

# Nigel Pentland's RACF Utilities

Documentation for v2.06

[www.racf.co.uk](http://www.racf.co.uk)

11 August 2007



# Acknowledgements

I'd like to acknowledge the support and encouragement I've received from various folks at various stages throughout the development an evolution of these utilities. In particular there are several who I'd like to mention by name and publicly thank them for their support.

Doc Farmer  
Mike Cairns  
Jeffery Loewenstein  
Mark Wilson  
Ulrich Boche

# Contents

<b>Acknowledgements</b> .....	<b>2</b>
<b>Contents</b> .....	<b>2</b>
<b>Introduction</b> .....	<b>5</b>
Background.....	5
Usage.....	5
Documentation structure.....	5
Overview.....	6
Other Complementary PC tools.....	7
Notes for existing users.....	7
Sample JCL.....	8
<b>One Line Summary</b> .....	<b>10</b>
<b>Miscellaneous</b> .....	<b>13</b>
racf00.....	13
racf01.....	13
racf02.....	13
racf11.....	14
racf21.....	14
racf32.....	15
racf42.....	15
racf52.....	15
racf58.....	16
racf59.....	16
racf89.....	16
racf118.....	17
racf124.....	17
racf128.....	17
racfawk.....	18
racfdiag.....	18
racfjcl.....	19
<b>Groups</b> .....	<b>20</b>
racf03.....	20
racf04.....	20
racf06.....	20
racf36.....	21
racf47.....	21
racf50.....	21
racf53.....	22
racf69.....	22
racf70.....	23
racf73.....	23
racf75.....	23
racf76.....	24
racf77.....	24
racf87.....	24
racf104.....	25
racf111.....	25
racf113.....	25
racf117.....	26
racf121.....	26
racf126.....	26
racf127.....	27
racf129.....	27
<b>Users</b> .....	<b>28</b>

racf05	28
racf08	28
racf09	28
racf18	29
racf24	29
racf38	29
racf46	30
racf66	30
racf79	31
racf82	31
racf84	31
racf86	32
racf88	32
racf92	32
racf93	33
racf102	33
racf103	33
racf105	34
racf106	34
racf110	35
racf122	35
racf130	35
racf131	36
<b>Datasets</b>	<b>37</b>
racf07	37
racf23	37
racf56	37
racf61	38
racf68	38
racf71	39
racf80	39
racf90	39
racf107	40
racf115	40
<b>General Resources</b>	<b>42</b>
racf12	42
racf16	42
racf19	42
racf20	43
racf22	43
racf25	43
racf28	44
racf30	44
racf33	44
racf34	45
racf35	45
racf37	45
racf48	46
racf49	46
racf51	47
racf62	47
racf64	47
racf65	48
racf67	48
racf72	48
racf85	49
racf91	49
racf114	49
racf116	50
racf123	50
racf125	50
<b>Certificates</b>	<b>52</b>
racf94	52
racf95	52
racf96	52
racf97	52
racf98	53
racf99	53
racf100	53
racf101	54
racf108	55
racf109	55
racf112	56

racf119 .....	57
racf120 .....	58
<b>Appendix A - Usage .....</b>	<b>59</b>
<b>Index.....</b>	<b>61</b>

# Introduction

## ***Background***

In 1995 I started to learn and work with RACF. The mainframe was new to me, but programming wasn't, hence I used my experience to build on. This meant trying to (a) learn about how to do RACF security, and (b) to understand the profiles that had already been defined in the RACF databases I started to work with.

This led me to develop and evolve a set of reporting tools, which also in some cases generated JCL which could be edited if required, and used to automate tedious, error prone tasks. In the early days of version 0 these reporting tools were all DOS based and produced fixed pitch formatted text reports.

Things moved on with version 1 in 2000, when the text reports were for the most part replaced with html reports giving many advantages and improvements. I'm now writing this documentation for version 2 in 2007, which represents another leap forward in terms of file handling, style of JCL produced, error reporting, and all reports now run as low priority tasks so they can run unobtrusively in the background allowing you to continue to do other tasks unimpeded (note: if there are any rogue processes which are burning up CPU cycles this means the utilities may appear to stall, as I found with iexplore.exe during testing).

## ***Usage***

All of the utilities will now accept any of the following arguments as an instruction to simply display usage information and not do any other processing:

```
-h  
/h  
-help  
/help  
-?  
/?
```

## ***Documentation structure***

Previously I just listed them sequentially in order. This made them hard to find the best utility for any given task. After much thought, I have opted this time for core of the documentation to consist of 6 headings under which the relevant utilities are listed sequentially. In addition to the Contents section, I have also added an index section to hopefully make things easier to lookup and find when required.

The 7 sections are:

- One Line Summary
- Miscellaneous
- Groups
- Users
- Datasets
- General Resources
- Certificates

The One Line Summary section has been re-instated by request. This is intended as an alternative way of finding utilities, especially ones you have used in the past but just can't quite remember the number.

Miscellaneous is for any utility which doesn't fit in only one of the other categories, e.g. a summary of all records, or an ID access cross reference which checks groups, datasets and general resources.

The Miscellaneous section tends to be rich in adding value, whereas the remainder tend more to be trusty tools for the administrator or auditor who knows exactly what they want.

Certificates are by rights covered by the General Resources category, but this is a sufficiently specialised, significant and different area, with its own unique challenges that I decided to give them their own section.

## Overview

The idea of these utilities is to take a flatfile download from the mainframe onto the PC and then process it to produce reports and or JCL. The JCL can be checked and when you are satisfied it will do what you want you then upload it onto the mainframe and submit it as with any other JCL.

To get started you will need to get the flat file download from RACF, one or more of these utilities, and a suitable **RACF.INI** file. I have supplied sample JCL at the end for obtaining the flat file from RACF although this is well documented in the IBM manuals. The utilities can be downloaded from my home page much as with this readme.htm file. The home page also has a sample **RACF.INI** file which should be downloaded. Note: its good practice to always run **RACF00.exe** after receiving an unload, before running any other reports. This only needs to be done once as it will update the unload file if required.

The sample **RACF.INI** file is expecting to find the flat file input in the sub-directory prod and with a filename of racfprod. The output refers to the prefix on the output files. For example utility **RACF02.exe** will write its output to file **???02.HTM** and **???02.JCL** where **???** is the output string defined. Hence the sample RACF.INI defines the output files as **RACF02.HTM** and **RACF02.JCL**.

When transferring the flat file to PC the ASCII and CR/LF options should be selected so that the PC file is a conventional ASCII text file. Note: it is important that this file is the raw output from the unload process and must not be sorted, otherwise the resultant reports will be very unpredictable!

Essentially, I keep all my programs in a '**RACF**' directory. Then under that in sub-directories called **DEV** and **PROD** I copy the flat file. Also residing in **DEV** and **PROD** are the **RACF.INI** files. The main purpose of these is to tell the programs which flat file to look at, although they contain some other self explanatory stuff as well.

I have my **..\RACF** directory defined in my **PATH** statement and simply change directory into the desired directory, i.e. **DEV** or **PROD** and the utilities look for an ini file in the current directory.

**PATH %PATH%;C:/DATA/RACF;C:/DATA/RACF/DEV;C:/DATA/RACF/PROD;**

- This is the **PATH** command to cover **DEV** and **PROD**

**C:\DATA\RACF\RACF\*.EXE**

- directory included in **PATH**
- This is also the directory used for comparison utilities, namely **RACF35** and **RACF40**

**C:\DATA\RACF\DEV\RACF.INI**

**C:\DATA\RACF\DEV\RACFDEV**

**C:\DATA\RACF\DEV\RACF\*.HTM**

**C:\DATA\RACF\DEV\RACF\*.JCL**

- ini file for **DEV**
- downloaded ASCII flat file for **DEV**
- generated html reports for **DEV**
- generated JCL for **DEV**

**C:\DATA\RACF\PROD\RACF.INI**

**C:\DATA\RACF\PROD\RACFPROD**

**C:\DATA\RACF\PROD\RACF\*.HTM**

**C:\DATA\RACF\PROD\RACF\*.JCL**

- ini file for **PROD**
- downloaded ASCII flat file for **PROD**
- generated html reports for **PROD**
- generated JCL for **PROD**

## ***Other Complementary PC tools***

In my opinion the most indispensable tool when working with text files in Windows, is Notepad++ (Notepad plus plus). This is a quite superb replacement for the relatively basic Windows Notepad program. Please investigate it at <http://notepad-plus.sourceforge.net/> You can even use the Settings / Preferences / File Association option to associate all '.jcl' files with Notepad++, and that's just the start. Then you might want to go to Settings / Preferences / Edit components and set 'Show Vertical Edge' at column 72. This is just such an easy to use, rich editor.

<http://unxutils.sourceforge.net/> is also an excellent selection of utilities. The one in particular here that has been invaluable for me has been gsr (Global Search And Replace) which allows you to a search and replace across multiple files.

Don't forget about Microsoft Excel. This can be used to open any html file and it will correctly interpret the columns and formatting. Very useful if you want a report sorted by an alternate column.

Last but not least are my own accompaniments, namely my DOSutils <http://www.racf.co.uk/dosutils.zip> (especially ts.exe for text search) and report <http://www.racf.co.uk/report.htm> (for selecting vertical columns based on a mask).

## **Notes for existing users**

Please ensure you have all sections defined within your racf.ini. The latest section to be added was for **RACF117**. If you don't have this section, please update using the latest version.

Also, if generating reporting for multiple SSIDs, why not amend either or both the output file name parameters to reflect this? E.g. instead of the default 'racf' change to 'SSID-racf'

Last but not least, if you just want to look up a utility quickly via the One Line Summary section, just click on the Chess image on the front page.

## Sample JCL

Sample JCL to copy and then unload RACF database to a sequential flat file.

```
//EXPORT JOB ((2331)),
// 'NIGEL',
// CLASS=A,
// MSGCLASS=X,
// MSGLEVEL=(1,1),
// NOTIFY=&SYSUID,
// TIME=1440
//*
//* CREATE SEQUENTIAL FLATFILE FROM THE RACF BACKUP DATABASE
//*
//UNLOAD EXEC PGM=IRRDBU00,PARM=NOLOCKINPUT
//SYSPRINT DD SYSOUT=A,COPIES=1,DEST=U1018
//INDD1 DD DSN=SYS1.RACF.DEVA.BACKUP,DISP=SHR
//OUTDD DD DSN=username.RACF.FLATFILE,DISP=SHR
//
```

Sample JCL to obtain **DSMON** output in a DATASET for downloading.  
(This is used in utility **RACF68** to identify APF authorised libraries.)

```
//DSMON JOB , 'DSMON', CLASS=F, MSGCLASS=X,
// NOTIFY=&SYSUID
//*
//DSMON EXEC PGM=ICHDSM00
//SYSPRINT DD SYSOUT=*
//SYSUT2 DD DSN=username.DSMON, DISP=SHR
//SYSIN DD *
FUNCTION ALL
//
```

Sample JCL to obtain **LISTCAT** input for **RACF107**. First stage is to obtain a list of user CATALOGs.

```
//RACFDSN1 JOB , 'LIST MASTCAT' , CLASS=A,
// REGION=0M, NOTIFY=&SYSUID
//*
//STEP01 EXEC PGM=IKJEFT01, REGION=25M
//SYSTSPRT DD DSN=xxxxxxx.CATLIST,
// DISP=(NEW, CATLG, DELETE) ,
// UNIT=3390,
// SPACE=(CYL, (100, 100), RLSE) ,
// LRECL=133, BLKSIZE=27930, RECFM=FB
//SYSOUT DD SYSOUT=*
//SYSTSIN DD *
PROFILE NOPREFIX
LISTCAT CATALOG('SYS1.xxxxxx.MASTCAT') UCAT
/*
```

Second stage is to list out all of those user CATALOGs.

```
//RACFDSN2 JOB , 'LIST CATALOGS' , CLASS=A,
// REGION=0M, NOTIFY=&SYSUID
//*
//STEP01 EXEC PGM=IKJEFT01, REGION=25M
//SYSTSPRT DD DSN=xxxxxxx.DSLIST,
// DISP=(NEW, CATLG, DELETE) ,
// UNIT=3390,
// SPACE=(CYL, (200, 200), RLSE) ,
// LRECL=133, BLKSIZE=27930, RECFM=FB
//SYSOUT DD SYSOUT=*
//SYSTSIN DD *
PROFILE NOPREFIX
LISTCAT CATALOG('SYS1.xxxxxxx') ALL
LISTCAT CATALOG('SYS1.xxxxxxx.xxxxxx') ALL
LISTCAT CATALOG('SYS1.xxxxxxx.xxxxxx') ALL
/*
```

# One Line Summary

(Courtesy of Doc Farmer)

Utility	Description	H T M L	J C L	T E X T	U S E R	G R P	D S N	G E N R	C E R T	M I S C
<a href="#">RACF00</a>	IRRD00 Flatfile Pre-Processor (Removes Binary Chars)									<input checked="" type="checkbox"/>
<a href="#">RACF01</a>	Summary of Unloaded Records (Totals By Record Type)									<input checked="" type="checkbox"/>
<a href="#">RACF02</a>	Profiles Owned By Non-Existent IDs or Groups									<input checked="" type="checkbox"/>
<a href="#">RACF03</a>	Group Tree Text Report					<input checked="" type="checkbox"/>				
<a href="#">RACF04</a>	List Groups With Installation Data And Create Date					<input checked="" type="checkbox"/>				
<a href="#">RACF05</a>	List Expired User IDs - Generate JCL To Delete				<input checked="" type="checkbox"/>					
<a href="#">RACF06</a>	List Group = ??? (With JCL To Remove Users)					<input checked="" type="checkbox"/>				
<a href="#">RACF07</a>	Recursive HLO Access Report With DSN Text File						<input checked="" type="checkbox"/>			
<a href="#">RACF08</a>	Search For User ID Using Specified String				<input checked="" type="checkbox"/>					
<a href="#">RACF09</a>	List All User IDs Starting With...				<input checked="" type="checkbox"/>					
RACF10	-----EMPTY-----									
<a href="#">RACF11</a>	Enhanced User ID Cross-Reference With JCL To Recreate				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
<a href="#">RACF12</a>	Access With User ID & Name for General Resource Profile							<input checked="" type="checkbox"/>		
RACF13	-----EMPTY-----									
RACF14	-----EMPTY-----									
RACF15	-----EMPTY-----									
<a href="#">RACF16</a>	List Profiles In Member/Group Class Pair							<input checked="" type="checkbox"/>		
RACF17	-----EMPTY-----									
<a href="#">RACF18</a>	Generate Data File For Fast Searches Of User IDs				<input checked="" type="checkbox"/>					
<a href="#">RACF19</a>	List General Resource Profiles With Access Lists							<input checked="" type="checkbox"/>		
<a href="#">RACF20</a>	List General Resource Profiles With Access Lists/Prefix (JCL)							<input checked="" type="checkbox"/>		
<a href="#">RACF21</a>	Enhanced User ID Cross-Reference With JCL To Remove				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>
<a href="#">RACF22</a>	List Class Profiles With Installation Data							<input checked="" type="checkbox"/>		
<a href="#">RACF23</a>	List Dataset Profiles With Prefix = ???						<input checked="" type="checkbox"/>			
<a href="#">RACF24</a>	List Revoked User IDs				<input checked="" type="checkbox"/>					
<a href="#">RACF25</a>	List General Resource Profiles With Access Lists/Prefix (JCL to Delete)							<input checked="" type="checkbox"/>		
RACF26	-----EMPTY-----									
RACF27	-----EMPTY-----									
<a href="#">RACF28</a>	List CICS Profiles With Transaction Prefix							<input checked="" type="checkbox"/>		
RACF29	-----EMPTY-----									
<a href="#">RACF30</a>	List All STARTED Profiles With STDATA							<input checked="" type="checkbox"/>		
RACF31	-----EMPTY-----									
<a href="#">RACF32</a>	List All Profiles With WARNING Attribute									<input checked="" type="checkbox"/>
<a href="#">RACF33</a>	List All CICS Profiles For Default CICS Class TCICSTRN							<input checked="" type="checkbox"/>		
<a href="#">RACF34</a>	List Duplicate Transaction Entries for Class TCICSTRN							<input checked="" type="checkbox"/>		
<a href="#">RACF35</a>	Compare General Resource Profiles With Specified Prefix							<input checked="" type="checkbox"/>		
<a href="#">RACF36</a>	Compare Two Groups And List User IDs Found In Both					<input checked="" type="checkbox"/>				
<a href="#">RACF37</a>	List General Resource Members With Specified ID/Group in Access List							<input checked="" type="checkbox"/>		
<a href="#">RACF38</a>	Generate Audit Report On Higher User IDs				<input checked="" type="checkbox"/>					
RACF39	-----EMPTY-----									
RACF40	-----EMPTY-----									
RACF41	-----EMPTY-----									
<a href="#">RACF42</a>	List All Discrete Profiles with ALTER									<input checked="" type="checkbox"/>

Utility	Description	H T M L	J C L	T E X T	U S E R	G R P	D S N	G E N R	C E R T	M I S C
RACF43	-----EMPTY-----									
RACF44	-----EMPTY-----									
RACF45	-----EMPTY-----									
<a href="#">RACF46</a>	Generate JCL To Remove List Of User IDs				<input checked="" type="checkbox"/>					
<a href="#">RACF47</a>	List Group With JCL To Change Owner/DFLTGRP To New Group					<input checked="" type="checkbox"/>				
<a href="#">RACF48</a>	List General Resource Profiles With Access Lists/Non-Prefix (JCL)							<input checked="" type="checkbox"/>		
<a href="#">RACF49</a>	List General Resource Profiles With Access Lists/Non-Prefix (DEL)							<input checked="" type="checkbox"/>		
<a href="#">RACF50</a>	List User IDs In Group With DFLTGRP & All Connected Groups					<input checked="" type="checkbox"/>				
<a href="#">RACF51</a>	List Prefixed General Resource Profile Pairs In Friendly Format							<input checked="" type="checkbox"/>		
<a href="#">RACF52</a>	List All Profiles/Access Lists For OwnerID									<input checked="" type="checkbox"/>
<a href="#">RACF53</a>	List Group With CICS Details					<input checked="" type="checkbox"/>				
RACF54	-----EMPTY-----									
RACF55	-----EMPTY-----									
<a href="#">RACF56</a>	List Datasets/JESSPOOLS with UACC > NONE						<input checked="" type="checkbox"/>			
RACF57	-----EMPTY-----									
<a href="#">RACF58</a>	List All Profiles With NOTIFY Set (With JCL)									<input checked="" type="checkbox"/>
<a href="#">RACF59</a>	List Profiles With Non-Default Audit Attribute									<input checked="" type="checkbox"/>
RACF60	-----EMPTY-----									
<a href="#">RACF61</a>	Grant Access To Some Dataset Profiles With Prefix & User ID						<input checked="" type="checkbox"/>			
<a href="#">RACF62</a>	Grant Access To Some General Resource Profiles With Prefix & User ID							<input checked="" type="checkbox"/>		
RACF63	-----EMPTY-----									
<a href="#">RACF64</a>	List General Resource Profiles With Access Lists/Prefix (JCL)							<input checked="" type="checkbox"/>		
<a href="#">RACF65</a>	List General Resource Class							<input checked="" type="checkbox"/>		
<a href="#">RACF66</a>	ListUser Command In Print For Specific ID				<input checked="" type="checkbox"/>					
<a href="#">RACF67</a>	Set Or Clear Notify On All Profiles In Prefixed General Resource						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
<a href="#">RACF68</a>	List APF Libraries Without Fully Qualified DSN Profiles						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
<a href="#">RACF69</a>	List All Revoked Group Connections (With JCL)					<input checked="" type="checkbox"/>				
<a href="#">RACF70</a>	Count The Number Of Users In A Group					<input checked="" type="checkbox"/>				
<a href="#">RACF71</a>	List All General Resource Profile Pairs With Access Lists						<input checked="" type="checkbox"/>			
<a href="#">RACF72</a>	Summary Breakdown of General Resource Profiles							<input checked="" type="checkbox"/>		
<a href="#">RACF73</a>	List Group With Names And All Other Group Connections					<input checked="" type="checkbox"/>				
RACF74	-----EMPTY-----									
<a href="#">RACF75</a>	List Group With Only User ID & Group On Report					<input checked="" type="checkbox"/>				
<a href="#">RACF76</a>	List Group With User ID, Name, Connect Owner, Authority					<input checked="" type="checkbox"/>				
<a href="#">RACF77</a>	List All Groups Where Connect Not Owned					<input checked="" type="checkbox"/>				
RACF78	-----EMPTY-----									
<a href="#">RACF79</a>	List User Where OWNER Not Equal To DFLTGRP				<input checked="" type="checkbox"/>					
<a href="#">RACF80</a>	List Discrete Dataset Profiles With JCL To Make Generic						<input checked="" type="checkbox"/>			
RACF81	-----EMPTY-----									
<a href="#">RACF82</a>	Annotate List Of User IDs				<input checked="" type="checkbox"/>					
RACF83	-----EMPTY-----									
<a href="#">RACF84</a>	List User IDs Which Are All Numeric				<input checked="" type="checkbox"/>					
<a href="#">RACF85</a>	List General Resource Profiles With Access Lists/Prefix (JCL)							<input checked="" type="checkbox"/>		
<a href="#">RACF86</a>	Create JCL To Delete User IDs Still Owning DSN Profiles				<input checked="" type="checkbox"/>					
<a href="#">RACF87</a>	GID Group Listing					<input checked="" type="checkbox"/>				
<a href="#">RACF88</a>	UID User Listing				<input checked="" type="checkbox"/>					

Utility	Description	H T M L	J C L	T E X T	U S E R	G R P	D S N	G E N R	C E R T	M I S C
<a href="#">RACF89</a>	List All Profiles With SECURITY In Installation Data									<input checked="" type="checkbox"/>
<a href="#">RACF90</a>	List All Dataset Profiles With JCL To Rebuild						<input checked="" type="checkbox"/>			
<a href="#">RACF91</a>	List All General Resource Profiles With Access Lists.							<input checked="" type="checkbox"/>		
<a href="#">RACF92</a>	List All User IDs With Revoked & Other Data				<input checked="" type="checkbox"/>					
<a href="#">RACF93</a>	List CICS Users Showing Any OPIDs				<input checked="" type="checkbox"/>					
<a href="#">RACF94</a>	List Digital Certificates By User ID & Label								<input checked="" type="checkbox"/>	
<a href="#">RACF95</a>	List All Digital Certificates (Unsorted)								<input checked="" type="checkbox"/>	
<a href="#">RACF96</a>	List All Digital Certificate Key Rings								<input checked="" type="checkbox"/>	
<a href="#">RACF97</a>	List All Digital Certificate Mappings								<input checked="" type="checkbox"/>	
<a href="#">RACF98</a>	List All Digital Certificate Trusts								<input checked="" type="checkbox"/>	
<a href="#">RACF99</a>	List All Digital Certificates Sorted By Expire Date								<input checked="" type="checkbox"/>	
<a href="#">RACF100</a>	List All Digital Certificates Sorted With Cut-off Date								<input checked="" type="checkbox"/>	
<a href="#">RACF101</a>	List All Digital Certificates Sorted By Expire Month								<input checked="" type="checkbox"/>	
<a href="#">RACF102</a>	Annotate List Of User IDs				<input checked="" type="checkbox"/>					
<a href="#">RACF103</a>	List User IDs With Duplicate Name Data				<input checked="" type="checkbox"/>					
<a href="#">RACF104</a>	Group Tree Text Output for Visio Input					<input checked="" type="checkbox"/>				
<a href="#">RACF105</a>	List STARTED User IDs & Check PROTECTED				<input checked="" type="checkbox"/>					
<a href="#">RACF106</a>	List UID(0) User IDs & Check PROTECTED				<input checked="" type="checkbox"/>					
<a href="#">RACF107</a>	Find Catalogues Without Dataset Profiles						<input checked="" type="checkbox"/>			
<a href="#">RACF108</a>	List of Digital Certificates By Owner User With JCL								<input checked="" type="checkbox"/>	
<a href="#">RACF109</a>	Digital Certificate Search (Up To Nine Search Strings)								<input checked="" type="checkbox"/>	
<a href="#">RACF110</a>	List All User IDs With TSO Segment by TSOPROC				<input checked="" type="checkbox"/>					
<a href="#">RACF111</a>	List Group for Sarbanes-Oxley Auditors					<input checked="" type="checkbox"/>				
<a href="#">RACF112</a>	List Digital Certificate Sorted By Certificate Auth								<input checked="" type="checkbox"/>	
<a href="#">RACF113</a>	Annotate A List Of Groups					<input checked="" type="checkbox"/>				
<a href="#">RACF114</a>	Reports On CDT Class							<input checked="" type="checkbox"/>		
<a href="#">RACF115</a>	List Dataset Profiles For Supplied Text File						<input checked="" type="checkbox"/>			
<a href="#">RACF116</a>	List General Resource Profiles For Supplied Text File							<input checked="" type="checkbox"/>		
<a href="#">RACF117</a>	Produce Summary Statistics For Groups					<input checked="" type="checkbox"/>				
<a href="#">RACF118</a>	List Discrete Profile With Generic Characters									<input checked="" type="checkbox"/>
<a href="#">RACF119</a>	Generate Digital Certificate JCL using RADCERT								<input checked="" type="checkbox"/>	
<a href="#">RACF120</a>	Report Digital Certificate Common Names From RACF119								<input checked="" type="checkbox"/>	
<a href="#">RACF121</a>	List Groups Showing Users That Never Logged On					<input checked="" type="checkbox"/>				
<a href="#">RACF122</a>	List Users Never Signed On & Not PROTECTED				<input checked="" type="checkbox"/>					
<a href="#">RACF123</a>	List General Resource Profiles/Members For Supplied Text File							<input checked="" type="checkbox"/>		
<a href="#">RACF124</a>	Generate Text Files of User IDs, Groups, DSN Profiles, GenRes Profiles				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
<a href="#">RACF125</a>	Generate Text File listing of Profiles in Class pair with UACC not NONE							<input checked="" type="checkbox"/>		
<a href="#">RACF126</a>	Summary report of groups referenced as DFLTGRPs in base user segments					<input checked="" type="checkbox"/>				
<a href="#">RACF127</a>	Summary report of groups containing Group-Special connections					<input checked="" type="checkbox"/>				
<a href="#">RACF128</a>	List details of SUBMIT profiles in SURROGAT class for SYSTEM SPECIAL									<input checked="" type="checkbox"/>
<a href="#">RACF129</a>	Multiple reports of users by DFLTGRP					<input checked="" type="checkbox"/>				
<a href="#">RACF130</a>	List all users, indicating TSO, Protected and Last Access				<input checked="" type="checkbox"/>					
<a href="#">RACF131</a>	List all users with Netview segments				<input checked="" type="checkbox"/>					
<a href="#">RACFAWK</a>	General Purpose Extract Tool									<input checked="" type="checkbox"/>
<a href="#">RACFDIAG</a>	IRRD00 Flatfile Diagnostic (Check For File Errors)									<input checked="" type="checkbox"/>
<a href="#">RACFJCL</a>	Build JCL Around RACF Commands (Similar To CLIST Option)									<input checked="" type="checkbox"/>

# Miscellaneous

## racf00

Description      Extract File Pre-processor - this should be run to ensure that no unprintable binary characters exist within the unloaded file. This can happen because of certificates or due to incorrect character mapping during ftp file transfer. As the rest of the utilities expect a pure text input data if any binary characters exist this can cause unpredictable behaviour. It also checks to see if the final CR/LF is missing and if so appends a CR/LF to fix.

Usage              racf00

Output

Keywords

Examples          C:\data\racf\dev\>racf00  
Checking to ensure unload hasn't been sorted...  
Unload appears unsorted as it should be, continuing...  
Compiling racf.tmp...  
Deleting racfdev  
Renaming racf.tmp as racfdev  
Number of bytes processed = 34050483  
Number of bytes altered    = 0  
If you require to know more about what is being altered then run RACFDIAG before running RACF00

C:\data\racf\dev\>

## racf01

Description      This produces a summary listing showing totals for each type of unloaded record.

Usage              racf01

Output             html

Keywords          summary

Examples          C:\data\racf\dev\>racf01  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling statistics...  
Parsing HTML...  
Removing temporary files...

The following output files have been created:  
    racf01.htm - HTML of summary of records

C:\data\racf\dev\>

## racf02

Description      This produces a report of profiles owned by non-existent IDs (both Groups and Users). Also generates JCL to remove redundant entries in access control lists and help with general RACF housekeeping. This utility has certain parallels with IBM's IRRRID00, but pre-dates it. It's not as comprehensive as IRRRID00, but the JCL produced is ready to use without requiring editing, and is simpler for routine ACL maintenance than IRRRID00.

Usage              racf02

Output             html jcl

Keywords          orphaned

## Examples

```
C:\data\racf\dev>racf02
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Searching for invalid access list entries...
Building access list array...
Completed building access list array...
Now processing DATASET profiles...
Completed processing DATASET profiles...
Now processing General Resource profiles...
Completed processing General Resource profiles...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf02.jcl - JCL to clean up ACLs
  racf02.htm - HTML of orphaned owners
  racf02     - raw extract of orphaned records

C:\data\racf\dev>
```

## racf11

### Description

Enhanced cross reference which produces (a) a report showing all access control list entries for a User ID or Group, and also produces JCL to reproduce. This JCL can be easily edited to replace the ID and use it to clone the access.

If a user, the User section will report:

- Name
- Default-Group
- Any Class Authorisations
- Any Attributes
- If password never expires
- And summary details from user segments

See also racf37 for enumerating members combined with access, e.g. GCICSTRN

### Usage

racf11 ID [all]

### Output

html jcl

### Keywords

XREF

### Examples

```
C:\data\racf\dev>racf11 ibmuser
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Preprocessing...
Compiling list of Access...
Do you wish to enumerate the 4 connected groups? [Y/N/?]
Enumerating access for user  IBMUSER  ...
Do you wish to enumerate the access for group SYSCTLG  ? [Y/N]
Do you wish to enumerate the access for group SYS1     ? [Y/N]
Enumerating access for group SYS1     ...
Do you wish to enumerate the access for group VSAMDSET ? [Y/N]
Do you wish to enumerate the access for group WORLD   ? [Y/N]
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf11.jcl - JCL to create ACL entries for WORLD
  racf11.htm - HTML list ACL entries for WORLD
  racf11-IBMUSER.htm - copy of racf11.htm

C:\data\racf\dev>
```

## racf21

### Description

List access of a User ID or Group. Similar to RACF11 but creates JCL to remove the access. Mainly used where a user has done a lot of RACF administration and their User ID has appeared on profiles which they have created in the days before this could be suppressed by SETROPTS and they didn't remove themselves from the access list.

### Usage

racf21 ID

### Output

html jcl

Keywords XREF

Examples C:\data\racf\dev>racf21 ibmuser  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list of Access...  
 Parsing HTML...  
 Parsing JCL...  
 Removing temporary files...  
  
 The following output files have been created:  
   racf21.jcl - JCL to remove all IBMUSER access  
   racf21.htm - HTML list of IBMUSER access  
   racf21-IBMUSER.htm - copy of racf21.htm  
  
 C:\data\racf\dev>

## racf32

Description List all profiles with WARNING attribute set.

Usage racf32

Output html

Keywords WARNING

Examples C:\data\racf\dev>racf32  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list of profiles with WARNING...  
 Parsing HTML...  
 Removing temporary files...  
  
 The following output files have been created:  
   racf32.htm - HTML list of profiles in WARNING mode  
  
 C:\data\racf\dev>

## racf42

Description List all discrete profiles with ALTER access. This is useful because normally if a user has ALTER access to a discrete profile this is bad as it means they have full administration access using the PERMIT command. This is unlike with generic profiles where it simply means the user has ALTER access to the resources.

Note: this looks at both DATASET profiles and General Resource profiles.

Usage racf42

Output html

Keywords discrete

Examples C:\data\racf\dev>racf42  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Processing discrete profile list...  
 Parsing HTML...  
 Removing temporary files...  
  
 The following output files have been created:  
   racf42.htm - HTML list of discrete profiles with ALTER access  
  
 C:\data\racf\dev>

## racf52

Description List all profiles and access lists belonging to OwnerID  
 Note: all irrcerta (i.e. CERTAUTH) digital certificates are owned by IBMUSER.  
 Used to just list Groups, Dataset profile and General Resource profiles. Has now been extended to include both Users and User connections.

Usage racf52 OwnerID

Output html

Keywords owner

Examples C:\data\racf\dev>racf52 ibmuser  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list of profiles for IBMUSER...  
 Parsing HTML...  
 Removing temporary files...  
  
 The following output files have been created:  
     racf52.jcl - JCL to which can be edited as suits  
     racf52.htm - HTML list of all profiles owned by IBMUSER  
     racf52-IBMUSER.htm - copy of racf52.htm  
  
 C:\data\racf\dev>

## racf58

Description List either all profiles with notify set, or just those with a specific User ID set to notify. JCL is produced for removing notify from listed profiles.

Usage racf58 Notify\_UserID | ALL

Output html jcl

Keywords notify

Examples C:\data\racf\dev>racf58 ibmuser  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list of profiles with NOTIFY...  
 Parsing HTML...  
 Parsing JCL...  
 Removing temporary files...  
  
 The following output files have been created:  
     racf58.jcl - JCL to remove NOTIFY from profiles  
     racf58.htm - HTML list of profiles with NOTIFY  
  
 C:\data\racf\dev>

## racf59

Description List profiles with non-default audit attributes set. Default audit attributes are defined as violation on READ or greater and no global audit attributes set. Gives a good picture of what is being effectively audited.

Usage racf59

Output html

Keywords audit-attributes globalaudit

Examples C:\data\racf\dev>racf59  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list of profiles with non-default audit attributes...  
 Parsing HTML...  
 Removing temporary files...  
  
 The following output files have been created:  
     racf59.htm - HTML list of profiles with non default audit  
  
 C:\data\racf\dev>

## racf89

Description List all profiles and access lists where the string 'SECURITY' is found in the installation data field. (i.e. groups, datasets and general resources, ignores users)

Usage racf89

Output html

Keywords security installation-data

Examples

```
C:\data\racf\dev>racf89
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of Access...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf89.htm - HTML list of 'SECURITY' profiles

C:\data\racf\dev>
```

## racf118

Description List discrete profiles containing generic characters (and hence invalid). I have to credit Mark Nelson for this one, as it was as a result of an example Mark gave during one of his presentations that gave me the idea for this one.

Usage racf118

Output html

Keywords discrete generics

Examples

```
C:\data\racf\dev>racf118
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of discrete profiles containing generic characters...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf118.jcl - JCL to delete and recreate invalid profiles
  racf118.htm - HTML list of discrete profiles containing generics

C:\data\racf\dev>
```

## racf124

Description To generate 4 text files with records, one per line, of all users, groups, datasets, and general resources. Note: all spaces are removed expect a single space between class and general resource profile name.

Usage racf124

Output text

Keywords

Examples

```
C:\data\racf\dev>racf124
Validating unload file racfdev...
Date of unload file = Thu Feb 22 23:09:06 2007
Compiling list of Groups...
Parsing TEXT...
Compiling list of User ID's...
Parsing TEXT...
Compiling list of Datasets...
Parsing TEXT...
Compiling list of General Resources...
Parsing TEXT...
Removing temporary files...

The following output files have been created:
  racf124-groups.txt - Text list of groups
  racf124-users.txt - Text list of users
  racf124-datasets.txt - Text list of datasets
  racf124-gen-res.txt - Text list of gen-res

C:\data\racf\dev>
```

## racf128

Description This utility first determines the users with any SYSTEM attribute and then looks for a matching .SUBMIT profile in the SURROGAT class.

Usage racf128

Output html  
 Keywords special  
 Examples C:\data\racf\dev>racf128  
 Validating unload file racfdev...  
 Date of unload file = Sun Mar 18 18:12:33 2007  
 Compiling list of IDs with SYSTEM SPECIAL...  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 [\*\*\*\*\* ]  
 Checking for SURROGAT profile for 7 users...  
 Found [SURROGAT] [\*\*\*\*\*.SUBMIT]  
 Found [SURROGAT] [\*\*\*\*\*.SUBMIT]  
  
 Parsing HTML...  
 Removing temporary files...  
  
 The following output files have been created:  
     racf128.htm - HTML list of SYSTEM SPECIAL SURROGAT profiles  
  
 C:\data\racf\dev>

## racfawk

Description General purpose tool to extract information from the flat file. Requires some knowledge of the format of the flat file which is defined in the IBM RACF Macros and Interfaces Manual. In its simplest form it can be used to pull out all records of a single type, e.g. racfawk 0200 1 0200 or a string can be searched for within the record type at a specified location. Can be very useful on occasions, especially in conjunction with Notepad++.

Usage racfawk record\_type start\_column [!]search\_string

Output text

Keywords adhoc

Examples C:\data\racf\dev>racfawk 0200 1 0200  
 Validating unload file racfdev...  
 Date of unload file = Wed Jan 17 22:27:29 2007  
 Compiling list...  
 Removing temporary files...  
  
 The following output files have been created:  
     racfawk.txt - Text extract of 0200 IRRDBU00 records  
  
 C:\data\racf\dev>

## racfdiag

Description This is an information gathering tool to assist in the event of problems. If you are unable to resolve a problem, then please use this tool to generate racfdiag.htm and send it to me for analysis with a description of the problem you are encountering.

Usage racfdiag

Output html

Keywords diagnostics summary

Examples

```

C:\data\racf\dev>racfdiag
Creating initial htm...
Populating getinfo section...
Compiling array...
Sorting groups...
Sorting counts...
Compiling diagnostic statistics...
Checking racf.ini...
Checking to ensure unload hasn't been sorted...
Analysing characters...
Analysing record types...
Analysing text line structure...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racfdiag.htm - HTML list of diagnostic information

C:\data\racf\dev>

```

## racfjcl

Description

This is similar to the clist option on the RACF search command and builds the header and footer information around it from the RACF.INI file. Very useful in conjunction with other utilities such as RACF75 where you can build a RACF command round a list of User IDs, e.g. RACFJCL RACF75 "ALU " " CICS(TIMEOUT(15))" where this will build the JCL to set everyone listed in the file RACF75 to a CICS timeout value of 15.

Usage

```
racfjcl text_input_file string1 string2 [string3]
```

Output

```
jcl
```

Keywords

```
clist
```

Examples

```

C:\data\racf\dev>racfjcl racf75 " ALU " " CICS(TIMEOUT(15))"
Compiling JCL...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racfjcl.jcl - JCL output file

C:\data\racf\dev>

```

# Groups

## racf03

Description	<p>Text report showing group tree structure. Similar to DSMON but also shows the number of connections to each group.</p> <p>See also RACF104 which generates group tree information in a suitable format to be loaded straight into Microsoft Visio.</p>
Usage	racf03
Output	html
Keywords	dsmon tree
Examples	<pre>C:\data\racf\dev&gt;racf03 Validating unload file racfdev... Date of unload file = Wed Jan 17 22:27:29 2007 Compiling group tree... Parsing HTML... Removing temporary files...  The following output files have been created:     racf03.htm - HTML of Group structure  C:\data\racf\dev&gt;</pre>

## racf04

Description	<p>List all groups showing the installation data and the create date of the group and the number of user IDs connected to the group. The example below lists all groups with one or more users. Another useful example is racf04 500 which would list only groups with more than 500 connected user IDs.</p> <p>Note: the racf04 text file always lists all groups. This is because it is intended for use with ts.exe (text search) enabling searching groups' names and installation data.</p>
Usage	racf04 [min_count] [max_count]   [no_install_data]
Output	html jcl text
Keywords	installation-data
Examples	<pre>C:\data\racf\dev&gt;racf04 1 Validating unload file racfdev... Date of unload file = Wed Jan 17 22:27:29 2007 Compiling group list... Parsing HTML... Parsing JCL... Removing temporary files...  The following output files have been created:     racf04.jcl - JCL with ALG commands which can be edited     racf04.htm - List of groups with descriptions etc.     racf04     - Simple text file with ALL groups and descriptions  C:\data\racf\dev&gt;</pre>

## racf06

Description	<p>List group showing User IDs, names, authority and flag if group-special attribute is operative. JCL is for removal of User IDs from the group and is useful when deleting large groups. Beware, however, that as it can also be edited to become a connect job instead of a remove job where ever group-special is operative this appears on the remove command although it is an invalid option.</p> <p>Fields listed are: User ID, Name, Authority, Group-Special, Revoke date</p> <p>The optional last field means if 'group-special' is found as a 2<sup>nd</sup> parameter then the report only lists those users with Group-Special in the group, ignoring all others.</p>
Usage	racf06 group [group-special]
Output	html jcl

Keywords authority group-special revoke-date

Examples C:\data\racf\dev>racf06 sys1  
Processing group list...  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
racf06.jcl - JCL to clean up orphaned ACL entries  
racf06.htm - HTML listing group SYS1  
racf06-SYS1.htm - copy of racf06.htm  
  
C:\data\racf\dev>

## racf36

Description Compares 2 RACF groups and reports any User IDs found in both groups. JCL produced is to remove users from group1.

Usage racf36 group1 group2

Output html jcl

Keywords common-groups

Examples C:\data\racf\dev>racf36 group1 group2  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group lists...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
racf36.jcl - JCL to remove from common users from GROUP1  
racf36.htm - HTML users common to groups GROUP1 and GROUP2  
  
C:\data\racf\dev>

## racf47

Description Lists a group and produces JCL to change the owner and dfltgrp of each member to that specified on the command line.

Usage racf47 group1 group2 [group3]

Output html jcl

Keywords DFLTGRP OwnerID group-special

Examples C:\data\racf\dev>racf47 sys1 vsamdset  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
racf47.jcl - JCL to change group connections assuming DFLTGRP  
racf47.htm - HTML list of group SYS1  
  
C:\data\racf\dev>

## racf50

Description List User IDs in a group with dfltgrp and all other connected groups. If many groups per user are involved then line length can grow to be awkwardly large, but can be useful. Similar to RACF73 which also shows names.

Fields listed are: DFLTGRP, other connected groups

Usage racf50 group  
Output html  
Keywords DFLTGRP  
Examples C:\data\racf\dev>racf50 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
racf50.htm - HTML list of connected groups for SYS1 users  
racf50-SYS1.htm - copy of racf50.htm  
  
C:\data\racf\dev>

## racf53

Description List group showing User ID, name, last access, if revoked, if TSO segment present, if CICS segment present, and CICS segment timeout value.

Fields listed are: Last Access, Revoked, TSO, CICS, Timeout

Usage racf53 group  
Output html  
Keywords  
Examples

C:\data\racf\dev>racf53 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
racf53.htm - HTML list of group SYS1  
racf53-SYS1.htm - copy of racf53.htm  
  
C:\data\racf\dev>

## racf69

Description List all revoked group connections. Doesn't quite work as intended as connections don't appear to become revoked until the connection is used after the revoke date. See also RACF83 Intended to pick up revoke attribute but this is not usually set in download. Hence have added a revoke date parameter to racf.ini file which works similar to expired users, i.e. compares the actual dates.

Fields listed are: Group, User ID, Name, Revoke date, Group-Special

Usage racf69  
Output html jcl  
Keywords revoked  
Examples

C:\data\racf\dev>racf69  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing list of revoked user-group connections...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
racf69.jcl - JCL to remove revoked connections  
racf69.htm - HTML list of revoked group connections  
  
C:\data\racf\dev>

## racf70

Description Count the number of users in a group. Very simplistic but useful never the less.

Usage Fields listed are: Group, Number of Connections, Installation Data  
racf70 group

Output html

Keywords group-count

Examples C:\data\racf\dev>racf70 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
SYS1 [1]  
  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
 racf70.htm - HTML count of members in SYS1  
  
C:\data\racf\dev>

## racf73

Description List group with names and all other group connections. Similar to RACF50

Usage Fields listed are: User ID, Name, DFLTGRP, All Groups or only Specified Groups  
racf73 group [connect1] [connect2] [connect3] [connect4] [connect5] [connect6] [connect7]

Output html

Keywords DFLTGRP

Examples C:\data\racf\dev>racf73 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
 racf73.htm - HTML list of group SYS1  
 racf73-SYS1.htm - copy of racf73.htm  
  
C:\data\racf\dev>

## racf75

Description List group with only User ID and group on report, one per line and separated by a single space.

Usage racf75 group

Output text

Keywords adhoc

Examples C:\data\racf\dev>racf75 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Number of ID's listed in group SYS1 = 1  
Removing temporary files...  
  
The following output files have been created:  
 racf75.txt - Text list of user ID and group appended to file  
 racf75 - Text list of just user IDs appended to file  
  
C:\data\racf\dev>

## racf76

Description List group showing User ID, name, connect owner, authority and if group-special attribute is present. This is essentially the same as a RACF06 but with the connect owner added. See also RACF111

Usage racf76 group

Output html jcl

Keywords connect-authority connect-owner

Examples C:\data\racf\dev>racf76 sys1  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing group list...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf76.jcl - JCL to recreate connects  
    racf76.htm - HTML listing group SYS1  
    racf76-SYS1.htm - copy of racf76.htm  
  
C:\data\racf\dev>

## racf77

Description List all group connections where connect is not owned by the group connected to. Shows where a connect has been done where the owner is not explicitly set to the same as the group.

Usage racf77

Output html jcl

Keywords connect-owner

Examples C:\data\racf\dev>racf77  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing list...  
.....  
.....  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf77.jcl - JCL to set all connect owners equal to group  
    racf77.htm - HTML list of connects owned by other than group  
  
C:\data\racf\dev>

## racf87

Description Generates list in essentially 3 columns, Group, Installation Data, and GID (any groups without a GID are not listed). Updated to include hex GID in addition to decimal GID

Usage racf87

Output html

Keywords GID

Examples C:\data\racf\dev>racf87  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling group list...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf87.htm - HTML of groups with GIDs  
  
C:\data\racf\dev>

## racf104

Description This utility generates a text (txt) file which can be directly imported into the Visio Organisational Chart Wizard. Note: if there is a large number of groups then it can take quite a long time for Visio to process this import! I have been asked where the idea for this utility came from, well Doc Farmer suggested it to me. Also, I have since discovered it is possible to specify the level within the tree to report from meaning you can generate a tree structure report on just part of the group structure which makes this more useful. See also RACF03

Usage racf104

Output text

Keywords Visio dsmon tree

Examples C:\data\racf\dev>racf104  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling group tree data...  
Parsing TEXT...  
Removing temporary files...  
  
The following output files have been created:  
    racf104.txt - Text file ready to importing into MS Visio  
  
C:\data\racf\dev>

## racf111

Description List group showing User IDs, names, last access date, creation date and if the revoked attribute is operative. This report was requested by Sarbanes Oxley Auditors specifically to see the user creation date. See also RACF06 and RACF76.

Usage racf111 group

Output html

Keywords create-date last-access-date

Examples C:\data\racf\dev>racf111 sys1  
Processing group list...  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf111.htm - HTML group list with user creation date  
    racf111-SYS1.htm - copy of racf111.htm  
  
C:\data\racf\dev>

## racf113

Description Annotate a list of groups. Where the input is in the form of one group per line and no spaces. The output from is similar to RACF04 which is full list of all groups.

Usage racf113 filename

Output html

Keywords annotate-groups

Examples C:\data\racf\dev>racf113 groups.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Annotating list of groups...  
  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf113.htm - HTML of annotated groups  
  
C:\data\racf\dev>

## racf117

Description Produces summary stats for groups listed in input file specified on command line

Usage racf117 filename

Output html

Keywords summary

Examples C:\data\racf\dev>racf117 groups.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling list of GROUP profiles...  
.....  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf117.htm - HTML summary statistics for Groups specified in  
    groups.txt  
  
C:\data\racf\dev>

## racf121

Description List groups only showing users who have never logged on (uses group connect counts)

Usage racf121 filename

Output html

Keywords never-logged-on

Examples C:\data\racf\dev>racf121 groups.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling list of GROUP profiles...  
.....  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf121.htm - HTML list of users, within Groups, that have never logged  
    on  
  
C:\data\racf\dev>

## racf126

Description List summary info of groups which are referenced as DFLTGRPs in base user segments. The idea behind this is to assist in user reviews so that by listing all DFLTGRPs and getting sign off means you have sign off on all users on the system.

Usage racf126

Output html text

Keywords DFLTGRP

Examples C:\data\racf\dev>racf126  
Validating unload file racfdev...  
Date of unload file = Mon Mar 05 11:13:28 2007  
Compiling group list...  
Parsing HTML...  
Parsing TEXT...  
Removing temporary files...  
  
The following output files have been created:  
    racf126.htm - List of DFLTGRPs with descriptions etc.  
    racf126.txt - List of DFLTGRPs  
  
C:\data\racf\dev>

## racf127

Description List summary info of groups which are referenced as Group-Specials.

Usage racf126

Output html

Keywords group-special

Examples C:\data\racf\dev>racf127  
Validating unload file racfdev...  
Date of unload file = Sun Mar 18 18:12:33 2007  
Compiling group list...  
Parsing HTML...  
Parsing TEXT...  
Removing temporary files...  
  
The following output files have been created:  
    racf127.htm - List of Group-Specials with descriptions etc.  
    racf127.txt - List of Group-Specials  
  
C:\data\racf\dev>

## racf129

Description This report has many subtleties which vary from the straight forward groups reports. First is that they only list users where the group is their DFLTGRP and hence doesn't list every user connected to the group. Then the Group-Special flag is set if the user has any group-special attributes, and not necessarily with the DFLTGRP. Last but not least are the optional connect parameters to list whether or not the users are connected to specified groups.  
  
This is a highly customised report which has been designed to assist with SOX reporting where every user has to be accounted for once, and only once.

Usage racf129 filename [connect1] [connect2] [connect3] [connect4] [connect5] [connect6] [connect7]

Output html (multiple)

Keywords dfltgrp

Examples C:\data\racf\dev>racf129 racf126.txt  
Validating unload file racfdev...  
Date of unload file = Fri Jun 01 23:30:55 2007  
Annotating group GROUP1...  
  
Parsing HTML...  
  
The following output files have been created:  
    racf129-SUPGROUP-GROUP1.htm - HTML of annotated groups  
Annotating group GROUP2...  
  
Parsing HTML...  
  
The following output files have been created:  
    racf129-SUPGROUP-GROUP2.htm - HTML of annotated groups  
Removing temporary files...  
  
C:\data\racf\dev>

# Users

## racf05

**Description** List expired User IDs where expiry is determined by listing those users who have not accessed the system since the date specified in the RACF.INI file. JCL is produced to delete all dormant users listed. Care needs to be taken not to delete apparently dormant, but required system User IDs such as started tasks etc. Enhanced to now accept optional command line parameter of a Group thus only listing the dormant users that are connected to that Group. It now suppresses the lower case irr user IDs which are IBM supplied and should never be deleted, as well as any IDs with the PROTECTED attribute.

**Usage** racf05 [group]

**Output** html jcl

**Keywords** expired dormant

**Examples**  
C:\data\racf\dev>racf05  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of expired users...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf05.jcl - JCL to clean up orphaned ACL entries  
    racf05.htm - HTML of profiles owned by orphans  
    racf05 - list of orphaned users  
  
C:\data\racf\dev>

## racf08

**Description** Search for User IDs using string supplied. Searches both User ID and name field from the database. Mainly used for searching on names field, e.g. find a user by their first name if you can't find them by their User ID. Includes Installation data now as well.

See also racf18 for a fast search capability.

**Usage** racf08 search\_string1 [search\_string2] .. [search\_string8]

**Output** screen text

**Keywords** search DFLTGRP

**Examples**  
C:\data\racf\dev>racf08 sys1

UserID	DFLTGRP	Name	Installation Data
IBMUERS	SYS1	SYS1.UADS	
TSO2	NOWORLD	SYS1.UADS	
TSO3	NOWORLD	SYS1.UADS	
TSO4	NOWORLD	SYS1.UADS	
TSO5	NOWORLD	SYS1.UADS	

-----  
Number = 5  
C:\data\racf\dev>

## racf09

**Description** List User IDs based on a mask. E.g. can list all User IDs starting with letter A or AB or ABC etc. Report shows User ID, name, if revoked, when last logged on, if TSO segment exists and if CICS segment exists. JCL is also produced for resuming all the User IDs listed.

**Usage** racf09 user\_mask

**Output** html jcl

**Keywords** userid-mask

Examples

```
C:\data\racf\dev>racf09 ibm
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of users...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf09.jcl - JCL to resume users listed
  racf09.htm - HTML of users matching mask

C:\data\racf\dev>
```

## racf18

Description Generate data file for allowing fast searching for User IDs. Output consists simply of all User IDs and associated names.

The output file can be used in conjunction with the 'ts.exe' text search facility to provide similar functionality to racf08 but with much faster response times since as it uses only a fraction of the total unload data when searching.

e.g. ts racf18.txt <search\_string\_1> <search\_string\_2>

Usage

racf18

Output

text html

Keywords

search

Examples

```
C:\data\racf\dev>racf18
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of User ID's...
Removing temporary files...

The following output files have been created:
  racf18.txt - Text list of users with default group and names
  racf18     - Text list of just user IDs

C:\data\racf\dev>
```

## racf24

Description List revoked users. JCL to delete if no TSO segment. List revoked users. JCL to delete if no TSO segment.

Usage

racf24

Output

html jcl

Keywords

revoked

Examples

```
C:\data\racf\dev>racf24
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of revoked users...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf24.jcl - JCL to remove CICS only users
  racf24.htm - HTML list of all revoked users

C:\data\racf\dev>
```

## racf38

Description General audit report on users with certain properties or attributes. The example below shows it with all sections being generated, although there is a section within the racf.ini file that enables suppression of any sections if required.

Usage

racf38

Output html

Keywords audit attributes authority NOPASSWORD PROTECTED RESTRICTED UAUDIT password-interval

Examples C:\data\racf\dev>racf38  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of users with System attributes...  
Compiling list of users with Group attributes...  
Compiling list of users with Class authority...  
Compiling list of users with Group connection other than USE...  
Compiling list of users with password interval greater than 31 days...  
Compiling list of users with NOPASSWORD...  
Compiling list of PROTECTED users...  
Compiling list of RESTRICTED users...  
Compiling list of users with UAUDIT...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
racf38.htm - HTML list of general audit information  
  
C:\data\racf\dev>

## racf46

Description Takes as input a list of users to be deleted in the form of one User ID per line with no spaces. This is designed to interface with output from our HR system and subsequently interfaces to an application. Likely to be of little use to anyone else.

Usage racf46 filename

Output html jcl

Keywords user-deletion

Examples C:\data\racf\dev>racf46 users.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling list of users...  
...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
racf46.jcl - JCL to revoke User IDs  
racf46.htm - HTML list of users to be revoked  
racf46.ts - Text list of just the User IDs  
  
C:\data\racf\dev>

## racf66

Description Similar to doing a ListUser command online. Similar to doing a ListUser command online

Usage racf66 userid

Output html

Keywords listuser

Examples C:\data\racf\dev>racf66 ibmuser  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling details of user...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
racf66.htm - HTML of user details  
racf66-IBMUSER.htm - copy of racf66.htm  
  
C:\data\racf\dev>

## racf79

Description List all User IDs where the owner of the User ID is not the same as the dfltgrp

Usage racf79

Output html jcl

Keywords DFLTGRP

Examples C:\data\racf\dev>racf79  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of users not owned by dfltgrp...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf79.jcl - JCL to make OWNER same as DFLTGRP  
    racf79.htm - HTML list of user IDs where DFLTGRP not same as OWNER  
  
C:\data\racf\dev>

## racf82

Description Annotate a list of User IDs. Where the input is in the form of one User ID per line and no spaces. The output from this is User ID, name, dfltgrp, date last accessed, etc. See also RACF102 Now reports in bold any User ID which cannot be found (previously dropped without any indication).

Fields listed are: User ID, Name, DFLTGRP, Last access, TSO, CICS

Usage racf82 filename

Output html

Keywords annotate

Examples C:\data\racf\dev>racf82 users.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Annotating list of users...  
...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf82.htm - HTML of annotated users  
  
C:\data\racf\dev>

## racf84

Description Generates a list of User IDs where the 2nd and 3rd positions are numeric but ignoring those which fit the mask 'Xnnnnn ' where X is any alpha and n is any numeric. Info list includes User ID, name, dfltgrp, create date, last access date and if TSO present. i.e. normally lists User IDs that are all numeric

Usage racf84

Output html

Keywords userid-mask

Examples C:\data\racf\dev>racf84  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of users...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf84.htm - HTML list of user IDs  
  
C:\data\racf\dev>

## racf86

**Description** Generates JCL intended for users who should be deleted from the system but can't be because they still own datasets. It assumes the group 'LIMBO' exists, connects the user to LIMBO with a revoked connection, makes it the DFLTGRP and OWNER of the User ID, removes all other connections, and all dataset and general resource access list entries. Also deletes any dataset profiles commencing with the User ID except userid.\*\* which has its access list reset.

**Usage** racf86 filename

**Output** html jcl

**Keywords** limbo user-deletion

**Examples**

```
C:\data\racf\dev>racf86 users.txt
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Searching for access list entries...
...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf86.jcl - JCL to put user IDs into 'LIMBO'
  racf86.htm - HTML list of any user IDs not found

C:\data\racf\dev>
```

## racf88

**Description** Generates list in essentially 6 columns, giving user details and OMVS segment details such as User ID, Name, default group, UID home path, program (any users without a UID are not listed). Update to include hex UID in addition to decimal UID.

Now, also accepts a group to be specified on the command line. In this mode it lists only users who are members of the group and lists all members of the group whether they have an OMVS segment or not.

**Usage** racf88 [group]

**Output** html

**Keywords** UID

**Examples**

```
C:\data\racf\dev>racf88
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of users...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf88.htm - HTML of users with UIDs

C:\data\racf\dev>
```

## racf92

**Description** List all User IDs along with name, owner, date created, date last accessed, and if revoked. Optionally can be restricted to just listing those with a matching DFLTGRP.

**Usage** racf92 [DFLTGRP]

**Output** html

**Keywords** DFLTGRP

Examples C:\data\racf\dev>racf92  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of users...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf92.htm - HTML list of user IDs  
  
C:\data\racf\dev>

## racf93

Description List all CICS users with showing any OPIDs. This utility was provided at the request of Rolf W Valters.  
Usage racf93  
Output html  
Keywords OPID  
Examples C:\data\racf\dev>racf93  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.tmp...  
Compiling list of users...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf93.htm - HTML of CICS users showing any OPIDs  
  
C:\data\racf\dev>

## racf102

Description Annotate a list of User ID. Where the input is in the form of one User ID per line and no spaces. The output from this is User ID, name, dfltgrp, date last accessed, etc. Similar to RACF82 but this report focuses on dates (creation, last access, password changed) and password (expiry, generation number). Now reports in bold any User ID which cannot be found (previously dropped without any indication).  
Usage racf102 filename  
Output html  
Keywords annotate create-date  
Examples C:\data\racf\dev>racf102 users.txt  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Annotating list of users...  
...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf102.htm - HTML of annotated users  
  
C:\data\racf\dev>

## racf103

Description Report on User IDs which have duplicate name data. Because there are likely to be User IDs with duplicate data which is intentional it looks for an input file of RACF103 for name data to be ignored.  
Usage racf103  
Output html  
Keywords duplicate-names

```

Examples      C:\data\racf\dev>racf103
              Validating unload file racfdev...
              Date of unload file = Wed Jan 17 22:27:29 2007
              Compiling racf.tmp...
              Names read in from racf103 to be ignored are
              [CICS DEFAULT USER  ]
              [#####]
              Analysing users, displaying names with duplicates...
              [NIGEL TEST      ]

              Parsing HTML...
              Removing temporary files...

              The following output files have been created:
                racf103.htm - HTML of userids with duplicate names

              C:\data\racf\dev>

```

## racf105

```

Description   This utility generates a report by first building a list of User IDs referenced from STARTED class profiles
              and then enumerating them with details, in particular do they have the PROTECTED attribute set?
              See also RACF106 and RACF30

Usage        racf105

Output       html

Keywords     STARTED PROTECTED

Examples     C:\data\racf\dev>racf105
              Validating unload file racfdev...
              Date of unload file = Wed Jan 17 22:27:29 2007
              Compiling list of IDs from STARTED profiles...
              Annotating list of 51 users...
              .....
              Parsing HTML...
              Removing temporary files...

              The following output files have been created:
                racf105.htm - List of User IDs referenced from STARTED profiles

              C:\data\racf\dev>

```

## racf106

```

Description   This utility generates a report by first building a list of User IDs with a UID(0) and then enumerating
              them with details, in particular do they have the PROTECTED attribute set?
              See also RACF106 and RACF88

Usage        racf106

Output       html

Keywords     UID(0) PROTECTED

Examples     C:\data\racf\dev>racf106
              Validating unload file racfdev...
              Date of unload file = Wed Jan 17 22:27:29 2007
              Compiling list of IDs with UID(0)...
              Annotating list of 8 users...
              .....
              Parsing HTML...
              Removing temporary files...

              The following output files have been created:
                racf106.htm - HTML list of User IDs with a UID of zero

              C:\data\racf\dev>

```

## racf110

**Description** Lists all users with a TSO segment sorted by the default TSOPROC. The report also lists both region size and max region size along with user details and default group. JCL is generated for each user using the currently defined region and max sizes. This can be easily edited and then submitted to effect changes to the TSO region sizes

**Usage** racf110

**Output** html jcl

**Keywords** TSOPROC

**Examples**

```
C:\data\racf\dev>racf110
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling table of TSOPROCs...
Compiling list of TSO users...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf110.jcl - JCL to set TSO region sizes
  racf110.htm - HTML list of TSO region sizes and TSOPROCs

C:\data\racf\dev>
```

## racf122

**Description** List all users who have never logged on and don't have protected attribute. DFLTGRP group connection count is used to determine if logged on before, and if zero assumed never logged on.

**Usage** racf122

**Output** html

**Keywords** never-logged-on

**Examples**

```
C:\data\racf\dev>racf122
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of user profiles...

Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf122.htm - HTML list of non-protected users that have never logged
on

C:\data\racf\dev>
```

## racf130

**Description** Lists all users and has columns to enable the reader to determine if the users are able to logon to TSO. Hence columns displayed are ID, Name, DFLGRP, Last access, Revoked, Protected, TSO.

**Usage** racf130 [DFLTGRP]

**Output** html

**Keywords** TSO DFLTGRP

**Examples**

```
C:\data\racf\dev>racf130
Validating unload file racfdev...
Date of unload file = Fri Jun 01 23:30:55 2007
Compiling list of users...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf130.htm - HTML list of user IDs

C:\data\racf\dev>
```

## racf131

Description	<p>Generates list in essentially 6 columns, giving user details and Netview segment details where each column heading equates to the key word used in the RACF ALU command to amend it.</p> <p>Now, also accepts a group to be specified on the command line. In this mode it lists only users who are members of the group and lists all members of the group whether they have a Netview segment or not.</p>
Usage	racf131 [group]
Output	html
Keywords	Netview
Examples	<pre>C:\data\racf\dev&gt;racf131 Validating unload file racfdev... Date of unload file = Wed Jan 17 22:27:29 2007 Compiling racf.tmp... Compiling list of users... Parsing HTML... Removing temporary files...  The following output files have been created:   Racf131.htm - HTML of users with Netview segments  C:\data\racf\dev&gt;</pre>

# Datasets

## racf07

Description Report showing access lists for DATASETs showing both User IDs and names. It is recursive and hence if you specify a high level qualifier it will list out all profiles starting with that HLQ. Because of the recursive name lookup this can be slow to run so if it is not important to show names RACF71 may be better. Also creates RACF07 containing simple text list of DATASET profile names.

Usage racf07 dataset

Output html text

Keywords mask filter

Examples

```
C:\data\racf\dev>racf07 sys1
Processing dataset list...
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Processing .....
.....
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf07.htm - HTML list of enumerated DATASET profiles
  racf07     - plain text list of DATASET profiles

C:\data\racf\dev>
```

## racf23

Description List all DATASET profiles with access lists where the prefix matches that specified. JCL is produced which will re-create profiles with that prefix. Sometimes useful for establishing environments by modelling existing ones. See also RACF90

Usage racf23 prefix

Output html jcl

Keywords addsd

Examples

```
C:\data\racf\dev>racf23 cicsal
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of dataset profiles prefixed with CICSAL...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf23.jcl - JCL to recreate profiles and access
  racf23.htm - HTML list of profiles and access

C:\data\racf\dev>
```

## racf56

Description List all DATASET and JESSPOOL profiles with a UACC greater than NONE. Produces 2 JCL output files RACF56A.JCL and RACF56B.JCL. The first grants the same access as the UACC to the profiles and the second sets the UACC to none.

Usage racf56 ID [ignore]

Output html jcl text

Keywords UACC

## Examples

```
C:\data\racf\dev>racf56 sys1
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of DATASET and JESSPOOL profiles without UACC(NONE)...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf56a.jcl - JCL to grant UACC access to SYS1
  racf56a.jcl - JCL to set UACC to NONE
  racf56.htm  - HTML list of profiles with UACC other than NONE
  racf56     - Text list of profiles

C:\data\racf\dev>
```

## racf61

**Description** Grant access to some (i.e. don't if already on access list) DATASET profiles with the specified prefix for the given ID and access level. In order to generate JCL for all profiles simply give a fictitious Group which doesn't appear on any of the access lists.

**Usage** racf61 prefix id access

**Output** html jcl

**Keywords** permit

**Examples**

```
C:\data\racf\dev>racf61 sys1 ibmuser read
Validating unload file racfdev...
Date of unload file = Thu Feb 22 23:09:06 2007
Compiling list of DATASET profiles prefixed with SYS1
excluding any with IBMUSER already on access list...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf61.jcl - JCL to grant IBMUSER READ access
  racf61.htm - HTML list of access

C:\data\racf\dev>
```

## racf68

**Description** In addition to taking the flat file as input this utility also requires the output from a DSMON. It then reports the fully qualified profiles and access lists for all APF authorised libraries. In addition to RACF68 listing all APFs, RACF68.APF has been added to list those APFs which do not have corresponding DATASET profiles. Note: if several LPARs all share the same DASD then concatenate the DSMON outputs as shown to ensure all APFs are protected from all LPARs.

```
type \data\racf\prda\dsmon > \data\racf\prda\dsmonx
type \data\racf\prdb\dsmon >> \data\racf\prda\dsmonx
type \data\racf\prdb\dsmon > \data\racf\prdb\dsmonx
type \data\racf\prda\dsmon >> \data\racf\prdb\dsmonx
and then run RACF68 as
RACF68 dsmonx
for both prda and prdb
i.e. this checks to ensure all APF authorised profiles are protected on both LPARs irrespective of which
LPAR they have been authorised on.
```

**Usage** racf68 filename

**Output** html text

**Keywords** APF dsmon

Examples

```

C:\data\racf\dev>racf68 dsmon
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of APF's (RACF68)...
Compiling racf.tmp...
Compiling list of 175 APF DATASET profiles...
.....
.....
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf68.htm - HTML of fully qualified APF authorised profiles
  racf68.apf - Text list of datasets without fully qualified profiles
  racf68     - Text list of all APF authorised datasets

C:\data\racf\dev>

```

### racf71

Description  
Usage  
Output  
Keywords  
Examples

```

List DATASET access but without names. Quicker running version of RACF07
racf71 dataset
html
mask filter
C:\data\racf\dev>racf71 sys1
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Processing dataset list...
.....
.....
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf71.htm - HTML of DATASET access

C:\data\racf\dev>

```

### racf80

Description  
Usage  
Output  
Keywords  
Examples

```

List discrete DATASET profiles and produce JCL to convert to generic (i.e. delete and re-create).
Ignores anything with HLO starting DFHSM. See also RACF42
racf80
html jcl
discrete generic
C:\data\racf\dev>racf80
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Processing dataset list...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf80.jcl - JCL to convert to generic profiles
  racf80.htm - HTML list of discrete DATASET profiles

C:\data\racf\dev>

```

### racf90

Description  
Usage  
Output  
Keywords

```

List all DATASET profiles with access lists. JCL is produced which will re-create all profiles. Similar to
RACF23 but without the limitation of prefix.
racf90
html jcl
addsd

```



Usage	racf115 filename
Output	html
Keywords	annotate audit
Examples	<pre>C:\data\racf\dev&gt;racf115 datasets Validating unload file racfdev... Date of unload file = Wed Jan 17 22:27:29 2007 Compiling racf.tmp... Compiling list of DATASET profiles... .. Parsing HTML... Removing temporary files...  The following output files have been created:   racf115.htm - HTML list DATASET profiles specified in datasets  C:\data\racf\dev&gt;</pre>

# General Resources

## racf12

**Description** Report showing access lists for General Resources showing both User IDs and names. This is similar to RACF07 expect for General Resources rather than DATASETs. It is recursive and hence if you specify a high level qualifier it will list out all profiles starting with that HLO.

**Usage** racf12 class profile

**Output** html

**Keywords** mask filter

**Examples**  
C:\data\racf\dev>racf12 facility irr  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Processing general resource list...  
.....  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf12.htm - HTML list of enumerated General Resource profiles  
  
C:\data\racf\dev>

## racf16

**Description** List all profiles in a member / group class pair with access lists. Useful for CICS classes. The racf19 report is the same as the racf16 except it doesn't have the installation data included.

The noaccess key word allows suppression of the access list information. This is intended for classes which have been defined for documenting information rather than controlling access, and can be useful for providing an installation data field for digital certificates.

**Usage** racf16 member-class group-class [noaccess]

**Output** Html

**Keywords**

**Examples**  
C:\data\racf\dev>racf16 tcicstrn gciestrn  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of TCICSTRN / GCICSTRN profiles...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf16.htm - HTML of list of TCICSTRN and GCICSTRN profiles  
    racf16-TCICSTRN.htm - copy of racf16.htm  
  
C:\data\racf\dev>

## racf19

**Description** List all General Resource profiles with access lists for the class and group-class specified. The racf19 report is the same as the racf16 except it doesn't have the installation data included.

**Usage** racf19 member-class group-class

**Output** html

**Keywords**

## Examples

```
C:\data\racf\dev>racf19 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf19.htm - HTML list of enumerated General Resource profiles

C:\data\racf\dev>
```

## racf20

### Description

List all General Resource profiles with access lists where prefix matches that specified. JCL is produced which will re-create profiles for that class / group-class pair and prefix. Useful for establishing environments such as modelling a prefixed CICS region or even taking a test set of profiles and implementing them on a live environment. See also RACF48,RACF51 and RACF91

### Usage

racf20 member-class group-class prefix

### Output

html jcl

### Keywords

prefixed recreate

### Examples

```
C:\data\racf\dev>racf20 tcicstrn gciestrn cicsal
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles prefixed with CICSAL...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf20.jcl - JCL to recreate profiles
  racf20.htm - HTML of TCICSTRN and GCICSTRN profiles
  racf20-TCICSTRN.htm - copy of racf20.htm

C:\data\racf\dev>
```

## racf22

### Description

List all member class profiles with installation data. Very simplistic but can be very useful. Now even better that it generates JCL which can be amended using your favourite editor and used to update the installation data field.

### Usage

racf22 class

### Output

html jcl

### Keywords

installation-data

### Examples

```
C:\data\racf\dev>racf22 gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling transactions list...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf22.htm - HTML list of class profiles with installation data
  racf22-GCICSTRN.htm - copy of racf22.htm

C:\data\racf\dev>
```

## racf25

### Description

Similar to RACF20 except JCL is for deleting profiles. Useful for cleaning up profiles no longer required. See also RACF49

### Usage

racf25 member-class group-class prefix

### Output

html jcl

### Keywords

delete prefixed

Examples

```
C:\data\racf\dev>racf25 tcicstrn gciestrn cicsal
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles prefixed with CICSAL...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf25.jcl - JCL to remove all CICSAL prefixed profiles
  racf25.htm - HTML list of all CICSAL prefixed profiles

C:\data\racf\dev>
```

## racf28

Description List General Resource class profiles for a specified class pair where the initial letter matches. e.g. most useful for listing all TCICSTRN / GCICSTRN where initial letter is 'C' signifying IBM supplied.

Also very useful for focusing on individual transactions, e.g. the example below which looks for profiles such as CEMT. Note: this will report any profile which could match, e.g. C\* and CE\* if defined. If both were defined then CE\* would be used and C\* would never be reached by CE trancodes. This matching is only done for the first 4 characters as it's intended for use with TCICSTRN / GCICSTRN type classes.

Usage racf28 member-class group-class first-character(s)

Output html

Keywords prefixed search CICS

Examples

```
C:\data\racf\dev>racf28 tcicstrn gciestrn cemt
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles...
Parsing HTML...
Removing temporary files...
```

The following output files have been created:  
 racf28.htm - HTML list of TCICSTRN / GCICSTRN profiles starting CEMT

C:\data\racf\dev>

## racf30

Description List all STARTED class profiles with STDATA

Usage racf30

Output html

Keywords STARTED STDATA

Examples

```
C:\data\racf\dev>racf30
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of STARTED profiles...
Parsing HTML...
Removing temporary files...
```

The following output files have been created:  
 racf30.htm - HTML list of STARTED class profiles

C:\data\racf\dev>

## racf33

Description List all Member / Group class profiles sorted. Useful for comparing profiles for similar CICS trancodes but with varying prefixes. Slow to run because of recursive sorting. See also RACF34

Usage racf33 member-class group-class

Output html

Keywords trancodes sorted

## Examples

```
C:\data\racf\dev>racf33 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of sorted TCICSTRN / GCICSTRN profiles
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf33.htm - HTML of TCICSTRN / GCICSTRN sorted by members

C:\data\racf\dev>
```

## racf34

**Description** List Member / Group class profiles which have duplicate entries. Useful for TCICSTRN profiles or GCICSTRN members. As with RACF33 this is unfortunately slow to run.

**Usage** racf34 member-class group-class

**Output** html

**Keywords** trancodes duplicate

**Examples**

```
C:\data\racf\dev>racf34 ccicscmd vcicscmd
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of duplicate CCICSCMD / VCICSCMD profiles...
Parsing HTML...
Removing temporary files...
```

The following output files have been created:  
 racf34.htm - HTML of CCICSCMD / VCICSCMD duplicate members

C:\data\racf\dev>

## racf35

**Description** Compare prefixed Member / Group profiles with specified prefix. Useful for comparing prefixed profiles in a test CICS region with those of the corresponding live region. Expects to find the flatfiles in sub-directories directly below. I normally run from \DATA\RACF with flat files in \DATA\RACF\DEV and \DATA\RACF\PROD This report details differences and gives a summary of the matching profiles.

**Usage** racf35 flatfile1 prefix1 flatfile2 prefix2 member-class group-class

**Output** html

**Keywords** prefixed compare

**Examples**

```
C:\data\racf\test>racf35 racftest cicsal racfprod cicsal tcicstrn gciestrn
Compiling racf.tmp...
Compiling list of differences in TCICSTRN / GCICSTRN profiles (pass1)...
Compiling racf.tmp...
Compiling list of differences in TCICSTRN / GCICSTRN profiles (pass2)...
Parsing HTML...
Parsing HTML...
Removing temporary files...
```

The following output files have been created:  
 racf35.htm - HTML of matching and differing profiles / members

C:\data\racf\test>

## racf37

**Description** List out all members with specified User ID or group on the access list. Similar to an XREF or RACF11 but only looks at Member / Group profiles and expands out the Group profiles to show all the actual Members. Useful for determining access to CICS trancodes.

**Usage** racf37 ID member-class group-class [all]

**Output** html jcl

**Keywords** XREF

## Examples

```
C:\data\racf\dev>racf37 ibmuser tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Mon Mar 05 11:13:28 2007
Compiling racf.tmp...
Compiling list of TCICSTRN / GCICSTRN Access...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf37.jcl - JCL to recreate access of IBMUSER for TCICSTRN / GCICSTRN
  racf37.htm - HTML of IBMUSER access for TCICSTRN / GCICSTRN showing
members
  racf37-IBMUSER.htm - copy of racf37.htm

C:\data\racf\dev>
```

## racf48

### Description

Like RACF20 but for non-prefixed General Resource pairs. See also RACF51

Note: non-prefixed means ignoring everything that is prefixed, hence this not the same as the entire class, subtle but important difference.

### Usage

racf48 member-class group-class

### Output

html jcl

### Keywords

non-prefixed recreate

### Examples

```
C:\data\racf\dev>racf48 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of non-prefixed resource profiles...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf48.jcl - JCL to recreate TCICSTRN / GCICSTRN non-prefixed
  racf48.htm - HTML list of TCICSTRN / GCICSTRN non-prefixed

C:\data\racf\dev>
```

## racf49

### Description

Like RACF25 but for non-prefixed General Resource pairs. See also RACF51

Note: non-prefixed means ignoring everything that is prefixed, hence this not the same as the entire class, subtle but important difference.

### Usage

racf49 member-class group-class

### Output

html jcl

### Keywords

non-prefixed delete

### Examples

```
C:\data\racf\dev>racf49 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of non-prefixed resource profiles...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf49.jcl - JCL to delete TCICSTRN / GCICSTRN non-prefixed
  racf49.htm - HTML list of TCICSTRN / GCICSTRN non-prefixed

C:\data\racf\dev>
```

## racf51

**Description** List prefixed profile pairs in friendly format. If using RACF20 or RACF25 to migrate profiles because they have to create profiles and then add members and then do any permits they are not very user friendly. This is a more readable report which can be used in conjunction with them. See also RACF48 and RACF49

**Usage** racf51 member-class group-class prefix

**Output** html

**Keywords** prefixed friendly

**Examples**  
C:\data\racf\dev>racf51 tcicstrn gciestrn cicsal  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of resource profiles prefixed with CICSAL...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf51.htm - HTML list of prefixed profiles for TCICSTRN / GCICSTRN  
    racf51-TCICSTRN.htm - copy of racf51.htm  
  
C:\data\racf\dev>

## racf62

**Description** Grant access to some General Resource profiles with prefix for ID and access. Some being where it is omitted if ID is already on the access list. In order to generate JCL for all profiles simply give a fictitious Group which doesn't appear on any of the access lists.

**Usage** racf62 member-class group-class prefix id access

**Output** html jcl

**Keywords** prefixed permit

**Examples**  
C:\data\racf\dev>racf62 tcicstrn gciestrn cicsal ibmuser read  
Validating unload file racfdev...  
Date of unload file = Thu Feb 22 23:09:06 2007  
Compiling list of resource profiles prefixed with CICSAL  
excluding any with IBMUSER already on access list...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf62.jcl - JCL to grant IBMUSER READ access  
    racf62.htm - HTML list of access  
  
C:\data\racf\dev>

## racf64

**Description** List all General Resource profiles with access lists. JCL is produced which will re-create profiles for that class / group-class pair. Similar to RACF20 but non-prefixed. See also RACF48 and RACF51, RACF85

Note: non-prefixed means ignoring everything that is prefixed, hence this not the same as the entire class, subtle but important difference.

**Usage** racf64 member-class group-class

**Output** html jcl

**Keywords** non-prefixed recreate

## Examples

```
C:\data\racf\dev>racf64 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of non-prefixed resource profiles...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf64.jcl - JCL to recreate non-prefixed TCICSTRN / GCICSTRN profiles
  racf64.htm - HTML list of non-prefixed TCICSTRN / GCICSTRN profiles

C:\data\racf\dev>
```

## racf65

### Description

List General Resource class.

### Usage

racf65 class

### Output

html

### Keywords

### Examples

```
C:\data\racf\dev>racf65 servauth
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf65.htm - HTML list of profiles
  racf65-SERVAUTH.htm - copy of racf65.htm

C:\data\racf\dev>
```

## racf67

### Description

Set or clear notify on all profiles in a prefixed General Resource class pair.

### Usage

racf67 member-class group-class prefix [id]

### Output

html jcl

### Keywords

notify

### Examples

```
C:\data\racf\dev>racf67 tcicstrn gciestrn cicsal ibmuser
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles prefixed with CICSAL...
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf67.jcl - JCL to set or remove NOTIFY on all TCICSTRN / GCICSTRN
profiles
  racf67.htm - HTML list of TCICSTRN / GCICSTRN profiles

C:\data\racf\dev>
```

## racf72

### Description

Summary breakdown of General Resource profiles.

### Usage

racf72

### Output

html

### Keywords

summary

Examples

```
C:\data\racf\dev>racf72
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling summary of General Resources...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf72.htm - HTML summary report of General Resource classes

C:\data\racf\dev>
```

## racf85

Description Similar to RACF20 / RACF64 but lists all profiles for a general resource class pair. Generates JCL to re-create for migrating to another platform.

Note: this report covers the entire class, not just non-prefixed or prefixed but everything. Also, handles members within the PROGRAM class as well.

Usage racf85 member-class [group-class]

Output html jcl

Keywords recreate

Examples

```
C:\data\racf\dev>racf85 program
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles...
Parsing HTML...
Parsing JCL...
Removing temporary files...
```

The following output files have been created:

```
  racf85.jcl - JCL to recreate all PROGRAM / profiles
  racf85.htm - HTML list of PROGRAM / profiles
  racf85-PROGRAM.htm - copy of racf85.htm
```

C:\data\racf\dev>

## racf91

Description List all General Resource profiles for the specified CLASS pair with access lists. JCL is produced which will re-create all profiles. Similar to RACF20 but without the limitation of prefix.

Usage racf91 member-class group-class

Output html jcl

Keywords recreate

Examples

```
C:\data\racf\dev>racf91 tcicstrn gciestrn
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of resource profiles ...
Parsing HTML...
Parsing JCL...
Removing temporary files...
```

The following output files have been created:

```
  racf91.jcl - JCL to recreate all TCICSTRN / GCICSTRN profiles
  racf91.htm - HTML list of all TCICSTRN / GCICSTRN profiles
```

C:\data\racf\dev>

## racf114

Description reports on CDT class

Usage racf114

Output html

Keywords CDT

Examples

```
C:\data\racf\dev>racf114
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of CDT profiles...
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf114.htm - HTML list CDT entries for class

C:\data\racf\dev>
```

## racf116

Description Lists out General Resource profiles listed in input file specified on command line. Input file to contain one entry per line, where each entry consists of class followed by profile. Useful for supplying tailored audit reports of specified profiles.

Usage racf116 filename

Output html

Keywords annotate audit

Examples

```
C:\data\racf\dev>racf116 general_resources
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of General Resource profiles...
.....
Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf116.htm - HTML list General Resource profiles specified in
general_resources

C:\data\racf\dev>
```

## racf123

Description Lists profiles listed in input file where input file has class followed by profile name, one per line  
Lists out members where racf116 doesn't.

Usage racf123 filename

Output html jcl

Keywords recreate annotate

Examples

```
C:\data\racf\dev>racf123 general_resources
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of General Resource profiles...
.....
Parsing HTML...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf123.jcl - JCL to recreate General Resource profiles specified
  racf123.htm - HTML list General Resource profiles specified in
general_resources

C:\data\racf\dev>
```

## racf125

Description Lists profiles in a class pair where the UACC is other than NONE. The output is a simple text file which can be used as the input file for racf123 to enumerate the members.

This was created with the intention of using it to review CICS classes where having UACC other than NONE can cause confusion when merging with profiles which do have a UACC NONE.

Usage racf125 member-class group-class

Output  
Keywords  
Examples

```
text
CICS UACC
C:\data\racf\dev>racf125 tcicstrn gicicstrn
Validating unload file racfdev...
Date of unload file = Thu Feb 22 23:09:06 2007
Compiling list of resource profiles...
Parsing TEXT...
Removing temporary files...

The following output files have been created:
  racf125.txt - Text list of profile names

C:\data\racf\dev>
```

# Certificates

## racf94

Description List all Digital Certificates by User ID and label.  
Usage racf94  
Output html  
Keywords label  
Examples C:\data\racf\dev>racf94  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of certificates...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf94.htm - HTML list of certificates by ID and label  
  
C:\data\racf\dev>

## racf95

Description List all Digital Certificates (unsorted). Shows key type and key size as well as start and end dates.  
Usage racf95  
Output html  
Keywords key-type key-size  
Examples C:\data\racf\dev>racf95  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of certificate data records...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf95.htm - HTML list of certificates, unsorted  
  
C:\data\racf\dev>

## racf96

Description List all Digital Certificate King Rings.  
Usage racf96  
Output html  
Keywords keyrings  
Examples C:\data\racf\dev>racf96  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of keyrings...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf96.htm - HTML list of certificate key rings  
  
C:\data\racf\dev>

## racf97

Description List all Digital Certificate Mappings.  
Usage racf97

Output html  
Keywords certificate-mapping  
Examples C:\data\racf\dev>racf97  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of Certificate Mappings Records...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf97.htm - HTML list of certificate mappings  
  
C:\data\racf\dev>

## racf98

Description List all Digital Certificate Trusts.  
Usage racf98  
Output html  
Keywords certificate-trusts  
Examples C:\data\racf\dev>racf98  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling list of certificate trusts...  
Parsing HTML...  
Removing temporary files...  
  
The following output files have been created:  
    racf98.htm - HTML list of certificate trusts  
  
C:\data\racf\dev>

## racf99

Description List all Digital Certificates sorted by expiry date. Now also includes User ID and certificate label which makes this a very useful report. The Expiry Date-Line shown in the report is calculated purely on the date and does not take into account the time. Now accepts optional command line parameter of a User ID to restrict the report to a subset of certificates owned by said User ID. Now also generates JCL to delete expired certificates. See also RACF109 and RACF112.  
Usage racf99 [OwnerID]  
Output html jcl  
Keywords expiry-date RACDCERT  
Examples C:\data\racf\dev>racf99  
Validating unload file racfdev...  
Date of unload file = Wed Jan 17 22:27:29 2007  
Compiling racf.\$\$\$...  
Compiling racf.tmp...  
Sorting expiry dates...  
Compiling list of certificate data records...  
Parsing HTML...  
Parsing JCL...  
Removing temporary files...  
  
The following output files have been created:  
    racf99.htm - HTML of all certificates sorted by expiry  
    racf99.jcl - JCL to delete expired certificates  
  
C:\data\racf\dev>

## racf100

Description Same as RACF99 but has addition of a cut-off-date so that you can list certificates up until the specified date. Note: unlike RACF99 which only creates JCL to delete expired certificates, RACF100 creates JCL to delete all listed certificates, so be careful you don't delete something you didn't mean to.  
Usage racf100 [yyyy-mm-dd]  
Output html jcl



## racf108

Description	This utility generates a report of certificates sorted by the owner user ID. It has a number of optional command line parameters. First is a user ID on which to report, no parameters means it generates a full report for all users. Then there are a couple of additional optional parameters of user ID to change to and HLQ for export datasets. Both these parameters are only relevant for the generated JCL. For each certificate found for the first user ID specified it generates the following RACDCERT commands, (1) a list of the cert on the user (2) an export to a dataset (3) a delete of the cert (4) an import back in from dataset to new user (5) finally a list of the cert on the new user.
Usage	racf108 [ID] [target ID for JCL] [HLQ for export datasets]
Output	html jcl
Keywords	owner RACDCERT
Examples	<pre>C:\data\racf\dev&gt;racf108 Validating unload file racfdev... Date of unload file = Wed Jan 17 22:27:29 2007 Compiling RACF.\$\$1... Compiling RACF.\$\$2... Compiling list of certificates... Parsing HTML... Parsing JCL... Removing temporary files...  The following output files have been created:   racf108.jcl - JCL to create export / import non CERTAUTH certificates   racf108.htm - HTML list of matching certificates  C:\data\racf\dev&gt;</pre>

## racf109

Description	This is a search facility for certificates. It accepts up to 9 search strings on the command line and reports back any certificates which match all search strings specified. The search is across the 4 fields that appear in the report, and scores a hit if found in any of the four fields, namely: User ID, label, common name (i.e. CN) within the certificate details, or expiry date. Only the User ID and label of any found certificates are displayed on the command line, but a full html report is always generated listing User ID, label, CN, and Expiry date. Updated to include listing of User ID who created the certificate. This has no relevance to RACF but can be useful to know. See also RACF112 and RACF99.
Usage	racf109 [search string1] [search string2] .. [search string9]
Output	html jcl
Keywords	search recreate RACDCERT

## Examples

```
C:\data\racf\dev>racf109 irrcerta
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling racf.tmp...
Compiling list of certificates...

UserID [irrcerta] Label [Verisign Class 3 Primary CA]
UserID [irrcerta] Label [Verisign Class 2 Primary CA]
UserID [irrcerta] Label [Verisign Class 1 Primary CA]
UserID [irrcerta] Label [RSA Secure Server CA]
UserID [irrcerta] Label [Thawte Server CA]
UserID [irrcerta] Label [Thawte Premium Server CA]
UserID [irrcerta] Label [Thawte Personal Basic CA]
UserID [irrcerta] Label [Thawte Personal Freemail CA]
UserID [irrcerta] Label [Thawte Personal Premium CA]
UserID [irrcerta] Label [IBM World Registry CA]
UserID [irrcerta] Label [Integrion CA]
UserID [irrcerta] Label [Verisign Class 1 Individual CA]
UserID [irrcerta] Label [Verisign International Svr CA]
UserID [irrcerta] Label [Identrus Interoperability CA]
UserID [irrcerta] Label [Entrust Secure Server Root CA]
UserID [irrcerta] Label [Equifax Secure CA]
UserID [irrcerta] Label [ICP-Brasil CA]
Parsing HTML...
Parsing JCL...
Removing temporary files...

Number of certificates = 17

The following output files have been created:
  racf109.jcl - JCL to create listed certificates
  racf109.htm - HTML list of certificates
  racf109-IRRCERTA.htm - copy of racf109.htm

C:\data\racf\dev>
```

## racf112

Description	List certificates showing owner, creator, certificate label and expiry date and sorted by certificate authority. Note: any CERTAUTH (or irrcerta) certificate which doesn't have certificates which have been issued under it (i.e. matching Common Names) are suppressed. See also RACF109 and RACF99.
Usage	racf112
Output	html
Keywords	certificates-by-issuer

## Examples

```
C:\data\racf\prod>racf112
Validating unload file racfprod...
Date of unload file = Mon Nov 20 18:16:42 2006
Compiling racf.tmp...
Compiling racf2.tmp...
Compiling list of certificates...

CA_common_name = [CN=Thawte Server CA]
.
CA_common_name = [CN=Thawte Premium Server CA]
.
CA_common_name = [CN=Thawte Personal Basic CA]
.
CA_common_name = [CN=Thawte Personal Freemail CA]
.
CA_common_name = [CN=Thawte Personal Premium CA]
.
CA_common_name = [CN=IBM World Registry Certification Authority]
.
CA_common_name = [CN=Integrion Certification Authority Root]
.
CA_common_name = [CN=Root CA]
..
CA_common_name = [CN=Root CA MQ]
..
CA_common_name = [CN=Identrus Root Interoperability Certificate Authority]
.
CA_common_name = [CN=Entrust]

CA_common_name = [CN=Autoridade Certificadora Raiz Brasileira]
.

Number of certificates = 4
Number of Certificate Authorities = 2

Parsing HTML...
Removing temporary files...

The following output files have been created:
  racf112.htm - HTML list of certificates by issuer

C:\data\racf\prod>
```

## racf119

### Description

Because the IRRDBU00 unload doesn't have any 'common name' information of signed certificates other than of the issuer, this report generates JCL to list all certificates using RACDCERT LIST commands. The output from this can then be used as input for racf120.

### Usage

racf119

### Output

jcl

### Keywords

common-name

### Examples

```
C:\data\racf\dev>racf119
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list JCL to list all certificates...
Parsing JCL...
Removing temporary files...

The following output files have been created:
  racf119.jcl - JCL with RACDCERT LIST commands to generate input file
for racf120 Common Name reporting

C:\data\racf\dev>
```

## racf120

**Description** Takes certificate output generated by RACF119 JCL and reports on Common Names. At the same time as generating the report based on the output of racf119, it also reconciles the certificates with the IRRDBU00 unload information and reports if the two are out of sync.

In addition to the html output it generates a text file of fixed format records that could be merged onto the IRRDBU00 for further processing if desired.

Format of text file is as follows:

Start	End	Comments
1	4	Record type for certificate common name (0010)
6	13	Certificate owner ID
15	46	Certificate label
48	111	Certificate common name

**Usage** racf120 filename

**Output** html text

**Keywords** common-name

**Examples**

```
C:\data\racf\dev>racf120 racdcert
Validating unload file racfdev...
Date of unload file = Wed Jan 17 22:27:29 2007
Compiling list of Certificate common names...
*** WARNING - racdcert output doesn't match unload output - refresh racdcert
listing using racf119 ***
Parsing HTML...
Removing temporary files...
```

The following output files have been created:

```
racf120.txt - Text unload records 0010 containing Common Names
racf120.htm - HTML list of certificates with Common Names
```

```
C:\data\racf\dev>
```

## Appendix A - Usage

racf00  
racf01  
racf02  
racf03  
racf04 [min\_count] [max\_count] | [no\_install\_data]  
racf05 [group]  
racf06 group [group-special]  
racf07 dataset  
racf08 search\_string1 [search\_string2] .. [search\_string8]  
racf09 user\_mask  
racf11 ID [all]  
racf12 class profile  
racf16 member-class group-class  
racf18  
racf19 member-class group-class  
racf20 member-class group-class prefix  
racf21 ID  
racf22 class  
racf23 prefix  
racf24  
racf25 member-class group-class prefix  
racf28 member-class group-class first-character(s)  
racf30  
racf32  
racf33 member-class group-class  
racf34 member-class group-class  
racf35 flatfile1 prefix1 flatfile2 prefix2 member-class group-class  
racf36 group1 group2  
racf37 ID member-class group-class [all]  
racf38  
racf42  
racf46 filename  
racf47 group1 group2 [group3]  
racf48 member-class group-class  
racf49 member-class group-class  
racf50 group  
racf51 member-class group-class prefix  
racf52 OwnerID  
racf53 group  
racf56 ID [ignore]  
racf58 Notify\_UserID | ALL  
racf59  
racf61 prefix id access  
racf62 member-class group-class prefix id access  
racf64 member-class group-class  
racf65 class  
racf66 userid  
racf67 member-class group-class prefix [id]  
racf68 filename  
racf69  
racf70 group  
racf71 dataset  
racf72  
racf73 group [connect1] [connect2] [connect3] [connect4] [connect5] [connect6] [  
racf75 group  
racf76 group  
racf77  
racf79  
racf80  
racf82 filename  
racf84  
racf85 member-class [group-class]  
racf86 filename  
racf87  
racf88 [group]  
racf89  
racf90  
racf91 member-class group-class

racf92 [DFLTGRP]  
racf93  
racf94  
racf95  
racf96  
racf97  
racf98  
racf99 [OwnerID]  
racf100 [yyyy-mm-dd]  
racf101 [OwnerID]  
racf102 filename  
racf103  
racf104  
racf105  
racf106  
racf107 filename  
racf108 [search ID] [target ID for JCL] [HLO for export datasets]  
racf109 [search string1] [search string2] .. [search string9]  
racf110  
racf111 group  
racf112  
racf113 filename  
racf114  
racf115 filename  
racf116 filename  
racf117 filename  
racf118  
racf119  
racf120 filename  
racf121 filename  
racf122  
racf123 filename  
racf124  
racf125 member-class group-class  
racf126  
racf127  
racf128  
racf129 filename [connect1] [connect2] [connect3] [connect4] [connect5] [connect6] [connect7]  
racf130 [DFLTGRP]  
racf131 [group]  
racfawk record\_type start\_column [!]search\_string  
racfdiag  
racfjcl text\_input\_file string1 string2 [string3]

# Index

## A

addsd.....	37, 39
adhoc.....	18, 23
annotate.....	31, 33, 41, 50
annotate-groups.....	25
APF.....	38
attributes.....	30
audit.....	30, 41, 50
audit-attributes.....	16
authority.....	21, 30

## C

catalogs.....	40
CDT.....	49
certificate-mapping.....	53
certificates-by-issuer.....	56
certificate-trusts.....	53
CICS.....	44, 51
clist.....	19
common-groups.....	21
common-name.....	57, 58
compare.....	45
connect-authority.....	24
connect-owner.....	24
create-date.....	25, 33

## D

delete.....	43, 46
dfltgrp.....	27
DFLTGRP.....	21, 22, 23, 26, 28, 31, 32, 35
diagnostics.....	18
discrete.....	15, 17, 39
dormant.....	28
dsmon.....	20, 25, 38
duplicate.....	45
duplicate-names.....	33

## E

expired.....	28
expiry-date.....	53, 54

## F

filter.....	37, 39, 42
friendly.....	47

## G

generic.....	39
generics.....	17
GID.....	24
globalaudit.....	16
group-count.....	23
group-special.....	21, 27

## I

installation-data.....	16, 20, 43
------------------------	------------

## K

keyrings.....	52
key-size.....	52
key-type.....	52

## L

label.....	52
last-access-date.....	25
limbo.....	32
listcat.....	40
listuser.....	30

## M

mask.....	37, 39, 42
-----------	------------

## N

Netview.....	36
never-logged-on.....	26, 35
non-prefixed.....	46, 47
NOPASSWORD.....	30
notify.....	16, 48

## O

OPID.....	33
orphaned.....	13
owner.....	16, 55
OwnerID.....	21

## P

password-interval.....	30
permit.....	38, 47
prefixed.....	43, 44, 45, 47
PROTECTED.....	30, 34

## R

RACDCERT.....	53, 55
racf00.....	13
racf01.....	13
racf02.....	13
racf03.....	20
racf04.....	20
racf05.....	28
racf06.....	20
racf07.....	37
racf08.....	28
racf09.....	28
racf100.....	53
racf101.....	54
racf102.....	33
racf103.....	33
racf104.....	25
racf105.....	34
racf106.....	34
racf107.....	40
racf108.....	55
racf109.....	55
racf11.....	14
racf110.....	35
racf111.....	25
racf112.....	56
racf113.....	25
racf114.....	49
racf115.....	40
racf116.....	50
racf117.....	26
racf118.....	17
racf119.....	57
racf12.....	42

racf120.....	58
racf121.....	26
racf122.....	35
racf123.....	50
racf124.....	17
racf125.....	50
racf126.....	26
racf127.....	27
racf128.....	17
racf129.....	27
racf130.....	35
racf131.....	36
racf16.....	42
racf18.....	29
racf19.....	42
racf20.....	43
racf21.....	14
racf22.....	43
racf23.....	37
racf24.....	29
racf25.....	43
racf28.....	44
racf30.....	44
racf32.....	15
racf33.....	44
racf34.....	45
racf35.....	45
racf36.....	21
racf37.....	45
racf38.....	29
racf42.....	15
racf46.....	30
racf47.....	21
racf48.....	46
racf49.....	46
racf50.....	21
racf51.....	47
racf52.....	15
racf53.....	22
racf56.....	37
racf58.....	16
racf59.....	16
racf61.....	38
racf62.....	47
racf64.....	47
racf65.....	48
racf66.....	30
racf67.....	48
racf68.....	38
racf69.....	22
racf70.....	23
racf71.....	39
racf72.....	48
racf73.....	23
racf75.....	23
racf76.....	24
racf77.....	24
racf79.....	31
racf80.....	39
racf82.....	31

racf84.....	31
racf85.....	49
racf86.....	32
racf87.....	24
racf88.....	32
racf89.....	16
racf90.....	39
racf91.....	49
racf92.....	32
racf93.....	33
racf94.....	52
racf95.....	52
racf96.....	52
racf97.....	52
racf98.....	53
racf99.....	53
racfawk.....	18
racfdiag.....	18
racfjcl.....	19
recreate.....	43, 46, 47, 49, 50, 55
RESTRICTED.....	30
revoked.....	22, 29
revoke-date.....	21

## S

search.....	28, 29, 44, 55
security.....	16
sorted.....	44
special.....	18
STARTED.....	34, 44
STDATA.....	44
summary.....	13, 18, 26, 48, 54

## T

trancodes.....	44, 45
tree.....	20, 25
TSO.....	35
TSOPROC.....	35

## U

UACC.....	37, 51
UAUDIT.....	30
UID.....	32
UID(0).....	34
user-deletion.....	30, 32
userid-mask.....	28, 31

## V

Visio.....	25
------------	----

## W

WARNING.....	15
--------------	----

## X

XREF.....	14, 15, 45
-----------	------------