BMC Mainframe: RACF for z/OS Systems Programmers

COURSE ABSTRACT

COURSE CODE

» MGRS-RAZP-2021

APPLICABLE VERSIONS

» Not Applicable

DELIVERY METHOD 🌐

» Instructor-led Training (ILT)

COURSE DURATION 🌐

» 3 Days

PREREQUISITES

» BMC Mainframe: RACF Administration & Auditing
» Full understanding of RACF at a conceptual level and be familiar with all the RACF commands and how they are utilized.

RECOMMENDED TRAININGS

» NA

Course Overview

The course is developed and delivered by © RSM Technology.

This advanced three-day course, designed, written, and presented by specialist RACF consultants, specifically focuses on RACF from a Systems Programmer's perspective.

The course provides a uniquely detailed insight into the technical architecture of RACF and RACF's relationship to z/OS. It describes and explains how RACF is implemented, and how it can be customized using standard RACF facilities.

Extensive hands-on practicals accompany the theory sessions, with each student having their own RACF system to customize.

Target Audience

» Systems Programmers
» Administrators

Learner Objectives

» Describe the RACF architecture, its components, and facilities
» Customise RACF to meet the requirements of their organisation and its environment
» Describe and use all the RACF Utilities, using JCL and REXX/CLIST
» Identify how the operation of RACF changes when running in a parallel sysplex
» Describe and explain the IPL process and the security issues associated with facilities such as APF, PPT, System Exits and Linklist
» Describe the components of the RACF database

For more information about BMC Education Services, visit **www.bmc.com/education**.

BMC Mainframe: RACF for z/OS Systems Programmers

COURSE ABSTRACT

COURSE ACTIVITIES
» Classroom Presentations
» Demonstration

BMC MAINFRAME INFRASTRUCTURE AND PLATFORMS LEARNING PATH
» https://www.bmc.com/education/courses/find-courses.html#filter/%7B%22type%22%3A%22edu-specific-types-159150236%22%7D

CERTIFICATION PATHS 🌐
» This course is not part of a BMC Certification Path

DISCOUNT OPTIONS 🌐
» Have multiple students? Contact us to discuss hosting a private class for your organization
» **Contact us for additional information** 🌐

Course Modules

What is RACF?
» Why do we need security?
» What does security provide?
» How does RACF work?
» RACF Profiles
» RACF classes
» How many RACF classes?
» Controlling access
» RACF commands

z/OS Technical Overview
» z/OS controls & drivers
» The IPL process
» PARMLIB & IPLPARM
» Display IPLINFO
» LOADxx & IODF
» System parameter list IEASYSxx
» What is APF?
» Defining an APF authorized library
» Program Properties Table
» Linklist
» Dynamic changes
» SMFPRMxx
» System exits
» In-storage profiles

» RACLIST & GENLIST
» Group tree in storage
» ACEE data in memory

The RACF Database
» The RACF database
» Database format
» Database templates
» RACF templates
» Issues
» Dynamic template objectives
» New template support
» RACF initialisation
» IRRMIN00
» Multiple database support
» RACF database sharing
» The RVARY command
» RVARY passwords
» RACF FAILSOFT processing
» Database backup & recovery

RACF in a Sysplex
» RACF and sysplex
» Basic sysplex
» Parallel sysplex
» RACF communication

» RACF data sharing
» RACF data sharing problems
» The four sysplex modes
» The RACF database name table
» Coupling Facility structures
» Defining Coupling Facility structures
» In-storage profiles reminder
» RACLISTed profiles via RACROUTE
» In-storage profiles and sysplex
» Introducing RACGLIST
» RACGLIST and REFRESH
» Using RACGLIST

RACF Modules
» RACF control tables
» RACF modules
» ICHRDSNT
» Using the IRRPRMxx member
» IRRPRMxx parameters
» ICHRDSNT - example
» ICHRRNG
» Class Descriptor Table (CDT)
» Dynamic CDT
» Defining a Dynamic CDT
» Rules

For more information about BMC Education Services, visit **www.bmc.com/education**.

BMC Mainframe: RACF for z/OS Systems Programmers

COURSE ABSTRACT

- POSIT values
- New segment CDTINFO
- CDTINFO options
- Managing Dynamic CDTs
- Migration Utility (CDT2DYN)
- ICHRFR01
- Normal rules apply
- ICHRIN03
- The Started Task Table
- ICHAUTAB
- ICHNCV00
- ICHSECOP

RACF Utilities
- IRRUT100
- IRRUT100 example output (Group)
- IRRUT100 example output (User)
- IRRUT200
- IRRUT200 example JCL
- IRRUT200 example output
- IRRUT400
- IRRUT400 example JCL
- IRRADU00
- IRRADU00 example JCL
- ICHDSM00
- ICHDSM00 example JCL
- IRRDBU00
- IRRRID00
- IRRRID00 JCL
- BLKUPD
- IRRBRW00
- IRRBRW00 JCL
- DFSORT ICETOOL utility
- Using ICETOOL
- ICETOOL - sample JCL
- ICETOOL report
- RACFICE package
- Sample report - ICETOOL keywords
- The Audit Reporting Tool
- ART main menu
- SMF UNLOAD
- XML output format support with SMF UNLOAD

- SMF UNLOAD - Job
- SMF UNLOAD - XML document
- Browsing an XML document
- Using an XSLT stylesheet

RACF Control Blocks
- RACF control blocks
- RACF Communications Vector Table (RCVT)
- Finding the RCVT
- Understanding the RCVT
- Data in the RCVT
- RCVT vs ICB
- SAF Vector Table (SAFV)
- Finding the SAFV
- Accessor Environment Element (ACEE)
- Where's my ACEE?
- ASXBSENV
- TCBSENV
- Local Control Block
- Which ACEE is used?
- Which ACEE do I need?
- Caveat ACEE
- Finding the active ACEE
- Security Token
- Security Token contents
- Security Token uses
- ACEE versus Token

RACF Macros
- RACF macros
- Macro interfaces
- The MVS router (SAF)
- RACF macros
- What do they DO?
- RACF macros: RACHECK, RACINIT, RACLIST, FRACHECK, RACDEF, RACSTAT
- RACROUTE additions
- ICHEINTY
- The RACROUTE interface
- RACROUTE MF= styles
- SAF Parameter list (SAFP)
- Initializing SAFP
- SAFP setup

- SAF Work Area (SAFW)
- SAFW setup
- History of REQSTOR & SUBSYS
- Using REQSTOR & SUBSYS
- Setting up REQSTOR and SUBSYS
- Other RACROUTE information
- The ACEE - AGAIN!
- Return codes
- REQUEST=Verify
- RACINIT ENVIR= options
- RACINIT ENVIR=CREATE
- Who do you create?
- RACINIT STAT=
- ENVIR=CREATE ACEE=
- Sample user/password=
- Sample with PASSCHK=NO
- Sample with Token
- Create SESSION=
- Create with TERMINAL=
- POE=
- TERMINAL= vs POE=
- Sample with POE=
- What about IP addresses?
- RACINIT ENVIR=DELETE
- ENVIR=DELETE ACEE=
- Sample DELETE
- REQUEST=AUTH
- CLASS=
- ENTITY/ENTITYX
- ENTITY(X) examples
- Sample RACHECK

RACF Exits
- Dataset RACF exits
- ICHRTX00/01
- Pre-processing for ICHRTX00
- ICHRTX00: input, output
- Pre-exit commonalities
- Post-exit commonalities
- Pre- to post- communication
- Work area pointer
- From post- to pre-

For more information about BMC Education Services, visit **www.bmc.com/education**.

BMC Mainframe: RACF for z/OS Systems Programmers

COURSE ABSTRACT

- » 'Gotchas' for SVC exits
- » Need some input
- » Finding the parameter list
- » Coding RACF exits
- » RACF command exit (IRREVX01)

- » What's a 'dynamic exit'?
- » RACF IRREVX01 dynamic exit
- » What can you do in the exit?
- » IRREVX01 parameter list
- » The exit command buffer

- » Using the ACEE passed in exit
- » Testing your command exit
- » Sample SETPROG command

For more information about BMC Education Services, visit **www.bmc.com/education**.