASE	21		ACLCNT	2	00	Decimal	5	
ACSCNT	DATASET	BASE	47		ACLCNT	2	00	Decimal	5	
ACSCNT	GENERAL	BASE	50		ACLCNT	2	00	Decimal	5	
ACSCNTL	DATASET	BASE	9			2	FF	Decimal	5	Stat
ACSCNTL	GENERAL	BASE	10			2	FF	Decimal	5	Stat
ACSREAD	DATASET	BASE	11			2	FF	Decimal	5	Stat
ACSREAD	GENERAL	BASE	12			2	FF	Decimal	5	Stat
ACSUPDT	DATASET	BASE	10			2	FF	Decimal	5	Stat
ACSUPDT	GENERAL	BASE	11			2	FF	Decimal	5	Stat
APPLDATA	GENERAL	BASE	42			varies	00		varies	
AUDIT	DATASET	BASE	14			1	00	Audit	7	
AUDIT	GENERAL	BASE	14			1	00	Audit	7	
AUDITQF	DATASET	BASE	23			1	FF	AudLvl	7	
AUDITQF	GENERAL	BASE	19			1	FF	AudLvl	7	
AUDITQS	DATASET	BASE	22			1	FF	AudLvl	7	
AUDITQS	GENERAL	BASE	18			1	FF	AudLvl	7	
AUTHDATE	GROUP	BASE	5			3	FF	Date	11	
AUTHDATE	USER	BASE	4			3	FF	Date	11	
AUTHDATE	DATASET	BASE		CREADATE						
AUTHDATE	GENERAL	BASE		DEFDATE						
AUTHOR	GROUP	BASE	6			8	FF		8	
AUTHOR	USER	BASE	5			8	FF		8	
AUTHOR	DATASET	BASE	5			8	FF		8	
AUTHOR	GENERAL	BASE		OWNER						
CATEGORY	USER	BASE	32		NUMCTGY	2	00	Decimal	5	
CATEGORY	DATASET	BASE	29		NUMCTGY	2	00	Decimal	5	
CATEGORY	GENERAL	BASE	36		NUMCTGY	2	00	Decimal	5	
CGAUTHDA	USER	BASE	52		CGGRPCT	3	FF	Date	11	
CGAUTHOR	USER	BASE	53		CGGRPCT	8	FF		8	
CGCREADT	USER	BASE		CGAUTHDA						
CGDEFDAT	USER	BASE		CGAUTHDA						
CGFLAG1	USER	BASE	58		CGGRPCT	1	00	Flag	3	
CGFLAG2	USER	BASE	59		CGGRPCT	1	00	Flag	3	
CGFLAG3	USER	BASE	60		CGGRPCT	1	00	Flag	3	
CGFLAG4	USER	BASE	61		CGGRPCT	1	00	Flag	3	
CGFLAG5	USER	BASE	62		CGGRPCT	1	00	Flag	3	
CGGRPAUD	USER	BASE	64		CGGRPCT	1	00	Flag	3	
CGGRPCT	USER	BASE	50			4	00	Decimal	5	
CGGRPNM	USER	BASE	51		CGGRPCT	8	00		8	Sorted
CGINITCT	USER	BASE	57		CGGRPCT	2	FF	Decimal	5	
CGLJDATE	USER	BASE	55		CGGRPCT	3	FF	Date	11	
CGLJTIME	USER	BASE	54		CGGRPCT	4	FF	Time	5	
CGNOTUAC	USER	BASE	63		CGGRPCT	1	00	Flag	3	
CGOWNER	USER	BASE		CGAUTHOR						
CGRESMDT	USER	BASE	66		CGGRPCT	varies	00	Date	11	
CGREVKDT	USER	BASE	65		CGGRPCT	varies	00	Date	11	
CGUACC	USER	BASE	56		CGGRPCT	1	00	Access	7	
CLASTYPE	GENERAL	BASE	4			1	FF		1	
CLCNT	USER	BASE	41			4	00	Decimal	5	
CLNAME	USER	BASE	42		CLCNT	8	00		8	
CONGRPCT	USER	BASE	43			4	00	Decimal	5	
CONGRPNM	USER	BASE	44		CONGRPCT	8	00		8	


```
CNRPRTMP CNRACF 1.1.a 01/11/91 09.35 T E M P L A T E   F I E L D   D E F I N I T I O N S  12 Feb 1991
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

Field    Entity   Segment   Id Alias-of Group     Bytes Dflt Format  Outlen Flags
CREADATE GROUP    BASE         AUTHDATE
CREADATE USER     BASE         AUTHDATE
CREADATE DATASET  BASE       4                        3  FF  Date      11
CREADATE GENERAL  BASE         DEFDATE
DATAAPPL GROUP    DFP          2                  varies  00            39
DATAAPPL USER     DFP          2                  varies  00            39
DATACLAS GROUP    DFP          3                  varies  00             8
DATACLAS USER     DFP          3                  varies  00             8
DEFDATE  GROUP    BASE         AUTHDATE
DEFDATE  USER     BASE         AUTHDATE
DEFDATE  DATASET  BASE         CREADATE
DEFDATE  GENERAL  BASE       5                        3  FF  Date      11
DEVTYP   DATASET  BASE      18                        4  FF  Hex        8
DEVTYPX  DATASET  BASE      19                        8  FF             8
DFLTGRP  USER     BASE      15                        8  FF             8
DSTYPE   DATASET  BASE      16                        1  00  DsType     5
FLAG1    USER     BASE       6                        1  00  Flag       3
FLAG1    DATASET  BASE      13                        1  00  Flag       3
FLAG2    USER     BASE       7                        1  00  Flag       3
FLAG3    USER     BASE       8                        1  00  Flag       3
FLAG4    USER     BASE       9                        1  00  Flag       3
FLAG5    USER     BASE      10                        1  00  Flag       3
FLAG6    USER     BASE      20                        1  00  Flag       3
FLAG7    USER     BASE      21                        1  00  Flag       3
FLAG8    USER     BASE      22                        1  00  Flag       3
FLDCNT   GROUP    BASE      12                        4  00  Decimal    5
FLDCNT   USER     BASE      37                        4  00  Decimal    5
FLDCNT   DATASET  BASE      38                        4  00  Decimal    5
FLDCNT   GENERAL  BASE      38                        4  00  Decimal    5
FLDFLAG  GROUP    BASE      15           FLDCNT       1  00  Flag       3
FLDFLAG  USER     BASE      40           FLDCNT       1  00  Flag       3
FLDFLAG  DATASET  BASE      41           FLDCNT       1  00  Flag       3
FLDFLAG  GENERAL  BASE      41           FLDCNT       1  00  Flag       3
FLDNAME  GROUP    BASE      13           FLDCNT       8  00             8
FLDNAME  USER     BASE      38           FLDCNT       8  00             8
FLDNAME  DATASET  BASE      39           FLDCNT       8  00             8
FLDNAME  GENERAL  BASE      39           FLDCNT       8  00             8
FLDVALUE GROUP    BASE      14           FLDCNT  varies  00        varies
FLDVALUE USER     BASE      39           FLDCNT  varies  00        varies
FLDVALUE DATASET  BASE      40           FLDCNT  varies  00        varies
FLDVALUE GENERAL  BASE      40           FLDCNT  varies  00        varies
GAUDIT   DATASET  BASE      20                        1  00  Audit      7
GAUDIT   GENERAL  BASE      16                        1  00  Audit      7
GAUDITQF DATASET  BASE      25                        1  FF  AudLvl     7
GAUDITQF GENERAL  BASE      21                        1  FF  AudLvl     7
GAUDITQS DATASET  BASE      24                        1  FF  AudLvl     7
GAUDITQS GENERAL  BASE      20                        1  FF  AudLvl     7
GROUPNM  DATASET  BASE      15                        8  FF             8
INITCNT  GROUP    BASE       7                        2  FF  Decimal    5
INSTDATA GROUP    BASE      10                  varies  00        varies
INSTDATA USER     BASE      18                  varies  00        varies
INSTDATA DATASET  BASE      21                  varies  00        varies
INSTDATA GENERAL  BASE      17                  varies  00        varies
JOBNAMES GENERAL  DLFDATA    4           JOBNMCNT      8  00             8
JOBNMCNT GENERAL  DLFDATA    3                         4  00  Decimal    5
KEYDATE  GENERAL  SESSION    4                         4  00  Date      11
KEYINTVL GENERAL  SESSION    5                         2  00  Decimal    5
LCHGDAT  DATASET  BASE       7                         3  FF  Date      11                Stat
LCHGDAT  GENERAL  BASE       8                         3  FF  Date      11                Stat
LEVEL    DATASET  BASE      17                         1  FF  Decimal    5
LEVEL    GENERAL  BASE      15                         1  00  Decimal    5
LJDATE   USER     BASE      17                         3  FF  Date      11                Stat
LJTIME   USER     BASE      16                         4  FF  Time       5                Stat
LOGDAYS  USER     BASE      35                         1  00  Logdays    7
LOGDAYS  GENERAL  BASE      32                         1  00  Logdays    7
LOGTIME  USER     BASE      36                  varies  00  Logtime    9
LOGTIME  GENERAL  BASE      33                  varies  00  Logtime    9
LOGZONE  GENERAL  BASE      34                  varies  00  Decimal    5
LREFDAT  DATASET  BASE       6                         3  FF  Date      11                Stat
LREFDAT  GENERAL  BASE       7                         3  FF  Date      11                Stat
MAGSTRIP USER     BASE      23                  varies  00        varies Masked
MAXFAIL  GENERAL  SESSION    7                         2  00  Decimal    5
MEMCNT   GENERAL  BASE      43                         4  00  Decimal    5
MEMLST   GENERAL  BASE      44           MEMCNT  varies  00            60
MGMTCLAS GROUP    DFP          4                  varies  00             8
MGMTCLAS USER     DFP          4                  varies  00             8
MODELNAM GROUP    BASE      11                  varies  00            44
MODELNAM USER     BASE      29                  varies  00            44
NAME     USER     BASE         PGMRNAME
```


CNRPRTMP CNRACF 1.1.a 01/11/91 09.35 T E M P L A T E F I E L D D E F I N I T I O N S 12 Feb 1991
(C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

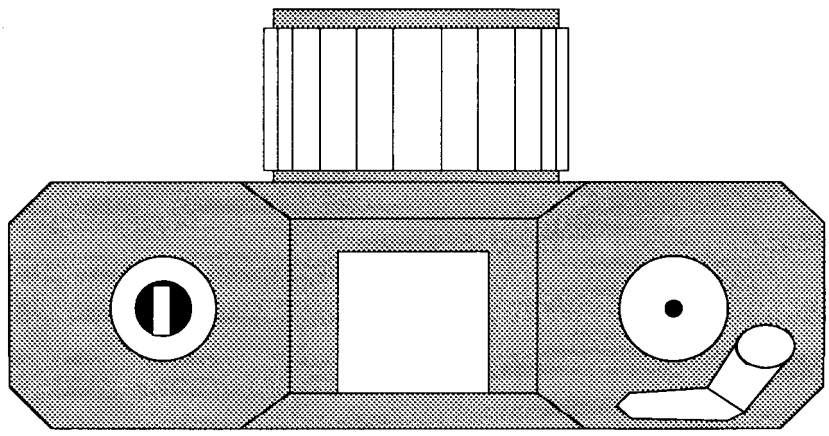
Field	Entity	Segment	Id	Alias-of	Group	Bytes	Dflt	Format	Outlen	Flags
NOTIFY	DATASET	BASE	30			varies	00		8	
NOTIFY	GENERAL	BASE	31			varies	00		8	
NOTRMUAC	GROUP	BASE	9			1	00	Flag	3	
NUMCTGY	USER	BASE	31			4	00	Decimal	5	
NUMCTGY	DATASET	BASE	28			4	00	Decimal	5	
NUMCTGY	GENERAL	BASE	35			4	00	Decimal	5	
OLDPWD	USER	BASE	27		PWDCNT	8	FF		8	Masked
OLDPWDNM	USER	BASE	26		PWDCNT	1	00	Decimal	5	
OWNER	GROUP	BASE		AUTHOR						
OWNER	USER	BASE		AUTHOR						
OWNER	DATASET	BASE		AUTHOR						
OWNER	GENERAL	BASE	6			8	FF		8	
PACSCNT	DATASET	BASE	36		ACL2CNT	2	00	Decimal	5	
PASSDATE	USER	BASE	13			3	FF	Date	11	
PASSINT	USER	BASE	11			1	FF	Decimal	5	
PASSWORD	USER	BASE	12			8	FF		8	Masked
PGMRNAME	USER	BASE	14			20	FF		20	
PROGACS	DATASET	BASE	35		ACL2CNT	1	00	Access	7	
PROGRAM	DATASET	BASE	33		ACL2CNT	8	00		8	
PWDCNT	USER	BASE	25			4	00	Decimal	5	
PWDGEN	USER	BASE	24			1	FF	Decimal	5	
RESFLG	GENERAL	BASE	23			1	00	Flag	3	
RESOWNER	DATASET	DFP	2			8	FF		8	
RESUMEDT	USER	BASE	34			varies	00	Date	11	
RETAIN	GENERAL	DLFDATA	2			1	00	Flag	3	
RETPD	DATASET	BASE	31			varies	00	Decimal	5	
REVOKECT	USER	BASE	28			1	FF	Decimal	5	Stat
REVOKEDT	USER	BASE	33			varies	00	Date	11	
SECLABEL	USER	BASE	49			8	00		8	
SECLABEL	DATASET	BASE	52			8	00		8	
SECLABEL	GENERAL	BASE	55			8	00		8	
SECLEVEL	USER	BASE	30			1	FF	Decimal	5	
SECLEVEL	DATASET	BASE	27			1	FF	Decimal	5	
SECLEVEL	GENERAL	BASE	37			1	FF	Decimal	5	
SENTCNT	GENERAL	SESSION	8			4	00	Decimal	5	
SENTFLCT	GENERAL	SESSION	10		SENTCNT	2	00	Decimal	5	
SENTITY	GENERAL	SESSION	9		SENTCNT	35	00		35	
SESSKEY	GENERAL	SESSION	2			varies	00		8	
SLSFAIL	GENERAL	SESSION	6			2	00	Decimal	5	
SLSFLAGS	GENERAL	SESSION	3			1	00	Flag	3	
STORCLAS	GROUP	DFP	5			varies	00		8	
STORCLAS	USER	DFP	5			varies	00		8	
SUBGRPCT	GROUP	BASE	16			4	00	Decimal	5	
SUBGRPNM	GROUP	BASE	17		SUBGRPCT	8	00		8	
SUPGROUP	GROUP	BASE	4			8	FF		8	
TACCNT	USER	TSO	2			varies	00		39	
TCOMMAND	USER	TSO	3			varies	00		varies	
TDEST	USER	TSO	4			varies	00		varies	
THCLASS	USER	TSO	5			varies	00		1	
TJCLASS	USER	TSO	6			varies	00		1	
TLPROC	USER	TSO	7			varies	00		8	
TLSIZE	USER	TSO	8			4	00	Decimal	5	
TMCLASS	USER	TSO	9			varies	00		1	
TMSIZE	USER	TSO	10			4	00	Decimal	5	
TOPTION	USER	TSO	11			1	00	Flag	3	
TPERFORM	USER	TSO	12			4	00	Decimal	5	
TRBA	USER	TSO	13			3	00	Hex	6	
TSCLASS	USER	TSO	14			varies	00		1	
TSOSLABL	USER	TSO	18			varies	00		8	
TUDATA	USER	TSO	15			2	00		2	
TUNIT	USER	TSO	16			varies	00		8	
TUPT	USER	TSO	17			varies	00	Hex	varies	
TVTOCCNT	GENERAL	BASE	24			4	00	Decimal	5	
TVTOCCRD	GENERAL	BASE	26		TVTOCCNT	3	00	Date	11	
TVTOCDSN	GENERAL	BASE	28		TVTOCCNT	varies	00		44	
TVTOCIND	GENERAL	BASE	27		TVTOCCNT	1	00	Flag	3	
TVTOCRDS	GENERAL	BASE	30		TVTOCCNT	varies	00		44	
TVTOCSEQ	GENERAL	BASE	25		TVTOCCNT	2	00	Decimal	5	
TVTOCVOL	GENERAL	BASE	29		TVTOCCNT	varies	00		6	

```
CNRPRTMP CNRACF 1.1.a 01/11/91 09.35 T E M P L A T E   F I E L D   D E F I N I T I O N S  12 Feb 1991
16:26
 (C) COPYRIGHT 1989, 1991, HANS SCHOONE AND CONSUL RISK MANAGEMENT B.V., VEENWEG 112, 2631 RB NOOTDORP,

Field     Entity   Segment   Id Alias-of Group     Bytes Dflt Format  Outlen Flags
UACC      GROUP    BASE       8                       1  00 Access       7
UACC      DATASET  BASE          UNIVACS
UACC      GENERAL  BASE      13                       1  00 Access       7
UAUDIT    USER     BASE      19                       1  00 Flag         3
UNIVACS   DATASET  BASE      12                       1  00 Access       7
USERACS   GROUP    BASE      20           ACLCNT      1  00 Access       7
USERACS   DATASET  BASE      46           ACLCNT      1  00 Access       7
USERACS   GENERAL  BASE      49           ACLCNT      1  00 Access       7
USERID    GROUP    BASE      19           ACLCNT      8  00              8
USERID    DATASET  BASE      45           ACLCNT      8  00              8
USERID    GENERAL  BASE      48           ACLCNT      8  00              8
USER2ACS  DATASET  BASE      34           ACL2CNT     8  00              8
USRCNT    GROUP    BASE      22                       4  00 Decimal      5
USRCNT    USER     BASE      45                       4  00 Decimal      5
USRCNT    DATASET  BASE      48                       4  00 Decimal      5
USRCNT    GENERAL  BASE      51                       4  00 Decimal      5
USRDATA   GROUP    BASE      24           USRCNT   varies  00          varies
USRDATA   USER     BASE      47           USRCNT   varies  00          varies
USRDATA   DATASET  BASE      50           USRCNT   varies  00          varies
USRDATA   GENERAL  BASE      53           USRCNT   varies  00          varies
USRFLG    GROUP    BASE      25           USRCNT      1  00 Flag         3
USRFLG    USER     BASE      48           USRCNT      1  00 Flag         3
USRFLG    DATASET  BASE      51           USRCNT      1  00 Flag         3
USRFLG    GENERAL  BASE      54           USRCNT      1  00 Flag         3
USRNM     GROUP    BASE      23           USRCNT      8  00              8
USRNM     USER     BASE      46           USRCNT      8  00              8
USRNM     DATASET  BASE      49           USRCNT      8  00              8
USRNM     GENERAL  BASE      52           USRCNT      8  00              8
VERSION   GROUP    BASE       3                       1  01              1
VERSION   USER     BASE       3                       1  01              1
VERSION   RSVTMP03 BASE       3                       1  01              1
VERSION   DATASET  BASE       3                       1  01              1
VERSION   GENERAL  BASE       3                       1  01              1
VOLCNT    DATASET  BASE      42                       4  00 Decimal      5
VOLCNT    GENERAL  BASE      45                       4  00 Decimal      5
VOLSER    DATASET  BASE      43           VOLCNT      6  00              6
VOLSER    GENERAL  BASE      46           VOLCNT      6  00              6
VOLUME    DATASET  BASE          VOLSER
WARNING   DATASET  BASE      26                       1  00 Flag         3
WARNING   GENERAL  BASE      22                       1  00 Flag         3
```


Index

CONSUL/COLLECT for MVS



User Reference Manual

Version 2.0

October 4, 1991

This manual applies to modification level 3 of version 2.0

The examples in this manual are not meant to be representative or particularly applicable to another situation. No guarantees are expressed or implied as to the accuracy, correctness, and applicability of any result produced by the CONSUL/COLLECT program.

The CONSUL/COLLECT program is distributed by

CONSUL Risk Management B.V.
Veenweg 112
2631 RB Nootdorp
The Netherlands
tel: +31 15 146670 or +31 1731 7989
fax: +31 15 146697 or +31 1731 7433

Please direct requests for information or maintenance to your service representative.
Problem reports are accepted formally by fax or mail and must contain sufficient information (summary dump, joblog, JCL, and input commands if abended).

Table of Contents

Summary of amendments

From 2.0.2 to 2.0.3

- Support for SP4.1 and SP4.2 features has been added. This includes dynamically reconfigured devices, a new record for XCF sysplex data, and EDT access.
- Ten new messages have been added for problems related to SP4 systems.
- Read Configuration Data is now done if it exists according to Sense Id, not based anymore on MVS control block settings.
- Logical Partition information has been added to aid in support of PR/SM. The current LPAR name has been added to the system record, and a new record type is generated describing all LPARs on the machine. This requires RMF to be active.
- The output now lists the currently active MVSCP configuration id.
- Command input processing has been enhanced to provide clearer error messages for syntax errors, in addition they now have 9 different message numbers instead of the same number.
- Minor enhancements to the summary dump.
- The parameter **OFFLINE=** has been added. It can be set to **YES** to obtain UCB information for offline devices.
- The parameter **SHARED=** has been added. It can be set to **NO** to limit data collection for disks to devices that have been generated as non-shared.

From 2.0.1 to 2.0.2

- The authorization checking has been changed to allow APF authorized operation if RACF is not active or present. In addition, FOCUS=CCWANAL and FOCUS=CONFIG are now allowed if no profile is present or if the class FACILITY is inactive. This is not the case for FOCUS=AUDIT: due to the sensitive nature of the information collected, operation is allowed only if class FACILITY is active and access is permitted, or if RACF is inactive or not present.
- Ten new messages have been added to assist in determining the cause of authorization failures.

From CONSUL/COLLECT 2.0.0 to 2.0.1

- The summary dump now shows the catalog name and PDS name for each I/O executor.

From IOCONFIG 1.5.6 to CONSUL/COLLECT 2.0.0

- The I/O operations and burst waits can now be performed in parallel. The amount of parallellism can be set by the new command PARALLEL= with the values NONE, PATHGROUP, and PATH (default).

- ICF catalog dumping can now also be done by VSAM instead of EXCP. This is called *slowdown* mode, and is automatically invoked if a catalog has more than 16 extents and the DFP level does not support EXCP on datasets with more than 16 extents or if the program is not running APF authorized on a pre-DFP V3 system.
- ICF catalog dumping in EXCP mode now supports catalogs with NOIMBED as well as IMBED.
- Storage above 16Mb is exploited for catalog processing with VSAM.
- Security and auditability for the use of the functions requiring APF authorization is achieved by consulting a number of FACILITY profiles starting with $CNF through the SAF interface. The proper FACILITY profile *must* be present and allow access to enable an authorized function, and may be used to create an audit trail of authorized operations. If the profile is not present or disallows access, the program refuses to operate with APF authorization.
- IDCSS03 is no longer used; the program now uses its own I/O driver in the cases that require a guaranteed device path or a non-prefixed channel program. Authorization to use the I/O driver must be granted through the FACILITY profiles $CNF.CCWANAL or $CNF.CONFIG.
- The requirement for an ALTER permit on the catalogs to be dumped has been removed. If run as an APF authorized program, CONSUL/COLLECT will now bypass the catalog dataset profiles, if authorized through the FACILITY profile $CNF.AUDIT.
- The selection commands have been extended to allow multiple selections. In addition, an EXCLUDE command has been added to suppress processing for one or more volumes.
- The dataset information being collected from VTOC, VVDS, and catalogs can be restricted based on an arbitrary length dataset name prefix.
- The specific data collection options are summarized in the new option FOCUS= that indicates the purpose of the data collection. Currently supported values are: CCWANAL, CONFIG, AUDIT, and ALL, to indicate that the data is meant for CONSUL/CCW, CONSUL/CONFIG, CONSUL/RACF, or all three, respectively.
- New functionality has been added to dump the contents of PDS directories. There currently are two uses for this data: auditing of PROGRAM profiles and AC=1 APF authorized modules by CONSUL/RACF, and I/O attribution on the member level by CONSUL/CCW.
- Security and integrity is enhanced for processing PDS directories by offering the possibility to dump the directory without having READ access. This enhances security because READ access might imply authority to execute all potentially dangerous AC=1 utilities. Authorization must be granted through the profile $CNF.AUDIT for dumping linklist, LPA list, and APF library directories.
- A report is now produced listing statistics on the amount and nature of the information collected both by volume and by catalog.

- The print output now contains page headers and page skips.
- Additional checks are incorporated to prevent some common abend situations, for instance 806 because the I/O appendage was not copied with the program to the APF library, and abend 113 because VVDS open requires APF authorization in DFP V3.
- The incore PPT can now be dumped. This can be used by CONSUL/RACF and requires APF authorization and access to the FACILITY profile $CNF.AUDIT.
- Attempts in non-APF mode to access all channels can now be customized by the parameters BURSTSIZE, BURSTWAIT, and BURSTS.
- Configuration information has been added on secondary RACF datasets, JES2 checkpoint datasets, and JES2 spool volumes. In addition, more releases are supported of MVS, DFP, and JES2.
- Release level information is now extracted for MVS, DFP, JES2, and VTAM. In addition the SMF id, the JES2 node name is extracted as well as the VTAM NETID and GRS system name (all three are used by RACF 1.9 and collected for use by CONSUL/RACF).

1 Introduction

The CONSUL/COLLECT program collects data on the connections in your I/O subsystem, as well as information on the way your MVS system is configured, using only a minimal amount of resources and elapsed time. To get a complete picture of your I/O subsystem, CONSUL/COLLECT must be run on all systems using shared DASD[1]. The data can be used to provide a mapping between the physical target of I/Os on DASD (string, device on string, cylinder, track) and the named entity which is known to the user or operating system (device number, volume name, dataset, file or data space, PDS member). This information is necessary for a variety of tasks like:

- interpreting the trace data collected by GTF correctly (e.g. which device connects to which channel, to which dataset and member is the I/O directed, mapping the I/O on a shared DASD from different addresses on different systems to the same device). This is done by CONSUL/CCW (CCWANAL);
- creating a system definition file for EREP, mapping errors on shared DASD to the same device (error counter);
- creating CONFIGXX members in an XA system for comparison between configurations;
- Creating a picture of the physical or logical connections of your I/O devices, controllers, and channels. This is done by CONSUL/CONFIG;
- accounting DASD space usage. Current accounting packages often create such a heavy I/O load on the system, that they are run only once a week, inducing user behaviour like dumping everything to tape for the weekend if that is cheaper. CONSUL/COLLECT can be run many times a day to collect DASD usage data without impressive I/O load or elapsed time.
- Matching resources (volumes, datasets, VSAM clusters) to (generic) profiles in RACF.
- Auditing the system protection. For instance information can be collected on current RACF options, RACF datasetnames, APF datasetnames, linklist datasets, LPA list datasets, incore MLPA and PLPA members, page- and swap datasets, SMF datasets, JES2 datasets, actual PPT contents, contents of APF datasets (AC=1 module information), etc.

CONSUL/COLLECT by itself does not perform any of the functions, it only collects data as fast as it can.

[1] CONSUL/RACF 1.1 does not support partially shared DASD yet. So for this release of CONSUL/RACF, CONSUL/COLLECT needs to be run on only 1 system.

The CONSUL/COLLECT program includes no reporting functions other than some messages and a summary report, because it is designed to use minimal resources during data collection. Data analysis must be done by a separate postprocessing program. The CONSUL/CCW, CONSUL/CONFIG, and CONSUL/RACF programs provide support for this purpose. In addition, you might write your own postprocessor for the collected data.

2 Usage guide

This chapter describes usage considerations. The complete command syntax description is part of the Reference chapter.

2.1 Getting started

The program has to be installed before it can be used. This is described separately in appendix A. When it has been installed, it can be used immediately. If CONSUL/COLLECT has been distributed as part of another package, then that package will contain sample JCL to run CONSUL/COLLECT. For operations requiring APF authorization, you will have to obtain a permit to the proper profile. See "2.3 Authorized or Unauthorized Execution?" for a discussion on this. However, for a first try you won't need APF authorization.

CONSUL/COLLECT takes its input from the parameter string, from the SYSIN file, or both. Neither is required. The most important parameter to learn about is the FOCUS= parameter. This parameter indicates to CONSUL/COLLECT for which purpose data has to be collected. If you do not specify any parameters, then CONSUL/COLLECT assumes FOCUS=CCWANAL. This means that it assumes that it has to collect data to be processed by the CONSUL/CCW product. Alternatives are CONFIG for CONSUL/CONFIG, AUDIT for CONSUL/RACF, and ALL if you want to include everything in one file.

Let's see some sample JCL saying that you would like to collect data for CONSUL/RACF, another product supported by CONSUL/COLLECT:

```
//*
//* Sample JCL to collect data for CONSUL/RACF
//*
//CNFAUDIT EXEC PGM=CNFCOLL,REGION=6M,PARM='FOCUS=AUDIT'
//SYSPRINT DD SYSOUT=*
//IOCONFIG DD DISP=(,CATLG),DSN=userid.name.IOCONFIG,
//         DD UNIT=SYSDA,SPACE=(CYL,(5,5),RLSE)
```

The program will generate messages and a report. If you want, you can browse the IOCONFIG output to get an impression; you will see that it is a variable blocked file with different kinds of records. You need not be concerned with its layout, all CONSUL programs can read and process it.

In the SYSPRINT file, you will probably find messages stating that information could not be collected or is missing. This need not be a problem for the programs analyzing the configuration, but they may be helpful to debug problems.

The following figure gives an example of output you may expect in an authorized run.


```
CNFCOLL 2.0.b 11/23/90 22.44 C O N S U L / C O L L E C T   C O N F I G U R A T I O N   C O L L E C T O R          page   1
 (C) COPYRIGHT 1986-1990 by Hans Schoone and CONSUL Risk Management B.V., Veenweg 112, 2631 RB Nootdorp, The Netherlands

PARM:  FOCUS=ALL,PAR=PATH,REPORT

CNF047I 00 Data collection started on 15 Jan 1991 15:54 for node DXAPST2  sysname IPO1     sid IPO1 netid DXAPS000
CNF039I 00 Running MVS/SP2.2.3  DFP 3.1.1  JES2 SP 2.2.0  VTAM 3.2
CNF000I 04 Control block IOCH omitted because of nil pointer
CNF000I 04 Control block IODN omitted because of nil pointer
CNF111I 00 Scheduler allocated 10 I/O executors
CNF017I 04 Path 13 to 3380 device 0345 SYSR22 not operational
CNF017I 04 Path 20 to 3380 device 0694 USER10 not operational
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0147 RES004
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0145 RES023
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0144 RES003
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0143 RES022
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0142 RES002
CNF022I 08 Storage director ID not returned by IOS for path 01 to 3350 device 0140 XX3350
CNF017I 04 Path 13 to 3380 device 034F ADMI13 not operational
CNF017I 04 Path 13 to 3380 device 034E USER53 not operational
CNF017I 04 Path 13 to 3380 device 034D CICS04 not operational
CNF017I 04 Path 13 to 3380 device 034B USER54 not operational
CNF017I 04 Path 13 to 3380 device 034A SMS002 not operational
CNF017I 04 Path 13 to 3380 device 0349 HSM002 not operational
CNF017I 04 Path 13 to 3380 device 0348 USER41 not operational
CNF017I 04 Path 13 to 3380 device 0347 ADMI15 not operational
CNF017I 04 Path 13 to 3380 device 0344 OVFL00 not operational
CNF017I 04 Path 13 to 3380 device 0342 USER46 not operational
CNF017I 04 Path 13 to 3380 device 0341 HSM003 not operational
CNF052I 08 Connected catalog MVSTST  CAT1.MVSTEST.#00 not found on volumes processed
```

```
CNFCOLL 2.0.b 11/23/90 22.44 C O N S U L / C O L L E C T   C O N F I G U R A T I O N   C O L L E C T O R          page   2
Volume overview
```

Volume	Dev	VTOC	trks	read	DSCB#	VSAM	ISAM	nonVSAM	F3	VVDS	trks	read	VVR#	BCS	ix	data	slw	trks	recs	PDS	no	trks	dirblk
ADMI09	0287		60	6	278			265	11												1	1	1
ADMI10	0925		45	14	656	193		448	13		5	4	288			1		20	1295				
ADMI11	0926		45	17	846	235		587	22		3	3	352										
ADMI12	0927		45	15	785	147		593	43		3	2	216										
ADMI13	034F		45	9	455	99		340	14		3	2	147										
ADMI15	0347		45	16	680	55		603	20		3	2	79										
CICS01	0286		14	2	61	41		16	2		7	1	43										
CICS02	0289		45	8	311	57		243	9		15	1	59										
CICS03	0920		15	2	64	13		49			15	1	13								1	1	1
CICS04	034D		15	2	71	1		63	5		15	1	1										
DASD02	028D		60	8	308	1		287	18		4	1	1										
DASD05	0281		60	5	245	15		220	8		3	1	19								3	4	82
DASD06	0924		60	9	340	11		322	5		3	1	16			4		103	11225		3	6	212
DASD10	028F		60	6	253	13		228	10		4	1	18								3	8	246
EDAS00	0D20		1	1	5	2		1			1	1	2										
EDAS01	0D21		1	1	5	2		1			1	1	2										
EDAS02	0D22		1	1	5	2		1			1	1	2										
EDAS03	0D23		1	1	5	2		1			1	1	2										
HSM002	0349		30	4	122	3		117			15	1	3										
HSM003	0341		30	4	194	3		189			15	1	3										
J220#0	0283		1	1	4			2															
MICS00	0288		15	4	169	1		142	24		15	1	1										
OVFL00	0344		12	4	106	1		103			10	1	100										
RES002	0142		15	1	3			1															
RES003	0144		15	1	4			2															
RES004	0147		15	1	4			2															
RES022	0143		15	1	3			1															
RES023	0145		15	1	3			1															
SMS002	034A		30	11	386	2		379	3		10	2	377										
SYSR18	0285		15	3	105	3		100			15	1	4								9	18	530
SYSR19	0923		15	3	139	19		112	6		15	1	28								7	10	253
SYSR20	028C		15	3	129	3		124			15	1	4										
SYSR21	0282		15	3	149	9		118	20		15	1	13										
SYSR22	0345		15	3	120	20		91	7		15	1	23			3		27	5274		13	15	145
SYSR23	028A		12	2	52	8		42			3	1	9										
USER10	0694		15	3	63	7		54			3	1	9		1	3		7	1014				
USER41	0348		15	7	215	1		193	19		15	1	1										
USER42	0280		15	10	351	1		320	28		3	1	1										
USER43	0921		30	11	326	2		296	26		7	1	2										
USER44	028E		15	7	217	1		197	17		15	1	1										
USER45	0922		30	11	323	2		294	25		7	1	2										
USER46	0342		15	9	353	1		326	24														
USER51	028B		45	16	568	20		529	17		7	1	21			1		8	1458				
USER52	0284		15	12	267	2		239	24		7	1	3										
USER53	034E		15	8	354	1		336	15		7	1	1										
USER54	034B		15	4	153	2		144	5		15	1	2										
XX3350	0140		30	1	4			2															
total	47		1153	272	10259	1001		8724	440		316	47	1868		1	12		165	20266		40	63	1470

```
CNF008I 00 Number of DASD devices interrogated: 47
CNF009I 00 Number of DSCB entries copied: 10259
CNF010I 00 Number of VVDS datasets processed: 38
CNF011I 00 Number of VVR entries copied: 1868
CNF041I 00 Number of ICF catalogs processed: 12
CNF042I 00 Number of BCS records copied: 20268
CNF035I 00 Number of PDS directories processed: 40
CNF034I 00 CNFCOLL used 5.7 CPU seconds, 33 elapsed seconds, and collected 4.992 Mb (0.153 Mb/s)
```


2.2 Operating System and DFP Release Dependency

The features included in CONSUL/COLLECT make the data heavily dependent on the operating system version in effect during the data collection. The releases currently supported are:

MVS/370 MVS/SP1.3.3 and upward with DFP 1.0 and up,

MVS/XA MVS/SP2.1.3 and upward with DFP 1.2 and up,

MVS/ESA MVS/SP3.1.0 and upward with DFP 2.3 and up.

Older releases probably work for most options. The CONSUL/COLLECT program has been designed to meet changes in the operating system without completely aborting work. Instead, some record types will be left out if the program does not know how to find them, and a message will be sent to the user.

The most important dependencies on the MVS and DFP release are summarized below:

- DFP version 3 protects the VVDS dataset, allowing only APF authorized programs to obtain information about VSAM datasets and SMS characteristics. A CONSUL/COLLECT program running from a user library cannot read the VVDS information. For CONSUL/CCW, this results in system generated VSAM dataset names that cannot be matched to cluster names. For CONSUL/RACF, since protection is based on cluster names, this will result in improper handling of VSAM datasets.
- MVS releases running under VM may not be able to retrieve all required status information from the storage directors and string controllers, resulting in an incomplete configuration description. Furthermore the message CNF017I (path not operational) may occur on MVS/370, meaning that no alternate path is available to MVS. This may be a problem for CONSUL/CCW. It is no problem for CONSUL/RACF.
- MVS/XA and MVS/ESA systems running under VM/XA, and systems without RMF, do not report Logical Control Unit numbers. This problem is generally circumvented by CONSUL/CCW. It is no problem for CONSUL/RACF.
- 3350 disks cannot return their storage director ID in /370 systems (see also the next section). This is a problem for CONSUL/CCW.

2.3 Authorized or Unauthorized Execution?

CONSUL/COLLECT provides support for both authorized and unauthorized operation.

To run authorized, the program must be copied to an APF library and run from this library. In addition, FACILITY profiles should be established to define who may use which authorized functions. For FOCUS=AUDIT, the program will *refuse* to operate with APF authorization unless the caller has sufficient authority on the proper FACILITY profile. For FOCUS=CCWANAL and FOCUS=CONFIG, the program will refuse to operate only if access is refused on the corresponding FACILITY profiles - if the profiles are not present or if authorization checking for class FACILITY is inactive, access will be allowed. In all cases, access wil be allowed if RACF is inactive or not installed (a message will be issued to indicate that no authorization checking was possible and why).

Callers that want to run the program without having access to the proper FACILITY profile, must drop the APF authorization, for instance by running it from a non-APF library, or by including a non-APF library in the STEPLIB.

The program issues a message if it is run without authorization, to note the reason for missing information. Clearly, not all configuration information can be obtained if the program is run without authorization. Specifically, APF authorization provides the following benefits:

1. *DPF version 3* requires authorization to read the VVDS dataset. The VVDS is used by CONSUL/CCW to report cluster names instead of system-generated names. The VVDS is required by CONSUL/RACF if VSAM dataset protection has to be taken into account. Use of this function requires authorization on either $CNF.CCWANAL or $CNF.AUDIT.

2. *Storage director IDs for 3350* may be obtained (if the OS release is sufficient, MVS/SP1.3.4 is not, MVS/SP2.1.3 is). Use of this function requires authorization on either $CNF.CCWANAL or $CNF.CONFIG.

3. Information on the current caching configuration for 3880 controllers is extracted. This includes all *cache sizes* and information on which devices have been *disabled for caching*. This information is always extracted for 3990 controllers. Use of this function requires authorization on either $CNF.CCWANAL or $CNF.CONFIG.

4. Information on *storage director IDs* for 3380s and *string controller IDs* is requested directly on the correct channel (*guaranteed device path*). The unauthorized version retries up to 80 times to issue I/Os over all paths, and sometimes fails anyway (especially in pre-XA systems that don't have specified channel rotation). Authorized execution results in no *WAITs* being issued for 3380s, and in complete information. Use of this function requires authorization on either $CNF.CCWANAL or $CNF.CONFIG.

5. To process catalog contents efficiently in a shared DASD environment and to guarantee completeness, authorization is required or ALTER access to the catalogs. However, in a DFP V3 environment authorization is required anyway because VVDS access is required by CONSUL/COLLECT to dump catalogs. CONSUL/RACF

requires the catalog contents to verify VSAM dataset protection, since the RACF-indicated bit resides in the catalog for VSAM datasets. Use of this function requires authorization on $CNF.AUDIT.

6. Security and integrity can be enhanced for CONSUL/COLLECT operations by not requiring READ access to APF libraries to dump the directory and by not requiring ALTER to dump ICF catalogs. This function requires APF to bypass the dataset profiles. Use of this function requires authorization on $CNF.AUDIT.

More authorized functions may be included in future releases.

For FOCUS=CONFIG and FOCUS=CCWANAL, if running without authorization results in missing path information, then you may want to try the program once with authorization, and see if it finds more information than without. If this makes no difference, your OS release may not be sufficient or not yet supported. Running the program authorized will speed up the processing, since no *WAITs* are issued.

As with any authorized program, it is recommended that you install the program on a test system first.

3 Reference material

This chapter contains two sections. The first describes the JCL considerations, and the second describes the command and parameter syntax.

3.1 Calling JCL

The call interface to CONSUL/COLLECT's main module CNFCOLL may follow the convention used for the JCL EXEC statement as well as the TSO command calling interface (CPPL). Note, however, that authorized execution from TSO requires an entry in the authorized command or authorized program list (IKJTSO*xx*). DDname/file names supported are:

SYSPRINT Lists messages and statistics. The record format is set by default to VBA for non-SYSOUT files, and to VA for SYSOUT files. However, record formats V, VB, F, FB, VA, FA, VBA, and FBA are all supported. The record length defaults, depending on the record format, to yield a printable line length 132. With the default VBA record format, this would yield 137 for the LRECL. A shorter line length is possible (e.g. 79) but will result in truncation of some headers, messages, and report lines. If you don't specify a blocksize, then a suitable default is chosen.

SYSTERM Lists status messages and message with a severity of 8 or higher. These messages are also printed on SYSPRINT. It may be omitted.

IOCONFIG All configuration information is written to a sequential file in VB format with a maximum LRECL of at least 5K. The LRECL specified limits the length of VVDS and catalog records copied. If no LRECL is specified, then it is set to 4 smaller than the blocksize. The blocksize should be large, but at least 5K. The recommended and default value is 32760. Thus, the DCB parameter should only be specified if you want to put the dataset on a device that does not support 32K blocksizes, like a 3350. In this case, specify DCB=BLKSIZE=19069.
The SPACE parameter should be expressed in cylinders and be sufficient to contain all *used* VTOC and VVDS space. Typically, a specification of (CYL,(5,5),RLSE) is more than sufficient. Most sites' configuration information only takes a few cylinders.

SYSIN This file can contain parameters. They are described in the next section.

No DDnames are required. However, no configuration information is collected if you omit the IOCONFIG DD statement, and no messages will be generated if SYSPRINT is missing. Sample JCL included is listed in "2.1 Getting started".

3.2 Parameters

The CONSUL/COLLECT program supports a number of parameters or commands to restrict the information collected to a subset of your I/O subsystem or to a specific purpose. Some restrictions are more limiting than others, and some restrictions can be combined to generate a subset.

Multiple parameters may be given, separated by commas and/or blanks. The commands are not case sensitive.

The parameters may be specified on the PARM field of the EXEC statement, or in the SYSIN file. From the SYSIN file, only positions 1 to 72 are read. Commands can be continued on the next line, but not in the middle of a word: the line end acts as a separator just like a blank or comma. If parameters are specified more than once, the value last given will be used. Parameters on the EXEC statement or passed on a TSO command are processed before the parameters in the SYSIN file. All parameters are listed on the SYSPRINT file, prefixed with their origin (PARM or SYSIN). The parameters are presented in the next sections in four categories:

1. Selection and exclusion commands
2. Options for both APF and non-APF
3. Options used for non-APF only
4. Options used for APF only.

3.2.1 Selection and exclusion commands

The commands SELECT and EXCLUDE can be used to select either on device/volume level or on dataset level. More than one SELECT or EXCLUDE command may be present. Multiple SELECTs on the same level imply an 'inclusive or' function. For each selection level, the commands are combined in the following way: first, the SELECT options are processed. If a SELECT was present, and the selection criteria of all SELECT commands fail, then the current candidate will be skipped. If the SELECT options are met, then the EXCLUDE options are processed (if present). If all the EXCLUDE commands fail, then the current candidate is selected and will be included, otherwise it will be skipped.

SELECT=*list*
SEL=*list*
S=*list*
The SELECT command accepts a list of selections, enclosed in parentheses, and separated by commas. If only one selection option is needed, then the parentheses may be omitted. The selections in the list indicated an AND condition if they are on the same level; multiple commands an OR condition.

EXCLUDE=*list*
EXCL=*list*
X=*list*
The EXCLUDE command accepts a list of selections, enclosed in parentheses, and separated by commas. If only one selection option is needed, then the parentheses may be omitted. The selections inside the list indicate an AND condition if they are on the same level; multiple commands an OR condition.

The device/volume level selection options are:

CH=*xx*
CHP=*xx*
CHANNEL=*xx*
C=*xx*
This can be used to restrict selection to the devices on a specified (physical) channel (path). The channel must be specified as two hexadecimal digits.

VOLUME=*xxxxxx*
VOLSER=*xxxxxx*
VOL=*xxxxxx*
V=*xxxxxx*
This can be used to restrict selection to the devices which have a volume serial starting with the specified string (1 to 6 characters can be specified).

DEV=*xxx*
DEVICE=*xxx*
U=*xxx*
This can be used to restrict selection to the devices which have a device number starting with the specified string (1 to 3 hexadecimal digits).

LCU=*xxx*
This can be used to restrict selection to the devices which are member of the specified logical control unit. This parameter applies to MVS/XA and MVS/ESA sites not running under VM. The LCU number of a device group is available from RMF reports and IOCP reports. The LCU must be specified as 3 hexadecimal digits.

The dataset level selection option is:

DSN=*prefix*
DSNPREF=*prefix*
D=*prefix*
This can be used to restrict selection to the datasets that have a name starting with the specified string.

An example of a selection option to collect information for one LCU only:


```
SELECT=LCU=005
```

The same example, but exclude a specific volume giving problems:

```
SELECT=LCU=005
EXCLUDE=VOL=DISK12
```

An example to select a number of volumes based on their prefix, abbreviated as shortly as possible:

```
S=V=SSD,S=V=SHR
```


3.2.2 General options

These options are valid both with and without APF authorization.

CAPS This can be used to request capitalization of output messages. It will not influence the first page header, however. Neither will it influence the parameter listing of previous parameters.

FOCUS=*focus*
F=*focus* This parameter indicates the intended use of the configuration data. It can be CCWANAL, CONFIG, or AUDIT. It automatically sets the proper combination of data collection options for the CONSUL/CCW, CONSUL/CONFIG, and CONSUL/RACF products, respectively.

NOREPORT This option can be used to suppress the volume and catalog reports.

PARALLEL=*option*
PAR=*option* This selects the amount of parallelism, and hence the amount of storage used by CONSUL/COLLECT. The values for *option* can be: **NONE, PATHGROUP,** and **PATH.** PATH is the default and recommended setting. It indicates that I/O is scheduled for at most one device per channel path. PATHGROUP requests a maximum of one I/O per group of paths (and hence per LCU). This option can be used if there is insufficient storage available for the default, or if PATH adds too much I/O to a heavily loaded LCU. Finally, NONE can be specified to force sequential operation, i.e. to access only one device at a time.

ALLOC=NO This can be used to prevent allocation of the DASD devices in your system. If no other parameters are specified, it will result in collecting only the configuration information that can be found in the OS control blocks. This includes all device numbers and volume names, but not the physical IDs necessary to map (shared) DASD, nor the contents of any VTOC, VVDS, catalog, or PDS.

PATH=NO This can be used to suppress processing to obtain information across all paths to a device. It is implied by FOCUS=AUDIT.

PATH=YES This can be used to include processing to obtain information across all paths to a device. It is implied by FOCUS=CCWANAL and FOCUS=CONFIG.

VTOC=NO This can be used to suppress collection of all DSCB and VVR information. If no other parameters are specified, then this will result in the complete configuration information with the device as the smallest entity. It is implied by FOCUS=CONFIG.

VVDS=NO This can be used to suppress collection of VVR and NVR information. The VVDS datasets will not be allocated and opened. If no other parameters are specified, then this will result in complete configuration information with the physical dataset extent as the smallest entity, but no information will be obtained on the cluster name and nature of VSAM physical components. This may result in non-informational system-generated VSAM names in your reports.

CAT=YES This can be used to cause the master catalog and all ICF user catalogs pointed to by the master catalog to be dumped. However, all password fields in all records will be overwritten with bytes containing hex EF to prevent security exposures. This is implied by FOCUS=AUDIT.

CAT=NO This can be used to suppress catalog dumps. It is implied by FOCUS=CCWANAL and FOCUS=CONFIG.

PDS=YES This can be used to cause collection of PDS directories of APF, linklist, and LPA list datasets. This is implied by FOCUS=AUDIT if the program is running APF authorized, otherwise it must be selected explicitly (see below).

PDS=NO This can be used to suppress PDS directory dumps. It is implied by FOCUS=CCWANAL and FOCUS=CONFIG and if the program is not running APF authorized (this is to prevent large number of 913-38 abends).

OFFLINE=YES Include UCB type information for offline devices.

SHARED=NO Do not process shared disk devices. The parameter is meant to reduce processing time in (partially) shared DASD configurations where only VTOC and VVDS information is needed. This can be accomplished by running Consul/collect normally on one system and with SHARED=NO on the other systems. Note that the parameter may result in missing catalog or PDS information, since the decision to dump these may be system-dependent.

SLOWDOWN This option can be selected to force the use of VSAM OPEN and GET to read all catalogs. Its main use is to maintain operation in a pre-DFP V3 system without APF authorization, but with ALTER permits on the catalogs.

INFO This can be used to obtain information on the progress of processing. The information is not presented in a structured way, but may be used to get an insight in the amount of parallellism. It is mainly meant for debugging purposes.

FREE=YES This can be used to free the files used dynamically as soon as processing has finished. This is much slower than letting them remain allocated until step termination.

3.2.3 Non-APF options

WAIT=NO This can be used to suppress retry waits used in attempts to collect data on all physical paths to a device. It can be specified if you are only interested in device contents layout (VTOC etc.), to speed up operation.

BURSTS=*nn* This can be used to modify the number of I/O bursts done in an attempt to access a device along all its paths. The default is 20.

BURSTWAIT=*nn* This can be used to modify the number of centiseconds waited between bursts. It should at least allow a reasonable chance that an active I/O has terminated. The default is 50.

BURSTSIZE=*nn* This can be used to modify the number of I/O's in a burst done in an attempt to access a device along all its paths. It should at least be equal to the maximum number of paths to any device. The default is 4.

3.2.4 APF options

3350=NO This can be used to prevent collection of storage director IDs for 3350 devices. In unauthorized state, this will prevent an error message for each 3350 device; in authorized state this will prevent IOS000I/IEA000I messages on the operator console (see also section 4.2 Other Problems).

NOSIO This option can be set to prevent use of the APF authorized I/O driver. This will cause fallback to non-authorized ways of finding paths.

NOBYPASS This option can be set to prevent bypassing dataset security for dumping catalogs and PDS directories.

UNCONNECTED This option can be set to include unconnected catalogs for dumping. Its main use is during volume selections that exclude the master catalog volume. OPEN for unconnected catalogs requires either APF authorization or inclusion of a STEPCAT or JOBCAT DDname for the unconnected catalog.

NOKEY0 This option can be set to prevent use of the APF authorized access to fetch protected control blocks (like the PPT). This will cause the fetch protected information to be missing.

3.3 Reports

This section explains the meaning of the columns on the two reports generated unless suppressed by the NOREPORT option. The first report is the volume report, the second report is the catalog report.

3.3.1 Volume report

CNFCOLL 2.0.0 01/19/91 18.09 C O N S U L / C O L L E C T C O N F I G U R A T I O N C O L L E C T O R page 7
Volume overview

Volume	Dev	VTOC trks	read	DSCB#	VSAM	ISAM	nonVSAM	F3	VVDS trks	read	VVR#	BCS ix	data	slw trks	recs	PDS no	trks	dirblk
CLB001	0C22	310	11	145			141	2										
CICSD0	04CD	310	18	346	128		204	12	75	2	140							
CICSD2	06B3	155	2	101	31		65	3	60	1	34					1	1	5
DSK102	04C7	155	3	83	13		62	6	60	1	15	1	40	2731		6	14	484
DSK103	0C2B	155	4	171			165	4								3	3	51
DSK200	0D7E	310	12	179	3		170	4	75	1	4					3	3	23
DSK201	0C21	310	14	210	3		202	3	75	1	4					4	4	40
DUR001	0D78	155	1	13	1		10		60	1	1							
DUR002	0D7A	155	1	4	1		1		60	1	1							
EMVCAT	0307	310	1	18	10		6		10	1	11	1	10	1259				
EMVSYS	0302	12	4	174	5		164	3	3	1	7	1	1	7		32	47	983
NORM01	04C9	155	7	165	9		139	15	60	1	10	1	34	1486				
NORM03	0D7F	155	19	906	230		668	5	10	3	263	1	20	3355				
NORM04	04CB	155	25	913	68	10	795	15	60	2	99	1	9	808				
PROD01	041C	310	5	213	124	12	51	12	75	2	181	1	52	9362				
PROD02	011C	310	1	37	11	1	20	2	75	1	15	1	141	9969				
PROD03	031B	310	1	11	3		5	1	75	1	4	1	81	5928				
PROD13	0418	310	5	233	226		5		75	2	226							
PROD14	0118	310	7	259	249		3	5	75	3	322							
SIMS01	0407	310	1	6	3		1		75	1	4	1	1	4				
SIMS02	0402	310	3	113	9	1	99	1	75	1	9					2	3	79
SMA001	0301	12	1	12	1		9		75	1	1							
SMC01E	030B	310	1	4	1		1		75	1	1							
SMC01F	0107	310	1	18	14		2		75	1	15	1	12	1248				
SMD89D	0304	12	5	203	9		190	2	3	1	11	1	1	35		1	1	6
SMF001	0109	310	1	40	3		30	5	75	1	3							
SMF002	0309	310	1	38	5		24	7	75	1	5							
SMS001	0305	12	2	92	31		57	2	3	1	43	1	1	37				
SMW001	0306	12	5	232	38		180	12	10	1	44					2	2	39
SPG001	0D76	310	16	165	34		120	9	75	1	39					2	4	114
SPG301	0308	310	3	112	2		107	1	75	1	2							
SPOL00	0300	14	2	3			1											
SPOL01	0101	12	1	4			2											
SYS101	0105	155	33	1706	27		1672	5	60	1	32	1	53	4798		37	54	1141
SYS102	0102	155	10	425	78		341	4	60	2	101	1	10	467		20	21	236
SYS301	0400	310	1	9	3		4		75	1	4	1	1	4		1	1	19
SYS302	0404	310	1	6	3		1		75	1	4	1	1	4				
TCLA01	0311	310	1	4	1		1		75	1	1							
TEST00	06B4	310	17	253	205		42	4	75	4	208	1	41	4395				
TEST01	0D73	310	15	193	168		23		75	3	180							
TEST02	06B6	310	16	250	197		51		75	3	230							
TEST06	06BE	310	18	483	311		162	8	75	4	318							
TSO001	0C24	155	24	638	397		206	33	60	7	399	1	30	3859				
TSO002	0C25	155	13	382	27	4	303	42	60	2	38	1	36	4428				
TSO003	0C26	155	16	348	19	1	318	7	60	2	20	1	21	2699				
TSO004	0C27	155	9	28	12		13	1	60	3	12	1	15	1311				
TSO005	0C28	155	16	279	33	14	210	6	60	2	34	1	21	1367				
TSO006	0C29	155	16	580	132		424	22	60	2	137	1	12	933				
TSO007	0EEE	155	25	356	132		217	5	60	5	169	1	20	1803				
TSO008	0EEF	155	15	392	70	11	283	15	60	2	101	1	18	1350				
TSO009	0C2C	155	14	532	44		469	17	60	1	45	1	1	14				
TSO010	0C2D	155	13	612	74	2	501	31	60	1	90	1	27	1249				
TSO011	06B0	155	10	284	14		255	13	60	1	15	1	37	6430				
total	171	1992	1097	27397	6609	104	19453	771	9888	212	7397	42	847	81128		138	180	3501

Column	*Meaning*
Volume	Volume serial
Dev	Device address
VTOC trks	Number of tracks allocated for VTOC
read	Number of VTOC tracks read
DSCB#	Number of DSCBs selected from VTOC
VSAM	Number of format 1 DSCBs with VSAM dataset organization (components)
ISAM	Number of format 1 DSCBs with ISAM dataset organization
nonVSAM	Number of non-ISAM, non-VSAM format 1 DSCBs
F3	Number of format 3 DSCBs (indicates multiple extents)
VVDS trks	Number of tracks allocated to the VVDS
read	Number of VVDS tracks read
VVR#	Number of NVR and VVR records selected
BCS ix	Number of non-imbedded indices on the volume processed with EXCP
data	Number of catalog data components on the volume processed with EXCP
slw	Number of catalogs on the volume processed with VSAM GET
trks	Number of tracks read from the catalog data components on the volume using EXCP
recs	Number of catalog BCS records selected
PDS no	Number of partitioned dataset directories on the volume processed with EXCP
trks	Number of tracks read from PDS directories on the volume
dirblk	Number of directory blocks read from partitioned datasets on the volume

3.3.2 Catalog report

```
CNFCOLL 2.0.0 01/19/91 18.09 C O N S U L / C O L L E C T   C O N F I G U R A T I O N   C O L L E C T O R                page   11
Catalog overview

Datasetname                          Dev  Volume IX blk  CISZ ext DA blk  CISZ CAtk ext | trks read records | flags
AEP1UCAT                             0EE4 SYAEP1 | 1024  1024   1 | 1024  1024          1 |                    | Unconnected
CSD0UCAT                             04CD CICSD0 | 1024  1024   1 | 1024  1024          1 |                    | Unconnected
DSYCAT.MVS530.MC.EMVCAT              0307 EMVCAT | 1024  1024   1 | 1024  1024    3     1 |   60   10    1259 | Mstr
DSYCAT.MVS530.MC.IPL1PK                   IPL1PK |                |                           |                    | Notfound
DSYCAT.MVS530.MC.SMC01F              0107 SMC01F | 1024  1024   1 | 1024  1024    3     1 |   60   12    1248 |
DSYCAT.MVS530.UC.EMVSYS              0302 EMVSYS |  512   512   1 |10240 20480   14     1 |   30    1       7 |
DSYCAT.MVS530.UC.FMVT02              0100 FMVT02 |  512   512   1 |14336 28672   14     1 |   30    1       7 |
DSYCAT.MVS530.UC.PROD01              041C PROD01 | 4096  4096   1 | 4096  4096   14     1 |  360   52    9362 |
DSYCAT.MVS530.UC.PROD02              011C PROD02 | 4096  4096   1 | 4096  4096   14     1 |  435  141    9969 |
DSYCAT.MVS530.UC.PROD03              031B PROD03 | 4096  4096   1 | 4096  4096   14     1 |  360   81    5928 |
DSYCAT.MVS530.UC.SDSK01              0401 SDSK01 | 4096  4096   1 | 4096  4096   14     1 |   60    1       4 |
DSYCAT.MVS530.UC.SYS301              0400 SYS301 | 4096  4096   1 | 4096  4096   14     1 |  165    1       4 |
DSYCAT.MVS530.UC.SYS302              0404 SYS302 | 4096  4096   1 | 4096  4096   14     1 |  165    1       4 |
DSYCAT.MVS530.UC.SIMS01              0407 SIMS01 | 4096  4096   1 | 4096  4096   14     1 |  120    1       4 |
DSYCAT.MVS530.UC.SMD89D              0304 SMD89D |  512   512   1 |10240 20480   14     1 |   30    1      35 |
DSYCAT.MVS530.UC.SMS001              0305 SMS001 |  512   512   1 |10240 20480   14     1 |   30    1      37 |
DSYCAT.MVS530.UC.TSO301              0411 TSO301 | 4096  4096   1 | 4096  4096   14     1 |  435    1      17 |
OD03UCAT                             06BA PROD03 | 1024  1024   1 | 1024  1024    2     1 |  120    1       9 |
OD05UCAT                             04C5 PROD05 | 1024  1024   1 | 1024  1024          1 |                    | Unconnected
OD10UCAT                             04CE PROD10 | 1024  1024   1 | 1024  1024    2     1 |  120    1       4 |
OD13UCAT                             04CA PROD13 | 1024  1024   1 | 1024  1024    3     1 |  120    1       6 |
OD01UCAT                             0C23 POD901 | 1024  1024   1 | 1024  1024          1 |                    | Unconnected
F101UCAT                             0EE1 SYF101 | 4096  4096   1 | 4096  4096   14     1 |  135   71    7716 |
G001UCAT                             0D76 SPG001 |  512   512   1 | 4096  4096          1 |                    | Unconnected
I001UCAT                             0EEA SYI001 | 1536  1536   1 | 4096  4096          1 |                    | Unconnected
I201UCAT                             0D77 SYI201 | 1024  1024   1 | 1024  1024    2     1 |  120    6     429 |
```

Column	*Meaning*	
Datasetname	Name of the catalog data component	
Dev	Device address of data component	
Volume	Volume serial of data component	
IX blk	Index physical block size (bytes)	
CISZ	Index control interval size (bytes)	
ext	Number of extents for index	
DA blk	Data component physical block size	
CISZ	Data component control interval size (bytes)	
CAtk	Number of tracks per control area for data component	
ext	Number of extents for data component	
trks	Number of tracks in use for data component (based on high used RBA)	
read	Number of tracks read from data component using EXCP	
records	Number of records selected from data component	
flags	Catalog attributes. Reported are:	
	Noimb	Noimbed
	Nospan	Non-spanned data component
	Norepl	Noreplicate
	Unconnected	No connector entry found in master catalog
	Notfound	Volume not mounted
	Mstr	Master catalog

4 Problems and their solution

If CONSUL/COLLECT abends in operation please consult the following section "4.1 Abends" for a list of common abend codes which can be easily corrected by taking the actions indicated.

If your installation includes 3350 devices please see "4.2 Other Problems" for advice on avoiding WTO Buffer Shortages.

4.1 Abends

The most common abend codes encountered with CONSUL/COLLECT are listed below with a suggestion for the possible cause and remedy. Of course your first check should be the appropriate message manual for your operating system, that will tell you the exact meaning of the abend and reason code.

001 Probably problems with blocksize. Look at the message in your joblog to determine the DDname.

002 Problems with the DCB parameters of a file. Look at the message in your job log to determine the DDname. Check your specification for DCB parameters with the reference material in "3.1 Calling JCL".

213-04 This abend may occur for the VTOC on online uninitialized volumes. It may also occur for APF libraries that are not physically present on the volume. CONSUL/COLLECT continues operation with the next volume.

213-20 An abend 213-20 may occur on pre-DFP V3 systems if a catalog has more than 16 extents. CONSUL/COLLECT will intercept the abend and enter slowdown mode (i.e. use VSAM GET running much slower on shared DASD systems).

322 CPU time limit exceeded. Check the joblog for prior abend messages with a different abend code. If a prior abend occurred, solve this abend. Otherwise, increase the TIME parameter on the JOB card, code less functions together, or split the input (e.g. per FOCUS).

522 Check in the joblog that the job was not waiting for a tape mount, offline, inaccessible, or reserved device. In the latter case, you may circumvent problems by excluding the inaccessible volume with an EXCLUDE command.

722 Too many output lines. Make your selection more specific or increase the output limit for your job (for instance with a `/*JOBPARM L=`*nn* card , where *nn* is thousands of lines allowed).

80A
878 GETMAIN error. Try to increase the REGION parameter on the EXEC or JOB card. If you have reached your site's maximum, code less functions together, reduce parallelism (e.g. by specifying PARALLEL=PATHGROUP), or split the input (e.g. per FOCUS or per LCU).

913-0C An abend 913-0C will occur if attempting to open an unconnected ICF catalog without bypassing RACF dataset security. This may happen if a selection is done that excludes the master catalog volume, while specifying CAT=YES. CONSUL/COLLECT will intercept the abend and continue with the next catalog.

913-38 An abend 913-38 will occur if a PDS is opened but you do not have authorization to do so, either through the DATASET profile or through FACILITY $CNF.AUDIT when running authorized.

D37
B37 One of the output datasets was too small, or there was no space left on the volume to extend the dataset. Look at the message in your job log to determine the DDname.

Most abends (except some I/O related abends) are accompanied by a summary dump. When an error occurs that may be caused by a CONSUL/COLLECT problem, you should report the error, along with a copy of the summary dump shown below, the JCL used, and the listing of the input commands to your service representative. The figure below shows a sample SYSPRINT log, for a CNFCOLL run that abended. The abend is indicated with a message and followed by a register dump and a traceback.

CONSUL/COLLECT will print the offset and CSECT name if the error occurs in a CNFCOLL module. It always prints the registers at the time of abend, a traceback with save areas, the MVS version and a processor name, the last IOCONFIG record written succesfully, and the status of some internal control blocks.

```
CNFCOLL 2.0.b 11/23/90 22.44 C O N S U L / C O L L E C T   C O N F I G U R A T I O N   C O L L E C T O R                page    1
 (C) COPYRIGHT 1986-1990 by Hans Schoone and CONSUL Risk Management B.V., Veenweg 112, 2631 RB Nootdorp, The Netherlands

PARM:  FOCUS=ALL,PAR=PATH,REPORT

CNF047I 00 Data collection started on 15 Jan 1991 15:49 for node DXAPST2  sysname IPO1     sid IPO1 netid DXAPS000
CNF039I 00 Running MVS/SP2.2.3  DFP 3.1.1  JES2 SP 2.2.0  VTAM 3.2
CNF111I 00 Scheduler allocated 10 I/O executors
CNF017I 04 Path 13 to 3380 device 0345 SYSR22 not operational
CNF017I 04 Path 20 to 3380 device 0694 USER10 not operational
Abend PSW 070C1000 81C4FA4C 00020001 00000000 code 322000
Registers at entry to abend:
R0  00000001 R1  00ABDE34 R2  00ABDE34 R3  00AC8960
R4  00ABDE38 R5  81C4EF7C R6  00ABDE34 R7  7FFF9890
R8  81C4EF78 R9  7FFF9AF5 R10 01C4FD6D R11 7FFF9B14
R12 81C4ED6E R13 7FFF9B14 R14 81C4EF7C R15 80AFE7D8
Save area at 0000C3B0 for procedure at 80006CE8 CNFCOLL 2.0.b 11/23/90 22.44 (C) COPYRIGHT 1 (not returned yet) R14 offset 0980
 WD1 00000000 HSA 00005F90 LSA 00022F30 R14 80007668 R15 000125C0 R0  00220029
 R1  00000000 R2  00000089 R3  00005508 R4  00005508 R5  00AFB8A8 R6  00AC9FF8
 R7  FD000000 R8  00AFB0F8 R9  0000BFB0 R10 00007CE8 R11 80006CE8 R12 0000BFB0
Save area at 00022F30 for procedure at 000125C0 CNFSCHED CNFCOLL 2.0.b 11/23/90 22.23 (C) CO (not returned yet)
 WD1 00000000 HSA 0000C3B0 LSA 0001F6F0 R14 70012B70 R15 0001E928 R0  00000000
 R1  00000008 R2  00000028 R3  000B6C20 R4  000B9CD0 R5  000B6D50 R6  00000000
 R7  FD000000 R8  00AFB0F8 R9  0000BFB0 R10 000BC978 R11 000125C0 R12 0000BFB0
Save area at 0001F6F0 for procedure at 0001E928 CNFVTOC CNFCOLL 2.0.b 11/23/90 22.30 (C) COP (not returned yet) R14 offset 0949
 WD1 00000000 HSA 00022F30 LSA 00008F60 R14 0001F271 R15 00008E9C R0  00000099
 R1  0008723A R2  00000028 R3  000B7950 R4  000B9CD0 R5  000B6D50 R6  00000000
 R7  000001BA R8  00AFB0F8 R9  000EE008 R10 000BC978 R11 0001E928 R12 0000BFB0
Save area at 00008F60 for procedure at 00008E9C CNFCOLL.PUT31 (returned)
 WD1 00000000 HSA 0001F6F0 LSA 00000000 R14 00000000 R15 00000000 R0  40008F1E
 R1  92D51360 R2  00000099 R3  0008723A R4  00000028 R5  000B7950 R6  000B9CD0
 R7  000B6D50 R8  00000000 R9  000001BA R10 00AFB0F8 R11 40D51484 R12 00D50D78
CNF031I 00 IOCONFIG runs on IPO1 with MVS/SP2.2.3, CPU model 3081
CNF031I 00 Last record written: ID=31, contents start 00010002 012C0060 E2E8E2F2

PGRP 000B9CD0 pathgroup 01FFFFFFFFFFFFFF online min=1 max=1
PGRP 000B9668 pathgroup 0222FFFFFFFFFFFF online min=2 max=2
PGRP 000B77D8 pathgroup 0310FFFFFFFFFFFF online min=2 max=2
PGRP 000B7918 pathgroup 062620FFFFFFFFFF online min=2 max=2
PGRP 000B6F60 pathgroup 0714FFFFFFFFFFFF online min=2 max=2
PGRP 000B85B0 pathgroup 1323FFFFFFFFFFFF online min=1 max=1

IOXC 000BD2F0 0345 SYSR22 pathgroup 1323FFFFFFFFFFFF state=VVDSRDY
IOXC 000BD978 0D23 EDAS03 pathgroup 0714FFFFFFFFFFFF state=CATRDY
IOXC 000BC2F0 0D22 EDAS02 pathgroup 0714FFFFFFFFFFFF state=CATRDY
IOXC 000BC978 0694 USER10 pathgroup 062620FFFFFFFFFF state=VTOC     <- current
IOXC 000BB2F0             state=DELETE
IOXC 000BB978 0927 ADMI12 pathgroup 0310FFFFFFFFFFFF state=VTOC
IOXC 000AA2E8 0926 ADMI11 pathgroup 0310FFFFFFFFFFFF state=VTOC
IOXC 000AA970 028F DASD10 pathgroup 0222FFFFFFFFFFFF state=ALLOCED
IOXC 000412E8 028E USER44 pathgroup 0222FFFFFFFFFFFF state=ALLOCED
```

Figure 1. *Sample CNFCOLL summary dump for a 322 abend.*

4.2 Other Problems

If you have many 3350 devices and you are running CONSUL/COLLECT authorized, then you may experience a *WTO buffer shortage* if your WTO buffer limit is set too low or your console mode and routing codes are set in a dangerous combination. Generally, you can recover from this situation by:

1. Make the limit higher than the amount of buffers currently used. This can be done by a D C,B command (display console backlog) and a K M,MLIM=*nnn* command (set a new WTO message buffer limit).
2. If there are any slow (hardcopy) devices with a long backlog, reroute the messages to a roll-delete device (for instance the master console) or to the hardcopy log (SYSLOG). This can be done by the command K Q,R=*mm*,L=*bb* where *bb* is the console id of the console with backlog and *mm* is HC for the SYSLOG or the console id for the master console.
3. If there are display consoles with long queues, check whether they have message deletion mode DEL=N. If they have, change the mode to DEL=RD. If these measures are still not successful, you may have to specify DEL=Y or DEL=R or reroute the message queue just like the hardcopy device.

These actions are taken automatically if you have installed the IBM SolutionPac® for Automated Console Operation with Netview.

The maximum number of messages generated by CONSUL/COLLECT is:

#3350 * burstsize * # bursts

where #3350 is the number of 3350 devices, burstsize is the burstsize set (default is 4), and # bursts is the maximum number of bursts (1 if WAIT=NO specified, 20 if not). You can compare this number against your WTO buffer limit (issue the K M command on the master console). You may also adjust the burstsize and the number of bursts used by CONSUL/COLLECT by specifying the BURSTSIZE and BURSTS parameter, respectively.

5 Messages

The messages issued by CONSUL/COLLECT have a message identifier of the form CNF*nnn*I where *nnn* is the message number. Behind the message identifier, a severity code is indicated. The program returns as completion code the highest severity code encountered. The general meaning of the severity codes and hence of the completion code is as follows:

00 Normal message, giving status or summary information.

04 Unusual condition found that may or may not result in missing information.

08 Unusual condition found that causes information that was requested to be missing. Subsequent processing may be impacted.

12 Unexpected condition during CONSUL/COLLECT processing.

16 Syntax error in command input.

24 Internal error or other unexpected and unsupported condition in CONSUL/COLLECT detected.

In the rest of this section, all error messages are listed with an explanation and possible actions to take.

5.1 CONSUL/COLLECT Messages

CNF000I 04 Control block *name* omitted, because of *reason*

This message is issued if the program fails to find an OS control block. This is not necessarily a problem, rather it notes the absence of some information which might have been useful, but which may not be available in your OS version at all. For instance, XA specific control blocks (like IOCH, IODN, CMCT, LLT and DCQ) will not be present in MVS/370, neither will the reverse (like CSTE, LCH). The name of the control block is given by *name*, the control block ID. The exact nature of the failure is given by *reason*, which may be:

invalid block id	The control block ID is not found in its proper place.
protection exception	A protection exception occurred during the walk through the pointer chain leading to the control block.
invalid length	A protection exception occurred during access to the last-to-be-used byte of the control block.
nil pointer	The pointer to the control block was found to contain binary zero.

This message may very well occur after conversion to a new release of the OS. The resulting configuration file may still be usable for your purpose.

Problems indicated with missing control blocks *names* include the following:

STGS	RMF is not active
EDT	device type information not retrieved
IODN	LCU and device number table missing (also RMF)

CNF001I 04 No generic unit name for *devclass* unit *dev*

This message indicates that your OS could not give a generic unit name for the device on address *dev*, and the device type is also not available in the hardcoded device table in CONSUL/COLLECT. The device class is *devclass*. This is not a problem, but warns you to expect question marks in the unit name fields.

CNF002I 04 LOCATE return code *rc* on LPA list dataset *datasetname*

This message indicates that the dataset in the LPA List Table called *datasetname* could not be found by the LOCATE service of MVS. The return code returned by the service is *rc*. The volume will be left blank or zero in the configuration file.

CNF003I 04 DEVTYPE RC nonzero for unit *dev*

The DEVTYPE SVC used to collect information on unit *dev* returned a nonzero return code. This may cause the device type record in the configuration file to be unusable.

CNF004I 16 Syntax error in input parameter at text "*parameter*"

This message indicates the program does not understand the parameter you specified. The parameter being processed at the time of the failure is given as *parameter* (first characters). Review "3.2 Parameters" for the syntax of the parameters. The program will stop processing after parsing the input parameters and not collect any configuration information.

CNF005I Please ignore CMD rejects

This message is displayed on the operator console to warn the operator that no action should be taken on the burst of IOS000I or IEA000I messages specifying a CMD reject on 3350 DASD devices. It is removed immediately after the program has finished processing the 3350 range of devices. It is displayed during authorized operation only.

CNF006I 12 CVAFDIR *type* **error, R15=***rc*, **CVSTAT=***code* **on device** *dev* **volume** *volume*

During access to the VTOC index, the CVAFDIR *type* (READ or RLSE) service returned a nonzero return code *rc* accompanied by CVAF return code *code*. See the appropriate IBM manual for the meaning of these codes. If the type of access was READ, the VTOC is read completely without taking into account the used DSCB map in the VTOC index.

CNF007I 00 Task is not APF authorized - only non-protected information can be collected

This message alerts you to the fact that the program could not obtain authorization. This issue is discussed in "2.3 Authorized or Unauthorized Execution?".

CNF008I 00 Number of DASD devices interrogated: *nn*

This message gives the number of devices that have been allocated and interrogated.

CNF009I 00 Number of DSCB entries copied: *nn*

This message gives the number of Dataset Control Blocks copied from VTOCs to the configuration file. It is somewhat larger than the number of datasets on the interrogated devices, because some extents are described in separate DSCBs for the same dataset. Note that only used DSCBs are copied.

CNF010I 00 Number of VVDS datasets processed: *nn*

This message gives the number of VVDS datasets for which an OPEN was attempted. Generally, this number is smaller than the number of DASD devices interrogated, because not every volume needs to have a VVDS.

CNF011I 00 Number of NVR/VVR entries copied: *nn*

This messages gives the number of VVRs (VSAM volume records) copied to the configuration file. This is roughly two times the number of VSAM datasets on the processed volumes.

CNF012I 12 Non-4K block size for VVDS not supported - volume *volume*

This message indicates a VVDS was encountered on volume *volume* with a blocksize other than 4K. This is not supported by this release of CONSUL/COLLECT. The VVDS has a 4k blocksize if it has been made automatically on 3330/3350/3380/3390 DASD with at least DFP 1.0 through DFP 3.2. If you encounter this message, then the VVDS information for the specified volume will not be read, and you will only see component names mentioned in the VTOC, not the cluster names.

CNF013I 12 EXCP failed on VVDS RC=*nn*, **IOBSEEK=***XXBBBBBCCCCHHHH*, **device** *dev* **volume** *volser*

This message indicates an unsuccessful I/O operation on a VVDS. This probably indicates an internal error in CONSUL/COLLECT. Please contact your service representative if you want something to be done about it. Your configuration file will probably miss some VVDS information, resulting in missing cluster names.

CNF014I 04 Device *dev volume* **online but not ready**

This message indicates the device number *dev* with volume serial *volume* was included in the configuration because it was online, but could not be interrogated because it was not ready. Instead of scheduling an I/O request, CONSUL/COLLECT has skipped the device. This may result in incomplete information for your purpose.

CNF015I 08 SYSEVENT DONTSWAP failed, return code hex *rc*

This message indicates that CONSUL/COLLECT failed to make itself nonswappable. As a result, no authorized I/Os can be scheduled and no cache size information and device level cache disablement information will be collected for 3880 devices. Neither will guaranteed device path I/O be used to eliminate WAITs.

CNF017I 04 Path *ch* **to** *type* **device** *dev volume* **not operational**

This message indicates that the installed physical channel (pre-XA) or channel path (XA) *ch* to the selected online and ready device number *dev* with volume serial *volume* was not operational. If this is not your normal working configuration, then you are measuring a reduced configuration with a higher contention than normal. Alternatively this may point at running MVS/370 under VM.

CNF018I 12 *parameter* **Parameter invalid in non-XA system.**

The *parameter* specified is not applicable to pre-XA systems.

CNF019I 12 BFLHFCHN invalid for *type* **device** *dev volser,* **VTOC processing skipped**

The forward chain pointer of next buffer list (BFLHFCHN) is not valid; i.e. no (more) VTOC information could be obtained for device *dev.*

CNF020I 08 Path information not gotten for unsupported device type *type,* **device** *dev volume*

This message indicates that you requested configuration information for a device type *type,* which is not currently supported by CONSUL/COLLECT. Requests for support for other DASD types than 3350, 3380, 3390, and compatibles should be directed to your service representative.

CNF021I 08 Storage director IDs unavailable for *type* **device** *dev volume* **because unauthorized**

This message indicates that physical storage director IDs for device number *dev* with volume serial *volume* can only be extracted by authorized programs because its device type is *type.* The result is missing storage director information which may prevent an automatic deduction of the configuration.

CNF022I 08 Storage director ID not returned by IOS for path *ch* **to** *type* **device** *dev volume*

This message indicates that the IOS version you have fails to return the complete sense information needed to find the storage director ID. This is, for instance, the case with 3350s in MVS/SP1.3.4, and may prevent automatic deduction of the configuration layout. The failure occurred on path *ch* to device number *dev* with volume serial *volume.* The device type is *type.* This message is issued for only one path, because CONSUL/COLLECT assumes the same failure will occur on the other paths to the device, and does not attempt I/O on these paths.

CNF023I 08 String controller ID not returned by IOS for path *ch* **to** *type* **device** *dev volume*
This message indicates that the controller ID was not found in its proper place. This message is not issued if the storage director ID is also missing. Currently no software level is known which omits only controller information. Because of redundancy of information, this will probably not have any effect on the ability to infer the configuration from the configuration file.

CNF024I 08 Path information still incomplete after *bs* **tries on** *type* **device** *dev volume*: **missing at least path** *ch*
This message indicates that after *bs* tries CONSUL/COLLECT still did not succeed in scheduling I/O along all paths to a device. This message only occurs if you specified or implied WAIT=NO and PATH=YES. The resulting configuration information is incomplete, and probably the configuration cannot be inferred from the configuration file.

CNF025I 08 Path information still incomplete after *bn bs*-**try bursts on** *type* **device** *dev volume*: **missing at least path** *ch*
This message indicates that after *bn* bursts of *bs* tries with a 0.5 second WAIT interval between the bursts, CONSUL/COLLECT still did not succeed in scheduling I/O along all paths to a device. This may happen on very busy shared DASD systems and on very empty pre-XA systems that don't have channel rotation. The number of bursts, burstsize, and inter-burst wait time can be adjusted by the appropriate BURST*xxxx* parameters.

CNF026I 12 Unexpected IOS return code *rc* **hex, CSW status** *hhhh* **sense** *ssss* **on path** *ch* **to** *type* **device** *dev volume*
This message indicates an unexpected error during EXCP processing. The IOS return codes are documented in the IBM debugging handbooks (IOB/IOSB) and in the appropriate DFP manuals. The resulting configuration file will be probably be incomplete.

CNF027I 04 Invalid DSCB FMTID=X'*xx***' on** *type* **device** *dev* **volume** *volser*
CCHHR=*0000000000* **DSN=***dsname*
The VTOC for the indicated volume contained an invalid DSCB, with format X'*xx*'. The only valid types are X'F0' .. X'F6'. The DSCB record is included in the configuration file, but not used by CONSUL/CCW. The *dsname* reported is the datasetname field (key area) of the DSCB in error. This has no consequences for MVS if the the DSCB is not in use according to the space map.

CNF028I 08 DAIRFAIL code *xxxx xxxx* **on** *dev volser*
This message will be followed by an IKJ-message on the problem. The error occurred in dynamic allocation or unallocation of a VTOC for device *dev*.

CNF029I 04 Device *dev volser* **online, but not mounted**
Device *dev* was not mounted public, storage or private, CONSUL/COLLECT does not attempt to allocate the VTOC and VVDS datasets.

CNF030I 08 OPEN abend *xxx-rc* **on device** *dev* **volume** *volser* **for** *dsname*
The dataset named *dsname* could not be opened for input on device *dev*. The VTOC is indicated with ** VTOC volser **. If the error occurs for a VTOC, both the VTOC and the VVDS for the volume will be missing. If the error occurs for a VVDS, the VTOC information has been read properly. See section "4.1 Abends" for a discussion of common abend codes.

CNF031I 00 CNFCOLL runs on *sid* with MVS/*version* CPU model *code*
CONSUL/COLLECT abended while running on the indicated system (SMF id) and MVS release level.

CNF031I 00 Last record written: ID=*hh*, contents start *hexstring*
CONSUL/COLLECT abended after writing the indicated record.

CNF032I 08 Number of record(s) truncated: *nn*
This message indicates that records were truncated on output. You might try increasing the record length, if problems arise. However, for most purposes the information needed is located at the beginning of the BCS records, and these truncated records do not therefore usually present a problem.

CNF033I 08 Module IGG019X1 missing, no configuration info for 3350 devices possible.
This message indicates that the appendage IGG019X1 could not be found.

CNF034I 00 CNFCOLL used *ss.t* CPU seconds, *ss* elapsed seconds, and collected *m.kkk* Mb (*m.kkk* Mb/s)
This message details the TCB time used as well as the wall clock time. In addition, the amount of data collected is summarized as well as the effective data rate.

CNF035I 00 Number of PDS directories copied: *nn*
This messages gives the number of PDS (Partition Data Set) directories copied to the configuration file.

CNF039I 00 Running MVS/*version* DFP *version* JES2 *version* VTAM *version* RMF *status*
This message indicates release levels or status of the software that CONSUL/COLLECT extracts information from.

CNF041I 00 Number of ICF catalogs processed: *nn*
This message gives the number of ICF catalogs for which an OPEN was attempted.

CNF042I 00 Number of BCS records copied: *nn*
This messages gives the number of BCS (Basic Catalog Structure) records copied to the configuration file.

CNF043I 08 VVDS information not collected, catalogs cannot be dumped
This message is issued if the VVDS datasets could not be accessed, but catalog processing was requested. CONSUL/COLLECT requires VVDS access to dump catalogs.

CNF044I 12 Name of master catalog not found in CAXWA. Abend 913-0C may result for unconnected catalogs
This message indicates that the master catalog name could not be determined. Consequently, it is impossible to determine which catalogs are connected. CONSUL/COLLECT will try to open all catalogs it encounters on the disks processed. This will result in abend 913-0C for each unconnected catalog.

CNF045I 08 Master catalog volume *volume* not selected. Abend 913-0C may result for unconnected catalogs.

This message indicates that the master catalog was not found on any of the disk volumes processed. Hence no user catalog connector information is accumulated by CONSUL/COLLECT. CONSUL/COLLECT will try to open all catalogs it encounters on the disks processed. This will result in abend 913-0C for each unconnected catalog.

CNF046I 04 Catalog cannot be dumped because not connected *volume catname*

This message indicates that no user catalog connector was found in the master catalog for the indicated user catalog. No OPEN will be attempted to prevent an 913-0C abend.

CNF047I 00 Data collection started on *date time* for mode *nodename* sysname *sysname* sid *smfid* netid *netid*
MVSCP conguration id *xx* LPAR *LPARname*

This message indicates the timestamp marking the start of data collection. It can be used to find the proper CONSUL/COLLECT output when presented with a specific configuration file. In addition, the various sytem identifiers are listed: the JES2 node name, the GRS system name, the SMF id, and the VTAM netid. On the second line, optional configuration information may be present to indicate the MVSCP configuration id and the Logical Partition name.

CNF048I 08 ACB OPEN failed for catalog *catalogname* rc=*nn*, code=*nn*

This message indicates a failure to open the catalog indicated and gives the return code and reason code. ACB OPENs are attempted only if the catalog has been defined with NOIMBED, if it has more than 16 extents on a pre-DFP V3 system, or if the run is unauthorized in a pre-DFP V3 system.

CNF049I 24 Internal error RDTRACK RC=16

Please report this error to your service representative.

CNF050I 00 TTR Conversion routined fails on TTTTTTRR

This informational message may occur when INFO is specifed, and indicates the end of a dataset was reached.

CNF051I 12 EXCP failed on *ddname*, RC=*hh*, IOBSEEK=*address* device *dev volser*

This message indicates an unexpected I/O failure on the indicated device and address. The return code is the EXCP return code in hex.

CNF052I 08 CI size for index of *catname* on *volume* (*value*) greater than 8K, skipped.

The current version of CONSUL/COLLECT does not support CI sizes for a catalog index larger than 8K. You can call your support representative for a fix to obtain a higher limit.

CNF053I 00 Slowdown mode invoked because not APF-authorized, catalog *catname*

This message indicates that the requested catalog dump will be tried with VSAM, because the faster EXCP mode requires APF authorization that is not present. ALTER authority is required to read the catalogs without APF authorization.

CNF054I 12 Catalog *catname* CA at rel track *tt* missing *nn* CIs in sequence set record
This message indicates that the number of CIs described by the index sequence set record was not the number of CIs per CA. If the error message is reproducible, perform EXAMINE on the catalog. If no strange things are found, please report this error to your service representative.

CNF055I 08 ACB OPEN abend *uuusss* for catalog *dev volume catalogname*
This message indicates an abend during an attempt to open the catalog indicated.

CNF056I 00 Slowdown mode invoked because more than 16 extents for *catname*
This message indicates that the catalog will be processed by VSAM instead by EXCP. In a shared DASD environment, this may perform 10 to 20 times slower (depending on the CI size).

CNF057I 08 Abend *uuusss* on *dev volser dsname*
A non-recoverable abend occurred opening dataset *dsname* for input on device *dev*. If the error occurs for a VTOC, the VTOC and all datasets on the volume will be missing. If the error occurs for a VVDS, the VTOC information has been read properly. See section "4.1 Abends" for a discussion of common abend codes.

CNF058I 12 Unexpected physical record length *decnum* in imbedded SSR with index blksize *decnum* for *catname*
This message indicates that a physical record (i.e. block) was read from the imbedded index track with a blocksize different from the blocksize indicated in the information in the VVR. Results will be unpredictable.

CNF059I 08 NOIMBED not supported, catalog *catname* on *volser*
This message indicates that for some reason the index was not read successfully. Consequently, the NOIMBED catalog cannot be processed.

CNF060I 12 VVDS space map extension at RBA *hexnum* ignored - expecting *hexnum*
CONSUL/COLLECT expects the space map chain to occur in order in the VVDS.

CNF061I 04 VVDS can only be accessed with APF authorization
In DFP V3 systems, APF authorization is required to read the VVDS.

CNF062I 08 Connected catalog *catname* not found on volumes processed
The master catalog processed contained a connector entry for catalog *catname*. However, the catalog was not found on the volumes processed. Catalog information may be incomplete.

CNF063I 08 Unexpected error: Master cat BCS not found on mastercat volume. Abend 913-0C may occur.
This message indicates that for some reason the master catalog was not found on the volume it was supposed to reside on. Consequently, it cannot be determined whether user catalogs are connected or not. Abend 913-0C results from trying to open an unconnected catalog if bypass-password processing is not being used.

CNF070I 24 Abend *sssuuu* on *dev volume dataset*
This message indicates a non-recoverable abend occurred during OPEN of the indicated PDS. See section "4.1 Abends" for a discussion of common abend codes.

CNF071I 08 Requested dataset not found or volume excluded - *volume dsname*
This message indicates that the indicated dataset could not be processed because it was not found in any of the VTOCs processed.

CNF072I 12 Unexpected IOCINFO return code *rc* reason code *rr* (decimal)
This message indicates that the IOCINFO service issued an unexpected return code. Results are unpredictable.

CNF073I 04 Dynamic configuration change occurred, UCBSCAN restarted - file may contain duplicate records
This message indicates that the UCBSCAN service indicated a configuration change while scanning all UCBs. The scan will be restarted, but this may make the configuration file unusable if your application does not support duplicate information. In this case, you will have to rerun the Consul/Collect step.

CNF074I 12 Unexpected UCBSCAN return code *rc* reason code *rr* (decimal)
This message indicates that the UCBSCAN service issued an unexpected return code.

CNF075I 04 Unexpected EDTINFO return code *rc* reason code *rr* (decimal) for *dev volume*
This message indicates that the EDTINFO service issued an unexpected return code while trying to obtain the generic device type for a device. The field will be filled with a default value.

CNF076I 12 Unexpected UCBSCAN return code *rc* reason code *rr* (decimal) on *dev volume*
This message indicates that the UCBSCAN service issued an unexpected return code when trying to obtain the last path used mask.

CNF077I 12 Unexpected UCBSCAN return code *rc* reason code *rr* (decimal) on *dev volume*
This message indicates that the UCBSCAN service issued an unexpected return code while trying to pin and obtain the address of a UCB. The intended authorized I/O function will not be performed.

CNF078I 12 Unexpected UCBPIN UNPIN rc *rc* reason code *rr* (decimal) on *dev volume*
This message indicates that the UCBPIN service issued an unexpected return code while trying to unpin an UCB after an authorized I/O operation.

CNF080I 04 Unexpected IXCQUERY return code *rc* reason code *rr* (decimal)
This message indicates that the IXCQUERY service issued an unexpected return code. The XCF sysplex record will be missing from the file.

CNF081I 04 Unexpected IXCQUERY return code *rc* reason code *rr* (decimal)
This message indicates that the IXCQUERY service issued an unexpected return code. The XCF sysplex record will be missing from the file.

CNF082I 04 Unexpected IXCQUERY abend *xxxxx*
This message indicates that the IXCQUERY service abended. The XCF sysplex record will be missing from the file.

CNF101I 00 Allocating *catname* on *volser*
This informational message is issued to indicate start of processing for the specified catalog. It is included if the INFO option is selected.

CNF102I 00 BCS *catname* on *volume* BLK *decnum* CISZ *decnum*, CASZ *decnum* bytes, *num* blk/CA, *num* trk/CA, *nn* CI/trk

This informational message gives the control interval size, the number of bytes, blocks, and tracks in a control area, and the number of control intervals per track for the specified catalog immediately before it is opened. It is issued only if the INFO option is selected.

CNF103I 00 No imbed - index *indexnam* BLK *decnum* CISZ *decnum*

This message indicates that the catalog about to be opened has the NOIMBED attribute, which makes it necessary to process the index. The message indicates the index component dataset name, as well as the physical blocksize and CI size. It is issued only if the INFO option is selected.

CNF104I 00 Closed IX *dev volume catname* **index incore *decnum* bytes**

This informational message summarizes the number of bytes that were read from the catalog index component prior to closing. It is issued only if the INFO option was selected.

CNF105I 00 Opened BCS *catname volume* size *num* tracks

This informational message contains the number of tracks in the catalog data component that has just been opened successfully. It is issued only if the INFO option was selected.

CNF106I 00 Master catalog is *catname*

This informational message indicates the name of the master catalog. It is issued only if the INFO option was selected.

CNF107I 00 Opened ACB *dev volume catname*

This informational message indicates the successful opening of the ACB for the indicated catalog. It is issued only if the INFO option was selected.

CNF108I 00 Closed ACB *dev volume catname* copied *decnum* records

This informational message indicates the number of records read from the indicated catalog. It is issued only if the INFO option was selected.

CNF109I 00 Opened PDS *dev volume dsname*

This informational message indicates the successful opening of the PDS indicated. It is issued only if the INFO option was selected.

CNF110I 00 Closed PDS *dev volume dsname* read *decnum* **tracks, copied *decnum* directory blocks**

This informational message indicates the number of directory tracks and blocks read from the indicated PDS. It is issued only if the INFO option was selected.

CNF111I 00 Scheduler allocated *decnum* I/O executors

This informational message shows the amount of parallellism introduced by the PARALLEL parameter or its default. It is issued only if the REPORT or INFO option was selected or defaulted.

CNF112I 00 Opened VTOC *dev volume* size *decnum* tracks

This informational message indicates the successful opening of the VTOC for the indicated volume. It is issued only if the INFO option was selected.

CNF113I 00 Closed VTOC *dev volume* **read *num* tracks, copied *decnum* DSCBs**

This informational message summarizes the number of tracks and records that were read from the VTOC prior to closing. It is issued only if the INFO option was selected.

CNF114I 00 Opened SYS1.VVDS.V*volume* size *decnum* tracks
This informational message indicates the successful opening of the indicated VVDS. It is issued only if the INFO option was selected.

CNF115I 00 Closed SYS1.VVDS.V*volume* read *num* tracks, copied *decnum* NVR/VVRs
This informational message summarizes the number of tracks and records that were read from the VVDS prior to closing. It is issued only if the INFO option was selected.

CNF116I 00 Closed BCS *catname volume* read *num* tracks, copied *decnum* records
This informational message summarizes the number of tracks and records that were read from the catalog data component prior to closing. It is issued only if the INFO option was selected.

CNF200I 12 Resource profile does not permit use of FOCUS=CCWANAL - FACILITY $CNF.CCWANAL
This message indicates that the user has insufficient authority on the indicated profile (RACF return code 8).

CNF201I 12 Access denied to one or more APF authorized features - adjust FOCUS or drop APF authorization.
This message indicates that the user has insufficient authority on the proper FACILITY profile. He either has to change the requested function, obtain a READ permit to the proper $CNF.*focus* profile, or drop APF authoriztion (e.g. by adding a non-authorized STEPLIB).

CNF202I 12 Resource profile does not permit use of FOCUS=AUDIT - FACILITY $CNF.AUDIT
This message indicates that the user has insufficient authority on the indicated profile (RACF return code 8).

CNF203I 12 Resource profile does not permit use of FOCUS=CONFIG - FACILITY $CNF.CONFIG
This message indicates that the user has insufficient authority on the indicated profile (RACF return code 8).

CNF204I 00 Resource not defined - FACILITY *profile*
This message indicates that the indicated profile cannot be found (RACF return code 4 while class is active). Access will be allowed for FOCUS=CCWANAL and FOCUS=CONFIG. For FOCUS=AUDIT, message CNF211I will follow.

CNF205I 00 RACF return code *nnnnnnnn* hex, reason code *nnnnnnnn* hex FACILITY *profile*
This message indicates the RACF return code and reason code returned in the first two fullwords of the RACROUTE REQUEST=AUTH parameter list by SAF. Generally, the meaning will be explained in additional messages, or, for return code 8, in an ICH408I message issued by RACF in the joblog. This message is mainly meant for debugging purposes. The meaning of the reason codes is documented in the proper RACF documentation.

CNF206I 00 RACF not installed, no authorization check possible
This message indicates that no resource access control is present on the system. All operations requested will be allowed.

CNF207I 00 RACF inactive, no authorization check possible
This message indicates that no resource access control is active on the system. All operations requested will be allowed.

CNF208I 00 RACF class FACILITY not defined in CDT, no authorization check possible
This message indicates that the resource class indicated is not defined in the Class Descriptor Table of RACF. All operations requested will be allowed.

CNF209I 00 RACF class FACILITY not active, no authorization check possible
This message indicates that protection for the resource class indicated has not been activated on the system. For FOCUS=CCWANAL and FOCUS=CONFIG, access will be allowed. For FOCUS=AUDIT, message CNF210I will be issued.

CNF210I 12 RACF authorization checking for class FACILITY must be active to use FOCUS=AUDIT
This message explains that CONSUL/COLLECT will refuse to collect auditing information for a user unless this is specifically allowed by a FACILITY profile. To be able to check the profile, the FACILITY class must be activated.

CNF211I 12 Resource profile must be present to use FOCUS=AUDIT - FACILITY $CNF.AUDIT
This message explains that CONSUL/COLLECT will refuse to collect auditing information for a user unless this is specifically allowed by a FACILITY profile. To be able to check the profile, the FACILITY class must be activated.

CNF80*x*I 00 *message*
These messages are issued as the result of debugging commands.

CNF900I 24 Internal error: CNFALLOC called in invalid state
This message indicates a serious internal error. User abend 900 will be issued. Please report this message to your service representative.

CNF901I 24 CNFAXCP: data areas too large, on *dev volume*
This message indicates an internal error: the data areas requested by the channel program passed to the I/O driver are too large. User abend 901 will be issued. Please report this message to your service representative.

CNF902I 24 I/O routine abend *sssuuu* **on** *dev volume*
This message indicates an internal error: the data areas requested by the channel program passed to the I/O driver are too large. User abend 902 will be issued. Please report this message to your service representative.

CNF903I 24 CNFCAT called in invalid state
This message indicates a serious internal error. User abend 903 will be issued. Please report this message to your service representative.

CNF904I 24 Internal error: BCS on wrong IOXC
This message indicates a serious internal error. The catalog will be skipped. Please report this message to your service representative.

CNF905I 24 Internal error: CNFPATH called in invalid state
This message indicates a serious internal error. User abend 905 will be issued. Please report this message to your service representative.

CNF906I 24 CNFPDS called in invalid state
This message indicates a serious internal error. User abend 906 will be issued. Please report this message to your service representative.

CNF907I 24 Internal error: PDS on wrong IOXC
This message indicates a serious internal error. The PDS will be skipped. Please report this message to your service representative.

CNF908I 24 CNFSCHED Internal error: hung I/O executor
This message indicates a serious internal error. A user abend 908 will be issued. Please report this message to your service representative.

CNF909I 24 Internal error: CNFVTOC called in invalid state
This message indicates a serious internal error. User abend 909 will be issued. Please report this message to your service representative.

CNF910I 24 Internal error: CNFVVDS called in invalid state
This message indicates a serious internal error. User abend 910 will be issued. Please report this message to your service representative.

CNF981I 12 Invalid *type* "*value*"
This message indicates that the text *value* is not a valid value in the context *type*.

CNF982I 12 Internal error: unknown error code at *ddname* line *number*
The input parser error routine encountered an invalid error code. Please report this error to your service representative.

CNF983I 12 Expecting list separator or terminator instead of *type* "*value*" at *ddname* line *number*
This message indicates that the input parser expected a list separator or terminator for the current list (this can for instance be a comma, blank, or end-of-line, depending on the context). Instead, it encountered the indicated token type *type* (and text *value*, if available). The input parser skips all input until it encounters a valid list separator or terminator for the current list.

CNF984I 12 Invalid list element type *type* "*value*" at *ddname* line *number*
This message indicates that the input parser expected a list element, but found a token of a type not supported as a list element in this context. If available, the offending text *value* is also listed in the message. The input parser skips all input until it encounters a valid list separator or terminator for the current list.

CNF985I 12 Required list element/parameter "*value*" missing at *ddname* line *number*
This message indicates that the input parser detected a missing required parameter or element in the list at the indicated line.

CNF986I 12 Duplicate parameter *value* at *ddname* line *number*
This message indicates that the input parser detected a duplicate occurrence of the parameter or list element *value* at the indicated line.

CNF987I 12 Syntax error: *type1* expected instead of *type2* at "*value*" on *ddname* line *number*
This message indicates that the input parser expected a specific token type *type1* in the current context. Instead of this, it found the token type *type2* (at the text *value*, if available) on the indicated input line.

CNF988I 12 Syntax error: "*c*" expected instead of *type* at "*value*" on *ddname* line *number*
This message indicates that the input parser expected a specific character "*c*" (presumably a delimiter) in the current context. Instead of this, it found the token type *type* (at the text *value*, if available) on the indicated input line.

CNF989I 12 Unexpected *type* "*value*" at *ddname* line *number*
This message indicates that the input parser expected any one of a number of specific token types, but found a different token type instead. If available, the offending text *value* is also listed in the message.

CNF991I 04 ESTAE return code *rc*
This message indicates that the program failed to establish an abend exit linkage.

CNF993I DIAGNOSTIC DUMP SUPPRESSED FOR ABEND *xxx*
This message indicates that the program abend exit did not attempt to make a diagnostic summary dump. This is done to prevent recursive abend conditions involving the print file.

CNF995I LRECL INVALID; NOT OVERRULED BECAUSE PARTITIONED
This message indicates that the print file open routine detected an invalid record length for the output file. This would have been overruled with a correct length for a Physical Sequential dataset, but this is not done for Partitioned datasets to prevent making any existing PDS members inaccessible. Subsequent 013 or 002 abends may be caused by the invalid record length.

CNF999I 16 GETMAIN FAILED FOR HEAP *name* - INCREASE REGION
This message indicates that the program needs more storage. If the heap name is **LOWHEAP**, then the request is for storage below the 16MB line.

Appendix A Installation

The CONSUL/COLLECT program is distributed in the same library as the package it accompanies. It ready for unauthorized operation immediately. For authorized operation, the following steps need to be taken:

1. Copy the CNFCOLL and IGG019X1 modules to an APF library that can only be read by you or the systems programming department for testing purposes.

2. Establish RACF or equivalent profiles used through the SAF (System Authorization Facility) in the class FACILITY. Only give permits to people on a need-to-use basis. The three resources checked are: $CNF.CCWANAL, $CNF.CONFIG, and $CNF.AUDIT. See section "2.3 Authorized or Unauthorized Execution?" for a discussion of the meaning of these profiles. The following is a sample RACF command sequence to define one of the profiles:

```
RDEFINE FACILITY $CNF.AUDIT
PERMIT $CNF.AUDIT CLASS(FACILITY) ID(user)
```

   Please check that class FACILITY has been activated. The class can be activated by the command

```
SETROPTS CLASSACT(FACILITY)
```

4. Run the program from the APF library using any test procedures you have for newly installed authorized programs (on a test system or during test time).

5. If the tests have run satisfactorily, copy CNFCOLL and IGG019X1 to an APF library that can be read by the people who have to run the program.

Index

CONSUL
risk management b.v.

document number
GC14-5288-01