

rac Command for z/OS UNIX

Author: Bruce R. Wells

RACF Security Server for z/OS

brwells@us.ibm.com

Last updated: 7/31/2007

<u>Change Date</u>	<u>Change Description</u>
08/18/2000	Introduction of rac
07/29/2007	Documented the prerequisite of implementing the RACF subsystem address space, and modernized the text.

Introduction

In the z/OS UNIX environment, one can only issue authorized TSO commands, including the RACF TSO commands, when entering the shell via the OMVS interface. This ability does not exist for users using telnet, rlogin, rsh, or rexec to establish their z/OS UNIX shell environment. Further, output from TSO commands, regardless of the environment, cannot be easily manipulated using standard shell programming techniques, such as the grep, sed, and awk commands, pipelines, and redirection of stdout.

The rac command allows RACF commands to be issued in any z/OS UNIX environment, and writes the command output to stdout, thus making it available for shell programming techniques.

Note that the tso cmd utility available on the [z/OS UNIX Tools and Toys](http://www.s390.ibm.com/products/oe/bpxa1toy.html) web page (<http://www.s390.ibm.com/products/oe/bpxa1toy.html>) can be used to issue RACF commands, but it requires APF authorization.

Description

rac is a z/OS UNIX executable that accepts a RACF command image and writes the command output to stdout. Entering rac with no arguments will display some brief help text. It can be invoked as a shell command, or from a shell script or REXX exec running in the UNIX shell. It uses the R_admin RACF callable service, and the issuer must have at least READ access to the IRR.RADMIN.<command-name> resource in the FACILITY class. The issuer is also subject to normal RACF command authorization.

The shell interprets several characters to have a special meaning. Some of these characters, particularly single quotes and parentheses, are frequently used in RACF commands. You can turn off the special meaning of a character (called *escaping* the character) by preceding it with a

backslash (“\”). This can be cumbersome however, and an alternative is to enclose the entire RACF command image within double quotes. See the examples below.

Note that R_admin limits its total output at 4096 lines. R_admin also dictates which commands can be issued through its interface (for example, the RACF TSO commands may be issued, but the RACF operator commands cannot). See z/OS Security Server RACF Callable Services for R_admin documentation.

Prerequisites

The R_admin callable service (IRRSEQ00) requires that the RACF subsystem address space be implemented. The RACF subsystem is required for other functions such as the RACF Remote Sharing Facility (RRSF) and RACF protection of operator commands. As such, the subsystem may already be implemented in your installation. If not, see z/OS Security Server RACF System Programmer’s Guide for instruction on implementing the subsystem.

Installation

Download the rac executable from the RACF downloads page. The executable is named rac.exe on the downloads page. Put it into the Hierarchical File System (HFS) on the z/OS system in the directory of your choice. The following examples use /bin/racf. When transferring the executable from your workstation to the HFS, make sure you do so in binary mode. Once on z/OS, you can rename it to anything you like, but we use “rac”.

Make rac available to whomever you wish.

If you wish to make rac available to all your users,

- Update the PATH variable in /etc/profile to include the directory in which you’ve stored the rac executable (e.g. Export PATH=/bin:/usr/lpp/java/J1.1.8/bin:/bin/racf:.
- Set the execute permission bits on for rac (e.g. chmod 755 /bin/racf/rac) and make sure users can search each component of the path name
- Define the appropriate FACILITY class profile(s): e.g. rac “rdefine facility irr.admin.* uacc(read)”

To restrict use to a subset of users, instruct them to update their PATH variable in their .profile, set the file permission bits appropriately, define the RACF profiles with UACC(NONE), and permit the users individually (or by group).

Examples

As a shell command

- 1) rac lu
- 2) rac rdefine unixpriv chown.unrestricted

3) rac altgroup authors omvs\(gid\451\)

4) rac "altgroup authors omvs(gid451)"

5) rac "addsd 'bruce.*' uacc(none)"

Note that, since rac writes command output to stdout, common UNIX techniques for processing command output can be employed for RACF commands:

6) rac listgrp sys1 > /u/mydir/sys1.output

7) rac "search class(facility)" | grep BPX

From REXX

The entire command must be enclosed within single quotes:

```
'rac lu barney'
```

In addition, parentheses must be escaped or quoted, just as above:

```
'rac alu barney name\(gumble\)'
'rac "alu barney name(gumble)" '
```

If you need to issue a RACF command which requires single quotes (as in example 5 above), it must be coded as

```
'rac "addsd 'bruce.*''"
```

That is, in addition to the entire command being enclosed within single quotes, and the RACF command image within double quotes, embedded single quotes must be doubled up.

Disclaimers, etc.

This program contains code made available by IBM Corporation on an AS IS basis. Any one receiving this program is considered to be licensed under IBM copyrights to use the IBM-provided code in any way he or she deems fit, including copying it and redistributing it, except that it may be neither sold nor incorporated within a product that is sold. No license under any IBM patents or patent applications is to be implied from this copyright license.

The software is provided "as-is", and IBM disclaims all warranties, express or implied, including but not limited to implied warranties of merchantability or fitness for a particular purpose. IBM shall not be liable for any direct, indirect, incidental, special or consequential damages arising out of this agreement or the use or operation of the software.

A user of this program should understand that IBM cannot provide technical support for the program and will not be responsible for any consequences of use of the program.