Introduction to RACF Administration on z/VM

James Vincent
Velocity Software, Inc
VM Workshop – June 2026

Agenda

- Overview – benefits of an ESM
- Security Policy
- Administration and Recommendations
- Security Think to RACF Reality
- Keeping RACF healthy

Some awesome references

- Alan Altmark
  https://www.ibm.com/support/pages/zvm/devpages/altmarka/present.html
  https://vmworkshop.org/2023/present/racfrtwy.pdf
- Bruce Hayden https://www.ibm.com/support/pages/zvm/education/roadmaps/int-racf.pdf Search the ‘net for: IBM bruce hayden racf

Overview

Why do you need an External Security Manager like RACF?

- There are miscreants out there (and maybe working with you)
- Companies insist – they do not want to be on a headline
  - Zero Trust – *sometimes* makes a lot of sense even if it is painful
- The system should be rock solid in regards to security

Basic z/VM (CP) security isn't enough in most cases

- An ESM provides you more
  - Controls and auditing
  - Encrypted passwords
  - Multi-Factor Authentication (MFA)
  - Access control for MDISKs instead of using passwords

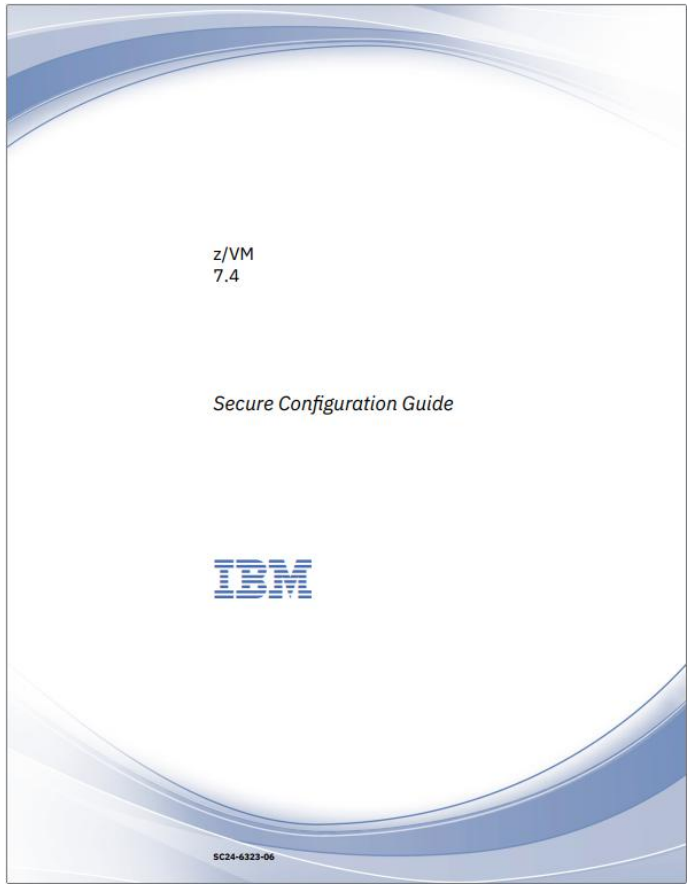
Overview

What you secure is by Security Policy

- Normally dictated by the CIO or CSO
  - They better have people involved that *understand* z/VM
- System audits determine compliance
  - Goal is to make the Auditors HAPPY
  - If not, plan on a lot of meetings and paperwork…

Who implements the Security Policy

- Usually a Security Admin group/department
  - Independent of “System Programmers” in a lot of cases
  - Access to resources is by request and justification (and waiting)
    - “Break Glass” for Emergencies



z/VM 7.4 Secure Configuration Guide SC24-6323-06

Overview

A 2 AM thought…

RACF is like your cell phone:

- Most people usually know how to use it for basic stuff (answer a call, send a text)
- It can do so many more things that most don't know about (or ever use)
- If you only use part of it once or twice a year, it is very easy to forget how to use it

Overview

The Basics

- RACF is a pre-installed feature of z/VM (priced option)
  - RACF releases are specific to the release of z/VM
- Maintenance userid (7VMRACrr) manages install & service
- **RACFVM** userid controls all resources, security DBs and SMF records (audit)
  - Primary and backup DBs
  - Primary and backup SMF disks
- RACMAINT userid for security services if RACFVM is broken (and more)
- RACFSMF used for auding reports and archiving SMF records

Overview

Installation and Configuration of RACF is a session itself…

- RACF is installed by default but not set up
- Follow the RACF Program Directory for *your* configuration!
- Read and follow Alan Altmark's "*RACF: The Right Way*"

IBM z/VM Education

z/VM: How to get started with RACF - Enablement

https://www.youtube.com/watch?v=i2pWf6ABFlk

z/VM: How to get started with RACF - Configuration

https://www.youtube.com/watch?v=JXdDfekYIr8

Some terminology

- A resource: a “thing” or place where data resides (MDISK) or passes through (terminal), or functions (commands)
- Classes – classifies a resource (semi-common ones):
  - VMBATCH - Alternate userid facility
  - **VMCMD - Some CP commands and more**
  - VMDEV – Real system devices
  - VMLAN – VSWITCHes and Guest LANs
  - **VMMDISK - Minidisks**
  - VMNODE - Target other RSCS nodes
  - VMRDR – Spooling to other users (e.g., RDR files)
  - VMSEGMT - Access to restricted segments (Class R)
  - **VMXEVENT – Profiles for commands and auditing**
  - FACILITY - RACROUTE
  - **SURROGAT – Logon-BY or LOGONBY**

Some terminology

- Some RACF options you may see on RAC commands:
  - CLASS() – the resource class you are working with; a collection of like ‘things’
  - ID() – one or more userids that the rule applies to or impacts
  - ACCESS() - the permission or type of access
  - UACC() – the default access for the resource
  - DFLTGRP() – for a user, the default group name
- Permissions for ACCESS() UACC()
  - ALTER – MW link (disks)
  - CONTROL – R/W with M (disks, no MW); promiscuous mode (VMLAN)
  - UPDATE – R/W access (disks); allows send/transfer of spool files
  - READ – R/O access (disks); allows command (VMCMD), allows logon-by (SURROGAT)
  - NONE – no access

Security Policy and RACF Setup

Remember, conform to your Security Policy!

Configure RACF for only resources users own and really use

- Easiest: Do this when you set up RACF
  - RPIDIRCT utility produces RPIDIRCT SYSUT1 file from your USER DIRECT source
    - RPIDIRCT is past-due for an update to align better to the needs of most sites
  - **See Alan Altmark's *"RACF: The Right Way"***
    - Turn off (NOCTL) the events you do not want/need; remove classes not needed
- Changes can be done later, but a *bit* more complicated
  - Adding is easy; removing later is a pain

Security Policy and RACF Setup

Passwords or Passphrases?

- 1 to 8-character passwords aren't enough anymore
- Use PHRASE('…') instead of PASSWORD(…)

  RAC ADDUSER JAMES DGLTGRP(SYS1) UACC(NONE) PASSWORD(TOOEASY)

  instead…

  RAC ADDUSER JAMES DGLTGRP(SYS1) UACC(NONE) PHRASE('SomeObscureString')
- You can add your own rules for passphrases (default min length 14)
  - Need to set up the ICHPWX11 exit
  - See the *z/VM RACF Security Server Systems Programmer's Guide*, Chapter 6

Security Policy and RACF Setup

Passwords/Passphrases

- Set up encryption (must be set **before** user profiles are created)
- Set minimum age / controls (example shown)
  - You can always change those later if needed

Example, near the top of RPIDIRCT SYSUT1, add:

```
SETROPTS PASSWORD(ALGORITHM(KDFAES))
SETROPTS PASSWORD (MINCHANGE(1) REVOKE(3) INTERVAL(30) HISTORY(45))
```

*Can only change password once a day, revoked on the 4th bad attempt, require new passwords every 30 days and keep 45 old passwords so they can't be re-used

Security Policy and RACF Setup

Set up for LOGON BY (Surrogate) access!

- Improves security, auditing and ease of use overall
- Set Up Generic Profiles
  SETROPTS GENERIC(*) GENCMD(*) NOADDCREATOR GRPLIST
- z/VM System Admins – usually there is more than one
  - Create a GROUP!
  - Allow the users in that group to use LOGON BY to any userid in the system that doesn't already have a discrete LOGONBY profile in the SURROGAT class
- Linux Admins – group them too
  - Allow the group to use LOGON BY to target userids with a pattern (ie, LNX*)
  - See RACFVARS if the target userids do not follow a pattern

Security Policy and RACF Setup

Adjust OPERATOR to be LOGON BY only

- PASSWORD(LBYONLY)
- Set NOPASSWORD NOPHRASE attributes
- Generic LOGONBY.** access for systems group should be set up

ADDUSER OPERATOR DGLTGRP(SYS1) UACC(NONE) PASSWORD(LBYONLY)
ALTUSER OPERATOR NOPASSWORD NOPHRASE

QUIZ: If RACF is down, then how do I get on OPERATOR ?!
QUIZ: What is the password for in the CP directory if RACF is active?

Security Think to RACF Reality

RACF is for the masses, is *not* intuitively obvious

There needs to be a connection between Security Thinking and RACF

- "Deny JAMES access to TCPMAINT 592"
  - RAC PERMIT TCPMAINT.592 CLASS(VMMDISK) ID(JAMES) ACCESS(NONE)
- "Turn on auditing for Diagnose 0x88"
  - RAC RALTER VMXEVENT EVENTS1 ADDMEM(DIAG088/AUDIT)
    - assuming you have EVENTS1 defined
  - RAC SETEVENT REFRESH EVENTS1
    - RACF usually will tell you a refresh is required before

Security Think to RACF Reality

The “thing” is the resource RACF will protect

- TCPMAINT 592
- TCPMAINT LOGONBY

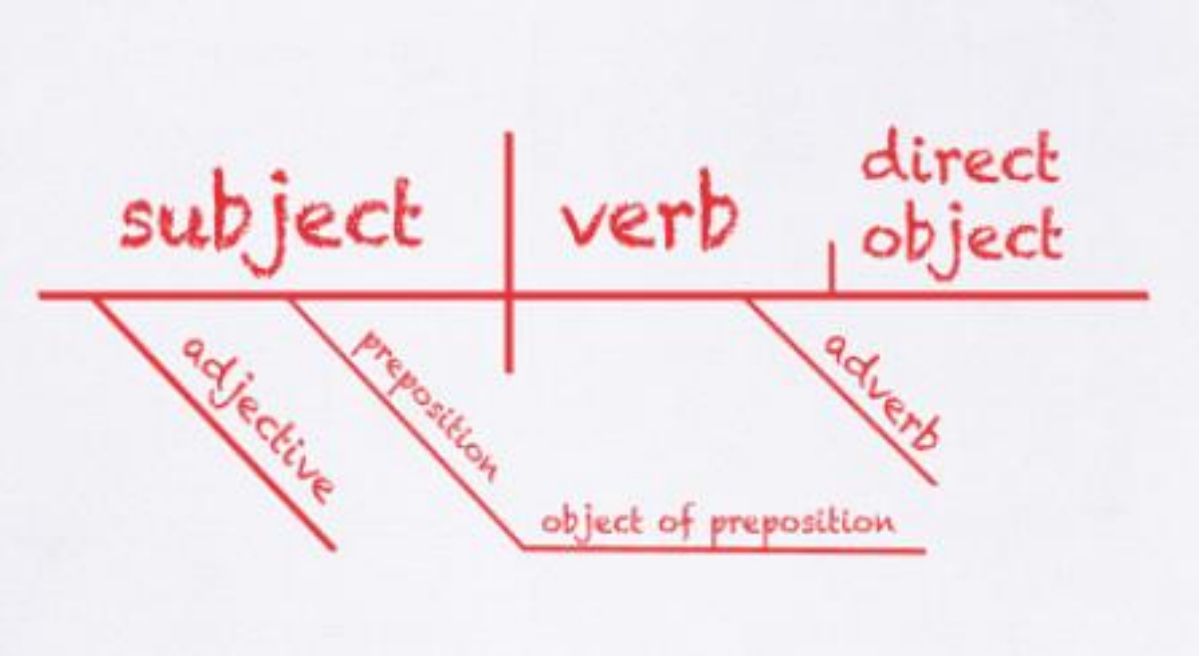
The “target” is the userid that the rule will apply to or impacts

Try to think of what is needed in “plain English”, then break it down for RACF

*Sentence Diagramming

Security Think to RACF Reality

Remember these?



Security Think to RACF Reality

"Deny JAMES access to TCPMAINT 592"

- **Deny** – the ACCESS() to the resource
- **JAMES** – the target ID() the rule will impact
- (to) **TCPMAINT 592** – the resource, which is an MDISK, which is in the class VMMDISK

Translates into "update the permission for TCPMAINT.592 in VMMDISK for JAMES to NONE":

RAC PERMIT TCPMAINT.592 CLASS(VMMDISK) ID(JAMES) ACCESS(NONE)

*If the base resource profile is not defined, use RDEFINE to create it

Good place for an AI…

Security Think to RACF Reality

Asked ChatGPT: Translate "Deny JAMES access to TCPMAINT 592" into a RACF command to run on z/VM

And its answer? WRONG! Use your brain, not an AI – please!

On z/VM RACF, the command would be:

```
PERMIT TCPMAINT CLASS(592) ID(JAMES) ACCESS(NONE)
```

If the resource profile does not already exist, you may need to define it first with `RDEFINE`, but for simply denying an existing permission, `PERMIT ... ACCESS(NONE)` is the standard RACF command.

Basic Use

VMMDISK - minidisk control is one of the commonly used classes

- A good Directory Manager will automatically interact with RACF
  - Adding/Deleting MDISKs will RDEFINE or RDELETE the profile for the owner/disk

The usual steps:

1. RAC RDEFINE VMMDISK owner.vaddr OWNER(rscown) UACC(default)
   - owner.vaddr is the userid and MDISK address – do not pad addresses with leading zeroes
   - rscown is set to the issuing userid if not defined; identifies who created the profile
   - UACC(NONE) is recommended for disks; UACC(READ) is good for public disks
2. RAC PERMIT owner.vaddr CLASS(VMMDISK) ID(owner) ACCESS(ALTER)
   - Give the owner of the disk the ability to access their own disk !!
3. RAC PERMIT owner.vaddr CLASS(VMMDISK) ID(userid) ACCESS(auth)
   - Define permissions for other userids – could be READ, UPDATE…etc
   - You can specify more than one userid like: ID(RICKB SAM JOE)

VMMDISK

Removing userid permissions

- RAC PERMIT owner.vaddr CLASS(VMMDISK) DELETE ID(userid)
  - ‘userid’ will default to the UACC permission on the profile

Query the profile and who has permissions set

- RAC RLIST VMMDISK owner.vaddr ALL
  - ALL will include all kinds of extra info; at the bottom is the list of users and their access for the resource that have been defined

Basic Use

SURROGAT - LOGONBY control is another of the commonly used classes

"Allow JAMES to use LOGONBY for RICH's userid"

- **Allow** – the ACCESS() to the resource
- **JAMES** – the target ID() the rule will impact
- **LOGONBY RICH** – the resource, which is an action-thing

Translates into "update the permission for LOGONBY.RICH in SURROGAT for JAMES to READ":

RAC PERMIT LOGONBY.RICH CLASS(SURROGAT) ID(JAMES) ACCESS(READ)

*If the base resource profile is not defined, use RDEFINE to create it

Basic Use

SURROGAT

- Remember, for easier controls for Admins, set up generic group profiles
- Any userid *with* a SURROGAT profile define is *excluded* from the generic group
  - "Real People" should always have a SURROGAT profile defined
    - **RAC RDEF SURROGAT LOGONBY.userid**
  - But wait… do you want that person to be able to logon with a passphrase (or password)?!
    - **RAC PERMIT LOGONBY.userid CLASS(SURROGAT) ID(userid) ACCESS(READ)**
- Any userid without a specific SURROGAT profile gets the generic profile by default (ie, LOGONBY.** (G))

Keeping RACF Healthy

RACF does not generally clean up after itself

- Finding the trash is not that easy

One small userid with a few RACF rules set

Rule Report for EXAMP on VSIVM3

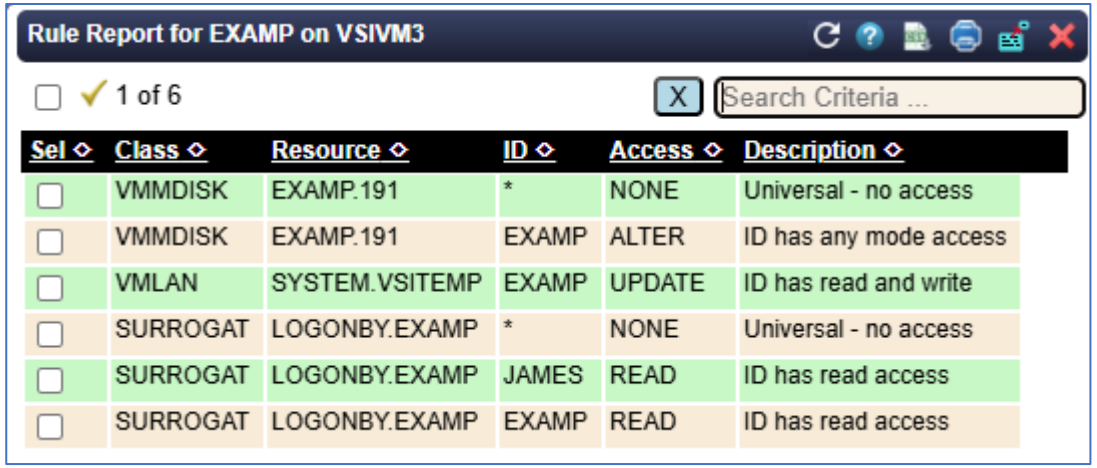
1 of 9

X Search Criteria ...

Sel	Class	Resource	ID	Access	Description
	USER	EXAMP			User entry in Security Manager
	GROUP	SYS1	EXAMP		DEFAULT Security Group the user is part of
	GROUP	@TEST1	EXAMP	USE	Security Group the user is part of
	VMMDISK	EXAMP.191	*	NONE	Universal - no access
	VMMDISK	EXAMP.191	EXAMP	ALTER	ID has any mode access
	VMLAN	SYSTEM.VSITEMP	EXAMP	UPDATE	ID has read and write
	SURROGAT	LOGONBY.EXAMP	*	NONE	Universal - no access
	SURROGAT	LOGONBY.EXAMP	JAMES	READ	ID has read access
	SURROGAT	LOGONBY.EXAMP	EXAMP	READ	ID has read access

No Directory Manager:

- You remove the directory
  - **ALL** the RACF rules are still there since there is no CP/RACF directory-change connection
- You remember to delete the RACF user profile (RAC DELUSER EXAMP)
  - You would *think* that if you delete the user, RACF would clean up itself!

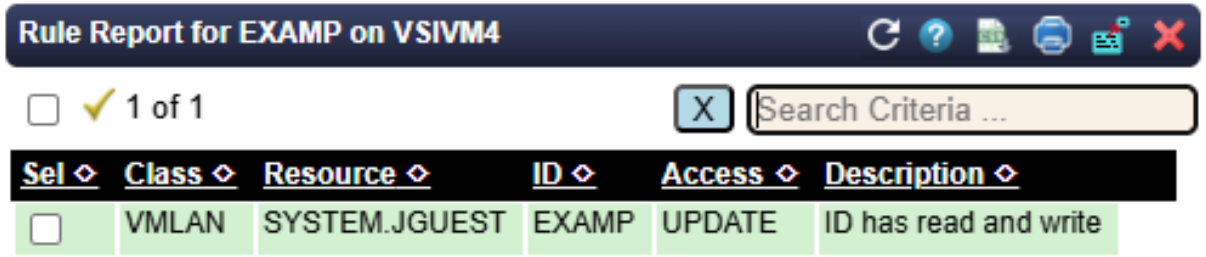


Sel	Class	Resource	ID	Access	Description
	VMMDISK	EXAMP.191	*	NONE	Universal - no access
	VMMDISK	EXAMP.191	EXAMP	ALTER	ID has any mode access
	VMLAN	SYSTEM.VSITEMP	EXAMP	UPDATE	ID has read and write
	SURROGAT	LOGONBY.EXAMP	*	NONE	Universal - no access
	SURROGAT	LOGONBY.EXAMP	JAMES	READ	ID has read access
	SURROGAT	LOGONBY.EXAMP	EXAMP	READ	ID has read access

Keeping RACF Healthy

With a Directory Manager

- zDIRECT (ZDIRECT DELUSER EXAMP or via zPRO UI)
  - RACF is clean – no dangling rules
- DIRMAINT (DIRM FOR EXAMP PURGE)
  - Cleaned up pretty well – except for VMLAN (VSWITCH/GLAN) rules



- DIRMAINT via zPRO
  - RACF is clean

Keeping RACF Healthy

Finding Orphaned RACF rules

- Use zPRO from Velocity Software
- Unload the RACF database - IRRDBU00
  - Post-process the output (requires a decoder ring and some coding)
- Scan the database manually
  - Requires some coding
    - Search for all Users and Groups ie, EXEC RAC SEARCH CLASS(USER) FILTER(*)
    - RAC RLIST for VMMDISK, VMCMD, VMBATCH, VMLAN, SURROGAT (etc)
    - Match userids found in the RLISTs with the Users/Groups from the search; any that fell out are orphans
      - From the RLIST output, determine all the rules for the orphan, then delete them after validating

Keeping RACF Healthy

Auditors want reports and historical data

- Critical for System Admin's when something *weird* happens
  - Missing/changed rules, etc

Set up RACFSMF to keep the audit disks clean and build reports

- Refer to and follow: "*RACF Security Server Auditor's Guide*"
- For very active systems, consider
  - When the active Audit disk fills, it will XAUTOLOG RACFSMF to switch & clear
  - Larger RACF 301/302 SMF disks
  - Running RACFSMF more than weekly

Keeping RACF Healthy

RACFSMF tips:

- Report on *everything* – violations *and* successes
  - Auditors or issues detected are usually from *n* weeks ago and they want to see details
- Move the reports off-platform and retain for up to one year
- Consider running snapshots of VMMDISK and SURROGAT definitions
  - Retain for recovery purposes in case someone goofs up

Summary

- Your Security group will likely be driving the bus; make *them* happy
- RACF may not be intuitively obvious to the occasional user
- With a little practice and ‘thinking about it’, things become clearer
- Setting up RACF correctly at the start is the very best way to have a less painful experience
- Keep the DB cleaned up; Run and retain the SMF reports
- If all else fails, take a deep breath, have an adult beverage and post a question on the IBMVM listserv
- Go deeper: Reading SMF reports for auditing; RACF dead or dying,

Questions?

(Ask Alan…)

james@velocitysoftware.com