



z/VM Directory Tricks and Techniques

Sam Cohen
Levi, Ray & Shoup, Inc.
Email: sam.cohen@lrs.com



Agenda

- z/VM Background
- z/VM Directory
- Smaller/Faster Directory Builds
- Strengthening the security environment
- Preparing for an external security manager

© 2025 Levi, Ray & Shoup, Inc.

This session is focused on learning more about the z/VM Directory, along with tips and techniques to improve directory compilation time, directory size and directory-managed security. These techniques should be considered regardless of implementing an external security manager.



z/VM Background

© 2025 Levi, Ray & Shoup, Inc.

This section is a very brief summary from the Introduction to z/VM presentation available to VM Workshop participants



z/VM Background

- z/VM's CP (Control Program) provides for management of real resources and definition of virtual machines with (only) virtual resources
- CP can define virtual hardware where there is no equivalent in the real hardware
- More granular/flexible than Logical Partitions (LPAR)

© 2025 Levi, Ray & Shoup, Inc.



z/VM Background

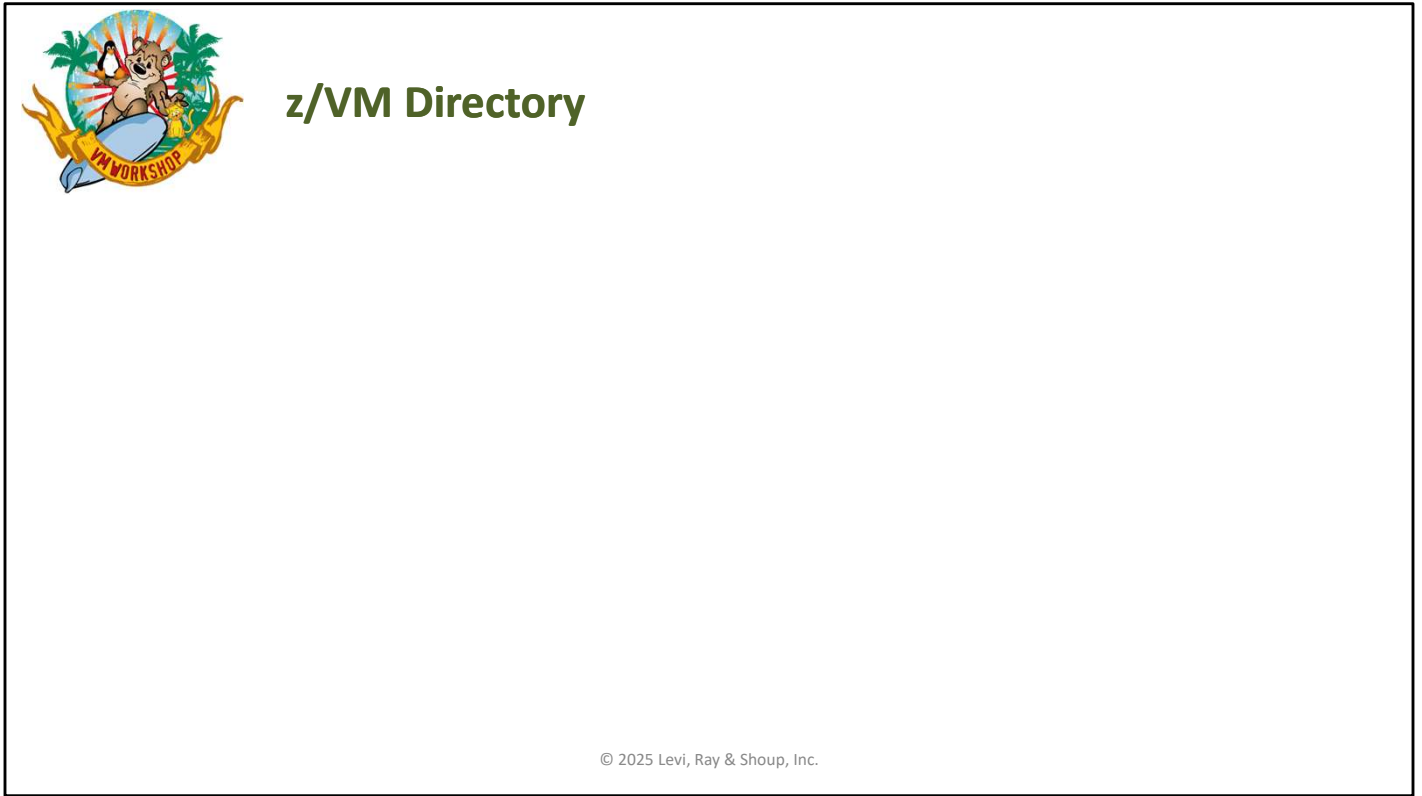
- z/VM's CP (Control Program) provides for management of real resources and definition of virtual machines with (only) virtual resources
- CP can define virtual hardware where there is no equivalent in the real hardware
- More granular/flexible than Logical Partitions (LPAR)

System Startup

- Load from device containing CP nucleus (&SYSRES)
- CP reads file on the System Parameter device (&SYSPARM) to determine resources and environment (default file: SYSTEM CONFIG)
- CP reads previously-compiled directory of virtual machines (allocated as DRCT space on &SYSRES)
- CP automatically starts virtual machines specified in SYSTEM CONFIG

© 2025 Levi, Ray & Shoup, Inc.

At system startup, you point to a text file (default name: SYSTEM CONFIG) that provides definitions for the z/VM environment. Some system variables (starting with "&") are automatically defined for use in SYSTEM CONFIG and in the VM Directory



Let's look at the VM Directory, its function and its capabilities



z/VM Directory

Function

- Define Virtual Machines to be created/managed/deleted by CP on command
- Define access to real and virtual resources for the specified Virtual Machine
- Provide mechanisms to allow:
 - Access to another virtual machine’s resources
 - Communications to/from other virtual machines
 - A different virtual machine to access this virtual machine (surrogacy)
 - Dynamic definitions of additional virtual resources
 - Visibility to the world of real resources instead of only virtual resources

© 2025 Levi, Ray & Shoup, Inc.

If a system is started without a z/VM Directory, the system operator (virtual machine “OPERATOR” by default) must manually access a device (disk, tape, card reader) that contains an operating system, load that operating system into its virtual machine, then build/assemble a Directory so other virtual machines can be started



z/VM Directory

V.M. Authentication

- Userid/Password combination
- Minidisk passwords

© 2025 Levi, Ray & Shoup, Inc.

Authentication is proving who you are by knowing an access “key” (e.g. password)



z/VM Directory

V.M. Authentication

- Userid/Password combination
- Minidisk passwords

V.M. Authorization

- Resources (Real and Virtual)
- CP Commands via CP Command Classes (A-G + installation-defined classes)

© 2025 Levi, Ray & Shoup, Inc.

Authorization is what you can access or do



V.M. Authentication

Each virtual machine is defined by a USER or IDENTITY statement

- Contains name of virtual machine (userid) and logon password
- Contains passwords for defined minidisks
 - Positional entries on MDISK statement
 - Read password
 - Write password
 - Multiuser password
 - Value of “ALL” means unrestricted access

© 2025 Levi, Ray & Shoup, Inc.

In a clustered (Single System Image (SSI)) environment, a USER can only be logged on a single time in the entire cluster, while an IDENTITY definition can be logged on to any/all cluster members at the same time. An IDENTITY user with CMS minidisks must have unique minidisks in each cluster member, since the CMS filesystem does not support multi-write access with integrity.



V.M. Authorization

Real Resources

- Access controlled by:
 - Hardware Activation Profile and I/O Subsystem Configuration
 - What is allocated/authorized to the Logical Partition (LPAR)
 - z/VM Directory
 - Use of those resources by virtual machines
 - » Minidisk definitions
 - » Link to other users' minidisks (mdisk passwords not required)
 - Issue out-of-class CP commands affecting real resources (such as "ATTACH")

© 2025 Levi, Ray & Shoup, Inc.

There is no requirement for an I/O Definition File (IODF) in z/VM, it is flexible enough to automatically add/delete devices upon a change to the IOCDS.



V.M. Authorization

Virtual Resources

- Access controlled by:
 - SYSTEM CONFIG file
 - Restricted vs. Unrestricted virtual devices (example: Guest LANs)
 - z/VM Directory
 - Virtual machine memory sizes
 - Inter-user communication
 - Virtual devices
 - Dynamically-defined virtual devices
 - Virtual terminals
 - Virtual NICs
 - Virtual CTCs
 - Virtual disks in memory
 - Are deleted when virtual machine is logged off

© 2025 Levi, Ray & Shoup, Inc.

Some connectivity can be restricted. For example, connecting a Virtual CTC to another virtual machine can be limited to a specific virtual machine based on the directory entry



z/VM Directory

© 2025 Levi, Ray & Shoup, Inc.

If a system is started without a z/VM Directory, the system operator (virtual machine "OPERATOR" by default) must manually access a device (disk, tape, card reader) that contains an operating system, load that operating system into its virtual machine, then build/assemble a Directory so other virtual machines can be started



z/VM Directory

Source

- Provided by IBM on MAINT 2CC disk (accessed as filemode “C” on IBM-supplied directory entry for MAINT)
- Editable via XEDIT

© 2025 Levi, Ray & Shoup, Inc.

If a system is started without a z/VM Directory, the system operator (virtual machine “OPERATOR” by default) must manually access a device (disk, tape, card reader) that contains an operating system, load that operating system into its virtual machine, then build/assemble a Directory so other virtual machines can be started



z/VM Directory

Source

- Provided by IBM on MAINT 2CC disk (accessed as filemode “C” on IBM-supplied directory entry for MAINT)
- Editable via XEDIT

Compilation

- Compiled using DIRECTXA module
- Output placed on DRCT space on virtual address defined in the directory source
- No listing output
- No disassembler

© 2025 Levi, Ray & Shoup, Inc.

If a system is started without a z/VM Directory, the system operator (virtual machine “OPERATOR” by default) must manually access a device (disk, tape, card reader) that contains an operating system, load that operating system into its virtual machine, then build/assemble a Directory so other virtual machines can be started



Auditing and Logging

- VM Event Records
- Operator Messages
- Secondary Console Interface (SCIF) Messages
- Virtual Machine Console Logs

© 2025 Levi, Ray & Shoup, Inc.

Note that updates to the VM Directory are not logged per se. So, this means that if you are focusing on just using the VM Directory for resource security management, more is needed.



Auditing and Logging

- VM Event Records
- Operator Messages
- Secondary Console Interface (SCIF) Messages
- Virtual Machine Console Logs
- Built-in Functions:
 - IBM-provided Programmable Operator (PROP) can record Operator and SCIF messages as well as act on certain types of messages or specific messages
 - Time-based actions can routinely capture spooled console logs and route them for storage or analysis
 - Journaling of improper logon attempts

© 2025 Levi, Ray & Shoup, Inc.

Retrieving the data that is being captured requires some local programming or some local customization of z/M-provided tools. There is no automatic logging of directory updates/changes or of CP-related activities within server virtual machines (a server v.m. is like a started task or a background process).



IBM-supplied Directory/security-related settings

© 2025 Levi, Ray & Shoup, Inc.

There is no "TOD Enable" button on current hardware



IBM-supplied Directory and security settings

Initial Authorization and Authentication:

- SYSTEM CONFIG file
 - Activates all sensed devices visible to the LPAR (by I/O Subsystem via IOCD5)
 - Prompts for spool startup mode
 - Ability to enter visible passwords (on command-line logon, link statements)
 - No notification of multiple logon attempts with invalid passwords
- VM Directory
 - Initial password specified at installation time
 - Same password used in most default virtual machine definitions
 - Limited use of special passwords to restrict access (more later)
 - All minidisk definitions have common or easily guessed passwords
 - READ/WRITE/MULTIPLE
 - Ruserid/Wuserid/Muserid

Initial Auditing and Logging:

- CP messages go to the userid defined to CP as the “System Operator”
 - Default ID = OPERATOR
- No logging of directory changes
- No logging of system changes made by a superuser

© 2025 Levi, Ray & Shoup, Inc.



Suggested System and Directory Updates

© 2025 Levi, Ray & Shoup, Inc.

These are my suggestions for improving execution/management/security in an “out-of-the-box” z/VM system. RACF or other security manager is not required to implement these changes; however, your security policies may require additional function via an external security manager (like RACF)



Suggested System and Directory Changes

- Bring the z/VM LPAR “up” faster
- Simplify startup/shutdown operations
- Keep track of often-changed parts of a file
- Use source-level “includes” to simplify source maintenance

© 2025 Levi, Ray & Shoup, Inc.



SYSTEM CONFIG file

- Remove the system operator from startup decisions during normal operations
 - Enable the following features,
 - AUTO_IPL
 - AUTO_IPL_AFTER_RESTART
 - AUTO_IPL_AFTER_SHUTDOWN_REIPL
 - If set to FORCE, the operator is only prompted if spool file destruction may occur
- Turn off PASSWORDS_ON_CMDS
- Define Virtual LANs/Switches here instead of AUTOLOG1
- Create new CP command classes allow subsets of IBM-supplied command classes.
 - Examples: FORCE, SET SECUSER, SIGNAL SHUTDOWN, XAUTOLOG for a help desk
- Enable Journaling to track invalid logon attempts
- Use IMBED files for frequently changed sections
- Use –system–, &SYSRES and &SYSPARM variables to reduce file complexity

© 2025 Levi, Ray & Shoup, Inc.

Recommendations are more “art” than “science” but are based on a long history of implementations at multiple customers. I like to use IMBEDs so I know when the contents have been updated (based on date/time). If I kept all data in the SYSTEM CONFIG file, then I would need to establish another method to know what changed from one save point to another. We all know how well sysprogs document their changes, right?



Example of modified SYSTEM CONFIG

```

.....
/*          Checkpoint and Warmstart Information          */
.....

System_Residence,
Checkpoint Valid &SYSRES From CYL 21 For 9 ,
Warmstart  Valid &SYSRES From CYL 30 For 9

.....
/* System-unique Volumes                                */
.....
IMBED -system- VOLSEERS

.....
/* Journaling                                           */
.....
Journal Facility ON Set_and_Query ON ,
Logon Lockout After 3 Attempts for 5 Minutes ,
VM_LOGO After 3 Attempts

.....
/*          Features Statement                          */
.....
Features ,
  Auto_IPL Force Drain ,           /* Startup options          */
  Auto_IPL_After_Restart Force Drain ,
  Auto_IPL_After_Shutdown_Reipl Force Drain ,
  Enable ,                         /* Enable the following features */
  STP_TZ ,
  New_Devices_Initialized_When_Added, /* Make new devices online */
  Disable ,                        /* Disable the following features */
  Dynamic_ID ,
  Set_Dynamic_ID ,
  Set_Privclass ,                  /* Disallow SET PRIVCLASS command */
  Clear_TDisk ,                   /* Don't clear TDisks at IPL time */
  Validate_Shutdown ,             /* Don't require system name */
  Retrieve ,                      /* Retrieve options          */
  Default 20 ,                    /* Default... default is 20  */
  Maximum 255 ,                   /* Maximum... default is 255 */
  MaxUsers noLimit ,              /* No limit on number of users */
  Passwords_on_Cmnds ,            /* What commands allow passwords? */
  Autolog no ,                    /* ... AUTOLOG does          */
  Link no ,                       /* ... LINK does              */
  Logon no ,                      /* ... and LOGON does, too   */
  Vdisk Userlim 144000 blocks,    /* Maximum vdisk allowed per user */
  Disconnect_Timeout 15 ,        /* Can be OFF, default is 15 min */

```

Contents of system-1 VOLSEERS:

```

User_Volume_List VM1WK1
User_Volume_Include VM1*
User_Volume_Exclude VM2*

```

Contents of system-2 VOLSEERS:

```

User_Volume_List VM2WK1
User_Volume_Include VM2*
User_Volume_Exclude VM1*

```

Alternate method of in-inline coding:

```

System-1: BEGIN
User_Volume_List VM1WK1
User_Volume_Include VM1*
User_Volume_Exclude VM2*
System-1: END

```

```

System-2: BEGIN
User_Volume_List VM2WK1
User_Volume_Include VM2*
User_Volume_Exclude VM1*
System-2: END

```

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

– NOPASS

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

– NOPASS

No password required for logon

© 2025 Levi, Ray & Shoup, Inc.

I would only do this for a tutorial virtual machine (for example, in conjunction with use of the CMS Primer)



VM Directory

Know and use “reserved” passwords

- NOPASS No password required for logon
- AUTOONLY

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

- NOPASS **No password required for logon**
- AUTOONLY **Similar to started task/process**

© 2025 Levi, Ray & Shoup, Inc.

Makes a virtual machine work like a started task or a background process



VM Directory

Know and use “reserved” passwords

- NOPASS **No password required for logon**
- AUTOONLY **Similar to started task/process**
- NOLOG

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

- NOPASS No password required for logon
- AUTOONLY Similar to started task/process
- NOLOG Logon not permitted

© 2025 Levi, Ray & Shoup, Inc.

Good for access control;



VM Directory

Know and use “reserved” passwords

- NOPASS **No password required for logon**
- AUTOONLY **Similar to started task/process**
- NOLOG **Logon not permitted**
- LBYONLY

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Know and use “reserved” passwords

- NOPASS No password required for logon
- AUTOONLY Similar to started task/process
- NOLOG Logon not permitted
- LBYONLY Use Surrogate Userid for logon

© 2025 Levi, Ray & Shoup, Inc.

Good for avoiding shared userids



VM Directory

Authentication Techniques

- Set all IBM-provided IDs that you don't use to NOLOG
 - Don't delete these definitions, otherwise system upgrades will be impacted
- Define "real" administrative users and LOGONBY to superuser virtual machines
 - Caution: These admin users should be subject to password management policies...but keep a "break-glass" password to MAINT in case all LOGONBY users get locked out.
- Set used IBM-provided service virtual machines to AUTOONLY
- Remove obsolete virtual machines after a version upgrade
- Delete **all** Minidisk passwords, except for certain limited disks needing the universal read password of ALL:
 - MAINT190/193/19D/19E/402
 - TCPMAINT 592
- Carefully consider impact of IUCV ANY
- Don't 'overauthorize' CP commands to a virtual machine
 - Define new command classes to avoid full CP CLASS authority when not needed

© 2025 Levi, Ray & Shoup, Inc.



VM Directory

Additional Directory Cleanup

- Use Directory Profiles
 - Use profile IBMDFLT for the entries that don't use any profile
 - Only use in-line values that differ from the profile entry
- Eliminate duplication within the IBM-supplied directory:
 - Use GLOBALOPTS MACHINE ESA and remove individual MACHINE ESA specifications
 - Move common TCPMAINT LINKS in individual TCP/IP entries to profiles TCPCMSU and TCPGCSU
 - Move non-version-specific LINK entries in SUBCONFIG clauses to the related USER or IDENTITY clauses
 - Keep version-specific links in SUBCONFIGs, since new versions are installed one LPAR at a time
- Cleanup like this speeds up DIRECTXA processing and reduces the size of the directory stored in DRCT space

© 2025 Levi, Ray & Shoup, Inc.

While these cleanup steps are not necessary, they reduce the size of the compiled directory and reduce complexity by eliminating information that already exists in a directory profile. A slimmer directory source compiles faster.



Example of Directory Cleanup

```

IDENTITY SYSMON WDSJU8QP 32M 32M DG
BUILD ON DEMOVM1 USING SUBCONFIG SYSMON-1
BUILD ON DEMOVM2 USING SUBCONFIG SYSMON-2
* BUILD ON @@member3name USING SUBCONFIG SYSMON-3
* BUILD ON @@member4name USING SUBCONFIG SYSMON-4
ACCOUNT 1 SYSMON
MACHINE ESA
IPL CMS PARM AUTO CR
CONSOLE 01F 3215
SPOOL 00C 2540 READER A
SPOOL 00D 2540 PUNCH A
SPOOL 00E 1403 A
SUBCONFIG SYSMON-1
LINK MAINT 190 190 RR
LINK MAINT 19D 19D RR
LINK MAINT 193 193 RR
MDISK 191 3390 03030 005 VM1RES MR RSYSMON WSYSMON MSYSMON
SUBCONFIG SYSMON-2
LINK MAINT 190 190 RR
LINK MAINT 19D 19D RR
LINK MAINT 193 193 RR
MDISK 191 3390 03030 005 VM2RES MR RSYSMON WSYSMON MSYSMON
*SUBCONFIG SYSMON-3
* LINK MAINT 190 190 RR
* LINK MAINT 19D 19D RR
* LINK MAINT 193 193 RR
*SUBCONFIG SYSMON-4
* LINK MAINT 190 190 RR
* LINK MAINT 19D 19D RR

```

```

IDENTITY SYSMON WDSJU8QP 32M 32M DG
INCLUDE IBMDFLT
BUILD ON DEMOVM1 USING SUBCONFIG SYSMON-1
BUILD ON DEMOVM2 USING SUBCONFIG SYSMON-2
* BUILD ON @@member3name USING SUBCONFIG SYSMON-3
* BUILD ON @@member4name USING SUBCONFIG SYSMON-4
ACCOUNT 1 SYSMON
IPL CMS PARM AUTO CR
LINK MAINT 193 193 RR
SUBCONFIG SYSMON-1
MDISK 191 3390 03030 005 VM1RES MR
SUBCONFIG SYSMON-2
MDISK 191 3390 03030 005 VM2RES MR
*SUBCONFIG SYSMON-3
*SUBCONFIG SYSMON-4

```

© 2025 Levi, Ray & Shoup, Inc.

Here’s an example of a directory entry before and after cleanup. This method shows unique details for a virtual machine while not repeating common entries in multiple virtual machines. Note that GLOBALOPTS MACHINE ESA and PROFILE IBMDFLT are referenced. I also would have LOGONBY users in the PROFILE IBMDFLT and change the password for SYSMON to LBYONLY



Auditing/Logging

- Use IBM Directory Maintenance Tool or similar
 - Logs all directory transactions
 - User password management (simple)
 - Limited policy enforcement
 - Number of characters
 - Password history
 - Expiration notices via reader notes
 - Userid is NOLOG'd upon expiration, administrator must reenable
 - IBM-provided exits synchronize directory changes with Security Server (RACF)
- Use CP Operator Message capturing tool
 - Programmable Operator (PROP)
 - Performance Toolkit
- Use virtual machine VMUTIL for time-based activities
 - Send daily virtual machine console logs to a collector
- Operations Manager for z/VM can also perform these non-directory functions

© 2025 Levi, Ray & Shoup, Inc.



Preparing for an external security manager

Why consider an external security manager?

- Limitations of z/VM Directory
 - 8 LOGONBY userids per virtual machine
 - Up to 8-character passwords
 - No passphrases
 - Passwords stored on disk in clear text (EBCDIC)
 - Need more granular access to resources for superusers
- Limitations of DirMaint
 - Limited password validation
 - Crude password change mechanism
- Single collection point for access logs
- Single point of authorization for CMS users

Note that an external security manager does not control security inside a “bare metal” operating system running in a virtual machine

© 2025 Levi, Ray & Shoup, Inc.

Superusers like MAINT have OPTION LNKNOPAS. This allows MAINT to access any minidisk without providing a password. While you can audit execution of CP LINK commands, your security requirements may not allow that much authorization. On the other hand, if there are only system maintainers logging directly onto z/VM and using CMS,



Preparing for an external security manager

Determine what resources you need to protect

- Do you really need to protect access to spool files?
- Do you really need to protect access to minidisks if there are no passwords associated with minidisks?
- Do you really need to protect resources for batch execution (under CMS)?
- Do you really need to protect CP commands if you have created custom command classes?

Prepare the z/VM Directory for loading the security database

- Use ACIGROUP directory statements to define virtual machines with a similar purpose
 - The ACIGROUP will be used to define the virtual machine's default group
 - Put the ACIGROUP statement in the PROFILE; override only on virtual machines that need to be in a different group

Run the IBM-supplied utility to build the initial RACF commands

- Remove the resource definitions that won't be monitored/managed
- Remove the class activations for resource groups that won't be monitored/managed

Update RACF exits to minimize searching of the security database

- Primarily access the VM directory for most authorizations
- Automatically authorize minidisks with universal READ access (ALL in the minidisk read password position)

Select DirMaint exits to send RACF updates only for resources that are being protected by RACF

- If you are only protecting userids/passwords with RACF, don't send directory updates for minidisks, spool, etc.

© 2025 Levi, Ray & Shoup, Inc.

You don't have to license HLASM for RACF; Assembler F (included with z/VM) will suffice



References

- CP Planning and Administration (SC24-6271)
- CMS Planning and Administration (SC24-6264)
- Directory Maintenance Facility Tailoring and Administration (SC24-6283)
- RACF Security Server Security Administrator's Guide (SC24-6311)
- RACF Security Server System Programmer's Guide (SC24-6312)

© 2025 Levi, Ray & Shoup, Inc.