z/OS
2.5

Planning for Multilevel Security and the Common Criteria

GA32-0891-50

Note

Before using this information and the product it supports, read the information in “Notices” on page 149.

This edition applies to IBM® z/OS® 2.5 (5650-ZOS) and to all subsequent releases and modi ications until otherwise indicated in new editions.

Last updated: 2025-08-21

Contents

Figures

Tables

About this document

This document describes the z/OS functions that can be used to provide multilevel security. multilevel security is a security policy that allows the classification of data and users based on a system of hierarchical security levels combined with a system of non-hierarchical security categories. This document provides a high-level overview of multilevel security and security labels, and information about how to migrate a z/OS system to a multilevel-secure system. This document does not provide all the details that an administrator needs to configure or operate a system to take advantage of multilevel security, but it references other documents where you can find the details.

This document also describes how you can configure IBM z/OS 2.5 to meet the requirements of the Common Criteria evaluated system.

Who should read this document

This document is intended primarily for the security administrator who is planning to establish a multilevel-secure system or a system that meets the requirements of the Common Criteria evaluated system.

Other users who should read this document include:

- Auditors preparing to audit a multilevel-secure system or a system that meets the requirements of the Common Criteria Protection Profile for General Purpose Operating Systems
- Operators preparing to operate a multilevel-secure system or a system that meets the requirements of the Common Criteria Protection Profile for General Purpose Operating Systems
- System programmers preparing to support a multilevel-secure system or a system that meets the requirements of the Common Criteria Protection Profile for General Purpose Operating Systems

How this document is organized

Chapter 1, “What is multilevel security?,” on page 1 contains a high-level overview of multilevel security.

Chapter 2, “Security labels,” on page 9 describes security labels and how you use them to implement multilevel security.

Chapter 3, “Establishing multilevel security,” on page 33 describes in detail how to establish a multilevel-secure environment.

Chapter 4, “Auditing a multilevel-secure system,” on page 99 discusses auditing considerations for a multilevel-secure environment.

Chapter 5, “Operating a system,” on page 105 discusses operations considerations for a multilevel-secure environment.

Chapter 6, “Adding authorized programs to a multilevel-secure system,” on page 107 discusses how to add authorized programs to a multilevel-secure system while maintaining the integrity of the system. It includes specific recommendations for CICS®, DB2®, DFSORT, IMS, ISPF, and WebSphere® MQ for z/OS.

Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117 discusses the Common Criteria and the certified configuration of z/OS Version 2 Release 5.

How to use this document

The security administrator should read the entire book. Skip Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117 if you are not interested in meeting the requirements of the Common Criteria. Be sure that you understand the concepts presented in Chapter 1, “What is multilevel security?,” on page 1 and Chapter 2, “Security labels,” on page 9. Use the information in

Chapter 3, “Establishing multilevel security,” on page 33 to plan for and implement multilevel security for your installation. You will need to reference *z/OS Communications Server: IP Configuration Guide* for information about planning for and implementing multilevel security for a TCP/IP network.

The auditor should read Chapter 1, “What is multilevel security?,” on page 1 and Chapter 2, “Security labels,” on page 9 and understand the concepts presented in these chapters. Then read Chapter 4, “Auditing a multilevel-secure system,” on page 99.

The operator should read Chapter 1, “What is multilevel security?,” on page 1 and Chapter 2, “Security labels,” on page 9 and understand the concepts presented in these chapters. Then read Chapter 5, “Operating a system,” on page 105.

The system programmer should read Chapter 1, “What is multilevel security?,” on page 1 and Chapter 2, “Security labels,” on page 9 and understand the concepts presented in these chapters. Then read Chapter 3, “Establishing multilevel security,” on page 33 to understand the tasks you will perform to assist the security administrator in planning for and implementing multilevel security for your installation. You will need to reference *z/OS Communications Server: IP Configuration Guide* for information about planning for and implementing multilevel security for a TCP/IP network. If you plan to install products on your system that are not part of the z/OS trusted computing base, read Chapter 6, “Adding authorized programs to a multilevel-secure system,” on page 107.

If you are planning to meet the requirements of the Common Criteria, all readers should read Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117.

Prerequisite and related information

The reader should have a detailed knowledge of RACF®, and be familiar with the other elements and features of z/OS, particularly MVS, JES, z/OS UNIX, TSO/E, Communications Server, DFSMS, zFS File System, RMF, and SDSF.

You will need to reference *z/OS Communications Server: IP Configuration Guide* for information about planning for and implementing multilevel security for a TCP/IP network.

Summary of changes

This information includes terminology, maintenance, and editorial changes. Technical changes or additions to the text and illustrations for the current edition are indicated by a vertical line to the left of the change.

Note: IBM z/OS policy for the integration of service information into the z/OS product documentation library is documented on the z/OS Internet Library under IBM z/OS Product Documentation Update Policy (www.ibm.com/docs/en/zos/latest?topic=zos-product-documentation-update-policy).

Summary of changes

New

August 2025
This edition includes editorial improvements.

May 2025
This edition includes editorial improvements.

January 2025
z/OS 2.5 is designed to meet the requirements of the Evaluation Assurance Level 4 (EAL4) level of assurance requirements. See Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117.

July 2023
z/OS 2.5 is designed to meet the requirements of Operating System Protection Profile (OSPP) Version 4.3. See Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117.

Chapter 1. What is multilevel security?

A fundamental requirement of a secure system is that there is a set of guidelines that specify the authorization of subjects to access specific objects. “Access” is a key concept; it implies a flow of information from a subject to an object or from an object to a subject. For example, when a user (a subject) updates a data set (an object), the information flows from the subject to the object. When a user reads a record from a data set, the information flows from the object to the subject.

The subject in these interactions is active; the subject is attempting to access an object (or the information that the object contains). The object, on the other hand, is passive; it contains the information that the subject wants to access, or it is the receiver of information from the subject. Each time a subject attempts to access an object, the system must decide whether to allow the access.

Two central concepts of security are security policy and accountability. A security policy is a set of laws, rules and practices that regulate how an organization manages, protects and distributes its sensitive data. It is the set of rules that the system uses to decide whether a particular subject can access a particular object. Accountability requires that each security- relevant event must be able to be associated with a subject. Accountability ensures that every action can be traced to the user who caused the action.

multilevel security is a security policy that allows the classification of data and users based on a system of hierarchical security levels combined with a system of non-hierarchical security categories. A multilevel-secure security policy has two primary goals. First, the controls must prevent unauthorized individuals from accessing information at a higher classification than their authorization. Second, the controls must prevent individuals from declassifying information.

History

In the 1980s, the United States Department of Defense provided guidelines and requirements for establishing data processing security in its computer installations. These criteria, as specified in *Department of Defense Trusted Computer System Evaluation Criteria*, DoD 5200.28-STD (also known as TCSEC or the Orange Book), applied also to the computer systems in companies working with a government contract. The criteria corresponded to a particular security designation, depending on the type and amount of security the system provides. The security designations ranged from D (the least amount of security) through C1, C2, B1, B2, B3, and A1. The National Security Agency (NSA) performed a formal evaluation to determine whether a data processing system adhered to the guidelines and requirements for a given security designation.

Between 1988 and 1990 IBM enhanced MVS, RACF, JES2, JES3, TSO, VTAM®, DFP, and PSF to meet the B1 criteria. MVS/ESA Version 3 Release 1 Modification Level 3 passed the formal evaluation that was performed by the National Security Agency and obtained a B1 security designation. For several years, subsequent versions of RACF, MVS/ESA, and OS/390® were designed to continue to meet the B1 criteria, although no formal evaluations were done. But over time new functions such as UNIX System Services were added to MVS that could not be used on a system with a B1 security designation. And customer configurations evolved to require networking, which could not be used on a B1 system. Eventually the Common Criteria and ISO 15408 superseded the older US Government standards that are described in the Orange Book.

IBM's multilevel security functions for z/OS build on the work done on MVS to meet the B1 criteria, and provide functions consistent with those described in the Common Criteria and some of the Common Criteria Protection Profiles.

Although the requirements for multilevel security arose from the classified data processing needed by government installations, the functions that are implemented for multilevel security (especially the basic ones of user and data classification through security labels) should be relevant to commercial installations too.

Note: As of z/OS V2R4, multilevel security is no longer a part of the Common Criteria evaluation.

Common criteria evaluations:

- z/OS V1R6 has been evaluated and certified under the Common Criteria Controlled Access Protection Profile (CAPP) at EAL3 augmented and the Labeled Security Protection Profile (LSPP) at EAL3 augmented. For information about the certified configuration at EAL3, see the second z/OS V1R6 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-02.
- z/OS V1R7 has been evaluated and certified at the stricter EAL4 level, under the CAPP and LSPP profiles. For information about the certified configuration for z/OS V1R7, see the final z/OS V1R7 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-05.
- z/OS V1R8 has been evaluated and certified at the EAL4 level, augmented by ALC_FLR.3, under the CAPP and LSPP profiles. For information about the certified configuration for z/OS V1R8, see the z/OS V1R8 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-06.
- z/OS V1R9 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3, for the CAPP and LSPP profiles. For information about the certified configuration for z/OS V1R9, see the z/OS V1R9 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-07.
- z/OS V1R10 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3, for the CAPP profile. For information about the certified configuration for z/OS V1R10, see the z/OS V1R10 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-09.
- z/OS V1R11 has been certified to meet the requirements of the Common Criteria Operating System Protection Profile (OSPP), BSI-CC-PP-0067, Version 2.0 (dated 2010-06-10) including the extended packages of OSPP:
  - Labeled Security (OSPP-LS), Version 2.0
  - Extended Identification and Authentication (OSPP-EIA), version 2.0

  For information about the certified configuration for z/OS V1R11, see the z/OS V1R11 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-10.
- z/OS V1R12 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3 for the following protection profiles:
  - Operating System Protection Profile (OSPP) Version 2.0 (dated 6/10/2010)
  - OSPP Extended Package - Labeled Security (OSPP-LS), Version 2.0 (dated 5/28/2010)
  - OSPP Extended Package - Extended Identification and Authentication (OSPP-EIA), version 2.0 (dated 5/28/2010)

  For information about the certified configuration, see the first z/OS V1R12 edition of this document, *z/OS Planning for Multilevel Security and the Common Criteria*, GA22-7509-10. The certification report is published on the OCSI web site (www.ocsi.gov.it/index.php/elenchi-certificazioni/prodotti-certificati.html).
- The RACF component of z/OS V1R12 is being evaluated for conformance to the requirements of the Common Criteria assurance level EAL5, augmented by ALC_FLR.3. At the time this information was published, the evaluation was not complete. To find out whether the certification report has been published, visit the OCSI web site (www.ocsi.gov.it/index.php/elenchi-certificazioni/prodotti-certificati.html).
- z/OS V1R13 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3 for the following protection profiles:
  - Operating System Protection Profile (OSPP), Version 2.0 (dated 6/1/2010)
  - OSPP Extended Package - Labeled Security (OSPP-LS), Version 2.0 (dated 5/28/2010)
  - OSPP Extended Package - Extended Identification and Authentication (OSPP-EIA), version 2.0 (dated 5/28/2010)
- z/OS V2R1 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3 for the following protection profiles:

  - Operating System Protection Profile (OSPP), Version 2.0 (dated 6/1/2010)
  - OSPP Extended Package - Labeled Security (OSPP-LS), Version 2.0 (dated 5/28/2010)
  - OSPP Extended Package - Extended Identification and Authentication (OSPP-EIA), version 2.0 (dated 5/28/2010)
- z/OS V2R2 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3 for the following protection profiles:
  - Operating System Protection Profile (OSPP), Version 2.0 (dated 6/1/2010)
  - OSPP Extended Package - Labeled Security (OSPP-LS), Version 2.0 (dated 5/28/2010)
  - OSPP Extended Package - Extended Identification and Authentication (OSPP-EIA), version 2.0 (dated 5/28/2010)
- z/OS V2R3 has been certified to meet the requirements of the Common Criteria assurance level EAL4, augmented by ALC_FLR.3 for the following protection profiles:
  - Operating System Protection Profile (OSPP), Version 2.0 (dated 6/1/2010)
  - OSPP Extended Package - Labeled Security (OSPP-LS), Version 2.0 (dated 5/28/2010)
  - OSPP Extended Package - Extended Identification and Authentication (OSPP-EIA), version 2.0 (dated 5/28/2010)
- z/OS V2R4.
- z/OS V2R5 is designed to meet the requirements of the Operating System Protection Profile (OSPP) Version 4.3 protection profile (dated September 27, 2022).
- z/OS V2R5 has been certified to meet the requirements of the Common Criteria assurance level EAL4.

For information about the certified configuration, see Chapter 7, “The certified configuration for the Common Criteria for z/OS 2.5,” on page 117. The certification report is published on the OCSI web site (www.ocsi.gov.it/index.php/elenchi-certificazioni/prodotti-certificati.html).

Characteristics of a multilevel-secure system

Characteristics of a multilevel-secure system include the following:

- The system controls access to resources.
- The system does not allow a storage object to be reused until it is purged of residual data.
- The system enforces accountability by requiring each user to be identified, and creating audit records that associate security-relevant events with the users who cause them.
- The system labels all hardcopy with security information.
- The system optionally hides the names of data sets, files and directories from users who do not have access to those data objects.
- The system does not allow a user to declassify data by "writing down" (that is, write data to a lower classification than the classification at which it was read) except with explicit authorization to do so.

The additional security processing required for these characteristics might result in some performance degradation.

Access controls

The system controls access to resources using mandatory access control and discretionary access control.

Subjects and objects

A *subject* is an entity that requires access to system resources. Examples of subjects are:

- Human users
- Started procedures

- Batch jobs
- z/OS UNIX daemons

The term *user* usually has the same meaning as the term *subject*, but sometimes implies a human subject. In this book, unless stated otherwise, the terms *user* and *subject* are used interchangeably.

An *object* is a system resource to which access must be controlled. Examples of objects are:

- Data sets
- z/OS UNIX files and directories
- Commands
- Terminals
- Printers
- DASD volumes
- Tapes

Mandatory access control (MAC)

Mandatory access control is the principle of restricting access to objects based on the sensitivity of the information that the object contains and the authorization of the subject to access information with that level of sensitivity. This type of access control is mandatory in the sense that subjects cannot control or bypass it. The security administrator (the user with the RACF SPECIAL attribute) defines the sensitivity of each object by means of a security label. This security label indicates the hierarchical level or classification of the information (such as Top Secret, Secret, Sensitive), and indicates to which non-hierarchical category the information belongs within that level (such as Project A, Project B). The security administrator also controls each subject's access to information by specifying which security labels the subject can use. A subject can access information in an object only when the subject's security label entitles the access. If the subject's security label does not have enough authority, the subject cannot access the information in the object. For more information about security labels, see Chapter 2, “Security labels,” on page 9.

Discretionary access control (DAC)

Discretionary access control is the principle of restricting access to objects based on the identity of the subject (the user or the group to which the user belongs). Discretionary access control is implemented using access control lists. A resource profile contains an *access control list* that identifies the users who can access the resource and the authority (such as read or update) the user is allowed in referencing the resource. The security administrator defines a profile for each object (a resource or group of resources), and updates the access control list for the profile. This type of control is discretionary in the sense that subjects can manipulate it, because the owner of a resource, in addition to the security administrator, can identify who can access the resource and with what authority.

Object reuse

In a multilevel-secure environment no subject can gain access to information remaining in a storage object (such as central storage or a DASD) after a previous subject has released the storage object back to the system. That is, the system must purge all residual data, including encrypted data, before reassigning the storage object to a new subject.

For example, when a user issues a command to scratch a data set on DASD, the next user assigned to use that same location could read the residual data. To prevent this, the system must erase the space on the DASD volume when the data set is deleted.

For tape volumes, the installation must either use a tape management system (such as DFSMSrmm) that can automatically erase tapes and configure it to do so when a volume is scratched, or put in place manual procedures to erase tape data as tapes are scratched.

To prevent object reuse for a zFS file system, the administrator must leave the NBS (new block security) option set to the default value (enabled) in the IOEFSPRM file. The NBS option must be enabled on any mount commands, and when attaching a multi-file system aggregate.

Accountability

The system can associate security-relevant events with the users who caused them.

Auditing

Audit records associate a security-relevant event with the user who caused the event to take place. In a multilevel-secure environment, the audit record must also indicate the security label of an object when a subject accesses, creates, or deletes the object.

The system auditor must be able to select for auditing those events necessary for efficient analysis. This selection includes auditing a user's actions either by the user's identity or by the security label of the accessed object.

Identification and authentication

Each user in the system must be identified. In a multilevel-secure environment, this requirement includes the MVS console operator, whose identity is verified through the LOGON and LOGOFF commands.

Additionally, each access of an object is dependent on the clearance and authorization of the user as specified by the security label of that user.

Labeling hardcopy with security information

In a multilevel-secure environment, the system prints a security notation associated with the security label on each page of the hardcopy output. The system also provides security information on both the top and the bottom of each page of printed output as a default. The system allows a user who has been authorized by the security administrator to request that no security information be printed; however, the system audits all such requests.

The name-hiding function

The names of data sets, files, and directories might contain information that must be protected from some users. The name-hiding function restricts the display of names to only those to which the user has authorization. The security administrator controls the name-hiding function by activating and deactivating the RACF MLNAMES option using the SETROPTS command.

The name-hiding function restricts the display only of names that the user does not already know; that is, if the user's request includes the name, the system does not hide the name. For example, assume that a user is not authorized to data set x.y.z. If the user asks to see all the names of all the data sets in a catalog that includes x.y.z, x.y.z is not displayed in the list. The system does not let the user know that the data set x.y.z exists. But, if the user asks to specifically see data set x.y.z, the system responds that the user does not have access to x.y.z, but does not hide the fact that x.y.z exists.

For files and directories, RACF does a mandatory access check to determine whether the user is authorized to see a name. For data sets in a multilevel-secure environment, RACF does both a discretionary access check and a mandatory access check to determine if a user is authorized to see a data set name.

The name-hiding function can be activated in an environment that is not multilevel-secure. *z/OS DFSMS Using the New Functions* describes the name-hiding function in an environment that is not multilevel-secure.

Write-down

To prevent users from accessing data that they are not authorized to see, a multilevel-secure system generally requires that in order to read data a user must have a security label that represents a level

of security at least as high as the data's level of security. (This statement is a generalization of the concept of "dominance", which is discussed in detail in “Dominance” on page 12.) To prevent users from declassifying data, a multilevel-secure system requires that in order to write data the data must have a security label that represents a level of security at least as high as the user's level of security – we say that the user cannot "write down". (This requirement is known as the star property, or *-property.) For example, if "Top Secret" is a higher level of security than "Secret", a user whose security label gives authorization to Top Secret data cannot be allowed to write Top Secret data into a data set with a security label that classifies the data as Secret. In order to both read and write data, the user and the data must have equivalent security labels. For more information about the star property, see “Preventing declassification of data” on page 12.

The security administrator controls whether write-down is allowed by activating and deactivating the RACF MLS option using the SETROPTS command.

To allow for controlled situations of write-down, z/OS allows the security administrator to assign a "write-down by user" privilege to individual users that allows those users to select the ability to write down. For more information about this privilege, see “Controlled write-down” on page 13.

Performance

You should expect some performance degradation in a multilevel-secure system. An installation must balance its need for a high level of security against performance requirements.

For most installations, the most noticeable degradation in performance will occur if the name-hiding function is active. When users try to list large VTOCs or catalogs, the system has to perform an authority check for each object to be listed that it wouldn't otherwise perform. Note that if a user uses ISPF 3.4 to list information about a single data set, ISPF reads the entire VTOC.

For many installations, the ERASE(ALL) SETROPTS option will also degrade performance. How much it does depends to a large extent on the kind of DASD controllers the installation has and how they're configured.

The trusted computing base

A trusted computer system is a computer system that uses both hardware and software to ensure that security criteria are met. The security-relevant portion of the system, called the "trusted computing base", is made up of separate hardware and software components. It is the combination of these components that enforce the security requirements in a system.

The z/OS multilevel-secure trusted computing base consists of all hardware attached to the processor(s) running z/OS, all microcode in the hardware, all elements and features of z/OS that support multilevel security, and all software products running on the system that run in an authorized state. The trusted computing base can violate the system's security policy for its own need, but enforces the security policy for all subjects outside the trusted computing base.

Any software that has at least one of these attributes, whether supplied by z/OS or installed separately by the customer, must be included in the trusted computing base.

- Runs in supervisor state
- Runs with protection keys 0-7 or with a PSW key mask allowing keys 0-7
- Runs authorized through the authorized program facility (APF).

All programs and software outside the trusted computing base are considered untrusted. The software that is part of the z/OS trusted computing base, whether supplied by IBM, the customer, or other vendors, must ensure that application software does not violate the security policy.

Hardware

All hardware that is supported by z/OS 2.5 is included in the z/OS trusted computing base.

Software

Most z/OS elements and features support multilevel security. A few do not, and are not part of the z/OS trusted computing base. The elements and features that do not support multilevel security are listed in “z/OS elements and features that do not support multilevel security” on page 35.

Some z/OS elements and features partially support multilevel security; for example, JES3 does not support the use of security labels on a per-system basis. See Table 5 on page 35 for information about z/OS elements and features that only partially support multilevel security.

Chapter 2. Security labels

A security label enables an installation to classify subjects and objects according to a data classification policy, identify objects to audit based on their classification, and protect objects such that only appropriately-classified subjects can access them. Objects in a multilevel-secure system have a security label that indicates the sensitivity of the object's data. Subjects in a multilevel-secure system also have a security label. This label determines whether the subject is allowed to access a particular object.

A security label is used as the basis for mandatory access control decisions. By assigning security labels, the security administrator can ensure that data of a certain classification is protected from access by a user of a lesser security classification. In addition, through the use of discretionary access control, the security administrator can further ensure that the data is protected from access by unauthorized users.

Security labels provide the capability to maintain multiple levels of security within a system. By assigning a security label to a resource, the security administrator can prevent the movement of data from one level of security to another.

Security labels can also identify the security of hardcopy output. The security label is associated with the security notation that is printed on the hardcopy output from the system. The security administrator associates the name of a security overlay with each security label in a multilevel-secure system. Print Services Facility (PSF) uses this association to print the proper label on the secure output.

Security labels for users, MVS data sets, and general resources are stored in the RACF database, in the profiles for the users and resources to which they apply. Security labels for zFS files and directories are stored in the file security packets (FSPs) for the files and directories to which they apply.

Defining security labels

A security *label* establishes an association between a RACF security *level* and a set of zero or more RACF security *categories*. For example, a system might have three security levels, unclassified, sensitive, and secret, and three security categories, Project A, Project B, and Project C. Then, PURPLE could be a security label name indicating Secret for Project A, Project B, and Project C. COLUMBIA could be a security label name meaning Sensitive for Project A and Project B; UNION could be a security label name indicating unclassified for Project C.

Table 1. An example of security labels

Security level	**Project A**	**Project B**	**Project C**
Secret	SECLABEL = PURPLE		
Sensitive	SECLABEL = COLUMBIA		*no label defined*
Unclassified	*no label defined*	*no label defined*	SECLABEL=UNION

The security administrator defines two profiles in the RACF SECDATA resource class that define the security levels and security categories for the system:

- The SECLEVEL profile contains a member for each hierarchical security level in the system.
- The CATEGORY profile contains a member for each non-hierarchical category in the system.

SECLEVEL: The hierarchical security level defines the degree of sensitivity of the data. “SECRET,” “SENSITIVE,” and “UNCLASSIFIED” are examples of levels you could define. You might define "SECRET" to be a security level of 30, "SENSITIVE" to be a level of 20, and "UNCLASSIFIED" to be a level of 10. The security administrator can define up to 254 security levels.

CATEGORY: The non-hierarchical categories further qualify the access capability. The security administrator can define zero or more categories that correspond to some grouping arrangement in the installation. "PROJECTA", "PROJECTB", and "PROJECTC" could all be categories defined.

Guideline: Although the system allows the definition of several thousand categories, define only the security categories you need. A large number of security categories can decrease performance, particularly at IPL time and for the SETROPTS RACLIST(REFRESH) command.

Security labels: After defining the SECLEVEL and CATEGORY profiles, the security administrator defines a profile in the SECLABEL resource class for each security label. The security label is a name of up to eight uppercase alphanumeric or national characters. The national characters are # (X'7B'), @ (X'7C'), and $ (X'5B'). The first character cannot be numeric. Each security label name must be unique. Each SECLABEL profile specifies the particular combination of a SECLEVEL member and zero or more members of the CATEGORY profile that applies to the security label. You do not need to define a security label for every possible combination of level and category.

There is no limit on the number of security labels that can be defined.

Guideline: Define only the security labels you need. A large number of security labels can decrease performance, particularly at IPL time and for the SETROPTS RACLIST(REFRESH) command.

Because implementation of MLS has consequences through all parts of your z/OS system, you should ensure that your plan addresses the following:

- the effects on Db2®,
- the effects on USS file systems and programs including terminals,
- the effects on TCP/IP and its programs, the implications of shared DASD,
- the implications of connection to a sysplex.

Because the scope of your MLS implementation extends to everything covered by the relevant RACF database, your plan should consider the implications of a shared RACF database or of restricting your MLS system to a RACF database not shared with other systems. The details on how to address these issues are contained in this and related manuals. Implementation of MLS should not be attempted without understanding its effects on each of these areas.

Two security labels with different names can have the same levels and sets of categories for administrative convenience. They are treated as equivalent for access control and name hiding purposes, but will not perform as efficiently as having only a single security label with that definition.

For more information about how to define security labels, see “Defining security labels” on page 36 or *z/OS Security Server RACF Security Administrator's Guide*.

Security labels that the system creates

The system creates four labels automatically at initialization time.

SYSHIGH: This label is equivalent to the highest security level defined by the security administrator, and all categories defined by the security administrator. It dominates all other security labels in the system. (For a discussion of dominance, see “Dominance” on page 12.) If another hierarchical security level is added to the system or if another non-hierarchical category is added, the system converts the SYSHIGH label to include the change. SYSHIGH should be restricted to special system-level address spaces such as consoles, and to system programmers, system operators, and system administrators.

SYSLOW: This label is equivalent to the lowest security level defined by the security administrator, and no categories. It is dominated by all other security labels. (For a discussion of dominance, see “Dominance” on page 12.) If a resource is not in a class that requires reverse mandatory access checks or equal mandatory access checks, assigning a security label of SYSLOW to the resource allows all subjects to pass a mandatory access check for read access to the resource. Subjects still must pass the discretionary access check in order to access the resource.

SYSLOW should be used only for resources that have no classified data content. It is appropriate to use SYSLOW for data sets that IBM supplies for which the following is true:

- Most users only need to read them.

- A limited number of users, such as system programmers, might need to update them. They should do so only when running at a very low classification, to prevent them from accidentally putting classified data into the data sets

SYSNONE: SYSNONE is treated as equivalent to any security label to which it is compared. SYSNONE, like SYSLOW, should be used only for resources that have no classified data content. It is different from SYSLOW in that all users might need to update resources with SYSNONE, even when running at a high classification. It is intended for use on resources that must be written to at different security labels when write-down is not allowed. It is used to ensure that a user is permitted read/write access to a data set such as a catalog.

Guidelines: Follow these guidelines for assigning the SYSNONE security label:

- Use SYSNONE for a data set only when some other process (such as catalog management, or a program via program access to data sets (PADS)) mediates the user's access to ensure that no classified data is written into the data set.
- Use SYSNONE for a z/OS UNIX file only when you limit discretionary access to the file to a specific UID, and the only access to the file is via a z/OS UNIX program with the setuid option that switches to that UID and ensures that no classified data is written into the data set.
- Do not use SYSNONE for users.

SYSMULTI: This label is considered to be equivalent to any defined security label. It is intended for use by the following:

- Server or daemon address spaces whose implementation and documentation explicitly support multilevel security, giving them the ability to perform and separate work for users running with different security labels
- zFS directories that can contain data with different security classifications, such as the root directory of a file system

Guidelines: Follow these guidelines for assigning the SYSMULTI security label:

- SYSMULTI is not an appropriate security label for a data set or z/OS UNIX file, unless access to the data set is mediated via PADS or some other mechanism to ensure that either of the following is true:
  - Users can only write, and not read
  - Users can only read appropriate parts of the data
- SYSMULTI is not generally appropriate for users. Table 6 on page 38 lists some user IDs for which SYSMULTI is appropriate.

Assigning a security label to a subject or resource

A subject can have authorization to more than one security label, but can only use one security label at a given time. (Note that although a subject can have only one security label active at a time for a session, the subject can access data with different security labels depending on the dominance relationship between the subject's and object's security labels.) To authorize a subject to use a security label, the security administrator permits that subject's user ID to the profile in the RACF SECLABEL resource class for the security label. The security administrator can also assign a default security label for the subject in the user profile. (Note that in order for the subject to use the default security label, the label must be one that the security administrator has authorized the subject to use.) The security label a subject uses at a given time can be assigned in a number of ways; for example, a TSO/E user can specify a security label on the logon panel, and a batch user can specify a security label on the JOB statement.

A resource can have only one security label. For most types of resources the security administrator assigns a security label to each resource in the system that is to be protected by ensuring that the resource is protected by a profile, and by specifying the security label in the profile.

Exceptions:

- The security label for a resource protected by a profile in the JESSPOOL class is stored by JES with the spool file. The resource inherits its security label from the job that creates it.

- The security label for a z/OS UNIX file or directory is stored in the file security packet (FSP). The system usually assigns the security label when it creates the file or directory. For more information, see “Security labels for z/OS UNIX files and directories” on page 20.
- The security label for an IPC object is stored in the IPC security packet (ISP). The system assigns a security label to an IPC object when it creates the object. For more information, see “IPC objects” on page 23.

Using security labels

After security labels have been created and assigned, the security administrator can activate the RACF SECLABEL resource class to cause the system to use the security labels for authorization checks. Then, when a user tries to access a resource, RACF checks whether the resource has a security label. If it does, RACF compares the security label of the user with that of the resource. If the security labels allow access, RACF then checks the access list of the profile that protects the resource. The decision as to whether or not to allow the access is based on both mandatory access control (MAC), based on security labels, and discretionary access control (DAC), based on access lists.

To activate the SECLABEL class, the security administrator issues the command:

```
SETROPTS CLASSACT(SECLABEL) RACLIST(SECLABEL)
```

Note that when the SECLABEL class is active, unless the security administrator has also set certain system options, a user needs a security label only if the resource the user wants to access has a security label, and resources are not required to have security labels. To increase security, the security administrator can use the SETROPTS command to set RACF system options that *require* that certain resources have security labels, and *require* that any user who tries to access those resources have a security label. For information about these RACF system options, see “SETROPTS options that control the use of security labels” on page 26.

Mandatory access control (MAC)

Mandatory access control is based on the theory of dominance, and is achieved through the use of security labels.

Dominance

One security label *dominates* a second security label when the following two conditions are true:

- The security level that defines the first security label is *greater than or equal to* the security level that defines the second security label.
- The set of security categories that define the first security label *includes* the set of security categories that defines the second security label.

Two security labels are said to be *disjoint* or *incompatible* if neither dominates the other because they have incompatible sets of categories. For example, if security label A has categories apples and pears, and security label B has categories pears and bananas, neither contains all the categories of the other and so neither dominates the other and they are disjoint.

Preventing declassification of data

A mandatory access check compares the security labels of the subject and object and grants the subject access to the object as follows:

- A subject can read an object if the subject's security label dominates the object's security label.
- A subject can write to an object if the object's security label dominates the subject's security label. A subject cannot write to an object whose security label the subject's security label dominates, unless the security labels are equivalent – we say that the subject is not allowed to *write down*.
- A subject can both read and write an object only if the subject's and object's security labels are equivalent.

To ensure that a user does not declassify data, a subject can read from and write to (alter) only an object with an equivalent security label; however, a subject can copy information from an object having a security label that the subject's security label dominates to an object having the subject's current security label. Or, for objects that support write-only processing, such as z/OS UNIX files, to an object with a security label that dominates the subject's security label.

Based on the examples shown in , and assuming that the subject passes the discretionary access check, the following actions can take place between subjects and objects with the assigned security labels:

- PURPLE can write to PURPLE; PURPLE can read COLUMBIA and copy it to PURPLE; PURPLE can read UNION and copy it to PURPLE.
- COLUMBIA can write to COLUMBIA. For z/OS UNIX files and directories, which support write-only processing, COLUMBIA can also write to PURPLE.
- UNION can write to UNION. For z/OS UNIX files and directories, which support write-only processing, UNION can also write to PURPLE.

The RACF SETROPTS MLS option controls whether write-down is allowed. For information about the MLS option, see “The MLS and NOMLS options” on page 30.

Controlled write-down

There might be cases where you want to allow for controlled situations of write-down. The security administrator can assign a "write-down by user" privilege to individual users or groups of users that allows them to select the ability to write down. The security administrator activates and deactivates the privilege by creating the profile IRR.WRITEDOWN.BYUSER in the FACILITY class. A user can activate write-down mode if the profile exists, the user has at least READ access to it, and the FACILITY class is active and SETROPTS RACLISTed. If the user has UPDATE or higher access to the profile, write-down mode is active by default when the user enters the system. RACF provides the RACPRIV command, and z/OS UNIX provides the **writedown** command, that allow users who are authorized to the write-down privilege to reset and query the setting of their write-down mode.

Access rules

Access rules depend on the purpose for which a subject accesses an object and whether the subject is allowed to write down.

The subject is not allowed to write down

A subject is not allowed to write down if the RACF MLS option is active in any of the following situations:

- The security administrator has not set up controlled write-down.
- The security administrator has set up controlled write-down, but has not given the subject authorization to write down.
- The security administrator has set up controlled write-down, and has given the subject authorization to write down, but the subject has not activated write-down mode.

This should be the case for most users in a multilevel-secure environment. In this case, the access rules, depending on the purpose for which the subject accesses an object, are:

- **Read only:** A subject can read an object when the subject's security label dominates the object's security label.
- **Write only:** A subject can write to an object when the object's security label dominates the subject's security label.
- **Read/Write**: A subject can read from and write to an object only if the security labels of the subject and object are equivalent.

To review: A subject is able to access a resource (for reading) only if the subject's security label dominates the security label of the resource. Remember that a security label is a particular combination of security level and categories. For security label A to dominate security label B, the hierarchical security level of A must be equal to or greater than the security level of B. In addition, the set of non-hierarchical categories

for security label A must include all categories for security label B. If both of these requirements are met, then security label A dominates security label B, and a user with security label A can read resources with security label B.

Example of security label dominance

The following example illustrates dominance of security labels when the subject is not allowed to write down:

Assume your system includes the following three security labels:

- SECLABEL = PURPLE
  - SECLEVEL = SECRET
  - CATEGORY = PROJECTA, PROJECTB, PROJECTC
- SECLABEL = COLUMBIA
  - SECLEVEL = SENSITIVE
  - CATEGORY = PROJECTA, PROJECTB
- SECLABEL = UNION
  - SECLEVEL = UNCLASSIFIED
  - CATEGORY = PROJECTC

SECLABEL	**SECLEVEL**	**CATEGORY**
PURPLE	SECRET	PROJECTA, PROJECTB, PROJECTC
COLUMBIA	SENSITIVE	PROJECTA, PROJECTB
UNION	UNCLASSIFIED	PROJECTC

In this example:

- PURPLE dominates COLUMBIA because the hierarchical security level SECRET is greater than the security level SENSITIVE and PURPLE's non-hierarchical categories include all of COLUMBIA's categories.

Security level	**Project A**	**Project B**	**Project C**
Secret	SECLABEL= PURPLE		
Sensitive	SECLABEL= COLUMBIA		*no label defined*

- PURPLE also dominates UNION because the hierarchical security level SECRET is greater than the security level UNCLASSIFIED and PURPLE's non-hierarchical categories include all of UNION's categories.

Security level	**Project A**	**Project B**	**Project C**
Secret	SECLABEL= PURPLE		
Unclassified	*no label defined*	*no label defined*	SECLABEL=UNION

- COLUMBIA does not dominate UNION. It is true that COLUMBIA's hierarchical security level is greater than the security level of UNION. However, COLUMBIA's non-hierarchical categories (Project A, Project B) do not include all categories for UNION (Project C).

Security level	**Project A**	**Project B**	**Project C**
Sensitive	SECLABEL= COLUMBIA		*no label defined*
Unclassified	*no label defined*	*no label defined*	SECLABEL = UNION

The subject is allowed to write down

A subject is allowed to write down if the RACF MLS option is not active, or if the RACF MLS option is active and the security administrator has set up controlled write-down and given the subject the write-down privilege, and the subject has activated write-down mode. In this case, the access rules, depending on the purpose for which the subject accesses an object, are:

- **Read only:** A subject can read an object when the subject's security label dominates the object's security label.
- **Write only:** A subject can write to an object when the object's security label dominates the subject's security label, or when the subject's security label dominates the object's security label.
- **Read/Write**: A subject can read from and write to an object only if the subject's security label dominates the object's security label.

Reverse and equal mandatory access checking

So far we have discussed dominance checking for data sets, z/OS UNIX files and directories, and the majority of RACF general resource classes. However, for some RACF general resource classes dominance checking works differently, involving either a reversed check (reverse mandatory access checking) or a strict equality check (equal mandatory access checking):

- *Reverse mandatory access checking* applies to resources in the classes listed in Table 2 on page 16. With reverse mandatory access checking access rules are the reverse of the access rules for mandatory access checking:
  - When the subject is not allowed to write down:
    - **Read only:** A subject can read an object when the object's security label dominates the subject's security label.
    - **Write only:** A subject can write to an object when the subject's security label dominates the object's security label.
    - **Read/Write:** A subject can read from and write to an object only if the security labels of the subject and object are equal.
  - When the subject is allowed to write down:
    - **Read only:** A subject can read an object when the object's security label dominates the subject's security label.
    - **Write only:** A subject can write to an object if the security label of the subject dominates the security label of the object, or the security label of the object dominates the security label of the subject.
    - **Read/Write:** A subject can read from and write to an object when the object's security label dominates the subject's security label.
- *Equal mandatory access checking* applies to resources in the classes listed in Table 3 on page 16. With equal mandatory access checking the security label of the user must be equivalent to the security label of the resource. *Equivalence* of security labels means that either the security labels have the same name, or they have different names but are defined with the same security level and identical security categories. The security label SYSMULTI is considered equivalent to any security label.

  Equal mandatory access checking is used for any class where two-way communication is expected.

The type of mandatory access check that is done for a resource depends on the definition of the class to which the resource belongs. The RACF class descriptor table contains the definitions of the resource classes for the system. The system programmer or security administrator can define additional classes using the class descriptor table macro, ICHERCDE. The RVRSMAC and EQUALMAC operands specify that reverse or equal mandatory access checking is done for resources in the class that the macro defines. For more information on the ICHERCDE macro, see *z/OS Security Server RACF Macros and Interfaces*.

Table 2 on page 16 lists the RACF resource classes that require reverse mandatory access checking. Table 3 on page 16 lists the RACF resource classes that require equal mandatory access checking.

Table 2. Resource classes that require reverse mandatory access checking		
Classes		
APPCPORT	CONSOLE	WRITER

Table 3. Resource classes that require equal mandatory access checking			
Classes			
APPL DSNR	JESINPUT MQCONN	SERVAUTH SERVER	TERMINAL

Discretionary access control (DAC) checking

Once the user passes the mandatory access check, a discretionary check follows. The discretionary access check ensures that the user is identified as having a “need to know” for the requested resource. The discretionary access check uses other access control information, such as the access control list in the profile protecting a resource, or z/OS UNIX access control (permissions, the access control list, and the UNIXPRIV class).

Security labels for data transferred to tape or DASD

An object retains its security label even when it is transferred from virtual storage to a device attached to the system, such as a tape volume or a DASD device.

A DASD volume can contain data sets with many different levels of information. The hardware ensures that, while accessing one data set, a user is not allowed to access a different part of the same volume.

Tape data sets, in general, have the security label of the first data set on the tape volume when more than one data set is written to tape. Only data sets with the same security label can be written to the same tape volume. This restriction exists because the system assumes that a user with access to one data set on the tape volume should have access to all data sets on that tape volume. An exception is a virtual tape server (VTS), which allows a physical tape cartridge to contain multiple logical volumes. Each of the logical volumes can have a different security label, and the tape hardware ensures that a user cannot access data outside a logical volume.

An installation using tapes should define profiles in the TAPEVOL class for all tape volumes and activate the TAPEVOL class.

Security labels and data set allocation

A user can allocate a data set if the security label of the user dominates the security label of the data set. However, if the RACF MLS option is active and a user allocates a data set having a security level lower than the user's security level, the user will be unable to write to that data set. For example, if the allocation is done with ISPF, ISPF attempts to write an end of file marker after allocating the new data set. The write fails if the SETROPTS MLS option is active, and ISPF issues an error message, but the allocation is successful.

Guideline: A user should allocate only data sets that have the same security label as the user's current security label. A user should not allocate data sets that the user's current security label does not allow him or her to write to.

Printing security information on hardcopy output

When a data set is transferred to a printer, the security notation associated with the security label is printed on each page or logical piece of the hardcopy output. The security administrator can give a user RACF authorization to override this requirement. If a user with this authority overrides the labeling requirement, Print Services Facility (PSF) creates an audit record specifying the event. If a user without this authority attempts to override labeling, PSF does not print the data set and an audit record is created.

The PSF security overlay file contains the definitions of the notations to be printed on the hardcopy output. There is a member in the security overlay file for each security label in the system. By associating a security label with a member in the security overlay file, the correct notation is printed on each output page. For instance, if PURPLE is defined to reference member A of the security overlay file, and member A specified that the overlay was “RESTRICTED DISTRIBUTION - MANAGEMENT ONLY”, then that overlay would print on every output page.

In addition to the security notation printed on each page of the hardcopy output being labeled, a separator page precedes and follows each data set that is printed. A random number is generated for each data set printed. This number appears on the header and trailer pages that begin and end the data set. The printer operator verifies that these numbers match before releasing the hardcopy output to the user.

In a multilevel-secure system, hardcopy output transferred to cards is not allowed because this requirement for printing the security notation for the security label is not met.

For detailed information about printed output, see “PSF” on page 65.

Changing a security label

The security administrator assigns security labels to subjects and objects in the system before a user attempts to access a resource. The security label must remain unchanged while the user accesses the resource. Enabling two RACF options ensures the integrity of the security labels:

- SECLABELCONTROL

  This RACF option prevents users other than those with the RACF SPECIAL attribute from changing a security label by doing any of the following:

  – Changing a profile in the SECLABEL class with the RALTER command
  – Altering the SECLABEL field of a profile in any class
  – Issuing an ADDSD, ALTDSD, or DELDSD command that causes the security label of a cataloged data set to change

- MLSTABLE

  This RACF option indicates that the installation is using "multilevel stability", or "tranquility". No one can change a security label by performing any of the actions controlled by the SECLABELCONTROL option, or by renaming a resource, unless the system is in a tranquil state.

  Before the system can enter a tranquil state, the console operator must quiesce the system so that all untrusted jobs are completed and all users accessing data sets covered by a security label that is being changed are logged off. All network access must be halted by stopping all TCP/IP stacks. (For information about TCP/IP stacks, see *z/OS Communications Server: IP Configuration Guide*.) When these actions are complete, the security administrator sets the RACF option MLQUIET. The system is now in a tranquil state and changes to the security labels can be made. While the system is in this tranquil state, only the security administrator, a started task or system address space with the trusted attribute, and the console operator are allowed to use the system.

See “SETROPTS options that control the use of security labels” on page 26 for more information about the RACF MLQUIET, SECLABELCONTROL and MLSTABLE options.

Note: Security labels for interprocess communication (IPC) objects and z/OS UNIX files and directories can never be changed, regardless of the settings of the SECLABELCONTROL and MLSTABLE options. For more information, see “Using security labels with z/OS UNIX System Services” on page 18.

Renaming a resource can change its security label, if the new name is protected by a different profile with a different security label than the old name. If renaming a resource would change its security label, the rename is allowed only if one of the following is true:

1. The RACF MLSTABLE option is active, the RACF MLQUIET option is active, and the user has the RACF SPECIAL attribute.
2. The RACF MLS option is active, and at least one of the following is true:
   a. The profile protecting the old name has a security label of SYSNONE.
   b. All of the following are true:
      - The user has access to a security label that dominates the security label of the profile that protects the old name.
      - The user has access to a security label that is equivalent to the security label of the profile that protects the new name.
      - The security label of the profile that protects the new name dominates the security label of the profile that protects the old name.

Whenever a change is made to a security label, or the security label in a profile is changed, RACF generates a type 80 SMF record. This record contains the old security label, the new security label, and the command that caused the security label change.

When the ADDSD, ALTDSD, and DELDSD commands are issued to change a security label, RACF generates the SMF type 83 record in addition to the type 80 record. The type 83 record contains a list of the resources affected by the security label change and a link to the type 80 record that contains the RACF command that caused the security label change.

To resume normal operations after changing a security label or the profile that defines a security label, the security administrator must reset the RACF option MLQUIET by issuing the command SETROPTS NOMLQUIET. Restart all TCP/IP stacks and network servers.

Using security labels with z/OS UNIX System Services

Traditionally, access to z/OS UNIX resources is based on POSIX permissions and access control lists (ACLs). In a multilevel-secure z/OS UNIX environment, authorization checks are performed for security labels in addition to POSIX permissions, to provide additional security.

Associating security labels with remote users

In a z/OS UNIX environment, a user might be entering the system from a remote IP address using an application such as rlogin. If the SECLABEL class is active, there are two ways that a security label can be associated with a remote user:

1. The system attempts to derive a security label from the user's port of entry. The system determines the security label associated with the user's IP address, and whether the user is authorized to use the security label.
2. If a port of entry is not available, the system uses the application's current security label to determine access for the remote user.

Profiles in the SERVAUTH class represent IP addresses, and require security labels if the MLACTIVE option is active. Mandatory access checks for profiles in the SERVAUTH class are equivalence checks.

For information about configuring a system for use by remote users, see *z/OS Communications Server: IP Configuration Guide*.

Assigning a home directory and initial program depending on security label

Because a user can use different security labels at different times, a z/OS UNIX user might need to have different home directories and initial programs, depending on the security label in use. The user's home directory and initial program are specified in the OMVS segment of the user's profile, in the HOME and

PROGRAM fields. You can set up a special symbolic link and use it in the HOME and PROGRAM fields to provide different values depending on the user's current security label. One or more symbolic links of this type can reside in various directories, so that when they are encountered in pathnames the security label is dynamically substituted into the pathname when it is resolved. The contents of these symbolic links must be in one of the following formats:

- $SYSSECA/ or $SYSSECA/*pathname*

  The symbol $SYSSECA indicates that the user's current security label should be substituted into the pathname as an absolute directory name. This means that pathname resolution continues at the ROOT with a directory name of the user's current security label.

  Example: Assume the OMVS segment of a user's profile defines the user's home directory to be /u/secsyma/ussuser, and /u/secsyma is a symbolic link defined to have the contents $SYSSECA/. If the user logs on with the security label ABCSEC, the user's home directory resolves to /ABCSEC/ussuser.
- $SYSSECR/ or $SYSSECR/*pathname*

  The symbol $SYSSECR indicates that the user's current security label should be substituted into the pathname as a relative directory. This means that pathname resolution continues in the directory in which the symbolic link is encountered, with a directory name of the user's current security label.

  Example: Assume the OMVS segment of a user's profile defines the user's home directory to be /u/secsymr/ussuser, and /u/secsymr is a symbolic link defined to have the contents $SYSSECR/. If the user logs on with the security label ABCSEC, the user's home directory resolves to /u/ABCSEC/ussuser.

Note: The security label name is substituted in uppercase. The directories in the file system must be defined with the security label in uppercase, because directory names are case-sensitive.

If a user logs on without a security label, when security label substitution is performed it replaces the $SYSSECA/ and $SYSSECR/ symbols with a dot (.), allowing pathname resolution to proceed from the ROOT or from the current pathname directory. **Examples:**

- If /u/secsyma is a symbolic link defined to have the contents $SYSSECA/, and if the user's home directory is /u/secsyma/ussusr, if the user logs on without a security label, the home directory resolves to /./ussuser, or /ussusr.
- If /u/secsymr is a symbolic link defined to have the contents $SYSSECR/, and if the user's home directory is /u/secsymr/ussuser, and the user logs on without a security label, the path resolves to u/./ussusr, or /u/ussusr.

Security label substitution is performed only if the SECLABEL class is active. When the SECLABEL class is not active, the $SYSSECA/ and $SYSSECR/ symbols are not substituted, but instead are left in the pathname as relative directory names.

Note that in order for a user's home directory to be protected by a security label and contain objects that have security labels, it must reside in a zFS file system. **Exception:** If the home directory is in an HFS file system mounted in read-only mode, in some cases the system assumes a security label for it. For information about security labels for z/OS UNIX files and directories, see “Security labels for z/OS UNIX files and directories” on page 20.

Example of using security label substitution with automount

You can use security label substitution with automount.

Assume the following:

- ABCSEC and XYZSEC are security labels
- The SECLABEL class is active.
- A symbolic link /u/secsymr is defined with the contents $SYSSECR/.
- The user's home directory is defined to be /u/secsymr/ussuser.
- The directories /u/ABCSEC and /u/XYZSEC are automount-managed.

To set up automount to perform security label substitution, do the following

- Add the following entries to the automount master file /etc/auto.master:

```
/u/ABCSEC        /etc/u_ABCSEC.map
/u/XYZSEC        /etc/u_XYZSEC.map
```

- Create the automount MapName file /etc/u_ABCSEC.map containing the following entry:

```
name            *
type            ZFS
filesystem      ZFS.ABCSEC.<uc_name>
mode            rdwr
duration        10
delay           0
```

- Create the automount MapName file /etc/u_XYZSEC.map containing the following entry:

```
name            *
type            ZFS
filesystem      ZFS.XYZSECC.<uc_name>
mode            rdwr
duration        10
delay           0
```

When the user logs on with security label ABCSEC, the home directory resolves to /u/ABCSEC/ussusr, causing the file system ZFS.ABCSEC.USSUSR to be automounted on directory /u/ABCSEC/ussusr. When the user logs on with security label XYZSEC, the home directory resolves to /u/XYZSEC/ussusr, causing the file system ZFS.XYZSEC.USSUSR to be automounted on the directory /u/XYZSEC/ussusr .

Security labels and the su command

The z/OS UNIX **su** command starts a new shell and lets the issuing user operate in it with the privileges of another user. The shell started by the **su** command inherits the security label of the user who issued the command. The new user must be authorized to the inherited security label or the **su** command fails.

The **su** command has an option to start the new shell as a login shell, which attempts to duplicate the login environment of the new user. However, the login shell does *not* use the security label of the new user. Even for the su login shell, the new shell inherits the security label of the issuing user.

Security labels for z/OS UNIX files and directories

z/OS UNIX files and directories can be protected by security labels. The security label for a file or directory is stored in the file security packet (FSP) for the file or directory.

The SETROPTS MLFSOBJ option controls whether security labels are required for z/OS UNIX files and directories. This option is described in “The MLFSOBJ option” on page 28.

The z/OS File System (zFS) and the hierarchical file system (HFS) are both z/OS UNIX file systems that can be used with security labels. However, there are restrictions for using security labels with HFS files in a multilevel-secure environment, which are described in “Security labels for HFS file systems” on page 21.

Assumed security labels

z/OS UNIX assumes a security label for a file system if the following conditions are met:

- A DATASET profile that specifies a security label protects the file system data set.
- You mount the file system in read-only mode.
- The RACF MLFSOBJ option is active when you mount the file system.
- The root directory of the file system does not already have a security label.

z/OS UNIX determines the assumed security label when the file system is mounted in read-only mode, from the security label specified in the data set profile, and assumes it only for the duration of the mount. After unmount, the file system again has no security label.

If you update the security label of the file system data set, the next time you mount the file system in read-only mode the assumed security label for the file system reflects the new security label for the data set – the assumed security label can vary upon mount according to the value of the security label specified in the profile for the containing aggregate. In contrast, a real security label assigned to a file system cannot be changed.

Security labels for zFS file systems and their contents

The z/OS File System (zFS) supports security labels. A zFS file system is contained in a VSAM linear data set. The security label of the root within each zFS file system data set is determined at the time the file system aggregate (container) is allocated, from the security label of the profile in the DATASET class that covers the aggregate. If the SECLABEL class is active when allocating the aggregate, all file systems subsequently created within that aggregate contain a root with the security label that is specified in the profile for that aggregate. If no profile exists for the aggregate, or if it exists but does not specify a security label, then if the MLFSOBJ option is *not* active the root for any file systems within that aggregate have no security label. If the MLFSOBJ option *is* active, requiring security labels on all file system objects, the user's security label is assigned to the root of the file system.

If a zFS file system or aggregate has a security label, changing the security label specified in the data set profile does not change the security label of any zFS file systems contained within it. Once a file system is created with a security label it has that security label forever. (However, if a zFS file system does not have security label and is mounted read-only, the security label that z/OS UNIX assumes for it can change, as explained in “Assumed security labels” on page 20.)

If the SECLABEL class is active, when a new z/OS UNIX file or directory is created within a zFS file system, the system assigns it a security label as follows:

- If the parent directory has no security label, the new file or directory is not assigned a security label.
- If the security label of the parent directory is SYSMULTI, the security label of the new file or directory is set to the security label of the requesting address space (the user). If the user has no security label (this could occur only if the MLACTIVE option is not active), the new file or directory is not assigned a security label.
- If the security label of the parent directory is not SYSMULTI, the security label of the new file or directory is set to the security label of its parent directory.

Symbolic links are also protected by security labels. When the name-hiding function is active a user's security label must dominate the security label of the symbolic link to read the link.

Hard links are also protected by security labels. An attempt to create a hard link fails if the security label of the directory containing the link is not equivalent to the security label of the file to which the link points. An attempt to create a link to a character special file fails if either the file or its directory has a security label.

If a z/OS UNIX file, directory, or symbolic link was created in a zFS file system without being assigned a security label (for example, if the SECLABEL class was not active when the file, directory, or symbolic link was created), the security administrator can assign a security label to it using the **chlabel** shell command.

Once a file or directory has been assigned a security label, there is no way to delete or change the security label assigned to it, other than copying the file or directory to another directory with a different security label. To accomplish this when the SECLABEL class is active, the user must have security label authority to both the old file and the directory for the new file, and must ensure that no declassification of data occurs. Alternatively, the copy could be done after deactivating the SECLABEL class.

Security labels for HFS file systems

The hierarchical file system (HFS) does not fully support security labels and multilevel security.

Rule: In a multilevel-secure environment, use HFS file systems only in readonly mode. If you want to use files from an HFS file system in read-write mode, and use security labels in the file system, you must copy or move those files to a zFS file system.

It is possible for an HFS file to have a security label. For example, if the SECLABEL class is active when you create an HFS file system, the system assigns the root of that file system a security label in the same way that it assigns a security label to a zFS file system aggregate, from the data set profile. Subsequently, files created within that HFS file system will adopt security labels under the same rules as for zFS files. However, the HFS physical file system (PFS) does not support some functions of multilevel security, such as the **chlabel** shell command and the name-hiding function. If you attempt to use an HFS file system in read-write mode, if it has security labels it will not always behave predictably. Furthermore, if the file system does not have security labels, and the MLFSOBJ RACF option is active, users who try to access the file system will receive failures.

If you mount an HFS file system that does not have a security label in read-only mode, the system can assume a security label for it, as described in "Assumed security labels" on page 20.

If a HFS file system has a security label, changing the security label specified in the data set profile does not change the security label of the HFS file system. Once a file system is created with a security label it has that security label forever. (However, if an HFS file system does not have security label, and is mounted read-only, the security label that z/OS UNIX assumes for it can change, as explained in "Assumed security labels" on page 20.)

Even with assumed security labels, an HFS file system does not support the name-hiding function.
Guideline: If the name-hiding option is active, do not use HFS file systems unless they contain files and directories whose names should be viewable by all users.

Security labels and mount points

You might get unexpected results if the root of a mounted file system and the directory it is mounted on have security labels that are not equivalent. A user's ability to see directory entries when the name-hiding function is in effect and search through a directory into the mounted file system might not be consistent.

Why inconsistencies might occur: A directory can contain directory entries that are mount points for other file systems. When a user reads a directory, access checking for the name-hiding function operates against the mount point directory entry within the directory being read, and ignores the security label of a file system root mounted on that directory entry. However, when a user accesses the file system mounted on that directory entry (for example, to search the directory) access checking is done on the root of the mounted file system, and ignores the security label of the mount point directory. As a result, when the mount point and the root directories do not have equivalent security labels, a user might be able to list a directory that cannot be otherwise accessed, or might not be able to list a directory that can be otherwise accessed.

Guideline: To ensure consistent results between the name-hiding function and directory search access, do one of the following:

- Mount a file system only on a directory that has a security label equivalent to the security label of the file system's root.
- Use the SYSMULTI security label for mount point directories. SYSMULTI is equivalent to any security label.

Example: This example illustrates the inconsistency that can occur if you don't follow the guideline. Assume that a zFS mount point has a security label of SYSHIGH, and the root of an HFS file system mounted on it has a security label of SYSLOW. The name-hiding function is in effect. zFS allows only users who have a security label of SYSHIGH or SYSMULTI to see the directory entry for the file system. But any user can search through the directory into the mounted HFS file system.

Security label processing for communications between z/OS UNIX processes

In a multilevel-secure environment, communication can only occur between processes with equivalent security labels. The security label associated with the communication path is the security label of the process that created the path. Any other process using the path must be running with an equivalent security label.

z/OS UNIX processes can communicate with each other using several methods, including:

- Interprocess communications (IPC) objects
- Signals
- PTRACE
- Sockets
- Pipes
- FIFO special files

IPC objects

For communications using IPC objects, when RACF creates an IPC security packet (ISP), if the SECLABEL class is active RACF copies the security label of the process, if one exists, into the ISP. RACF rejects requests for subsequent connections if the connecting process does not have a security label equivalent to the security label in the ISP. Once a security label has been assigned to an IPC object, it cannot be changed.

To establish multilevel security for IPC objects, activate the SECLABEL class and activate the MLIPCOBJ RACF option. If the SECLABEL class is active, and the MLIPCOBJ RACF option is not active, the system assigns a security label to an IPC object only if the creating process had one. If the IPC object does not have a security label, the system does not require a security label for connecting processes. However, if the connecting process does have a security label, the connection fails. If the SECLABEL class is active, activating the MLIPCOBJ option causes the system to require a security label for all IPC objects and for all connecting processes.

Signals

The signal services kill() and pthread_kill() allow very limited information to be passed from one process to another, and do not use security labels. However, sigqueue(), although it allows only 4 bytes of information to travel with each signal, allows thousands of signals to be queued, and so can transmit a significant amount of data to another process. For sigqueue(), if the SECLABEL class is active, the signalling process and the target process must have equivalent security labels.

ptrace

The ptrace service allows one process to view and change the storage in another process. When the SECLABEL class is active, access checking for ptrace requires that the process initiating the ptrace and the target process have equivalent security labels.

Sockets

Like other files, a socket is assigned a security label when it is created. There are callable services that can pass sockets between processes, and those processes must have equivalent security labels for the sockets to be passed between them. The takesocket (BPX1TAK) callable service returns an error of EINVAL if the process receiving the socket does not have a security label equivalent to that of the process that issued the givesocket (BPX1GIV) callable service. Additionally, the recvmsg (BPX1RSM) callable service succeeds, but fails to create the desired file descriptors when the receiving process does not have a security label equivalent to that of the process that issued the sendmsg (BPX1SMS) callable service.

Pipes

A pipe is assigned a security label when it is created. The creation of the pipe opens two file descriptors, one for reading from the pipe and one for writing to the pipe. It is common to pass one end of the pipe to another process via spawn or fork and exec. A file descriptor for a pipe can be passed, like that for any other file, between processes that might or might not have equivalent security labels. See “Passing open file descriptors when a process changes identity” on page 24 for the expected behavior when passing file descriptors when a process changes identity.

FIFO special files

FIFO special files are named pipes. They differ from pipes in that they are specifically opened for read or for write. Access to a FIFO special file for read and write is checked when the FIFO special file is opened, and if the process does not have a security label appropriate for accessing the FIFO in the desired mode, access is denied. This processing is the same as for access to any other file.

Passing open file descriptors when a process changes identity

Any process that is to receive an open file descriptor should have a security label equivalent to that of the originating process. z/OS UNIX Systems Services does not allow a process to change its security label via an identity changing function unless the process is already running with a security label of SYSMULTI. However, if a process is running with SYSMULTI, it would be possible to change the security label to another, and give the process access to file descriptors opened when the process was running as SYSMULTI. This scenario could lead to the declassification of data. When the potential for this condition is detected, message BPXP022I is issued to indicate that a violation might have occurred. The text of the message is as follows: `BPXP022I PROCESS pid CHANGED FROM SYSMULTI TO A NON-SYSMULTI SECLABEL WITH AN OPEN FILE OR SOCKET DESCRIPTOR.` However, the message does not eliminate the possibility that data could be declassified. **Guideline:** To prevent data declassification, evaluate processes running with a security label of SYSMULTI to ensure that prior to an identity change all open files are closed.

Using system-specific security labels in a sysplex

In a sysplex it can be useful to limit the use of certain security labels to certain members of the sysplex. This allows one member of the sysplex to run work at security label A, while another handles work at security label B, keeping work separated based on security classification while still sharing the RACF database. The SETROPTS option SECLBYSYSTEM allows you to use security labels on a per-system basis.

Restrictions: The following restrictions apply to system-specific security labels:

- JES3 does not support the use of system-specific security labels. Do not activate the SECLBYSYSTEM SETROPTS option if you are using JES3.
- JES2 does not support using system-specific security labels for systems that perform NJE and OFFLOAD processing. These systems must have all security labels active. In addition, JES2 printers can not process output unless the security label associated with the output is active on the system controlling the printer.
- If you define system-specific security labels, be aware that using a generic TSO system name at logon might not work, because the user could be allocated to a system where the user's security label is not active.
- If you use Application Restart Manager (ARM) to manage applications and you use system-specific security labels, ensure that the systems that you've told ARM to use when restarting an application are systems that have the appropriate security label active. Otherwise, ARM might try to restart an application requiring a particular security label on a system where the security label is not active, and the application restart will fail.

Defining and activating system-specific security labels

To define system-specific security labels, the security administrator specifies on which systems a security label is to be active by adding a member list to the SECLABEL resource class profile. The member names are system SMF IDs containing 1–4 characters. For example, to define the security label named SECRET as being active only on the systems with SMF system IDs SYSA and SYSB, the security administrator could define SECRET with a command like:

```
RDEFINE SECLABEL SECRET....ADDMEM(SYSA,SYSB)
```

If no member list of system IDs is added, the security label is considered to be active on all systems sharing the RACF database.

The security labels SYSHIGH, SYSLOW, SYSNONE, and SYSMULTI are always considered to be active on all systems. If a member list is added to one of their profiles, it is ignored.

To activate the use of system-specific security labels, activate the SECLBYSYSTEM option and refresh the SECLABEL class:

```
SETROPTS SECLBYSYSTEM
SETROPTS RACLIST(SECLABEL) REFRESH
```

Shared file system environment and system-specific security labels

Shared file systems refers to an environment for sharing z/OS UNIX file systems in a sysplex environment, and applies to any file system type configurable for z/OS UNIX, including HFS and zFS. If you have a sysplex and are using shared file systems, you might get unexpected results if you define system-specific security labels and activate the RACF SECLBYSYSTEM option. Access checks might succeed or fail in ways that you don't expect if a user is running on a system other than the system where the data is owned, and either of the systems has security labels defined that are not defined on the other system.

If you want to use the SECLBYSYSTEM option in a shared file system environment, be aware of the following:

- Security checking for most operations occurs on the system where the user is running. For a shared file system client most checking is done on the client system. However, security checking for the name-hiding function for a command that lists the contents of a directory is done on the system that owns the data, and that might not be the system where the user is running.
- If you use automount or automove, a file system might be moved to a system on which one or more of its security labels are not defined, causing data to not be available. You can use the automove system list (SYSLIST) to ensure that if the owning system leaves the sysplex, the file system is not moved to a system on which a security label contained in the file system is undefined. The SYSLIST option of automove is not supported for automounted file systems, but should not be necessary. Automount unmounts the file system if it is not in use from another system, or automatically moves it to a system where it is currently being used.
- If an object has a SYSMULTI security label, RACF grants a subject access to the object without doing further checks on the subject's security label, because SYSMULTI is equivalent to any other security label. As a result, in a shared file system environment with SECLBYSYSTEM active, if the user's security label is not defined on the system on which the access check is being done, RACF does not determine that the security label is undefined.

Examples: The following examples illustrate unexpected results that can occur in a shared file system environment if you activate the RACF SECLBYSYSTEM option:

1. A user might not be given access to objects to which the user's security label grants access.

   Assume the following:

   - SYSA and SYSB are two systems in a shared file system environment.
   - The RACF SECLBYSYSTEM option is active.
   - The name-hiding function is in effect (the RACF MLNAMES option is active).
   - The security label LBLA is defined on SYSA but not on SYSB.
   - A user, JOEUSER, is logged on to SYSA, with the security label LBLA.
   - JOEUSER is trying to access data owned by SYSB.

   Because security checking for most operations occurs on the system where the user is running, SYSA, for most operations JOEUSER's security label would be defined. For example, JOEUSER can search a directory on SYSB that has security label LBLA, even though LBLA is not defined on SYSB. But if JOEUSER tries to display data items in that directory that have security label LBLA, the checks for the name-hiding function are done on SYSB where the LBLA security label is undefined, and data items on SYSB with the LBLA security label are not displayed.

2. A user might be able to see the names of objects to which the user's security label does not grant access.

   Assume the following:

   - SYSA and SYSB are two systems in a shared file system environment.
   - The RACF SECLBYSYSTEM option is active.
   - the name-hiding function is in effect (the RACF MLNAMES option is active).
   - The security label LBLA is defined on SYSA but not on SYSB.
   - The security label LBLB is defined on SYSB but not on SYSA.
   - The security label LBLB dominates LBLA.
   - A user, JOEUSER, is logged on to SYSB, with the security label LBLB.
   - JOEUSER is trying to access data owned by SYSA.

   If JOEUSER tries to list the contents of a directory owned by SYSA that has a SYSMULTI security label, RACF gives JOEUSER access to the SYSMULTI directory without checking JOEUSER's security label, because SYSMULTI is equivalent to any security label. RACF does not find out that JOEUSER's security label, LBLB, is not defined on SYSA. When the system lists the objects in the directory, it lists all objects that JOEUSER's security label dominates, including any that have security label LBLA, even though LBLA is not defined on JOEUSER's system. This happens because the access checks for the name-hiding function are done on SYSA, and LBLA is defined on SYSA. However, if JOEUSER tries to access one of the objects listed that has security label LBLA, the access check is done on SYSB, where LBLA is not defined, and the attempt fails.

   Note that if the system-specific security labels LBLA and LBLB are disjoint (that is, neither dominates the other because they each have at least one security category that the other does not) the name-hiding function hides the names of the objects with security label LBLA, because JOEUSER's security label does not dominate LBLA. Defining system-specific security labels to be disjoint ensures that JOEUSER does not see the names of objects to which JOEUSER is not authorized.

Guidelines: If you want to activate the RACF SECLBYSYSTEM option in a shared file system environment, you can minimize problems by doing the following:

- Mount a file system only on a system on which all security labels contained in the file system are defined. For example, if you have a sysplex with members SYSA, SYSB, and SYSC, and you have defined the security label "XYZ" to be valid only on SYSA and SYSB, do not mount a file system with the security label "XYZ" on SYSC or use SYSC as a backup for SYSA or SYSB.
- When mounting a file system, use the automove system list (SYSLIST) to ensure that if the owning system leaves the sysplex, the file system is not moved to a system on which a security label contained in the file system is undefined.
- Define system-specific security labels to be disjoint from security labels to be used on other systems, especially when the name-hiding function is in effect (the RACF MLNAMES option is active). Doing this ensures that the file system can be used from only one system, and will not automove.
- Do not use system-specific security labels (the RACF SECLBYSYSTEM option is active) for automount-managed file systems.
- If the RACF SECLBYSYSTEM option is active and you define security labels that are equivalent, define them to be active on the same systems.

SETROPTS options that control the use of security labels

You use the RACF SETROPTS command to control how security labels are used on your system. Different SETROPTS options control different aspects of security label function. These options are described in the following sections:

- COMPATMODE and NOCOMPATMODE
- MLACTIVE and NOMLACTIVE

- MLFSOBJ
- MLIPCOBJ
- MLNAMES and NOMLNAMES
- MLQUIET and NOMLQUIET
- MLS and NOMLS
- MLSTABLE and NOMLSTABLE
- SECLABELAUDIT and NOSECLABELAUDIT
- SECLABELCONTROL and NOSECLABELCONTROL
- SECLBYSYSTEM and NOSECLBYSYSTEM

Before you can activate most of these SETROPTS options, the SECLABEL class must be active. Activating the SECLABEL class is discussed in "Using security labels" on page 12.

For more information on SETROPTS options, see *z/OS Security Server RACF Security Administrator's Guide*. For information about the SETROPTS command, see *z/OS Security Server RACF Command Language Reference*.

The COMPATMODE and NOCOMPATMODE options

The COMPATMODE option allows a user to access a resource if the user is authorized to use a security label that would allow the access, regardless of whether the user is using the security label at the time of the authorization check. The NOCOMPATMODE option requires that the user be using a security label that allows the access in order to be granted access. NOCOMPATMODE is in effect when a RACF database is first initialized using IRRMIN00.

The COMPATMODE option only applies if the creator of the ACEE for the user had an old RACINIT parameter list. The only reason to use COMPATMODE is to prevent an old application from failing while you test. Correct the application to use a current RACINIT protocol, or replace it with an application that does.

Guideline: Do not set the COMPATMODE option.

The MLACTIVE and NOMLACTIVE options

Use the MLACTIVE and NOMLACTIVE options to control whether security labels are required for certain resources. The MLACTIVE option requires security labels for most resources other than resources related to z/OS UNIX, and for all users entering the system. The MLACTIVE option has two suboptions, FAILURES and WARNING:

- MLACTIVE(FAILURES) specifies that RACF is to reject any request to access a resource in the classes listed in Table 4 on page 28 that does not have a security label. For a detailed description of the access checking methodology, see *z/OS Security Server RACF Security Administrator's Guide*. RACF also rejects any attempt by a user to enter the system without a security label, unless the user is authorized to use the SYSLOW security label, in which case the user runs with the SYSLOW security label. In addition, a user task running in a server address space must have a security label that is equivalent to the security label of the address space.
- MLACTIVE(WARNING) specifies that RACF is to issue a warning for any request to access a resource in a class listed in Table 4 on page 28 that does not have a security label, but allow the access if the request passes the discretionary access check. RACF also issues a warning for any attempt by a user to enter the system without a security label, but allows the user to enter the system. In addition, a user task running in a server address space must have a security label that is equivalent to the security label of the address space.

The NOMLACTIVE option specifies that security labels are not required for resources in the classes listed in Table 4 on page 28.

You can set MLACTIVE(WARNING) temporarily when you are setting up multilevel security, to verify that you have assigned security labels to all of the users and resources that require them. Then set MLACTIVE(FAILURES) when you are ready to enforce the use of security labels.

Before you activate MLACTIVE(FAILURES), ensure that you have done the following tasks:

1. You have defined security labels by defining profiles in the RACF SECLABEL class.
2. You have authorized all users to use the security labels they will need.
3. You have assigned security labels to all applications and started tasks that act as servers, authenticating users in their address spaces. If the users can operate at multiple security labels, you should assign SYSMULTI to the started tasks. This includes started tasks that are marked trusted, such as JES2. The security label for a started task must be assigned prior to the task starting. If a task is only started at IPL time, you might need to re-IPL before you activate MLACTIVE(FAILURES), to ensure that the task has a security label.
4. You have assigned security labels to all data sets.
5. You have assigned security labels to all profiles in the classes shown in Table 4 on page 28.
6. You have activated and RACLISTed the SECLABEL class.
7. You have temporarily run with MLACTIVE(WARNING) set to verify that you have completed your setup correctly.

Requirement: The SECLABEL class must be active before you can activate the MLACTIVE option.

Guideline: Run with the MLACTIVE(FAILURES) option active.

Table 4. Resource classes that require a security label when MLACTIVE(FAILURES) is active

Classes				
APPCPORT APPCSERV APPCTP APPL DATASET DEVICES DIRECTRY* DSNADM DSNR	FILE* GDSNBP GDSNCL GDSNDB GDSNJR GDSNPN GDSNSC GDSNSG GDSNSM	GDSNSP GDSNTB GDSNTS GDSNUF MDSNBP MDSNCL MDSNDB MDSNJR MDSNPN	MDSNSC MDSNSG MDSNSM MDSNSP MDSNTB MDSNTS MDSNUF SERVAUTH SERVER	TAPEVOL TERMINAL VMLAN* VMMAC* VMMDISK* VMSEGMT* WRITER

Attention: Classes marked with a * are VM classes and should not be relevant on a z/OS system.

When the MLACTIVE option is active, if a user creates a profile in any of the classes listed in Table 4 on page 28, the system assigns a security label to the profile. If the command that creates the profile does not specify a security label, the issuing user's current security label is used.

The MLFSOBJ option

Use the MLFSOBJ option to control whether security labels are required for z/OS UNIX files and directories. The MLFSOBJ option has two suboptions, ACTIVE and INACTIVE:

- MLFSOBJ(ACTIVE) specifies that when the SECLABEL class is active, only trusted or privileged started tasks can access files and directories that do not have security labels. For information about how security labels are assigned to z/OS UNIX files and directories, see “Security labels for z/OS UNIX files and directories” on page 20.
- MLFSOBJ(INACTIVE) specifies that files and directories do not require security labels.

If you issue the command SETROPTS MLSFOBJ without specifying ACTIVE or INACTIVE, ACTIVE is the default.

Before you activate MLFSOBJ(ACTIVE), ensure that you have done the following tasks:

1. You have defined security labels by defining profiles in the RACF SECLABEL class.
2. You have authorized all users to use the security labels they will need.
3. You have assigned security labels to all z/OS UNIX files and directories.
4. You have activated and RACLISTed the SECLABEL class.
5. It is a good idea to re-IPL after you have assigned security labels to all users and activated the SECLABEL class, to ensure that all file systems have security labels.

Requirement: The SECLABEL class must be active before you can activate the MLFSOBJ option.

Guideline: Run with MLFSOBJ(ACTIVE) set.

The MLIPCOBJ option

Use the MLIPCOBJ option to control whether security labels are required for interprocess communication. The MLIPCOBJ option has two suboptions, ACTIVE and INACTIVE:

- MLIPCOBJ(ACTIVE) specifies that when the SECLABEL class is active, all IPC objects must have a security label. Those that don't can only be accessed by trusted or privileged started tasks.
- MLIPCOBJ(INACTIVE) specifies that IPC objects do not require a security label.

If the SECLABEL class is active, security labels are assigned to IPC objects during object creation, and security labels are checked before access is allowed to an IPC object that has a security label. However, as long as the MLIPCOBJ option is not active, any IPC object that is running without a security label can be accessed. When you activate the MLIPCOBJ option, IPC objects running without a security label can no longer be accessed. Before you activate the MLIPCOBJ option, let your system run with the SECLABEL class active, to allow the system to assign security labels to IPC objects as they are created. Run until you are sure that all active IPC objects have been created by users who have a security label. Or, re-IPL to be certain that all IPC objects have security labels.

Before you activate MLIPCOBJ(ACTIVE), ensure that you have done the following tasks:

1. You have defined security labels by defining profiles in the RACF SECLABEL class.
2. You have authorized all users to use the security labels they will need.
3. You have activated and RACLISTed the SECLABEL class.
4. You have ensured that all IPC objects have security labels, by either re-IPLing after you assigned security labels to all users and activated the SECLABEL class, or running with the SECLABEL class active until you are sure that all IPC objects have security labels.

Requirement: The SECLABEL class must be active before you can activate the MLIPCOBJ option.

Guideline: Run with MLIPCOBJ(ACTIVE) set.

The MLNAMES and NOMLNAMES options

Use the MLNAMES and NOMLNAMES options to control whether the name-hiding function is in effect. For a description of the name-hiding function, see “The name-hiding function” on page 5.

- The MLNAMES option specifies that the name-hiding function is active, with the following results:
  - Users cannot view the names of z/OS UNIX files and directories that their current security label does not give them authority to read.
  - Users cannot view the names of data sets that a mandatory access check followed by a discretionary access check does not allow them to read.
  - Users listing catalogs or directories cannot see the names of resources that they cannot currently read.
  - Users cannot read a VTOC directly, unless they have been given authorization to the profile in the FACILITY class that protects the VTOC.

  - If a zFS directory is read as a file (for example, through the **strings** command), zero bytes are returned.
- The NOMLNAMES option specifies that the name-hiding function is not in effect. Users can view the names of files, directories, and data sets regardless of their authority to read them.

Guideline:

- If you do not have a need to protect the names of data sets, files, and directories, run with the NOMLNAMES option set. Because the MLNAMES option can adversely affect performance, do not run with it active unless you need the protection it provides.
- If you need to protect the names of data sets, files, and directories, run with the MLNAMES option set.

The MLQUIET and NOMLQUIET options

Use the MLQUIET and NOMLQUIET options to control whether the system is in a tranquil state. When the MLSTABLE option is active, authorized users cannot make changes to security labels or change the security labels associated with resources until the security administrator sets the MLQUIET option.

- The MLQUIET option prevents users other than SPECIAL users, console operators, and started procedures from logging on, starting new jobs, or accessing resources. This option prevents these users from using the RACROUTE AUTH, DEFINE, and VERIFY requests.
- The NOMLQUIET option resumes normal processing.

Requirement: The SECLABEL class must be active before you can activate the MQUIET option.

Guideline: Run with NOMLQUIET active. Set the MLQUIET option temporarily when you need to change profiles in the SECLABEL class or change the SECLABEL field in profiles.

The MLS and NOMLS options

Use the MLS and NOMLS options to control whether users who do not have the write-down privilege can write down. The MLS option, together with the write-down privilege, helps prevent declassification of data. The MLS option has two suboptions, FAILURES and WARNING:

- MLS(FAILURES) specifies that RACF is to reject any request to write down, unless the user issuing the request has write-down mode active. For a description of write-down, see “Preventing declassification of data” on page 12. For a description of write-down mode, see “Controlled write-down” on page 13.
- MLS(WARNING) specifies that RACF is to issue a warning for any request to write down, and allow the request.

  Exception: z/OS UNIX files and directories do not support the WARNING mode. MLS(WARNING) has the same effect as MLS(FAILURES) for z/OS UNIX files and directories.
- NOMLS specifies that requests to write down are allowed, and users can copy data to a lower security label.

You can set MLS(WARNING) temporarily when you are setting up multilevel security, to verify that you have given all users who need it the write-down privilege. Then set MLS(FAILURES) when you are ready to prevent write-down by unauthorized users.

When the MLS option is active, a user running with a security label cannot write to a data set that does not have a security label.

Before you activate MLS(FAILURES), ensure that you have done the following tasks:

1. You have defined security labels by defining profiles in the RACF SECLABEL class.
2. You have authorized all users to use the security labels they will need.
3. You have assigned security labels to all data sets.

   Note: It is possible to run with the MLS(FAILURES) option active, without assigning security labels to all data sets and without activating the MLACTIVE option. However, you must be careful about how you assign security labels, because a user without a security label will not be able to write to a data set

that has a security label, and a user with a security label will have to log on without a security label to write to a data set that does not have a security label.

4. If you need to allow some users to write-down, you have activated the write-down by user privilege by creating the profile IRR.WRITEDOWN.BYUSER in the FACILITY class, you have given users who require the write-down privilege access to the profile, and you have activated and RACLISTed the FACILITY class.
5. You have activated and RACLISTed the SECLABEL class.
6. You have temporarily run with MLS(WARNING) set to verify that you have completed your setup correctly.

Requirement: The SECLABEL class must be active before you can activate the MLS option.

Guideline: Run with the MLS(FAILURES) option active.

The MLSTABLE and NOMLSTABLE options

These options control whether authorized users can make changes to security labels or change the security labels associated with resources while the system is not quiesced.

- The MLSTABLE option prevents authorized users from doing the following while the system is not quiesced:
  - Changing profiles in the SECLABEL class with the RALTER command
  - Changing the SECLABEL field in profiles

  Security labels can be changed only when any possible users of the security labels are logged off and the security administrator has issued the RACF command SETROPTS MLQUIET.
- NOMLSTABLE specifies that there are no restrictions on when authorized users can change security labels.

Guideline: Run with the MLSTABLE option active.

The SECLABELAUDIT and NOSECLABELAUDIT options

You can specify auditing options for a specific security label. You specify these options in the profile in the SECLABEL class that defines the security label. The auditing options specified for the security label are used in addition to the auditing options specified for the user or resource. The additional auditing occurs whenever an attempt is made to access or define a resource protected by a profile, file security packet (FSP), or IPC security packet (ISP) that has a security label specified, or whenever a user running with a security label attempts to access or define a resource. If the user and resource have different security labels, auditing occurs if either security label's options specify auditing. If both security labels' options specify auditing, the auditing done is based on the options specified for the resource's security label.

Example: To specify auditing of all failed accesses to resources that have a security label of EAGLE, and all failed accesses by users that have a security label of EAGLE, issue the following command:

```
RALTER SECLABEL(EAGLE) AUDIT(FAILURES(READ))
```

The SECLABELAUDIT and NOSECLABELAUDIT options determine whether RACF does the additional auditing specified on the SECLABEL profiles. You must have the RACF AUDITOR attribute to issue a SETROPTS command specifying these options.

- SECLABELAUDIT specifies that RACF is to do the additional auditing specified in the profiles in the SECLABEL class.
- NOSECLABELAUDIT specifies that RACF is not to do the additional auditing specified in the profiles in the SECLABEL class.

The SECLABELAUDIT option also determines whether audit records for object creation and access contain the object's security label. These audit records always contain the subject's security label. They

provide the object's security label when the SECLABELAUDIT option is active and the auditing options for the label's profile in the SECLABEL class require auditing.

For more information about the SECLABELAUDIT option, see *z/OS Security Server RACF Auditor's Guide*.

The SECLABELCONTROL and NOSECLABELCONTROL options

These options control whether users other than those with the RACF SPECIAL attribute can make changes to security labels and change the security labels associated with resources.

- SECLABELCONTROL specifies that users other than those with the RACF SPECIAL attribute cannot do the following:
  - Change a profile in the SECLABEL class with the RALTER command.
  - Change the SECLABEL field of a profile.
  - Issue an ADDSD, ALTDSD, or DELDSD command that causes the security label of a data set to change.
  - Designate a resource as delegated. For information about delegated resources, see Defining delegated resources in *z/OS Security Server RACF Security Administrator's Guide*.
- NOSECLABELCONTROL specifies that users other than those with the RACF SPECIAL attribute are not restricted from changing security labels and can perform the actions listed above if they have the appropriate authorization. For example, the owner of a resource profile (or someone with ALTER authorization for a discrete profile) can change the security label of a data set or resource profile using an ALTDSD or RALTER command, or can change the security label of a data set by issuing an ADDSD or DELDSD command that causes a different profile to protect the data set. (However, if the MLSTABLE option is active, that option prevents profile owners from issuing ALTDSD, ADDSD, or DELDSD commands to change the security label of a data set.) When NOSECLABELCONTROL is active, a user issuing the ADDSD command can explicitly specify a security label, rather than having the system assume the user's current security label.

Guideline: Run with the SECLABELCONTROL option active.

The SECLBYSYSTEM and NOSECLBYSYSTEM options

Use these options to control activation of security labels on a system image basis in a sysplex. For more information on using system-specific security labels, see "Using system-specific security labels in a sysplex" on page 24.

- The SECLBYSYSTEM option specifies that security labels are defined on a system basis. When SECLBYSYSTEM is active, the SMF ID values specified in the member list of the profiles in the SECLABEL class determine whether or not a security label is valid for a system. A security label that is not valid for a system is considered inactive and cannot be used on that system. It can be listed on that system only by a user with SPECIAL or AUDITOR authority. After you activate the SECLBYSYSTEM option, you must issue a `SETROPTS RACLIST(SECLABEL) REFRESH` command to complete the activation of security labels by system.
- The NOSECLBYSYSTEM option specifies that security labels are not defined on a system basis. All security labels are valid on all systems that share the RACF database.

Requirement: The SECLABEL class must be active before you can activate the SECLBYSYSTEM option.

Guideline: Run with the NOSECLBYSYSTEM option active unless you need to use system-specific security labels.

Chapter 3. Establishing multilevel security

This topic is a planning guide to establishing multilevel security on a system. The security administrator should use it as a checklist to make sure that all necessary hardware and software definitions are established correctly and that any dependencies are taken into account. This topic outlines the tasks required and references specific documents where more detailed information can be found.

In this topic

About this task

The tasks required to establish multilevel security are summarized as follows, and discussed in detail in the remainder of this chapter.

Subtask	See . . .
Set up the physical environment.	"The physical environment" on page 33
Establish the hardware configuration.	"The hardware configuration" on page 33
Establish the software configuration.	"The software configuration" on page 34
Define security labels.	"Defining security labels" on page 36
Assign security labels.	"Assigning security labels" on page 37
Set up software.	"Setting up your software for multilevel security" on page 46
Activate multilevel security.	"Activating multilevel security" on page 94

The physical environment

To enable full multilevel security protection, most hardware devices should be located in a physically secure area. The processors, control units, DASD, tape devices, printers, and console terminals process data at different levels of security, and should be in a secure location. Only people who have a system high security clearance should have access to this secure area.

You do not need to place TSO terminals in a location that is physically secure, because the security administrator defines the TSO/E users and specifies the commands that each TSO/E user can submit.

You can place remote printers in a secure area where needed. Only users with the appropriate security clearance for the output that prints there should have physical access.

The hardware configuration

IBM zSeries servers provide facilities to protect the contents of processor storage from unauthorized access. You can use each of the following configurations in a multilevel-secure environment:

- Processor complex

  A processor complex can be any processor running z/OS. The processor must have one of the following:

  – A single LPAR
  – Multiple LPARs that belong to the same JES complex or sysplex
  – Multiple LPARs that do not belong to the same JES complex or sysplex, but meet either of the following conditions:

    - They share the same RACF database.

- The hardware I/O definition is such that the LPARs share no DASD.

  Additionally, each LPAR must treat the other as a single-level system for networking (TCP/IP, JES NJE) purposes. With these conditions multiple LPARs can be treated as though they are separate stand-alone machines or a multi-system complex.

- Multiple system complexes and sysplexes

  A sysplex consists of multiple z/OS systems coupled together by hardware elements and by software services.

Use only printers that support guaranteed print labeling for output requiring security labels and secure separator pages. Other printers can be used in a multilevel-secure system, but only for non-secure output. For information about how to determine whether a printer supports guaranteed print labeling, see *PSF for z/OS: Security Guide*.

The software configuration

Some z/OS components do not fully support multilevel security. In order to configure a z/OS system for multilevel security, you need know:

- Which z/OS components are required in a multilevel-secure environment
- Which z/OS components do not support multilevel security, and cannot be used
- Which z/OS components partially support multilevel security, and how to configure them in amultilevel-secure environment

Required software

In order to establish multilevel security on a z/OS system, you must install the RACF component of the Security Server optional feature.

In addition, you should install the following optional feature, for backing up and restoring z/OS UNIX files:

- DFSMSdss

If you want to implement system-specific security labels, you must install the following software:

- JES2

You must also install the following two IBM products that are not part of z/OS, for secure printing:

- IBM Print Services Facility Version 4 Release 7 for z/OS (PSF V4.7.0). The program number is 5655-M32.
- Overlay Generation Language/370 Version 1 (OGL/370 V1R1, 5688-191)

If your system is a shared DASD environment (multiple z/OS systems sharing workload and DASD), be aware of these restrictions:

- All z/OS systems must share the RACF database in order to make identical security decisions.
- The z/OS systems in the global resource serialization complex must be the same set of z/OS systems that are sharing the RACF database.
- The JES complex (JES2 multi-access spool or JES3 complex) must be either the same set of z/OS systems that share the RACF database, or a subset of these systems. Note that, if your system is a sysplex, your JES2 subsystem must be part of a multi-access spool environment.

This document does not discuss how to install the required software or how to set it up for a non-multilevel-secure environment. Use the instructions you receive with your software. After you complete the installation and setup of your system, use this book to determine the additional setup required for multilevel security.

Before you install z/OS

If you are installing z/OS for the first time, and if you have defined a security label named SYSMULTI on your system, you must define a new security label to replace the SYSMULTI security label, update every profile that specifies the SYSMULTI security label to specify the new security label, and then delete the SYSMULTI security label. You must do this *before* the first time that you install z/OS V1R5 or a later release; you will not be able to do it after the install.

Tip: You can use the SEARCH command with the SECLABEL and CLIST operands to do this. For each resource class issue the following command to build a CLIST:

```
SEARCH CLASS(class) SECLABEL(SYSMULTI)
       CLIST('RALTER class ' ' SECLABEL(new_seclabel)')
```

To execute the CLIST issue the command:

```
EXEC EXEC.RACF.CLIST
```

z/OS elements and features that do not support multilevel security

The following z/OS elements and features do not support multilevel security:

- Infoprint Server
- Bulk data transfer facility (BDT)
- Tivoli® Directory Server

To insure the integrity of your multilevel-secure system, you should not install these elements and features. If they are already installed, you should remove them or disable them, or ensure that no one can use them for sensitive data.

z/OS elements and features that partially support multilevel security

Other z/OS elements and features support multilevel security only partially. The functions that they do not support are summarized in Table 5 on page 35. The actions that you need to take to use those elements and features safely in a multilevel-secure environment are identified in "Setting up your software for multilevel security" on page 46.

Table 5. z/OS elements and features that do not completely support multilevel security

z/OS element or feature	Support not provided for multilevel security
zFS File System	• Server message block (SMB) server does not support multilevel security.
DFSMS	• DFSMShsm does not support the name-hiding function. • DFSMSdss does not support the name-hiding function. • OAM object support for content management-type applications does not support multilevel security.
JES2	• Network job entry (NJE) does not support multilevel security if there are any NJE links to devices or systems that have more than one security label. • Remote job entry (RJE) does not support multilevel security if there are any RJE links to devices or systems that have more than one security label.

Table 5. z/OS elements and features that do not completely support multilevel security (continued)	
z/OS element or feature	**Support not provided for multilevel security**
JES3	• JES3 does not support system-specific security labels. • Network job entry (NJE) does not support multilevel security if there are any NJE links to devices or systems that have more than one security label. • Remote job processing (RJP) does not support multilevel security if there are any RJP links to devices or systems that have more than one security label.
TCP/IP	For a list of applications and commands that do not support multilevel security, see *z/OS Communications Server: IP Configuration Guide*.
TSO/E	• The Information Center Facility does not support multilevel security.
z/OS UNIX System Services	• The hierarchical file system (HFS) does not fully support security labels and multilevel security. You should not mount an HFS file system in read-write mode. You can copy or move an HFS file system to a z/OS File System (zFS) if you need to mount it in read-write mode. • The cron daemon does not support multilevel security. • The pax and tar commands do not copy, save, or restore security labels. When pax or tar restores or copies a directory tree, it might assign security labels as described in "Security labels for z/OS UNIX files and directories" on page 20, and possibly over-classify some data. • The UUCP (UNIX-to-UNIX copy program) group of commands does not support multilevel security.

Software applications

Software applications run on z/OS systems, but are not z/OS elements and features. For information about running software applications on a z/OS multilevel-secure system, see Chapter 6, "Adding authorized programs to a multilevel-secure system," on page 107.

Defining security labels

You must define security labels in order to establish multilevel security. For information about security labels, see Chapter 2, "Security labels," on page 9.

Steps for defining security labels

Before you begin:

- You must have established the software configuration, as described in "The software configuration" on page 34. You must be using RACF as your security product.
- You must know the security levels, security categories, and security labels your installation will use.

  Tip: Minimize the number of security labels you define that have different names but are equivalent - the system can quickly determine that a subject's and object's security labels are the same if they have the same name, but must do additional processing if their names are not the same to determine if they are equivalent.

Perform the following steps to create security labels for your installation.

1. Define the SECLEVEL profile in the SECDATA class. For example:

```
RDEFINE SECDATA SECLEVEL UACC(NONE)
```

2. Define your security levels as members of the SECLEVEL profile. For example, to define the security levels UNCLASSIFIED, CONFIDENTIAL, and SECRET:

```
RALTER SECDATA SECLEVEL  ADDMEM(UNCLASSIFIED/10, CONFIDENTIAL/20, SECRET/30)
```

3. Define the CATEGORY profile to the SECDATA class. For example:

```
RDEFINE SECDATA CATEGORY UACC(NONE)
```

4. Define your security categories as members of the CATEGORY profile. For example, to define the categories PROJECTA, PROJECTB, and PROJECTC:

```
RALTER SECDATA CATEGORY  ADDMEM(PROJECTA, PROJECTB, PROJECTC)
```

5. Define your security labels. Use the RACF RDEFINE command to specify the SECLABEL class profiles. The profile names are the security labels and are limited to eight characters. For example, to define the security labels EAGLE and SPARROW:

```
RDEFINE SECLABEL EAGLE SECLEVEL(SECRET) ADDCATEGORY(PROJECTA,PROJECTB)
    UACC(NONE)
RDEFINE SECLABEL SPARROW SECLEVEL(UNCLASSIFIED) UACC(NONE)
```

   You do not need to create the SYSHIGH, SYSLOW, SYSNONE, and SYSMULTI security labels; they are created automatically during RACF initialization.

   Guideline: Do not define security labels that start with the characters "SYS". System-defined security labels start with "SYS". Using other characters for your security labels ensures that in the future if RACF expands the list of security labels it creates automatically, a new system-defined label will not have the same name as an installation-defined label and you will not have to rename your label and update all of the profiles that use it.

When you are done, your security levels, security categories, and security labels are defined, but the system does not use them for access checks because the SECLABEL class is not active. Note that after you activate the SECLABEL class, if any user without a security label tries to access a resource that has a security label, the access fails.

Guideline: To avoid access failures, assign security labels to all users before you activate the SECLABEL class.

Assigning security labels

You should assign security labels to all users before you activate the SECLABEL resource class. You should assign security labels to all data sets, all z/OS UNIX files and directories, and most resources before you activate the MLACTIVE, MLFSOBJ, and MLIPCOBJ SETROPTS options.

Assigning security labels to users

You need to assign user IDs to users such as started procedures, z/OS UNIX daemons, and other processes not associated with human users, as well as to human users. In a multilevel-secure

environment, it is important that human users do not share user IDs and passwords, in order to ensure accountability. Make sure that every human user who can access the system has a unique user ID. User IDs assigned to started procedures, daemons, and processes not associated with human users can be shared without the same level of concern as for human users. In particular if any started procedures, daemons, or processes must run with UID(0), they might want to share the same user ID, because a RACF database running in Application Identity Mapping stage 3 has a limit of roughly 130 user IDs with UID(0).

Note: To create a user ID, use the RACF ADDUSER command. To add a default security label to a user's profile, or otherwise modify the profile, use the ALTUSER command. To assign a user ID to a started procedure, define a profile in the STARTED class (or use the started procedures table, ICHRIN03). For more information on administration tasks related to users, see *z/OS Security Server RACF Security Administrator's Guide*.

You must assign at least one security label to each user in a multilevel-secure environment. To assign a security label to a user, use the PERMIT command to add the user to the access control list for the profile for the security label in the SECLABEL class. For example, to authorize USER05 to use the EAGLE and SPARROW security labels:

```
PERMIT EAGLE CLASS(SECLABEL) ACCESS(READ) ID(USER05)
PERMIT SPARROW CLASS(SECLABEL) ACCESS(READ) ID(USER05)
```

Guideline: Add a default security label to each user's profile. Be sure that you permitted the user to the SECLABEL profile for the default security label. For example, to define a default security label of SPARROW for user USER05:

```
ALTUSER USER05 SECLABEL(SPARROW)
```

Tip: To add default security labels to a large number of user profiles, use the SEARCH command to generate a TSO CLIST that you can tailor (by editing) and then run. For example:

```
SEARCH CLASS(USER) CLIST('ALTUSER ' ' SECLABEL(most-common-security-label)')
```

Edit the CLIST, and change the SECLABEL field to the appropriate default security label where necessary. After tailoring the CLIST, run it with the command:

```
EXEC EXEC.RACF.CLIST
```

Recommended security labels for users

Table 6 on page 38 lists the security labels that should be assigned to specific users.

Table 6. Recommended security labels for users. Each user listed should be authorized to use its recommended security label, and should have its recommended security label defined to be its default security label.

User	Recommended security label	Notes
BPXAS started procedure	SYSMULTI	
CIM server	SYSMULTI	See the topic about required RACF setup in *z/OS Common Information Model User's Guide*.
Console operator	SYSHIGH	Because a console can contain output from any address space in the system, it can contain data of any security label.
JES2 or JES3 started procedure	SYSMULTI	This security label allows ACEEs with differing security labels to be anchored in TCBs in the JES2 or JES3 address space.

Table 6. Recommended security labels for users. Each user listed should be authorized to use its recommended security label, and should have its recommended security label defined to be its default security label. (continued)

User	Recommended security label	Notes
MVRSHD server		Do *not* run with SYSMULTI. Run a separate instance for each security label you need to support, under a job name assigned to a user ID with the appropriate security label. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
OMPROUTE	SYSMULTI	Run one instance for each TCP/IP stack that is using dynamic route configuration. The user ID must have UPDATE authority to the EZB.STACKACESS profile in the SERVAUTH resource class. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
OMVS started procedure	SYSMULTI	
OpenSSH daemon	SYSMULTI	For more information about the OpenSSH daemon, see *z/OS OpenSSH User's Guide*.
OpenSSH privilege separation user (SSHD)	SYSMULTI	For more information about this user ID, see *z/OS OpenSSH User's Guide*.
RACF started procedure	SYSMULTI	This security label allows ACEEs with differing security labels to be anchored in TCBs in the RACF address space.
Resolver address space	SYSMULTI	The resolver task is started by OMVS and runs under the same identity. For information about resolvers, see *z/OS Communications Server: IP Configuration Guide*, and *z/OS UNIX System Services Planning*.
Security administrator	SYSHIGH	At least one user with the RACF SPECIAL attribute should be authorized to use the SYSHIGH security label, in case of a problem with security labels. If both the MLS and MLACTIVE options are in FAILURES mode and a RACF SPECIAL user logs on with the SYSHIGH security label, RACF processes all security label checks for that user as if the MLS and MLACTIVE options are both in WARNING mode.
SYSLOG daemon (syslogd)	SYSMULTI	Run one instance per system. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
System programmers who work with system dump and trace data	SYSHIGH	

Table 6. Recommended security labels for users. Each user listed should be authorized to use its recommended security label, and should have its recommended security label defined to be its default security label. (continued)

User	Recommended security label	Notes
TCP/IP stack, restricted	The security label of the stack	A restricted TCP/IP stack ensures that all sockets are opened by applications running with the security label of the stack. Configure the stack with a user ID that has the security label of the stack. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
TCP/IP stack, unrestricted	SYSMULTI	An unrestricted TCP/IP stack allows sockets to be opened by applications with any security label. Configure the stack with a user ID that has the SYSMULTI security label. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
TN3270 server on a restricted stack, NACUSERID	Must be equivalent to the security label of the stack user ID	In a multilevel-secure environment, all TN3270 ports must have a NACUSERID configured to ensure correct LU mapping. The NACUSERID can be the same user ID used to run the stack. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
TRMD	The same security label as the stack it is servicing, or SYSMULTI	Run one instance of TRMD for each TCP/IP stack that has IDS functions configured. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.
zFS administrators	SYSHIGH	
zFS started procedure	SYSMULTI	
z/OS UNIX policy agent (pagent)	SYSMULTI	Run one instance per system. The user ID that pagent is running under should have READ access to the EZB.STACKACCESS profiles in the SERVAUTH resource class for all stacks on the system. For more information, see Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide*.

Listing security labels for users

The LISTUSER command displays information about a user's security label:

- If a user issues the LISTUSER command and does not specify a user ID, the command displays the security label with which the user logged on.
- If a security administrator issues the LISTUSER command for another user's user ID, the command displays the default security label from the user's USER profile. A security administrator cannot determine which security label a user is currently using.

Assigning security labels to data sets

When MLACTIVE(FAILURES) is active, if a data set is not protected by a profile, or if the profile that protects a data set does not have a security label assigned to it, every attempt to access the data set fails.

Therefore, you need to ensure that every data set is protected by a profile in the DATASET class, and that every profile in the DATASET class has a security label, before you activate MLACTIVE(FAILURES).

Tip: The RACF PROTECTALL option ensures that a user can create or access a data set only if it is RACF-protected. If you are not already running with the PROTECTALL option in FAILURES mode, activate it in WARNING mode while you are assigning security labels to your data set profiles:

```
SETROPTS PROTECTALL(WARNING)
```

RACF will issue a warning message if a user attempts to create or access a data set that is not RACF-protected. When you are sure that all of your data sets are RACF-protected, activate the PROTECTALL option in FAILURES mode:

```
SETROPTS PROTECTALL(FAILURES)
```

If a user attempts to create or access a data set that is not RACF-protected, the attempt fails.

Guidelines: To determine the security label to assign to a system data set, consider the data that the data set contains:

- Data that has no classified content and can be read by all users can have a security label of SYSLOW (or an installation-defined security label) and a UACC of READ, or an entry in the global access checking table specifying READ access. Data sets such as SYS1.LINKLIB and SYS1.PROCLIB are in this category.
- Data that has no classified content and needs to be accessed by only certain users can have a security label of SYSLOW (or an installation- defined security label) and a UACC of NONE. If a user requires access to the data set, the user must be permitted specifically. The access authority (for example, to READ or to UPDATE) can be set for each individual user allowed to access the data set. Examples of this type of data set are SYS1.PARMLIB and SYS1.VTAMLST.
- Assign all catalogs a security label of SYSNONE.
- Assign the SYSHIGH security label to data sets that contain multiple levels of data. To further protect these data sets from unauthorized access, specify a UACC of NONE and permit only certain users to access the data set.

Note: Regardless of the protection established for data sets in the LPA concatenation, any user can read most of the data set contents by examining the link pack area (LPA) in virtual storage. Because the data sets' contents are exposed, it is important to note that data sets classified higher than SYSLOW should not be in the LPA concatenation.

Tip: To add default security labels to a large number of data set profiles, use the SEARCH command to generate a TSO CLIST that you can tailor (by editing) and then run. For example, to generate a CLIST that sets all discrete profiles to the most common security label, use the command:

```
SEARCH CLASS(DATASET) CLIST('ALTDSD ' ' SECLABEL(most-common-seclabel)') NOGENERIC
```

Edit the CLIST, and change the SECLABEL field to the appropriate security label where necessary. After tailoring the CLIST, run it with the command:

```
EXEC EXEC.RACF.CLIST
```

To generate a CLIST that sets all generic profiles to the most common security label:

```
SEARCH CLASS(DATASET) CLIST('ALTDSD ' ' SECLABEL(most-common-seclabel)') GENERIC
```

Table 7. Recommended security labels for profiles in the DATASET class		
Data set	**Recommended security label**	**Notes**
Catalogs	SYSNONE	Define a UACC of READ or UPDATE, as appropriate. Give ALTER access only to users who maintain the catalogs, because ALTER access allows users to list the names of data sets cataloged in the catalogs.
DFSMShsm control data sets and their logs and journals	SYSHIGH	Define a UACC of NONE
DFSMSrmm control data sets and their logs and journals	SYSHIGH	Define a UACC of NONE
Dump analysis and elimination (DAE) data sets	SYSHIGH	Define a UACC of NONE
Dump job data sets	SYSHIGH	Define a UACC of NONE
JES2 checkpoint data set	SYSHIGH	Define a UACC of NONE
JES2 spool offload data set	SYSHIGH	Define a UACC of NONE
JES3 checkpoint data sets	SYSHIGH	Define a UACC of NONE
JES3 dump job data set	SYSHIGH	
JES3 job control table (JCT) data set	SYSHIGH	
Log data sets	SYSHIGH	Define a UACC of NONE
Page data sets	SYSHIGH	Define a UACC of NONE
PSF security libraries (overlay, font, page segment, security definitions)	SYSHIGH	Define a UACC of NONE
SMF data sets	SYSHIGH	Define a UACC of NONE
SMS configuration data sets (CDS), source control data set (SCDS) and active control data set (ACDS)	SYSHIGH	Define a UACC of NONE
Spool data sets	SYSHIGH	Define a UACC of NONE
Spool offload data sets	SYSHIGH	Define a UACC of NONE
Swap data sets	SYSHIGH	Define a UACC of NONE
SYS1.dump data sets	SYSHIGH	Define a UACC of NONE
SYS1.LINKLIB	SYSLOW	Define a UACC of READ
SYS1.IMAGELIB	SYSLOW	Define a UACC of READ

Table 7. Recommended security labels for profiles in the DATASET class (continued)

Data set	Recommended security label	Notes
SYS1.PARMLIB	SYSLOW or installation-defined	Define a UACC of NONE
SYS1.PROCLIB	SYSLOW	Define a UACC of READ
SYS1.VTAMLIST	SYSLOW or installation-defined	Define a UACC of NONE
System data sets that have no classified content and can be read by all users	SYSLOW	Define a UACC of READ
System data sets that contain multiple levels of data	SYSHIGH	Define a UACC of NONE
System data sets that have no classified content and need to be accessed by only certain users	SYSLOW or installation-defined	Define a UACC of NONE
Trace data sets	SYSHIGH	Define a UACC of NONE
TSO/E broadcast data set	SYSLOW	Define a UACC of READ
TSO/E NAMES data set	The lowest security label to which the user has access	Allows TRANSMIT and RECEIVE to access the data set, and the user can update the data set when logged on at the security label assigned to it. (The data set is named *userid*.NAMES.TEXT.)
TSO/E log data set	User's most commonly used security label	A user authorized to more than one security label requires a log data set for each of those security labels, and when using a security label other than the one assigned to LOG.MISC must use the LOGDSNAME or LOGDATASET keyword on the TRANSMIT or RECEIVE command to specify the data set to use for logging. (The data set is named *userid*.LOG.MISC.)
TSO/E user message log data set (*logname.userid*)	SYSHIGH	The log can contain any level of information.
XCF couple data sets	SYSHIGH	Define a UACC of NONE
zFS debug settings data set	SYSLOW	The debug_settings_dsn option in the IOEFSPRM file specifies the data set name.
zFS IOEFSPRM file	SYSLOW	
zFS output message data set	SYSLOW	The msg_output_dsn option in the IOEFSPRM file specifies the data set name.

Table 7. Recommended security labels for profiles in the DATASET class (continued)		
Data set	**Recommended security label**	**Notes**
zFS root file system	SYSHIGH	Set the security label for the VSAM data set to SYSMULTI when you create the VSAM data set and format it as a zFS file system, to assign SYSMULTI to the root. Then change the security label to SYSHIGH.
zFS trace table	SYSLOW	The trace_dsn option in the IOEFSPRM file specifies the data set name.
zFS translated message data set	SYSLOW	The msg_input_dsn option in the IOEFSPRM file specifies the data set name.
z/OS UNIX file systems	See Table 13 on page 89	The security label for a z/OS UNIX data set should be consistent with the security label for the mountpoint.

Assigning security labels to system resources

Table 4 on page 28 lists the resource classes that require security labels when MLACTIVE(FAILURES) is active. If a resource is protected by a profile in one of these classes, and that profile does not have a security label assigned to it, any attempt to access the resource when MLACTIVE(FAILURES) is active fails. Therefore, you need to update every profile that you have created in the classes listed in Table 4 on page 28 to add a security label.

Choose the security labels of the TERMINAL, SERVAUTH and JESINPUT classes carefully, because their security labels override the user's default security label if the user doesn't specify a security label when entering the system from these ports of entry.

Tip: To add default security labels to a large number of resource profiles, use the SEARCH command to generate a TSO CLIST that you can tailor (by editing) and then run. For example:

```
SEARCH CLASS(resource-class) CLIST('RALTER ' ' SECLABEL(most-common-seclabel)')
```

Protecting data

In addition to assigning security labels to profiles in the DATASET class (see "Assigning security labels to data sets" on page 40), there are other steps you should take to protect data in a multilevel-secure environment.

Ensuring that user data sets are erased when scratched or released

The erase-on-scratch attribute should apply to all user data sets. Set the ERASE(ALL) option described in "SETROPTS options" on page 71.

Protecting DASD volumes

A user who has DASDVOL authority to a direct access device could potentially dump the volume or rename any residual data sets still left on the pack. To protect DASD volumes, ensure that profiles in the DASDVOL class are defined with the security label SYSHIGH and UACC(NONE), and that only appropriate trusted users are on the access lists.

Protecting data on tape

To protect the data on tape volumes, the security administrator defines RACF profiles in the TAPEVOL resource class. The profiles apply both to private and to scratch pool volumes. Use only tapes with a TAPEVOL profile in a multilevel-secure system.

DFSMSrmm provides support for creating and maintaining TAPEVOL profiles, using the TPRACF parmlib option. For more information, see "Using DFSMSrmm" on page 49.

Private volumes

A private volume has an associated security label in the TAPEVOL profile. Only data with the same security label can be written to the tape volume.

The security administrator can establish access lists for the TAPEVOL profile via the RACF PERMIT command, to allow specific users to access the tape volume or prevent specific users from accessing the tape volume. The discretionary access control check is performed after the mandatory access control check is passed.

Scratch pool volumes

The security administrator defines TAPEVOL profiles with the TVTOC option for scratch tapes. RACF places the user ID of the user who first writes information to that tape into the TAPEVOL profile. Other users can add data sets to the volume only if they have been placed in the volume's access list with at least UPDATE authority. The security label of the first data set written to the tape is assigned as the security label of the tape volume. As on a private volume, a scratch pool volume cannot contain data of different security labels.

When the user no longer requires the scratch pool volume, the security administrator ensures that the data on the tape is erased, the tape volume is reinitialized, and a new profile is created for the tape volume. The security administrator can use one of two methods to erase the data on the tape:

- Degauss the tape. This method does not work for all types of tape.
- Use a tape management system such as DFSMSrmm that supports erasing via the hardware data security erase (DSE) function when tapes become scratch. This method works for all types of tape. For more information, see "Using DFSMSrmm" on page 49.

For information about protecting data sets on both DASD and tape, see *z/OS Security Server RACF Security Administrator's Guide*.

Protecting temporary data sets

The security administrator must protect temporary data sets in a multilevel-secure system. To achieve this temporary data set protection, activate the TEMPDSN resource class. This class controls who can access and delete temporary data sets. There are no profiles defined in the TEMPDSN resource class.

Normally, temporary data sets are considered protected from any accesses except by the job or session that created them, because the job or session has an exclusive ENQ on the data set from creation through deletion, and therefore the data sets do not need to be protected by a RACF profile. However, the following situations could leave a temporary data set unprotected:

- System failure
- Initiator failure or initiator terminated by the FORCE command
- Automatic restarts during the time between the failure and the restart
- A GRS configuration that suppresses multi-system ENQs for temporary data sets

To protect data in these situations, TEMPDSN restricts access to a temporary data set such that only the creating job or session, or a user with OPERATIONS, can access them. A user with OPERATIONS is only allowed to scratch the data sets.

See *z/OS Security Server RACF Security Administrator's Guide* for information about protecting temporary data sets.

Protecting catalogs

Any user might need to create MVS data sets and update one of the system catalogs to point to the created data sets. Because MVS manages the data content of the catalogs, they do not contain any sensitive information (except possibly data set names, which an installation can protect via the name-hiding function). Thus it is appropriate to specify a security label of SYSNONE for catalogs in order to ensure that users can update them regardless of a user's security label at the time the update occurs.

If you are implementing the name-hiding function, keep in mind that ALTER access to a catalog allows a user to see all names of data sets cataloged in that catalog, while UPDATE access does not. UPDATE access to a catalog is adequate for all users except catalog administrators. ALTER access is only necessary for administrators who maintain and repair catalogs.

Setting up your software for multilevel security

Each z/OS element, feature, and software product has specific requirements and directions for its setup. This topic describes considerations for multilevel security that affect how you set up your software. This topic also suggests additional topics that contain detailed information about how to set up the software.

Common Information Model (CIM)

The Common Information Model (CIM) is a standard data model developed by a consortium of major hardware and software vendors. CIM for z/OS provides for support for CIM on z/OS, and includes the CIM server, which manages communication between clients and providers.

The CIM server supports multilevel security. The security administrator can specify security labels on WBEM profiles that control access to CIM function and providers. The security administrator can control client connections to the CIM server by defining SERVAUTH profiles and specifying security labels.

Where to find more information

z/OS Common Information Model User's Guide

User ID for the CIM started procedure

Authorize the user ID associated with the CIM started procedure to use the SYSMULTI security label, and define its default security label to be SYSMULTI.

zFS File System

The zFS File System element consists of two components, only one of which supports multilevel security:

- Server Message Block (SMB) server – does *not* support multilevel security
- zFS – supports multilevel security

zFS supports the creation of security labels on individual files and directories stored in zFS file systems. It invokes RACF during authorization checking to determine whether the user is authorized to the object. It supports the name-hiding function.

Where to find more information

z/OS File System Administration

User ID for the zFS started procedure

Authorize the user ID associated with the zFS started procedure to use the SYSMULTI security label, and define its default security label to be SYSMULTI.

User IDs for zFS administrators

Authorize all zFS administrator user IDs to use the SYSHIGH security label.

zFS configuration data sets

Assign a security label of SYSLOW to the following zFS configuration data sets:

- The IOEFSPRM file
- The translated message data set (specified in the msg_input_dsn option in the IOEFSPRM file)
- The debug settings data set (specified in the debug_settings_dsn option in the IOEFSPRM file)
- The output message data set (specified in the msg_output_dsn option in the IOEFSPRM file)
- The trace table (specified in the trace_dsn option in the IOEFSPRM file)

zFS File System restrictions

To ensure that security is not compromised in a multilevel-secure environment:

- Ensure that the Server Message Block (SMB) server is not activated.

Checklist for zFS File System setup

Use the following checklist to ensure that you complete all the tasks required to set up zFS File System for multilevel security:

- Assign a default security label of SYSMULTI to the user ID associated with the zFS started procedure.
- Authorize each zFS administrator to use the SYSHIGH security label.
- Assign a security label of SYSLOW to each of the following zFS configuration data sets:
  - The IOEFSPRM file
  - The translated message data set (specified in the msg_input_dsn option in the IOEFSPRM file)
  - The debug settings data set (specified in the debug_settings_dsn option in the IOEFSPRM file)
  - The output message data set (specified in the msg_output_dsn option in the IOEFSPRM file)
  - The trace table (specified in the trace_dsn option in the IOEFSPRM file)
- Ensure that the Server Message Block (SMB) server is not activated.

DFSMS

DFSMS supports storage management for DASD and tape. DFSMS has four components:

- DFSMSdfp controls data, DASD, and tape storage for the operating system. DFSMSdfp acts as the link between the processor and the storage devices to provide storage, data, and device management functions. DFSMSdfp is a base element of z/OS.
- DFSMSdss provides functions including moving or copying data between volumes, managing DASD space, data backup and recovery, and converting data sets and volumes to system-managed storage. DFSMSdss is an optional feature of z/OS.
- DFSMShsm provides space management which improves DASD space usage, availability management, and application backup and recovery for disaster recovery purposes. DFSMShsm is an optional feature of z/OS.
- DFSMSrmm manages removable media resources, including tape cartridges and reels. DFSMSrmm is an optional feature of z/OS.

Where to find more information

- *z/OS DFSMSdfp Diagnosis*
- *z/OS DFSMSdfp Storage Administration*

- *z/OS DFSMS Using the New Functions*
- *z/OS DFSMSrmm Managing and Using Removable Media*
- *z/OS DFSMSrmm Implementation and Customization Guide*
- *z/OS MVS Initialization and Tuning Guide*
- *z/OS MVS Initialization and Tuning Reference*
- *z/OS Security Server RACF Security Administrator's Guide*

DFSMS provides the following support for multilevel security:

- DFSMSdfp supports multilevel security, including the name-hiding function.
- DFSMSrmm uses resource profiles in the RACF FACILITY, DATASET, and TAPEVOL classes to authorize access to information in the DFSMSrmm control data set about volumes and data sets. DFSMSrmm also supports the name-hiding function.
- DFSMShsm supports mandatory access control. It does not support the name-hiding function. DFSMShsm allows security administrators to control authorization to DFSMShsm storage administrator and user commands, using profiles in the RACF FACILITY class. For more information on the FACILITY class profiles, see *z/OS DFSMS Using the New Functions*.
- DFSMSdss supports mandatory access control. It preserves security labels when it dumps or restores zFS file systems. It does not preserve security labels when it dumps or restores MVS data sets. It does not support the name-hiding function.
- OAM functions in support of system-managed or automated manual tape libraries support multilevel security, because OAM's support is at the tape volume level versus the data set level.

The name-hiding function

When the name-hiding function is active (the MLNAMES option is active), DFSMSdfp does not display the name of, or any other information about, a data set that a user requests using a generic name unless the user has authorization to the data set. For example, if a user issues a LISTCAT command with the LEVEL keyword, LISTCAT displays only the names of data sets to which the user has authorization. Requests for information about a specific data set name, such as a LISTCAT command with the ENTRY keyword, or specifying an exact data set name on an ISPF catalog or VTOC listing panel, are not affected by the name-hiding function.

A user who can read the VTOC or VTOC index can read the data set names listed in them. When the name-hiding function is active DFSMS limits read access to the VTOC and VTOC index, to protect the names of data sets. DFSMS protects the VTOC with resources in the FACILITY class named STGADMIN.IFG.READVTOC.*volser*. When the name-hiding function is active, a user who does not have FACILITY class authorization to a volume cannot read the VTOC or VTOC index for that volume directly. (The user can still read a VTOC indirectly using system services and functions such as the ISPF panels that allow listing VTOCs, but is restricted to retrieving information only for those data sets she can access.)

Ways in which a user might access the VTOC include:

- The IEHLIST utility
- ISMF in ISPF
- The DSLIST utility for printing or displaying lists of data set names in ISPF

If you need to allow some users to read the complete VTOC for a volume when the name-hiding function is active, bypassing name-hiding restrictions, create a profile in the FACILITY class protecting the volume. Specify UACC(NONE) to prevent users who aren't in the access control list from accessing the VTOC, and add users who are allowed to read the VTOC to the access control list.

Example: To give the user USER10 authorization to read the VTOC for the volume with volume serial 123456:

```
RDEFINE FACILITY STGADMIN.IFG.READVTOC.123456 UACC(NONE)
PERMIT STGADMIN.IFG.READVTOC.123456 CLASS(FACILITY) ID(USER10) ACCESS(READ)
```

The system's GQSCAN and ISQUERY functions can allow users to see data set names that they do not already know. Therefore, if you are setting up name-hiding, you should protect global resource serialization services. For information on how to do this, see “Protect global resource serialization services” on page 61.

Using DFSMSrmm

DFSMSrmm has the capability to force tapes to be erased before they are returned to the scratch pool. To configure DFSMSrmm to do this, use the DFSMSrmm SECCLS parmlib option for parmlib member EDGRMM*xx*, specify the data set masks to be used, and specify the erase option. For example:

```
SECCLS NUMBER(30) NAME(CC) DESCRIPTION('CONF') MASK('**') SMF(N) MESSAGE(N) ERASE(Y)
```

At OPEN time, DFSMSrmm uses the masks to determine which data sets get the ERASE release action set. When the data on the volume expires, the volume is set pending release for return to scratch. At that time the ERASE action is set on and the EDGINERS tape labeling and erasing utility is used to erase the tape. For more information on the SECCLS parmlib option, see *z/OS DFSMSrmm Implementation and Customization Guide*.

Ensure that the DFSMSrmm parmlib option TPRACF is set to AUTOMATIC or PREDEFINED, so that DFSMSrmm ensures that all tape volumes have a profile in the RACF TAPEVOL class.

To prevent users from altering tape labels, you can use DFSMSrmm facilities and also set up automated operations replies. Use the STGADMIN.EDG.LABEL.* and STGADMIN.EDG.NOLABEL.* profiles in the FACILITY class to prevent the changing of labels on tape volumes. Use the EXPDTCHECK(Y) operand of the VLPOOL parmlib command to prevent expiration dates in tape labels being overridden. Then set up automated operations replies to the following WTORs, which are issued when a user attempts to alter a tape label. The responses cause the system to reject and dismount the tape volume:

- IEC507D reply 'M' to unload
- IEC534D reply 'M' to unload
- IEC704A reply 'M' to unload

You can use MPF (message processing facility) and NetView to automate operations replies. For more information, see *NetView Automation Planning*, SC31-7082-00.

Storage Management Subsystem (SMS)

SMS, a facility of DFSMS, provides centralized management of external storage as specified by the currently active storage management policy. The Interactive Storage Management Facility (ISMF), which is shipped as part of DFSMSdfp, provides the interactive interface to define an installation's storage management policy in an SMS configuration.

The SMS configuration is stored in a data set called the source control data set (SCDS). The security administrator uses ISMF options to define and modify the SCDS.

ISMF also provides for the creation and modification of ACS (automatic class selection) routines that are used in the selection of SMS classes and groups for SMS- managed data sets. These routines are stored in an SCDS in object format using ISMF.

The security administrator also uses ISMF functions to validate the set of ACS routines and SMS classes and groups and to test the ACS routines.

The security administrator must protect SMS to prevent unauthorized users from modifying the information in the control data sets. You can use the RACF STORCLAS and MGMTCLAS resource classes to protect the ability of a resource owner to use management and storage classes. Use the RACF PROGRAM resource class to prevent unauthorized users from running selected ISMF programs. Use the RACF FACILITY resource class with STGADMIN resources to control such functions as activating a configuration or performing catalog functions on SMS data sets.

See *z/OS DFSMSdfp Storage Administration* for information on using ISMF to define storage management functions.

See *z/OS Security Server RACF Security Administrator's Guide* and *z/OS DFSMSdfp Storage Administration* for information on implementing protection for SMS.

SMS-managed temporary data sets

To protect SMS-managed temporary data sets in a JES3 environment, specify REVERIFY=YES in the IGDSMS*xx* member of SYS1.PARMLIB.

See *z/OS MVS Initialization and Tuning Reference* for a description of the IGDSMS*xx* member of SYS1.PARMLIB.

Separation of DASD data with different security labels

In an SMS-managed environment, you can separate data having different security labels at a storage group level. To do this, use the storage class ACS exit to retrieve the security label from the object's profile, determine the storage class based on the security label, and redrive the storage class ACS routine to have the storage class assigned. Then use the storage class to derive the storage group in the storage group ACS routine.

DFSMS restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Do not use the Object Access Method (OAM) for access to OAM objects. OAM is a component of DFSMSdfp. OAM object support for content management-type applications does not support multilevel security.
- DFSMSdss does not support the name-hiding function. If you plan to activate the name-hiding function, protect DFSMSdss functions from all users except those required to do storage management functions. You can use RACF program control to do this. For information on program control, see *z/OS Security Server RACF Security Administrator's Guide*. In addition, you can protect certain DFSMSdss keywords by defining FACILITY class resource profiles and restricting access to those profiles. For information on using RACF FACILITY class profiles to protect DFSMSdss, see *z/OS DFSMSdfp Storage Administration*.
- DFSMShsm does not support the name-hiding function. If you plan to activate the name-hiding function, you should protect the DFSMShsm commands LIST and QUERY from all users except those required to do storage management functions. You can use profiles in the FACILITY class to protect these commands. For example, to prevent any user other than USER5 from issuing the LIST command:

```
RDEFINE FACILITY STGADMIN.ARC.LIST UACC(NONE)
PERMIT STGADMIN.ARC.LIST CLASS(FACILITY) USER(USER5) ACCESS(READ)
```

Checklist for DFSMS setup

Use the following checklist to ensure that you complete all the tasks required to set up DFSMS for multilevel security:

- Protect SMS.
- Ensure that OAM is not used for access to OAM objects. Verify that there are no object storage groups defined or enabled on the system.
- Ensure that DFSMSdss administrator functions are protected from all users except those required to do storage management functions.
- If you plan to activate the name-hiding function, create profiles in the FACILITY class to protect DFSMShsm commands that display data set names, such as LIST and QUERY.
- If you plan to activate the name-hiding function, create profiles in the FACILITY class to protect VTOCs. Add users who need to be able to read a VTOC or VTOC index to the access control list of the profile that protects that VTOC.
- Set up DFSMSrmm to force tapes to be erased when scratched
- Ensure that the DFSMSrmm parmlib option TPRACF is set to AUTOMATIC or PREDEFINED

- Ensure that the STGADMIN.EDG.LABEL.* and STGADMIN.EDG.NOLABEL.* profiles in the FACILITY class prevent the changing of labels on tape volumes.
- Ensure that the EXPDTCHECK(Y) operand is specified on each VLPOOL parmlib command to prevent expiration dates in tape labels being overridden. If you do not have a VLPOOL defined for PREFIX(*), which is the default pool, you must define it and include EXPDTCHECK(Y).
- Set up automated operations replies to the following WTORs to reject and dismount tape volumes:
  - IEC507D reply 'M' to unload
  - IEC534D reply 'M' to unload
  - IEC704A reply 'M' to unload

JES2

Where to find more information

- *z/OS JES2 Commands*
- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS JES2 Initialization and Tuning Reference*
- *z/OS JES2 Installation Exits*
- *z/OS JES2 Macros*
- *z/OS Security Server RACF Security Administrator's Guide*

JES2 provides the following support for multilevel security:

- All JES2 operator commands are auditable.
- JES2 SYSIN and SYSOUT data sets can be accessed only by the user who created the data sets. However, the user can give explicit permission to access the data set to another user. When any user attempts to access the data, whether the original creator or someone else, JES2 ensures that if the data has a security label, the accessor has a security label that dominates the data's security label.
- The security administrator can protect the JES2 system data sets by assigning them a security label of SYSHIGH. The security administrator can audit accesses to these data sets.
- The security administrator can control submission of work through a particular JES2 input device.
- The security administrator can control what data is output to a particular output device. To provide even more control, the security administrator can permit only certain owners of data to print the data at a particular output device.
- The security administrator can control which systems can run certain jobs.

JES2 user ID

The user ID associated with JES2 must have a default security label of SYSMULTI to avoid failures. This security label allows ACEEs with differing security labels to be anchored in TCBs in the JES2 address space.

JES2 commands

As with the MVS commands, the security administrator in a multilevel-secure system must be able to audit all JES2 operator commands. This audit information then is available to the security auditor for monitoring the system.

To protect each JES2 command, define RACF profiles in the OPERCMDS resource class. Each JES2 command has a standard profile name, the first qualifier of which is the name of the JES2 subsystem. The security administrator can restrict the use of the commands to certain operators by establishing an access list of users who are allowed to issue the command. RACF creates an audit record each time an operator issues the command.

See *z/OS JES2 Initialization and Tuning Guide* for a list of JES2 commands, their associated RACF profile names, and the minimum operator authority needed to enter the command. See *z/OS Security Server RACF Security Administrator's Guide* for information on creating profiles for JES2 commands.

JES2 system data sets

Some JES2 system data sets must be defined to RACF with a security label of SYSHIGH. Examples of such data sets are listed as follows. For a list of the JES2 data sets that you must protect with RACF, see *z/OS JES2 Initialization and Tuning Guide*.

- The JES2 spool data sets contain spool files managed by JES2. Some of these spool files are JES2 system or private data sets; others contain SYSIN or SYSOUT data for jobs in the system. The security administrator provides protection for the JES2 spool data sets by defining profiles in the DATASET resource class with a security label of SYSHIGH. "JES2 spool files" on page 52 discusses protection of the spool files.
- JES2 checkpoint data set

  The JES2 checkpoint data set is read and written by JES2. Define a profile for the checkpoint data set in the DATASET resource class with a security label of SYSHIGH.
- JES2 spool offload data set

  The JES2 spool offload data set contains copies of spool data. Define a RACF DATASET profile for the spool offload data set with a security label of SYSHIGH.

  In order to successfully dump and then restore the spool offload data set, activate and RACLIST the RACF RACFVARS resource class, and define an &RACLNODE statement to identify the node.

See *z/OS Security Server RACF Security Administrator's Guide* for use of the DATASET and JESSPOOL resource classes.

Note: To protect temporary data sets in a multilevel-secure system, see "Protecting temporary data sets" on page 45.

JES2 spool files

To provide protection for certain data sets on spool, the security administrator defines RACF profiles in the JESSPOOL resource class.

Note: Security labels on profiles in the JESSPOOL class are not used. The security label for a resource protected by a profile in the JESSPOOL class is extracted from the job that creates the resource, and stored in JES control blocks. The security label of the user ID of the job determines the security label of the SYSOUT data set.

- SYSIN/SYSOUT Data Sets

  A SYSIN/SYSOUT data set is given the security label of the job for which it was created. The user who created the data set is always able to access it, provided that the security label with which the user logged on dominates the security label of the data set. If a user wishes to give access to this data set to another user, a profile for the data set must be defined in the JESSPOOL resource class and other users must be permitted access to the data set. The security label of the user who is accessing the SYSIN/SYSOUT data set must dominate the security label of the SYSIN/SYSOUT data set.
- JESNEWS Data Set

  JESNEWS contains information to be printed by the JES2 print processor with the output from a job. All JES2 users should be able to access JESNEWS, but only certain users should have the authority to update it.

  The JESNEWS data set is updated by running a job that specifies a program name of JESNEWS on a SYSOUT DD statement. The program writes data into the data set. The user who updates JESNEWS must be authorized through a profile in the RACF OPERCMDS class. The security label for the JESNEWS data set is the security label of the job that creates it. The security label of this job must be SYSLOW, and it should not put any sensitive information into the JESNEWS data set.

In a multilevel-secure system, assign the JESNEWS data set a UACC of READ in the RACF JESSPOOL resource class.

Each access of JESNEWS can be audited, depending upon the audit options that were set.

- SYSLOG

  The SYSLOG data set contains job- and operation-related data. Define a profile for SYSLOG in the JESSPOOL resource class. The SYSLOG data set inherits a security label of SYSHIGH from the master address space.

- Special JES2 Spool Data Sets

  JES2 creates several data sets for a job, and assigns the security label of the job to them. They are printed as part of a job's output. The system reserves names for these data sets so that a RACF profile can be defined for each of them. The format of the profile name is the same as for a SYSOUT data set. The last qualifier of the profile name is one of the following values: JESYSMSG, JESJCL, JESJCLIN, and JESMSGLG. These values are reserved for both JES2 and JES3 for consistency and cannot be specified using DSN= on a DD statement.

JES2 input devices

Commands and jobs are input to JES2 through a JES2 input device. To control the submission of work through these devices, define a RACF profile in the JESINPUT resource class for each device. In the profile, list the users allowed to issue commands or submit batch jobs from this device.

If the RACF JESINPUT resource class is active and you do not create a profile for the input device, RACF rejects any work from this device.

See *z/OS JES2 Initialization and Tuning Guide* for the profile names to associate with the input device. See *z/OS Security Server RACF Security Administrator's Guide* for more information about protecting JES input devices.

JES2 output processing

With the JESSPOOL resource class active in a multilevel-secure system, RACF can produce an SMF type 80 audit record when a data set is printed. Depending on the audit options specified, the audit record contains the name of the device on which the data set was printed.

Optionally, the security administrator can control which output device a user can access to print a data set. The security administrator can define a profile in the WRITER resource class to protect a printer and then permit a user to access that printer. The data set is printed only if the security label of the printer (in the WRITER profile) dominates the security label of the data. The owner of the data also must be permitted to the WRITER profile.

See *z/OS Security Server RACF Security Administrator's Guide* for information on using the WRITER resource class.

Controlling which systems can run certain jobs

The security administrator can define a security label to be active only on certain members of a sysplex. Using security labels on a per-system basis allows the installation to separate work based on security classification while still sharing the RACF database. The security administrator activates the use of system-specific security labels by activating the SETROPTS SECLBYSYSTEM option. When SECLBYSYSTEM is active, JES2 insures that no job is run on a member that does not have an appropriate security label active. If no system is available on which a job's security label is active, the job remains in the conversion phase. For a description of the conversion phase, see *z/OS JES2 Introduction*.

Restrictions:

- JES2 does not support using system-specific security labels for systems that perform NJE and OFFLOAD processing. These systems must have all security labels active.
- JES2 printers are not able to process output unless the security label associated with the output is active on the system controlling the printer.

NJE and RJE

You can use network job entry (NJE) and remote job entry (RJE) in a multilevel-secure system if you configure them correctly. Security labels assigned to jobs arriving via RJE are restricted by the security label assigned to the RJE reader's JESINPUT profile. The default security label for jobs read in on an RJE reader is obtained from the RMT*xxxx* user ID assigned to the remote device. SYSOUT sent on RJE devices is controlled using WRITER class profiles in the same way as on a local printer. Jobs and SYSOUT arriving via NJE are also restricted by the security label assigned to the adjacent node profile in the JESINPUT class. The default security label assigned to the job is determined by the security label sent by the originating node as interpreted by the NODES class profiles. WRITER class profiles also control what jobs and SYSOUT can be sent to other nodes. For NJE and RJE, a JOB or SYSOUT that has a security label that is not active on the system that receives the JOB or SYSOUT fails job validation, and the system purges the job or SYSOUT. For more information on NJE and RJE security considerations, see *z/OS JES2 Initialization and Tuning Guide*.

Protect NJE and RJE input resources with profiles in the JESINPUT class, assigning a security label to each of them. Protect NJE and RJE printers with profiles in the WRITER class, assigning a security label to each. JES transmits work to an NJE or RJE printer only if the printer's security label dominates the work's security label.

JES2 restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Network job entry (NJE) does not support multilevel security if there are any NJE links to devices or systems that have more than one security label.
- Remote job entry (RJE) does not support multilevel security if there are any RJE links to devices or systems that have more than one security label.
- JES2 does not support using system-specific security labels for systems that perform NJE and OFFLOAD processing. These systems must have all security labels active.
- Only PSF-managed printers provide for separator pages and security labeling. Therefore, ensure that this type of printer processes all secure output from the system. Printers that do not have the capability of providing separator pages and security labeling can be used in a multilevel-secure system, but only for non-secure output.
- Remove all installation routines that you have written and added to your current JES2 library.
- Remove any modifications to JES2 source code.
- Do not enter system level commands through the input job stream. Only READ authority commands can be issued. This restriction is controlled by specifying AUTH=(DEVICE=NO,JOB=NO,SYSTEM=NO) on the RDR*nnn* and INTRDR initialization statements.

JES2 setup checklist

Use the following checklist to ensure that you complete all the tasks required to set up JES2 for multilevel security:

- Assign a default security label of SYSMULTI to the user ID associated with JES2.
- Protect and audit all JES2 commands. Define RACF profiles in the OPERCMDS resource class to protect all JES2 commands. Specify which operators are authorized to issue which commands by adding authorized operators to the access control lists for the commands that they are authorized to use.
- Assign the JESNEWS data set a UACC of READ in the RACF JESSPOOL resource class.
- Assign the job that creates the JESNEWS data set a security label of SYSLOW.
- Protect the SYSLOG data set by defining a profile in the RACF JESSPOOL resource class.
- Protect JES2 system data sets by defining profiles in the RACF DATASET or TAPEVOL resource classes.
- Ensure that all NJE and RJE input sources are protected by profiles in the RACF JESINPUT class, and assign a security label to each profile.

- Ensure that all NJE and RJE printers are protected by profiles in the RACF WRITER class, and assign a security label to each profile.
- Ensure that only PSF-managed printers process secure output.
- Remove any modifications that you have made to JES2 source code.
- Remove any installation routines that you have written and added to your current JES2 library.
- Specify AUTH=(DEVICE=NO,JOB=NO,SYSTEM=NO) on the RDR*nnn* and INITRDR initialization statements to prevent system level commands from being entered through the input job stream.
- If you want to restrict jobs to running on certain systems based on security classification, set up system-specific security labels.

MVS

Where to find more information

- *PSF for z/OS: Security Guide*
- *z/OS MVS Initialization and Tuning Guide*
- *z/OS MVS Initialization and Tuning Reference*
- *z/OS MVS Installation Exits*
- *z/OS MVS JCL Reference*
- *z/OS MVS System Commands*
- *z/OS Security Server RACF Security Administrator's Guide*

MVS supports multilevel security by providing the following support:

- All console operators must LOGON before issuing any commands.
- All operator commands are auditable.
- All accesses to named protected objects from operator commands are audited.
- Only users defined to RACF are allowed to access MVS.
- The use of terminals, printers, and other unit record equipment is controlled through RACF.
- The security administrator can restrict the use of particular commands to a particular operator at a specific console

Note: The Hardware Management Console (HMC) and support element console both allow entry of z/OS operator commands, but neither supports the MVS LOGON command. Therefore there is no operator accountability when an operator uses these consoles.

Guidelines: You must take extra care to protect these consoles:

- Use physical security (for example, place them in a locked room)
- Limit distribution of passwords for these consoles
- Use these consoles for z/OS operation only in an emergency

For more information about the Hardware Management Console and the support element, see *S/390® Hardware Management Console Operations Guide*, GC38-0470.

Establish operator LOGON/LOGOFF

The system must be able to audit the commands that operators enter. To meet this auditing requirement, all operators must log on to the system. The LOGON command establishes the security environment for the console operator.

You must specify that a LOGON is required in the CONSOLxx member of the SYS1.PARMLIB data set. For more information, see the description of CONSOLxx in *z/OS MVS Initialization and Tuning Reference*.

To delete the security environment, the operator enters the LOGOFF command. This action leaves the console in a secure state; the system does not accept commands from this console until another operator logs on by entering the LOGON command.

Note: When you IPL a multilevel-secure system, multilevel security is not in effect until RACF is active. Until that time, only the master console can issue commands. Commands that are issued from secondary consoles are rejected, with one exception:

- You might need to allow for operator intervention during IPL before RACF is fully initialized, for example, if you need to establish a master console to complete RACF initialization. If so, an operator can establish a master console by entering the following command from any secondary console:

```
VARY CN(name),AUTH=MASTER
```

After RACF completes initialization, the operators must log on before they can enter commands successfully. See *z/OS MVS System Commands* for a description of the LOGON command.

To identify the console operator to RACF, create a RACF user profile for each operator. Assign a default security label of SYSHIGH to each console operator profile and permit the operator to the SYSHIGH security label in the SECLABEL resource class. To ensure that operators log on, set the LOGON parameter on the DEFAULT statement in the CONSOLxx member of SYS1.PARMLIB to REQUIRED. When an operator logs on to the system and is identified to RACF, the operator's security information becomes available for authorizing any subsequent commands that the operator enters.

Set up RACF control for each MCS, EMCS, and SMCS console that an operator can use.

- To define an MCS or SMCS console to RACF, create a profile in the RACF CONSOLE resource class, specifying the SYSHIGH security label and UACC(NONE). To authorize an operator to log on to a particular console, use the PERMIT command to give the operator (or operator group) READ authority to the profile that protects the console.
- To control use of an EMCS console, create a profile in the OPERCMDS class for the console, with the form MVS.MCSOPER.*console-name*, and assign a security label of SYSHIGH to that OPERCMDS profile.

  Example:

  ```
  RDEFINE OPERCMDS MVS.MCSOPER.console-name UACC(NONE) SECLABEL(SYSHIGH)
  ```

  Then, use the PERMIT command to grant the appropriate users READ access to the OPERCMDS profile, which allows them to establish an EMCS console with that name. Only users with the SYSHIGH label can establish an EMCS console. Create a profile of MVS.MCSOPER.** to prevent the use of other EMCS console names that you do not define.

  Example:

  ```
  RDEFINE OPERCMDS MVS.MCSOPER.** UACC(NONE) SECLABEL(SYSHIGH)
  ```

For information about how to create a user profile, see *z/OS Security Server RACF Security Administrator's Guide*.

For a description of the CONSOLxx member of SYS1.PARMLIB, see *z/OS MVS Initialization and Tuning Reference*.

Audit operator commands

In a multilevel-secure system the security administrator and auditor must be able to audit all operator commands. These commands include not only MVS commands, but also commands related to specific elements, features, or subsystems (such as JES2 or JES3). The audit information enables the security auditor to monitor the text of the command issued, who issued the command, from which MCS console, when, and whether the operator had the authority to issue the command. (In the last instance, the audit record indicates that the operator was not authorized to issue the command. The system does not permit the command to be processed.)

To define the auditing to be done for commands, create RACF profiles in the OPERCMDS resource class. Each MVS command has a standard RACF resource name, the first qualifier of which is 'MVS'. You can restrict the use of the command to certain operators through the access list in the command profile.

Note: The 'K' command, when entered with no operands from an MCS console, is not audited. It is interpreted as a 'K E,1' command, which only erases the display area of the console screen.

See *z/OS MVS Planning: Operations* for information about securing access to system commands, including a list of the MVS commands that can be audited, their associated RACF resource names, and the suggested operator authority needed to enter the command.

See *z/OS Security Server RACF Security Administrator's Guide* for information on creating profiles for MVS commands.

Program properties table

In a multilevel-secure system, all accesses to resources that are protected must be monitored. Therefore, in a multilevel-secure environment, a program that accesses a protected resource is not allowed to bypass any type of access control. The program properties table (PPT) contains entries for special attributes of programs. One of these entries indicates whether or not the program is allowed to bypass password protection. IBM supplies in SYS1.SAMPLIB a sample SCHED00 that contains PPT statements. Each entry in the PPT supplied by IBM has the 'PASS' option specified or defaulted to, meaning that the program is not allowed to bypass password protection. If there have been modifications to your PPT, check the options specified in the PPT statement in the SCHED*xx* member of SYS1.PARMLIB. For each program name parameter listed, you must specify or default to the PASS option. Do *not* specify the NOPASS option.

See *z/OS MVS Initialization and Tuning Reference* for a description of the SCHED*xx* member of SYS1.PARMLIB.

Establish SMF controls

In a multilevel-secure system, system management facilities (SMF) log records are used to audit security-relevant events. For information on auditing, see Chapter 4, "Auditing a multilevel-secure system," on page 99. SMF records can be written to SMF data sets, or log streams, or both.

Using SMF data sets

When you use SMF data sets, SMF maintains SMF records in buffers until they are written to DASD. In a multilevel-secure system, you should ensure that SMF records are not lost when no buffers are available to SMF or when the last allocated SMF data set is full. Specify the following parameters in the SMFPRM*xx* member of SYS1.PARMLIB:

- NOBUFFS(HALT) — If SMF runs out of space for buffers in its address space, halt processing.
- LASTDS(HALT) — If the last allocated SMF data set is full, halt processing.

These parameters specify that before either condition occurs, warning notices are issued to the console operator. The warning notices are:

- IEE986E SMF HAS USED *nn*% OF AVAILABLE BUFFER SPACE
- IEE985A SMF IS PROCESSING THE LAST AVAILABLE DATA SET

For a complete explanation of these messages, see *z/OS MVS System Messages, Vol 7 (IEB-IEE)*.

If the system issues one of these messages, the operator has the option of either slowing down the system so that not as many SMF records are being maintained in the buffers or emptying an SMF data set so that the DASD is available to the system.

If either condition does occur, the system is put into a restartable wait state. The wait state codes are:

- D0D-00 No SMF Buffer Space Available
- D0D-01 No SMF Data Sets Available

The data in the SMF buffers is lost if the system runs out of SMF buffer space. However, if the wait state was caused by the unavailability of SMF data sets, it is possible to recover SMF records from the buffers that were not written to DASD before the system failure.

See Chapter 5, "Operating a system," on page 105 for information about using the IPCS **SMFDATA** subcommand to recover SMF records from buffers that were not written to the SMF data set before a system failure.

See *z/OS MVS Initialization and Tuning Reference* for a description of the SMFPRM*xx* member of SYS1.PARMLIB.

Using SMF logging

If you use SMF logging, SMF writes records to log streams that are managed by the system logger. Operators do not need to switch SMF data sets, nor dump them to archive storage, nor clear them.

Note: If you use SMF logging, SMF does not honor the NOBUFFS(HALT) and LASTDS(HALT) configuration parameters. If you require the SMF data loss prevention provided by these parameters, you should configure SMF to use SMF data sets, not log streams.

Protect resources

You must identify to RACF not only the users of the multilevel-secure system but also the resources that you want RACF to protect.

Unit record, communication, and graphic devices

In a multilevel-secure system, only programs that are part of the trusted computing base can allocate secure unit record, communication, and graphic devices. This ensures that hardcopy output contains the required security labeling and that terminal and graphic device users are identified and authenticated.

The programs that are part of the trusted computing base that are trusted to allocate secure devices are:

- VTAM for communication and graphic devices (channel-to-channel adapters and terminals)
- PSF for unit record devices (printers)
- JES for printers and communication services

In a multilevel-secure environment, activate the DEVICES resource class and allow only those programs that are part of the trusted computing base to allocate secure unit record, communication, and graphic devices.

See *z/OS Security Server RACF Security Administrator's Guide* for information about controlling the allocation of devices to define the unit record, communication, and graphic devices to RACF and to permit users to their access lists.

LLA parmlib data sets and LLA-managed data sets

Library lookaside (LLA) is an MVS service that maintains a copy of the PDS directory entries for an installation-specified group of libraries in its address space virtual storage. This group of libraries includes the set of LNKLST data sets and the data sets that are specified in CSVLLAxx parmlib members. LLA uses virtual lookaside facility (VLF) to keep the most active LLA-managed modules in a data space to avoid program fetch I/O.

If you want your multilevel-secure system to use LLA, define RACF profiles in the DATASET resource class for the LLA parmlib data sets. These data sets are the parmlib data sets that contain CSVLLAxx members that specify which libraries LLA is to manage and how it is to manage them. Permit operators to access the data sets with READ authorization.

Also, define RACF profiles for your LLA-managed data sets. These data sets are the libraries that are specified in the CSVLLAxx and LNKLSTxx members of parmlib, and by LNKLST statements in the PROGxx members of parmlib. For the **MODIFY LLA,UPDATE,LIBRARY=** command, the data set profiles must exist in the FACILITY class; otherwise, the data set profiles can exist in either the FACILITY or the

DATASET class. In both cases, UPDATE access is required so that the operator can revise the current version of the list of LLA-managed data sets.

Except for the **MODIFY LLA,UPDATE,LIBRARY=** command, LLA checks the FACILITY resource class before it checks the DATASET resource class. If the FACILITY class profile exists and the operator is defined in the access list with update access, or, if no FACILITY class profile exists for the data set, access is granted and LLA does not check the DATASET class. If a profile exists in the FACILITY class, but the operator is not defined in the access list, RACF checks the DATASET class. Access is granted through the DATASET resource class if the operator is defined in the access list, or if no profile exists at all for the data set.

For the **MODIFY LLA,UPDATE,LIBRARY=** command, LLA checks the FACILITY resource class only. Access is granted if a matching profile exists and the profile grants at least UPDATE access.

Observe the following considerations for defining RACF profiles for LLA-managed data sets:

- If you use the FACILITY class, you can define a generic profile that can cover all LLA-managed data sets. Permit the appropriate operators (those who are allowed to revise the LNKLST and other LLA-managed data sets) to the profile with at least UPDATE access.
- If you use the DATASET class, you must define a profile for each LLA-managed data set. Because the check in the DATASET class is done only when a FACILITY class profile exists (and it denies the operator access), you must also have such a FACILITY class profile in order to use the DATASET class for this purpose. If you want to add a data set to the list of LLA-managed data sets, a profile for that new data set must exist.

If access is denied, LLA issues message CSV244I, which provides information about the class and resource that was checked. Prior to message CSV244I, you might also receive RACF message ICH408I. For cases in which LLA checks both the FACILITY and DATASET classes for the resource, message ICH408I might indicate the DATASET class. Regardless, observe the following guideline for using the FACILITY class to control LLA access.

Guideline: Use a generic profile and the FACILITY class for the LLA-managed data sets in your installation. This method allows the operator to perform the `LLA START` and `LLA MODIFY` commands without having to have access to each data set.

For a description of the CSVLLAxx member of SYS1.PARMLIB, see *z/OS MVS Initialization and Tuning Reference*.

For a description of how to create RACF profiles for LLA parmlib and LLA-managed data sets, see *z/OS Security Server RACF Security Administrator's Guide*.

System data sets

Define system data sets to RACF with either a discrete or a generic profile in the DATASET resource class. To determine the security label to assign to a system data set in a multilevel-secure system, consider the data that the data set contains.

- Data that can be accessed by all users should have a security label of SYSLOW (or an installation-defined security label) and a UACC of READ. Data sets such as SYS1.LINKLIB, SYS1.IMAGELIB, and SYS1.PROCLIB are in this category.
- Data that needs to be accessed by only certain users should have a security label of SYSLOW (or an installation-defined security label) and a UACC of NONE. Users must be permitted specifically to access the data set. The access authority (for example, to READ or to UPDATE) can be set for each individual user allowed to access the data set. Examples of this type of data set are SYS1.PARMLIB and SYS1.VTAMLST.

Assign the SYSHIGH security label to data sets that contain multiple levels of data. To further protect these data sets from unauthorized access, specify a UACC of NONE and permit only certain users to access the data set. In a multilevel-secure system, examples of data sets that should be assigned a security label of SYSHIGH with a UACC of NONE are:

- Log data sets

- SYS1.dump data sets (user dump data sets should have the security label of the user)
- Trace data sets
- SMF data sets
- Page and swap data sets
- Spool data sets
- Dump analysis and elimination (DAE) data sets
- Spool offload and dump job data sets
- JES checkpoint data sets
- PSF security libraries (overlay, font, page segment, security definitions)
- XCF couple data sets
- SMS configuration data sets (CDS)

In a multilevel-secure system, assign all catalogs a security label of SYSNONE.

See *z/OS Security Server RACF Security Administrator's Guide* for a description of how to protect system data sets and a list of system data sets to protect in a multilevel-secure system.

APF-authorized libraries

The authorized program facility (APF) helps your installation protect the system. APF-authorized programs can access system functions that can affect the security and integrity of the system. The APF list identifies libraries that contain APF-authorized programs.

Your installation can create and maintain the APF list in a dynamic or static format. If the APF list format is dynamic, the system administrator or the installation can update the APF list at any time during normal processing or at IPL, and enter as many libraries in the APF list as storage limitations allow. For more information about APF, see *z/OS MVS Programming: Authorized Assembler Services Guide*.

To protect the APF list, set up RACF FACILITY resource class profiles that protect the following resources:

- CSVAPF.*libname*
- CSVAPF.MVS.SETPROG.FORMAT.STATIC
- CSVAPF.MVS.SETPROG.FORMAT.DYNAMIC
- Grant UPDATE authority to CSVAPF.*libname* (truncated to a total length of 39 characters if the length of *libname* exceeds 32 characters) to users who are allowed to add the specified library to, or delete it from, the APF list.
- Grant UPDATE authority to CSVAPF.MVS.SETPROG.FORMAT.DYNAMIC to users who are allowed to change the format of the APF list to dynamic.
- Grant UPDATE authority to CSVAPF.MVS.SETPROG.FORMAT.STATIC to users who are allowed to change the format of the APF list back to static.

If you authorize users to update the APF list using some other method (such as the SETPROG operator command), you must ensure that there is no FACILITY class profile that matches a profile previously listed. If there is such a profile, the system uses it to determine if the requestor is authorized.

Dynamic exits facility

The exit services tables contain lists of exits with associated exit routines. To protect the exit services tables, set up RACF FACILITY resource class profiles that protect the following entities:

- CSVDYNEX.*exitname*.DEFINE
- CSVDYNEX.*exitname.modname*
- CSVDYNEX.*exitname*.UNDEFINE
- CSVDYNEX.*exitname*.ATTRIB
- CSVDYNEX.LIST

- CSVDYNEX.*exitname*.CALL
- CSVDYNEX.*exitname*.RECOVER

For more information about protecting the dynamic exits facility, see *z/OS MVS Installation Exits*.

Protect global resource serialization services

It is possible for the global resource serialization ENQ and GQSCAN services and the corresponding 64-bit services ISGENQ and ISGQUERY to be used as covert communication mechanisms to declassify data. These services can be issued by unauthorized callers. Both ENQ and ISGENQ take character data as input in serializing abstract resources. The GQSCAN and ISGQUERY macros enable programs to scan for resource requests across the global resource serialization complex. Therefore ENQ and ISGENQ are potential transmit mechanisms where GQSCAN and ISGQUERY would be used for receiving, and the abstract resource names would be the data.

To protect these services, create a profile in the FACILITY class whose name is `ISG.QSCANSERVICES.AUTHORIZATION`. When the MLACTIVE option is active and an unauthorized program issues a GQSCAN or ISGQUERY ReqInfo=SCAN, the request fails if the user running the program does not have READ access to the profile. The request also fails if the in-storage profiles for the FACILITY class are not available, so you must RACLIST the FACILITY class before you activate the MLACTIVE option.

The DISPLAY GRS system command can internally issue a GQSCAN. Because this command runs authorized, global resource serialization processing does not check the FACILITY class profile for authorization to issue this GQSCAN. The installation must protect the DISPLAY GRS operator command and the consoles from which it can be issued. For information about protecting operator commands and consoles, see *z/OS Security Server RACF Security Administrator's Guide*.

The global resource serialization ENQ/RESERVE/DEQ monitor runs authorized as either a batch job or a started task. If you do not protect it, any unauthorized user can submit the job to start the monitor, and gather ENQ and DEQ data. To prevent unauthorized users from doing this, use the RACF PROGRAM class to protect the program ISGAUDIT in the library SYS1.LINKLIB. For information about using RACF program control and the PROGRAM class, see Protecting programs in *z/OS Security Server RACF Security Administrator's Guide*.

For more information on global resource serialization, see *z/OS MVS Planning: Global Resource Serialization*.

Check job control language (JCL)

The JOB OUTPUT and DD job control statements have keywords that are security-related. To ensure that your existing job control statements are multilevel-secure, determine if you need to use these keywords.

- JOB statement

  All jobs must be submitted by a RACF-defined user or be associated with a RACF-defined user. Use the SECLABEL keyword to specify a security label at which the job executes. If SECLABEL is not specified, the job inherits the security label of the submitting user.

  If a job is submitted through TSO/E, then a user ID is propagated by JES.

- OUTPUT statement

  The security administrator can authorize a user to specify the DPAGELBL keyword on the OUTPUT statement. The DPAGELBL keyword indicates whether the system should print the notation associated with the security label on each page of printed output.

  The security administrator can authorize a user to specify the SYSAREA keyword on the OUTPUT statement. The SYSAREA keyword indicates whether or not the system should reserve an area on each page of printed output for the security label.

  Note: The DPAGELBL and SYSAREA keywords are valid also on the TSO OUTDES command. For more information about these keywords, see "Authorize users allowed to override print labeling" on page 66.

- DD statement

  Use the DSNAME keyword to assign the fifth qualifier in a JES SYSIN/SYSOUT data set. If you do not specify a name, the system assigns a '?' as the last qualifier of the name of the data set. By defining a specific name, you can give access to this data set to other users who have the appropriate security label and who have access via profiles in the RACF JESSPOOL class.

See *z/OS MVS JCL Reference* for a description of the JOB, OUTPUT and DD statements.

See *PSF for z/OS: Security Guide* for additional information about specifying print options in a multilevel-secure system.

See *z/OS Security Server RACF Security Administrator's Guide* for information about how to assign a user a default security label.

MVS supplied exit routines

The following default installation exits that are shipped with MVS can be used on a multilevel-secure system without compromising security:

- Allocated/Offline Device Installation Exit
- Specific Waits Installation Exit
- Volume ENQ Installation Exit
- Volume Mount Exit
- ASREXIT—SYMREC Authorization Exit
- IEALIMIT—Limiting User Region Size
- IEAVTSEL—Post Dump Exit Name List
- IEFDOIXT—Edit/Check A Caller's Text Units
- ISGGREX0—Scanning the ENQ/DEQ/RESERVE Resource Name Lists

For a description of these exits, see *z/OS MVS Installation Exits*.

Protect sensitive privileges in IPCS

The interactive program control system (IPCS) has two resources in the FACILITY class that should be granted only in extraordinary circumstances and to very trusted users:

- BLSACTV.ADDRSPAC protects the ability to examine sensitive storage.
- BLSACTV.SYSTEM protects the ability to examine storage in other address spaces.

These resources should have UACC(NONE), and a security label of SYSHIGH.

MVS restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Remove any installation-written exit routines or modifications that you have added to your current system.
- Do not allow an operator to place the system console in problem determination mode.
- Do not use APPC/MVS. It does not ensure that both ends of the conversation have the same security label.
  - Use RACF program control to disable the APPC/MVS programs ATBINMIG, ATBSDEPE, ATBSDFMU, ATBSDFCS, and ATBSDFM1. Do not add any users to the access lists for the PROGRAM class profiles protecting these programs. For example, if RACF program control is already active on your system, you could issue the following commands:

```
RDEFINE PROGRAM ATB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKLIB') UACC(NONE)
RDEFINE PROGRAM ASB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKLIB') UACC(NONE)
SETROPTS WHEN(PROGRAM) REFRESH
```

  - Do not start the APPC or ASCH address spaces.

Checklist for MVS setup

Use the following checklist to ensure that you complete all the tasks required to set up MVS for multilevel security:

- Set the DEFAULT statement in the CONSOL*xx* member of SYS1.PARMLIB to LOGON(REQUIRED), to specify that operators must log on.
- Create a RACF user profile for each console operator. Assign a default security label of SYSHIGH to each console operator, and permit each operator to the SYSHIGH security label in the SECLABEL class.
- Create a profile in the CONSOLE RACF resource class for each MCS and SMCS console, specifying the SYSHIGH security label and UACC(NONE). Update the access control lists for the profiles to control which operators can log on to particular consoles.
- Create a profile in the OPERCMDS class for each EMCS console, of the form MVS.MCSOPER.*console-name*, and assign the profile a security label of SYSHIGH.
- Create a profile in the OPERCMDS class of the form MVS.MCSOPER.** to prevent the use of EMCS console names that you have not defined:

```
RDEFINE OPERCMDS MVS.MCSOPER.** UACC(NONE) SECLABEL(SYSHIGH)
```

- Create RACF profiles in the OPERCMDS resource class for MVS operator commands, and update the access control lists to identify users authorized to issue the commands.
- Update the program properties table (PPT) to specify the PASS option for each entry.
- Specify NOBUFFS(HALT) and LASTDS(HALT) in the SFMPRM*xx* member of SYS1.PARMLIB.
- Create RACF profiles in the DEVICES resource class to allow only programs in the trusted computing base to allocate unit record, communication, and graphic devices, and activate the DEVICES class.
- If you are using LLA, create a generic profile in the FACILITY class to protect all LLA-managed data sets. Give operators allowed to revise the LNKLST and other LLA-managed data sets at least UPDATE access to the profile.
- Create RACF profiles in the DATASET resource class to protect system data sets that can be accessed by all users. Specify a security label of SYSLOW and a UACC of READ. These data sets include:
  - SYS1.LINKLIB
  - SYS1.IMAGELIB
  - SYS1.PROCLIB
- Create RACF profiles in the DATASET resource class to protect system data sets that only certain users need to access. Specify a security label of SYSLOW and a UACC of NONE. Update the access control lists to give users who need access the appropriate authority. Theses data sets include:
  - SYS1.PARMLIB
  - SYS1.VTAMLST
- Create RACF profiles in the DATASET resource class to protect system data sets that contain multiple levels of data. Specify a security label of SYSHIGH and a UACC of NONE. Update the access control lists to give users who need access the appropriate authority. Theses data sets include:
  - Log data sets
  - SYS1.dump data sets (user dump data sets should have the security label of the user)
  - Trace data sets
  - SMF data sets
  - Page and swap data sets
  - Spool data sets
  - Dump analysis and elimination (DAE) data sets

  - Spool offload and dump job data sets
  - JES checkpoint data sets
  - PSF security libraries (overlay, font, page segment, security definitions)
  - XCF couple data sets
  - SMS configuration data sets (CDS)
- Assign all catalogs a security label of SYSNONE.
- Protect APF-authorized libraries. Ensure that you have profiles in the RACF FACILITY class protecting the following resources:
  - CSVAPF.*libname*
  - CSVAPF.MVS.SETPROG.FORMAT.STATIC
  - CSVAPF.MVS.SETPROG.FORMAT.DYNAMIC
- Protect the dynamic exits facility. Ensure that you have profiles in the RACF FACILITY class protecting the following resources:
  - CSVDYNEX.*exitname*.DEFINE
  - CSVDYNEX.*exitname.modname*
  - CSVDYNEX.*exitname*.UNDEFINE
  - CSVDYNEX.*exitname*.ATTRIB
  - CSVDYNEX.LIST
  - CSVDYNEX.*exitname*.CALL
  - CSVDYNEX.*exitname*.RECOVER
- Protect global resource serialization services:
  - Create a profile in the FACILITY class to protect GQSCAN and ISGQUERY:

    ```
    RDEFINE FACILITY ISG.QSCANSERVICES.AUTHORIZATION UACC(NONE)
    ```

    If any unauthorized callers need to issue the protected requests, give them READ access to the profile. If the FACILITY class is active and RACLISTed, refresh the in-storage profiles:

    ```
    SETROPTS RACLIST(FACILITY) REFRESH
    ```

    If the FACILITY class is not active or RACLISTed, make sure that you activate and RACLIST it before you activate the MLACTIVE option.
  - Protect the ENQ/RESERVE/DEQ monitor by using the RACF PROGRAM class to protect the program ISGAUDIT in the library SYS1.LINKLIB.
- Check job control language (JCL)
  - Ensure that all JOB statements specify a user ID.
  - Add the SECLABEL keyword to JOB statements to specify the security label at which the job executes. If the SECLABEL keyword is not specified, the job uses the user's default security label.
- Remove any installation-written exit routines or modifications that you have added to your system.
- Create profiles to protect the BLSACTV.ADDRSPAC and BLSACTV.SYSTEM resources in the FACILITY class, specifying UACC(NONE) and SECLABEL(SYSHIGH). Ensure that only highly trusted users are on the access list.
- Do not allow an operator to place the system console in problem determination mode. Use the RACF OPERCMDS resource class to disable the VARY CN command with the ACTIVATE option.
- Use RACF program control to disable APPC/MVS programs.

  ```
  RDEFINE PROGRAM ATB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKLIB') UACC(NONE)
  RDEFINE PROGRAM ASB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKLIB') UACC(NONE)
  SETROPTS WHEN(PROGRAM) REFRESH
  ```

PSF

Print Services Facility (PSF) for z/OS provides the capability to print security information on all hardcopy output and to audit any attempt by a user to override this security labeling.

The security information, or *identification label*, that PSF prints on each output page corresponds to the security label that is associated with the data to be printed. (The security label is specified on the JOB statement, or specified at LOGON, or inherited from a session when a job is submitted. For more information, see "Check job control language (JCL)" on page 61.) The identification label is composed of text, graphics, or a combination of text and graphics. You define and maintain these identification labels for your installation. You also specify where on a page the identification label is to be positioned by using a security overlay. The labels are placed outside the area on which the user is allowed to print so that the user cannot subvert the labels.

Where to find more information

- *Overlay Generation Language/370: User's Guide and Reference*
- *PSF for z/OS: Customization*
- *PSF for z/OS: Security Guide*
- *z/OS MVS JCL Reference*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS TSO/E Command Reference*

PSF provides the following support for multilevel security:

- Identification labels are printed on each page of hardcopy output. The security administrator can authorize users to override this labeling requirement, but PSF generates an audit record for the override.
- PSF can produce job separator pages that contain the identification label of the job for all hardcopy output. Users cannot override this requirement, but the security administrator can authorize operators to specify that separator pages are not to be produced.
- The header and trailer pages for the hardcopy output of each job contain a matching random number generated by PSF. The operator can use these numbers to ensure that output from jobs is properly separated.

Note: Printers used exclusively by the system, such as the MVS hardcopy console, that contain only system information and that are physically protected, do not need to have security labels on each page.

Setting up PSF print labeling

About this task

Before you begin you must know the names of the security labels for your installation.

This section lists the subtasks that you must complete to set up the print labeling function of PSF in a multilevel-secure environment, and provides a high-level description of each subtask. For detailed information on completing these tasks, see *PSF for z/OS: Security Guide*.

Subtask	See . . .
Create the security libraries	"Create the security libraries" on page 66
Install PSF exit routines for the security separator pages	"Install PSF exit routines for separator pages" on page 66
Modify the PSF startup procedures	"Modify the PSF startup procedure" on page 66
Authorize users allowed to override print labeling	"Authorize users allowed to override print labeling" on page 66

Subtask	See . . .
Authorize operators allowed to override separator pages	"Authorize operators allowed to override separator pages" on page 67
Enable guaranteed print labeling	"Enable guaranteed print labeling" on page 67

Create the security libraries

For print labeling, PSF requires four security libraries that contain the printer resources and the definitions of your identification labels:

- The *security overlay library* contains an overlay for each paper size used in applications that require print labeling. The security administrator can create this library using the Overlay Generation Language/370 program (OGL/370). For more information about OGL/370 see *Overlay Generation Language/370: User's Guide and Reference*.
- The *security font library* contains the fonts required for printing the identification labels.
- The *security page segment library* contains the page segments that might be required for printing the identification labels.
- The *security definitions library* contains a member for each security label in your system. The member name is the security label. For each member, the security administrator defines the identification label to be printed by specifying the name of the corresponding security overlay file member and the physical characteristics of the page on which the overlay prints.

See *PSF for z/OS: Security Guide* for information about how to define these libraries. Define these libraries to RACF with a security label of SYSLOW.

Install PSF exit routines for separator pages

As a method of ensuring that the pages separating job output are not deliberately tampered with in an attempt to violate security, PSF generates a random number and places that number on both the header separator page and the corresponding trailer separator page. The printer operator, who is responsible in a multilevel-secure system for separating and distributing printer output, must ensure that these numbers match. If the numbers do not match, the operator must continue searching through the output until the correct trailer separator page is found.

PSF provides two exit routines in SYS1.SAMPLIB for printing random numbers on job separator pages. These routines are APSUX01S for security job headers and APSUX02S for security job trailers. To use these routines, replace the PSF-supplied default job header and trailer routines, APSUX01 and APSUX02.

See *PSF for z/OS: Security Guide* for information on replacing the separator pages installation exit routines.

Modify the PSF startup procedure

The PSF startup procedure specifies PSF initialization parameters and libraries that tell PSF where security resources and security definitions are located. PSF supplies a sample startup procedure in SYS1.SAMPLIB. Modify this startup procedure for each functional subsystem application (FSA) to reference the security libraries and to define print labeling.

The PRINTDEV statement in the PSF startup procedure specifies the security libraries for an FSA and indicates whether separator page labeling, data page labeling, and a system defined user-printable area are in effect.

See *PSF for z/OS: Security Guide* and *PSF for z/OS: Customization* for more information about the PRINTDEV statement.

Authorize users allowed to override print labeling

The security administrator can define profiles in the RACF resource class PSFMPL to protect data page labeling (PSF.DPAGELBL) and the user printable area (PSF.SYSAREA). To authorize a user to override either of these print labeling requirements, give the user READ access to the appropriate profile in the

PSFMPL class and activate the class. An authorized user can override data page labeling by specifying DPAGELBL=NO on the OUTPUT JCL statement or the OUTDES TSO/E command, and can override the user printable area by specifying SYSAREA=NO on the OUTPUT JCL statement or the OUTDES TSO/E command.

Example: To allow USER1 to override data page labeling, and USER2 to override the user printable area:

```
RDEFINE PSFMPL PSF.DPAGELBL UACC(NONE)
RDEFINE PSFMPL PSF.SYSAREA  UACC(NONE)
PERMIT PSF.DPAGELBL CLASS(PSFMPL) ID(USER1) ACCESS(READ)
PERMIT PSF.SYSAREA  CLASS(PSFMPL) ID(USER2) ACCESS(READ)
SETROPTS CLASSACT(PSFMPL) RACLIST(PSFMPL)
```

See *PSF for z/OS: Security Guide* and *z/OS Security Server RACF Security Administrator's Guide* for information about using the PSFMPL resource class.

Authorize operators allowed to override separator pages

An operator can specify that separator pages are not to be produced by entering JES commands from the console.

Example: For JES2: $T PRT(*nnnn*),SEP=NO

To authorize an operator to use the commands, the security administrator must give the operator UPDATE access to the appropriate command profile in the OPERCMDS resource class.

See *z/OS JES2 Initialization and Tuning Reference* for the syntax of the profile names for these commands.

Enable guaranteed print labeling

Guaranteed print labeling ensures the integrity of the identification label by preventing the user from changing the label. Before the printers are started, specify that the system uses guaranteed print labeling by activating the RACF PSFMPL resource class:

```
SETROPTS CLASSACT(PSFMPL) RACLIST(PSFMPL)
```

See *PSF for z/OS: Security Guide* for information about guaranteed print labeling, and *z/OS Security Server RACF Security Administrator's Guide* for information on setting up the PSFMPL resource class.

Auditing PSF

RACF provides the capability to generate an SMF type 80 log record each time an authorization check is performed to determine if a user is allowed to override the data page labeling or the system area requirements. To limit the number of audit records cut, use the SETROPTS LOGOPTIONS command or audit options in the profiles for the PSFMPL resource class.

Example: To cut an audit record for only those authorization checks that failed, issue the command:

```
SETROPTS LOGOPTIONS(FAILURES(PSFMPL))
```

RACF also provides the capability to generate an SMF type 80 log record each time an authorization check is performed to determine if an operator has issued a command to specify that no separator pages are to be produced.

At the end of each print operation, PSF generates an SMF type 6 log record. This record contains security-relevant information about the printing of the data set.

For information about setting options for auditing see *z/OS Security Server RACF Security Administrator's Guide* and *z/OS Security Server RACF Auditor's Guide*.

PSF restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Do not define the direct printing subsystem (DPSS). DPSS is not supported in a multilevel-secure environment. For more information about DPSS, see *PSF for z/OS: Customization*.

- Use only printers that support guaranteed print labeling for output requiring security labels and secure separator pages. Other printers can be used in a multilevel-secure system, but only for non-secure output. For information about how to determine whether a printer supports guaranteed print labeling, see *PSF for z/OS: Security Guide*.
- Do not use the job-header and job-trailer exit routines that are shipped with PSF. Replace these exit routines with APSUX01S and APSUX02S.

Checklist for PSF setup

Use the following checklist to ensure that you complete all the tasks required to set up PSF for multilevel security.

Check off	Item	For more information, see . . .
	Set up PSF print labeling. Use the following checklist to ensure that you complete all the tasks to set up PSF print labeling: • Create the security libraries. • Install the PSF exit routines APSUX01S and APSUX02S. • Modify the PSF startup procedure to reference the security libraries and to define print labeling. • Authorize users allowed to override print labeling • Authorize operators allowed to override separator pages • Activate the RACF PSFMPL resource class	"Enable guaranteed print labeling" on page 67
	Set audit options for PSF.	"Auditing PSF" on page 67
	Disable the direct printing subsystem (DPSS) if it is defined.	*PSF for z/OS: Customization*

RACF

RACF is a component of the z/OS Security Server. RACF, as the z/OS security manager, is responsible for making all access control decisions in z/OS. You can install another security product, but this book does not address the use of a security manager other than RACF, and IBM cannot make any statement about whether a system with another security product would support multilevel security. This book assumes that you have installed RACF and have it working on your system.

Where to find more information

- *z/OS Security Server RACF Auditor's Guide*
- *z/OS Security Server RACF Command Language Reference*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS Security Server RACF System Programmer's Guide*

RACF provides the following support for multilevel security:

- RACF authorizes access to protected resources based on the clearance of the user and the classification of the resource.
- RACF authorizes access to protected resources so that users cannot declassify information.

- RACF determines which data sets, files, and directories the user is authorized to see the names of when the name-hiding function is in effect.
- RACF restricts certain security-oriented functions to a security administrator.
- RACF provides the capability to audit all security-relevant events.
- RACF provides the capability to audit the list of data sets affected by a change in the security label of a particular data set profile.
- RACF allows users to query whether they have the write-down privilege, and activate and deactivate write-down mode if they do.

RACF profiles

The RACF database contains profiles for all the users who can access the protected resources in the system, for all the tape and DASD data sets that you want to protect, and for other general resources. In a multilevel-secure environment, you must define profiles for all users, including console operators and started procedures, and activate the appropriate RACF resource classes to protect all resources. Table 8 on page 70 lists the RACF resource classes that you should activate for multilevel security.

The STARTED class

The purpose of the RACF STARTED class is to assign RACF identities to started procedures to give the started procedures authority to access RACF-protected resources. The STARTED profile for a procedure can specify that the procedure is "trusted", and therefore can bypass security checking.

A "trusted" procedure can be assigned either the trusted attribute or the privileged attribute. Both of these attributes allow the procedure to bypass RACF authorization checking. The trusted attribute indicates that auditing of the access should not be bypassed. The privileged attribute causes auditing to be bypassed. Therefore, in a multilevel-secure system, use the trusted attribute instead of the privileged attribute for bypassing RACF authorization checking. Because authorization checking is bypassed, the audit options associated with individual resources are ignored. Use the SETROPTS LOGOPTIONS command or assign the UAUDIT attribute to the user ID associated with the started procedure to request auditing.

Profiles in the STARTED class have a segment, STDATA, that contains fields for the trusted and privileged attributes. You should have defined profiles in the STARTED class for all of your started procedures when you installed RACF, but you might have specified the privileged attribute for some started procedures. If you have defined profiles in the STARTED class, check them for any profiles that define started procedures to be privileged. If you find any, update the profiles to define the started procedures as trusted.

For information about the STARTED class, see *z/OS Security Server RACF Security Administrator's Guide*.

The started procedures table is an alternative to the STARTED class that you can use to assign RACF identities to started procedures and jobs. Check your started procedures table for procedures that are defined to be privileged, and if you find any, change them from privileged to trusted.

For information about the started procedures table, see *z/OS Security Server RACF System Programmer's Guide*.

Security labels

For information on defining security labels and assigning them to users, data sets, and other system resources, see “Defining security labels” on page 36 and “Assigning security labels” on page 37.

Surrogate job submission

In a multilevel-secure system it is important that users do not share user IDs and passwords, in order to ensure accountability. If you need to allow a user to submit a job on behalf of another user, you can set up surrogate job submission. Profiles in the SURROGAT resource class specify that a user (the surrogate

user) is able to submit a job on behalf of another user (the execution user). The surrogate user does not need to supply the execution user's password, but must have read access to the security label under which the job runs. The job runs with the user ID that the jobcard specifies, not the surrogate user's user ID. The audit record for surrogate job submission identifies both the surrogate user and the jobcard user ID.

To define which jobs are allowed to be submitted by surrogate users, the security administrator creates a profile for each appropriate job in the SURROGAT resource class, and permits the submitting user to the access list in the specific job profile with at least READ access.

For information about surrogate job submission, see *z/OS Security Server RACF Security Administrator's Guide*.

Audit requirements

A complete audit trail requires the following:

- SMF type 30 records, subtypes 1 and 5. The SMF type 30 record contains information about job start and termination.
- SMF type 80 records
- SMF type 81 records
- SMF type 83 subtype 1 records

The RACF SMF data unload facility creates a sequential file from these records, which you can use to generate reports. For information about the SMF data unload facility, see *z/OS Security Server RACF Auditor's Guide*. For information about SMF records, see *z/OS MVS System Management Facilities (SMF)*.

RACF resource classes

Table 8 on page 70 lists RACF resource classes that should be active in a multilevel-secure system. It indicates whether the classes must be RACLISTed, or whether RACLISTing is recommended. RACLISTing a class reduces processing overhead because RACF copies the profiles for the class into storage, eliminating subsequent I/O when RACF needs to use the information. Use the SETROPTS command with the CLASSACT option to activate a class, and with the RACLIST option to make the profiles resident in storage. Classes that might be recommended or required for general usage of the system, but that have no significance for multilevel security, are not listed here. Before you activate a class, be sure that you have created the required profiles in the class. For information about defining profiles and activating and RACLISTing RACF resource classes, see *z/OS Security Server RACF Security Administrator's Guide*.

Guideline: Do *not* activate the SECLABEL class until you have defined security labels and assigned them to all users that require them. For more information about defining and assigning security labels, see "Defining security labels" on page 36 and "Assigning security labels" on page 37. For information about activating the SECLABEL class, see "Activating multilevel security" on page 94.

Table 8. RACF resource classes that should be active in a multilevel-secure system.

Class	Description	RACLIST required	RACLIST recommended
ACCTNUM	TSO/E account numbers		X
DEVICES	Unit record, teleprocessing, and graphics devices	X	
DIRAUTH	Receive messages (No profiles in this class)		
FACILITY	LLA-managed data sets and other system facilities		X

Table 8. RACF resource classes that should be active in a multilevel-secure system. (continued)			
Class	**Description**	**RACLIST required**	**RACLIST recommended**
JESSPOOL	JES spool files relating to jobs and the system (for example, SYSLOG, SYSIN and SYSOUT)		
OPERCMDS	MVS and JES commands associated with an operator	X	
PSFMPL	Users permitted to bypass security labels on hardcopy output	X	
SECLABEL	Definition of security labels	X	
SERVAUTH	Clients permitted to use a server or the resources controlled by a server	X	
SMESSAGE	List of users permitted to send messages to a user		X
TAPEVOL	Tape volumes		
TEMPDSN	DFP-managed temporary data sets (No profiles in this class)		
TERMINAL	TSO/E terminal		
VTAMAPPL	Controls authorization to open VTAM ACBs from non-APF authorized programs	X	

SETROPTS options

To ensure that security is not compromised in a multilevel-secure system, the RACF SETROPTS options listed in Table 9 on page 71 should be active. The user with the RACF SPECIAL attribute activates these options using the SETROPTS command.

Table 9. SETROPTS options that should be active in a multilevel-secure environment	
SETROPTS option	**Description**
CATDSN(FAILURES)	Use this option to prevent users from accessing data sets that are not cataloged or that are not system temporary data sets. FAILURES specifies that RACF is to reject any request to access a data set that is not cataloged.
ERASE(ALL)	Use this option to erase (overwrite with binary zeroes) the contents of any scratched or released data set extents that are part of a DASD data set regardless of the erase indicator set.
GENERICOWNER	Use this option to prevent an administrator from creating a profile that is more specific than an existing profile, for all general resource classes except the PROGRAM class and the grouping classes, except in the case where the administrator is the owner of the existing less specific profile.
JES(BATCHALLRACF, XBMALLRACF)	Use this option to require that all batch jobs run with a RACF-defined identity.

Table 9. SETROPTS options that should be active in a multilevel-secure environment (continued)	
SETROPTS option	**Description**
MLACTIVE(FAILURES)	Use this option to require that all resources protected by profiles in certain classes have a security label assigned to them. The classes are listed in Table 4 on page 28.
MLFSOBJ(ACTIVE)	Use this option to require that files and directories have security labels. Those that do not can only be accessed by trusted or privileged started tasks.
MLIPCOBJ(ACTIVE)	Use this option to require that all IPC objects have a security label. Those that do not can only be accessed by trusted or privileged started tasks.
MLS(FAILURES)	Use this option to prevent users from downgrading data by writing it to a lower security label, unless they have activated write-down mode.
MLSTABLE	Use this option to prevent authorized users from changing profiles in the SECLABEL class with the RALTER command, or changing the SECLABEL field in profiles, while the system is not quiesced.
NOMLQUIET	Run with the NOMLQUIET option set for normal operations. Set the MLQUIET option temporarily when you need to change profiles in the SECLABEL class or change the SECLABEL field in profiles.
PROTECTALL (FAILURES)	Use this option to ensure that a user can create or access a data set only if the data set is RACF-protected.
SECLABELAUDIT	Use this option to log access attempts to resources that have a security label assigned and access attempts by users who have a security label assigned. The profile in the SECLABEL class that defines a security label specifies the auditing that is done.
SECLABELCONTROL	Use this option to prevent users who do not have the RACF SPECIAL attribute from changing profiles in the SECLABEL class using the RALTER command or changing the SECLABEL field of profiles.

The following options control the use of security labels, and are optional in a multilevel-secure environment.

Table 10. SETROPTS options that are optional in a multilevel-secure environment	
SETROPTS option	**Description**
MLNAMES	Use this option to activate the name-hiding function. The name-hiding function can degrade system performance because it requires authorization checks for every object for which a non-SPECIAL user attempts to list the name. You should balance the performance impact against the possibility of exposing sensitive information in the names of data sets, files, and directories on your system to decide whether you want to activate the MLNAMES option.
SECLBYSYSTEM	Use this option to activate the use of system-specific security labels. The SECLBYSYSTEM option can sometimes cause unexpected results from authorization checks, because the security labels used on different systems in a sysplex are not consistent. (For examples, see “Shared file system environment and system-specific security labels” on page 25). Activate this option only if you need to run work on specific systems on a sysplex based on security classification.

The following options control the use of security labels, and are not recommended in a multilevel-secure environment.

Table 11. SETROPTS options that are not recommended in a multilevel-secure environment

SETROPTS option	Description
COMPATMODE	This option allows a user to access a resource if the user is authorized to use a security label that would allow the access, regardless of whether the user is using the security label at the time of the authorization check.

For information about the SETROPTS options that control the use of security labels, see "SETROPTS options that control the use of security labels" on page 26. For information about setting these options, see "Activating multilevel security" on page 94. For information about SETROPTS options in general, see *z/OS Security Server RACF Security Administrator's Guide*. For information about the SETROPTS command, see *z/OS Security Server RACF Command Language Reference*.

Sysplex considerations

If you implement multilevel security in a sysplex without sysplex communications enabled, some SETROPTS commands are not automatically propagated to the other systems in the sysplex. SETROPTS functions that are not propagated include RACLIST, REFRESH, and WHEN(PROGRAM). These functions can affect multilevel security. If you issue one of these SETROPTS commands, you need to issue the command on every system in the sysplex.

Guideline: Use sysplex communications to propagate SETROPTS commands. Doing so avoids excessive ENF signaling caused by issuing SETROPTS commands on multiple systems, and keeps the in-storage security label information current on all systems.

RACF exit routines

The following installation exits that are shipped with RACF can be used on a multilevel-secure system without compromising security:

- The sample new password phrase exit, ICHPWX11 (from SYS1.SAMPLIB(RACEXITS))
- The REXX exec IRRPHREX (from SYS1.SAMPLIB(IRRPHREX)), which is invoked by ICHPWX11

For information about these exits, see *z/OS Security Server RACF System Programmer's Guide*.

RACF restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Global access checking can be used to allow access to protected resources that are accessed on a regular basis by many users at an installation. Global access checking is performed before security label checking or access list checking is performed. Global access checking does not permit auditing of the access to the protected resource. Therefore, if a user is allowed access to a resource based on a global access checking table entry, security label checking and access list checking are not performed for that user and there is no audit record of the user's access to the resource.

  To avoid the security exposure to a sensitive resource, define entries in the global access checking table for only those resources that do not require security label checking or access list checking and for which an audit record is not required. Your global entries should be made only for resources whose profiles specify a security label of SYSLOW. In addition, the global entry should specify an access level of READ, so that attempts to update the resource will require appropriate authorization using a profile.
- Do not create a profile in the FACILITY class protecting the resource IEC.TAPERING. If the FACILITY class is active and the profile exists, a programmer with read authorization could potentially write on a tape. For information about IEC.TAPERING, see *z/OS DFSMS Using Magnetic Tapes*.
- Do not use the RACF remote sharing facility (RRSF) in remote mode. If you use RRSF in local mode, ensure that command direction cannot be used by taking one of the following actions:

  - Ensure that the RRFSFDATA class is not active.
  - Define the profile DIRECT.* in the RRSFDATA class with UACC(NONE) and no users in the access list.

Checklist for RACF setup

Use the following checklist to ensure that you complete all the tasks required to set up RACF for multilevel security.

- Ensure that you have created a USER profile for each user and started procedure that can access protected resources in your system, and create new profiles if required.
- Ensure that there is an entry in the started procedure table or a profile in the STARTED class for every started procedure that accesses protected resources or authenticates users, and that each of these started procedures is assigned a RACF- defined user ID that is authorized to use a security label.
- Ensure that no entries in the started procedure table or profiles in the STARTED class specify the privileged attribute.
- Set up surrogate job submission, if you haven't already and you need to allow users to submit jobs on behalf of other users.
- Ensure that any profiles defined in the DASDVOL resource class specify the security label SYSHIGH and UACC(NONE) and include only appropriate trusted users in the access lists. (For more information, see “Protecting DASD volumes” on page 44.)
- Ensure that you do not have entries in the global access checking table for any of the following:
  - Resources that require mandatory access control checking for access
  - Resources that require discretionary access control checking for access
  - Resources with a security label other than SYSLOW
  - That specify an access level other than READ
- Ensure that you do not have a profile in the FACILITY class protecting the resource IEC.TAPERING.
- Ensure that you are not using the RACF remote sharing facility (RRSF) in remote mode.
- Specify auditing options.
- Define profiles in the ACCTNUM class.
- Activate and RACLIST the ACCTNUM class.
- Define profiles in the DEVICES class. Ensure that all profiles specify a security label.
- Activate the DEVICES class.
- Activate the DIRAUTH class.
- Define profiles in the FACILITY class.
- Activate and RACLIST the FACILITY class.
- Define profiles in the JESSPOOL class.
- Activate the JESSPOOL class.
- Define profiles in the OPERCMDS class.
- Activate and RACLIST the OPERCMDS class.
- Define profiles in the PSFMPL class.
- Activate and RACLIST the PSFMPL class.
- Define profiles in the RACFVARS class.
- Activate and RACLIST the RACFVARS class.
- Define profiles in the SERVAUTH class. Ensure that all profiles specify a security label.
- Activate and RACLIST the SERVAUTH class.
- Define profiles in the SMESSAGE class.
- Activate and RACLIST the SMESSAGE class.

- Define profiles in the TAPEVOL class. Ensure that all profiles specify a security label.
- Activate the TAPEVOL class.
- Activate the TEMPDSN class.
- Define profiles in the TERMINAL class. Ensure that all profiles specify a security label.
- Activate the TERMINAL class.
- Define profiles in the TSOAUTH class.
- Activate and RACLIST the TSOAUTH class.
- Define profiles in the TSOPROC class.
- Activate and RACLIST the TSOPROC class.
- Define profiles in the VTAMAPPL class.
- Activate and RACLIST the VTAMAPPL class.
- Activate the CATDSN(FAILURES) option
- Activate the ERASE(ALL) option
- Activate the GENERICOWNER option
- Activate the JES(BATCHALLRACF,XBMALLRACF) option
- Activate the PROTECTALL(FAILURES) option

See "Activating multilevel security" on page 94 for a description of other SETROPTS options that you need to activate to complete the activation of multilevel security.

RMF

The Resource Measurement Facility (RMF) helps you to manage the performance of your system by providing tools that monitor and report performance data.

Where to find more information

- *z/OS Resource Measurement Facility Programmer's Guide*
- *z/OS Resource Measurement Facility User's Guide*

RMF does not perform mandatory access checks for objects it presents in tabular report displays, or hide data set names in reports. Monitor II SMF type 79 data and Monitor III set-of-samples data might contain data set names. If you are implementing the name-hiding function and want to protect data set names from being exposed by RMF, create profiles in the RACF FACILITY class to protect the following resources:

- ERBSDS.MON3DATA - controls access to set-of-samples data by controlling who can invoke the ERB3XDRS service
- ERBSDS.MON2DATA - controls access to SMF type 79 data by controlling who can invoke the ERB2XDGS service.

If you do not define a profile to protect one of these resources, RACF does not restrict any user ID from invoking the service it protects.

Example: To allow only the user ID PERFMON to invoke the ERB3XDRS service and the ERB2XDGS service:

```
RDEFINE FACILITY ERBSDS.MON3DATA UACC(NONE)
RDEFINE FACILITY ERBSDS.MON2DATA UACC(NONE)
PERMIT ERBSDS.MON3DATA CLASS(FACILITY) ID(PERFMON) ACCESS(READ)
PERMIT ERBSDS.MON2DATA CLASS(FACILITY) ID(PERFMON) ACCESS(READ)
SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
```

You could also do this using a generic profile:

```
RDEFINE FACILITY ERBSDS.* UACC(NONE)
PERMIT ERBSDS.* CLASS(FACILITY) ID(PERFMON) ACCESS(READ)
SETROPTS CLASSACT(FACILITY) RACLIST(FACILITY)
```

Checklist for RMF setup

Use the following checklist to ensure that you complete all the tasks required to set up RMF for multilevel security:

- If you are implementing the name-hiding function, create profiles in the FACILITY class to protect the resources ERBSDS.MON3DATA and ERBSDS.MON2DATA.

SDSF

The System Display and Search Facility (SDSF) displays system information.

Where to find more information

- *z/OS SDSF Operation and Customization*
- *z/OS Security Server RACF Security Administrator's Guide*

SDSF provides the following support for multilevel security:

- All actions performed through SDSF are checked to ensure that the user has sufficient authority to perform the request.
- All access attempts to protected SDSF resources can be audited.
- Rows in tabular panels are filtered based on security label dominance when the MLACTIVE option is active. If SDSF cannot obtain a security label for an object, it assigns one for the purpose of filtering.

SAF and ISFPARMS

You can provide security for SDSF using either the system authorization facility (SAF), which is part of the z/OS environment, or ISFPARMS, which is provided by SDSF. To use SDSF's support for multilevel security, you must use SAF to provide security for SDSF. For information about how to use SAF for SDSF security, see *z/OS SDSF Operation and Customization*.

If you use the SDSF server to process ISFPARMS, assign a security label of SYSLOW to the server, or whatever security label is appropriate to access PARMLIB.

Security label assignments

For some panels, such as the job and output panels, SDSF can obtain the security labels of the objects (such as jobs and output) and filter what it displays based on security label dominance. For other panels, SDSF cannot obtain a security label for the objects. In these cases, SDSF assigns a security label to the objects, and uses the assigned security label for purposes of filtering. The security labels of objects displayed by SDSF are shown in Table 12 on page 76.

Table 12. Security labels for objects displayed by SDSF

Panel	Object	Security label	SDSF class resource that protects access to the panel
DA	Address space	Obtained from JES. For jobs not running under JES, SYSHIGH.	ISFCMD.DSP.ACTIVE.*jesx*
I	Job	Obtained from JES	ISFCMD.DSP.INPUT.*jesx*
ST	Job	Obtained from JES	ISFCMD.DSP.STATUS.*jesx*
O	Output	Obtained from JES	ISFCMD.DSP.OUTPUT.*jesx*
H	Output	Obtained from JES	ISFCMD.DSP.HELD.*jesx*
JDS	Data set	Obtained from JES	
OD	Output descriptor	Obtained from JESS	

Table 12. Security labels for objects displayed by SDSF (continued)			
Panel	**Object**	**Security label**	**SDSF class resource that protects access to the panel**
MAS	System	SYSNONE	ISFCMD.ODSP.MAS.*jesx*
JC	Job class	SYSNONE	ISFCMD.ODSP.JOBCLASS.*jesx*
INIT	Initiator	Obtained from JES. If initiator idle, uses SYSNONE	ISFCMD.ODSP.INITIATOR.*jesx*
SP	Spool volume	SYSNONE	ISFCMD.ODSP.SPOOL.*jesx*
PR See note following the table.	Printer	Obtained from JES	ISFCMD.ODSP.PRINTER.*jesx*
PUN	Punch	Obtained from JES	ISFCMD.ODSP.PUNCH.*jesx*
RDR	Reader	Obtained from JES	ISFCMD.ODSP.READER.*jesx*
LINE	NJE line	Obtained from JES	ISFCMD.ODSP.LINE.*jesx*
NO	Node	Obtained from JES	ISFCMD.ODSP.NODE.*jesx*
NC	Network connection	Obtained from JES	ISFCMD.ODSP.NC.*jesx*
NS	Network server	Obtained from JES	ISFCMD.ODSP.NS.*jesx*
SO	Offloader	Obtained from JES	ISFCMD.ODSP.SO.*jesx*
ENC	Enclaves	SYSHIGH	ISFCMD.ODSP.ENCLAVE
PS	Process	Obtained from z/OS	ISFCMD.ODSP.PROCESS
SE	Scheduling environment	SYSNONE	ISFCMD.DSP.SCHENV.*system*
RES	WLM resource	SYSNONE	ISFCMD.ODSP.RESOURCE.*system*
LOG	Syslog	None	ISFCMD.ODSP.SYSLOG.*jesx*
LOG	Operlog	None	ISFCMD.ODSP.SYSLOG.*jesx*
ULOG	EMCS console	None	ISFCMD.ODSP.ULOG.*jesx*
SR	Reply and action messages	SYSHIGH	ISFCMD.ODSP.SR.*system*
CK, CKH	Check for IBM Health Checker for z/OS and Sysplex	SYSNONE	ISFCMD.ODSP.HCHECKER.*system*
RM	JES2 resource	SYSNONE	ISFCMD.ODSP.RESMON.*jesx*

Note: If the printer is configured to process multiple security labels, the row is displayed only if the security label of the user dominates the security label of the device. Thus the device might be printing the user's job but the row might be suppressed from the panel.

Controlling access to SDSF panels

Guidelines:

- When an SDSF panel shows objects that have a security label of SYSNONE, consider using a security label to control access to the panel. When you protect the panel, protection of the individual objects on the panel is less important. To protect a panel, add a security label to the profile for the SAF resource that SDSF checks to determine if a user can open the panel. The security label should be SYSHIGH; otherwise, a user whose security label is less than SYSHIGH could access the panel, and it would show all rows. Table 12 on page 76 shows which panels show objects that have a security label of SYSNONE, and the resources that protect access to those panels.

  Example: SDSF assigns a security label of SYSNONE to the JES members that are shown on the MAS panel. You could protect access to the MAS panel by defining a profile to protect the resource representing the MAS command and adding a security label to the profile:

  ```
  RDEFINE SDSF ISFCMD.ODSP.MAS.* SECLABEL(SYSHIGH)  UACC(NONE)
  ```

  For more information about SDSF resources, see *z/OS SDSF Operation and Customization*.
- ULOG panel: Control activation of the EMCS console by adding a security label of SYSHIGH to the profile in the OPERCMDS resource class that protects the resource MVS.MCSOPER.*console_name*. SYSHIGH is recommended because the EMCS console can receive unsolicited messages. Also, control access to the panel by adding a security label of SYSHIGH to the profile in the SDSF class that protects the resource ISFCMD.ODSP.ULOG.*jesx*.
- LOG panel: Control access to the SYSLOG and OPERLOG panels by adding a security label of SYSHIGH to the profile in the SDSF resource class that protects the resource ISFCMD.ODSP.SYSLOG.*jes_system_name*. SYSHIGH is recommended because the log contains messages issued by all address spaces in the system.
- SR panel: Control access to the panel by adding a security label of SYSHIGH to the profile in the SDSF resource class that protects the resource ISFCMD.ODSP.SR.*system*. SYSHIGH is recommended because the panel shows messages issued by all address spaces in the system.

SDSF Restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Do not configure SDSF to use IBM MQ. If your installation uses SDSF's sysplex support, which is provided through IBM MQ, disable it by removing the server group definitions from ISFPARMS. Server groups are defined with SERVERGROUP, SERVER, and COMM statements.

Checklist for SDSF setup

Use the following checklist to ensure that you complete all the tasks required to set up SDSF for multilevel security:

- Ensure that SDSF is set up to use SAF to provide security.
- If you use the SDSF server to process ISFPARMS, assign a security label of SYSLOW to the server, or whatever security label is appropriate to access PARMLIB.
- Control access to SDSF panels that show objects having a security label of SYSNONE.
- Control activation of the EMCS console.
- Control access to the SYSLOG and OPERLOG panels.
- Control access to the SR panel.
- Ensure that SDSF is not configured to use IBM MQ.

TCP/IP

TCP/IP is a component of the z/OS Communications Server.

Where to find more information

- *z/OS Communications Server: IP Configuration Guide*

- *z/OS Communications Server: IP Configuration Reference*
- *z/OS Security Server RACF Security Administrator's Guide*

Preparing for IP networking in a multilevel secure environment in *z/OS Communications Server: IP Configuration Guide* contains detailed information on the topic. Read that topic for background information on networking in a multilevel-secure environment, as well as for detailed information on setting up TCP/IP networking in a multilevel-secure environment.

In a multilevel-secure system, applications can use sockets to communicate securely with other applications within a multilevel-secure network. IP addresses are configured into named network security zones. These security zone names are used to construct resource names in the SERVAUTH class. Security labels are assigned to the profiles that map these network security zone resource names. Communications are only allowed between applications running under user sessions with equivalent security labels. A network security zone profile name can also be used by an authorized socket server program as a port of entry identifier during user login processing. This use ensures that the resulting user session runs under a security label equivalent to the network security zone containing the client IP address. The TN3270 server maps TCP connections to SNA sessions. In a multilevel-secure system, it ensures that the mapped LU name has the same security label as the client's network security zone. 3270 applications, such as TSO/E, can then use the LU name as a port of entry resource identifier in the TERMINAL class. Use of a TCP/IP stack can be restricted to users running with a security label equivalent to a specific security label or can be unrestricted, that is, available to users running with different security labels. A z/OS multilevel-secure system can be configured to run with up to eight concurrent TCP/IP stacks in any mix of restricted and unrestricted configurations.

TCP/IP provides the following support for multilevel security:

- A TCP/IP stack can have a security label.
- Application socket creation on a stack can be controlled by security labels on profiles in the SERVAUTH resource class that protect EZB.STACKACCESS resources.
- The ability of applications to issue gethostid() and gethostname() can be controlled by security labels on profiles in the SERVAUTH resource class that protect EZB.STACKACCESS resources.
- Socket commands accept(), bind(), connect() and all commands that read or write data through a socket can be controlled by security labels on profiles in the SERVAUTH resource class that protect EZB.NETACCESS resources.
- All user application socket command attempts can be audited.
- Socket peer port of entry information can be extracted (SIOCGSOCKPOEATTRS ioctl).
- TN3270 server LU name mapping can be controlled by network security zone security labels.

Mandatory access checking for stacks in a multilevel-secure environment

TCP/IP supports two types of stacks in a multilevel-secure environment:

- Restricted stacks run with a specific security label, and ensure that only applications running with a security label that is equivalent to the stack's security label open sockets.
- Unrestricted stacks run with the SYSMULTI security label, and allow applications with any security label to open sockets.

When the RACF MLACTIVE option is active, mandatory access checks for TCP/IP stacks include the following processing:

- A stack allows new sockets only if the stack is protected by a STACKACCESS profile in the SERVAUTH class.
- Network access is allowed only to IP addresses that are mapped into network security zones protected by NETACCESS profiles in the SERVAUTH class.
- Restricted stacks do not allow tasks that have the SYSMULTI security label to have network access to security zones with security labels that are not equivalent to the stack's security label.

- Unrestricted stacks transmit packet labels both internally and externally to enable a mandatory access check between the sending task's security label and the receiving task's security label, when both IP addresses are in security zones with a SYSMULTI security label.
- Distributing stacks consider security labels in choosing target applications
- TN3270 servers consider security labels in mapping connections to LU names.

Setting up a multilevel-secure TCP/IP network

In a multilevel-secure environment, the network and security administrators must plan the secure networking environment. The steps required are described in detail in the topic on preparing for TCP/IP networking in a multilevel-secure environment in *z/OS Communications Server: IP Configuration Guide*.

TCP/IP restrictions

All programs in a multilevel-secure environment must be inspected for conformance to certain programming techniques before being used. In addition to the communications protocol stack, z/OS Communications Server provides many client and server applications and user commands. Not all of these programs have been inspected and approved for use in a multilevel-secure environment. Those that are and are not approved are listed in *z/OS Communications Server: IP Configuration Guide*.

TSO/E

In a multilevel-secure system, a TSO/E user creates a security environment by supplying a security label at logon time. This security label determines the security classification associated with the user's TSO/E session. The system uses the security label to control access to protected resources.

Where to find more information

- *z/OS MVS Initialization and Tuning Guide*
- *z/OS MVS Initialization and Tuning Reference*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS TSO/E Command Reference*
- *z/OS TSO/E Customization*

TSO/E provides the following support for multilevel security:

- All users can be identified and can have a security label.
- All user logon attempts can be audited.
- Messages flowing between untrusted users can be protected.
- The security administrator can restrict a user's ability to submit or cancel a job based on the name of the job.

TSO/E user identification

In a multilevel-secure environment, the security administrator must define all TSO/E users to TSO/E through the RACF database rather than through SYS1.UADS. (However, it is recommended that one duplicate user be defined in SYS1.UADS for recovery purposes.) If you are using SYS1.UADS to define your TSO/E users, convert your SYS1.UADS database to RACF. All TSO/E users must be authorized to use at least one security label.

For information on converting an existing UADS database to RACF, see *z/OS TSO/E Customization*.

Security label on logon

A TSO/E user can be authorized to more than one security label, but can log on with only one security label at a time. Both the LOGON command and the LOGON panel allow a user to specify a security label.

The security label that a TSO/E user sets at logon time is valid for the length of the TSO/E session. The TSO/E user can change the security label at subsequent logons if authorized to use another security label.

- Line-Mode LOGON Command

  If the TSO/E user does not specify a security label on the LOGON command, the system uses the security label of the terminal from its profile in the TERMINAL class. If the terminal does not have a security label, the system uses the default security label from the user's RACF profile.

- Full-Screen LOGON Panel

  The full-screen LOGON panel displays the security label from the user's TSO segment, if the segment contains one. The user can choose to use that security label, or specify another one. In either case, the system saves the security label in the user's TSO segment for use during the next full-screen LOGON.

Generic TSO system names

If you are using system-specific security labels (the RACF SECLBYSYSTEM option is active), be aware that using a generic TSO system name at logon might not work, because the user could be allocated to a system where the user's security label is not active. You should disable the use of generic TSO system names, or use care in defining on which systems security labels are active to ensure that a user is not allocated to a system on which the user's security label is not active. For more information on TSO generic resource support, see *z/OS TSO/E General Information*.

Audit all logon attempts

MVS generates an audit record (SMF type 30) whenever a TSO/E user attempts to logon. RACF records information about each failed logon in the SMF type 80 record.

Protect user messages

In a multilevel-secure system, the broadcast data set should not be used to contain messages for individual users. Instead, each user should have an individual user log. This user log contains the messages for only the particular user. The user log cannot be browsed or edited by the owning user. Access to the user log is via the LISTBC command.

To ensure that the user's messages are protected in an individual user log and that access to the messages is based on security label, specify the following operands in the SEND parameter of the IKJTSO*xx* member of SYS1.PARMLIB:

- Set the LOGNAME operand to “logname”

  Note: The “logname” should not be the name of the broadcast data set (for example, SYS1.BRODCAST) because individual user logs should be used in a multilevel-secure system.

- Set the USEBROD operand to “OFF”
- Set the MSGPROTECT operand to “ON”

You can change TSO/E parmlib settings using the TSO/E PARMLIB command or the SET IKJTSO command, or by IPLing.

To protect a user's messages, the security administrator must define a RACF profile covering the user logs. The name of the profile for each user's log is of the form *logname.userid*, where *logname* is an installation-specified name in the LOGNAME parameter of the SEND PARMLIB statement. Define the user log as SYSHIGH because the log can contain any level of information.

The security administrator also must define a RACF profile for the broadcast data set, which contains notices for TSO/E users. Define a RACF profile for this data set in the DATASET resource class with a universal access (UACC) of READ and with a security label of SYSLOW.

If a user does not have an individual user log, the message is not stored in the broadcast data set and the user does not receive the message.

See *z/OS MVS Initialization and Tuning Reference* for a description of the IKJTSO*xx* member of SYS1.PARMLIB.

See *z/OS TSO/E Customization* for information on the use of individual user logs and changing the SEND PARMLIB parameter.

Sending and receiving messages

RACF can audit the sending of messages and can control the sending and viewing of messages to prevent the declassification of information. A user should be able to send messages only to another user who has access to a security label that dominates the sender's security label. Sending messages to a user with a lower security label would declassify the information.

- Sending messages (SEND command)

  The RACF SMESSAGE resource class allows the security administrator to audit and control which users can send messages to other users via the TSO/E SEND command. Define a profile in this class for each user to which you want to restrict the sending of messages. The profile name must be the user's user ID. For other users to send a message to the first user, they must be permitted to the first user's profile.

  If you do not create a profile for a user in the SMESSAGE class, then anyone can send a message to that user. The event will be audited if the SMESSAGE class is specified for auditing with the SETROPTS LOGOPTIONS command.

  Guideline: Define a generic SMESSAGE ** profile to ensure proper message traffic control for all TSO/E users:

  ```
  RDEFINE SMESSAGE ** UACC(NONE)
  SETROPTS RACLIST(SMESSAGE) REFRESH
  ```

  Whenever a user SENDs a message to a user who has a profile in the SMESSAGE resource class, RACF performs two checks. The first check is a discretionary access check to ensure that the sender is included in the access list of the receiver's profile. If not the message is not sent and the sender is notified of that event.

  If the discretionary access check completes successfully, the system handles messages as follows:

  – Messages sent directly (NOW operand on SEND command)

    RACF performs a mandatory access check to ensure that the security label of the receiver dominates that of the sender. If the mandatory access check completes successfully, the message appears immediately on the receiver's display screen. Otherwise, the message is cancelled and the sender is notified.

  – Messages to be saved (SAVE operand on the SEND command)

    The system stores these messages in the receiver's user log. RACF performs the mandatory access check for these messages when the receiver issues the LISTBC command.

  See *z/OS Security Server RACF Security Administrator's Guide* for information about using the SMESSAGE class to control the use of the TSO/E SEND command.

  See *z/OS TSO/E Command Reference* for information about use of the SEND command.

- Receiving messages (LISTBC command)

  A user issues the LISTBC command to retrieve messages from the broadcast data set and from the individual mail log. The notices from the broadcast data set precede those in the individual mail log. In a multilevel-secure system, the user can retrieve only those messages for which the user's security label dominates that of the sender.

  The RACF directed authorization (DIRAUTH) resource class allows the security administrator to prevent users from viewing messages at a higher security label than the security label with which they logged on. Activating this class acts as a switch to turn on mandatory access checking for messages. When the class is active, before the user receives a message RACF performs a mandatory access check to ensure that the receiver's security label dominates that of the sender. RACF audits the event according to the options selected in the RACF SETROPTS LOGOPTIONS command for the DIRAUTH resource class. Do not define any profiles in the DIRAUTH class.

When the user issues the LISTBC command, if there are messages in the user log that have a security label higher than the security label with which the user logged on, two possibilities exist:

- If the user has authorization to a security label that dominates the security label of the messages in the user log, LISTBC processing notifies the user that the user log contains these messages:

  ```
  IKJ56962I  YOUR USER LOG CONTAINS MESSAGES THAT CANNOT BE VIEWED AT YOUR
             CURRENT SECURITY LABEL
  ```

  By logging on at an appropriate security label, the user can retrieve these messages.
- If the user does not have authorization to a security label that dominates the security label of a message in the user log, LISTBC processing erases the message from the user log and sends a message to the security administrator's console specifying that the receiver does not have the authorization necessary to receive a message from this sender.

See *z/OS Security Server RACF Security Administrator's Guide* for information about using the DIRAUTH class and auditing the LISTBC command.

See *z/OS TSO/E Command Reference* for information about using the LISTBC command.

Control access to spool data sets via TSO/E commands

The TSO/E OUTPUT, TRANSMIT, and RECEIVE commands provide access to spool data sets. To ensure that the security of these data sets is maintained, usually only the user who created the data set is allowed to access it. However, the security administrator can authorize other users access to these data sets, with the appropriate auditing controls.

OUTPUT command

To authorize a TSO/E user to access the data sets created by a job, define a profile in the JESSPOOL resource class. If the user owns the job, the user has ALTER access and can perform any actions that the OUTPUT command allows. If the user is not the owner of the job, the access assigned can be either READ or ALTER, and the OUTPUT command is executed accordingly. Whether the user is the owner or not, the security label with which the user is currently logged on must dominate the security label of the SYSOUT data set.

For more information about protecting SYSOUT data sets, see *z/OS Security Server RACF Security Administrator's Guide*.

TRANSMIT and RECEIVE commands

TSO/E users use the TRANSMIT and RECEIVE commands to send data sets to other users. Access to these data sets is based on the user's security label.

The security label of a data set that is sent with the TRANSMIT command is that of the logged-on user who sent the data set. To RECEIVE the data set, the security label of the receiver must dominate that of the data set.

If the receiving user's security label does not dominate the security label of the data set, the user must log on at an appropriate security label in order to RECEIVE the transmitted data set. The user is not notified that this data set is present. If the user does not have access to a security label that dominates that of the transmitted data set, an audit record is written and the data set is deleted. Neither the sender nor the intended receiver is notified.

Two data sets are used by the TRANSMIT and RECEIVE commands:

- Log data set

  The log data set (*user.*LOG.MISC) contains information about TRANSMIT and RECEIVE activity. In a multilevel-secure system, if you are authorized for more than one security label, you must have more than one log data set. Each log data set contains only information transmitted or received at a particular security label.

Define the log data set at the user's most commonly used security label. For messages or data of a different security label, the user must specify either the LOGDSNAME or LOGDATASET keyword on the TRANSMIT or RECEIVE command.

See *z/OS TSO/E Command Reference* for a description of the LOGDSNAME and LOGDATASET keywords on the TRANSMIT and RECEIVE commands.

- NAMES Data Set

  The NAMES data set (*user.*NAMES.TEXT) contains information that both the TRANSMIT and RECEIVE commands can reference or update. Define the NAMES data set with a security label that is the lowest to which the user has access. Then, both TRANSMIT and RECEIVE are able to access the data set, and the user can update the data set when logged on at that level.

Control who can submit and cancel jobs

The security administrator can define profiles for jobnames in the RACF JESJOBS resource class. These profiles are used to restrict who can submit a particular jobname or who can be authorized to cancel another user's jobname.

SUBMIT command

To define what jobnames a TSO/E user is allowed to submit, create a profile for each jobname to be protected in the JESJOBS resource class. It is necessary to use this class only when you want to restrict the use of a particular jobname. Specify the users allowed (or not allowed) to submit the jobname in the access list of the jobname's profile.

For more information about controlling who can submit jobs, see *z/OS Security Server RACF Security Administrator's Guide*.

CANCEL command

To define what jobnames a TSO/E user is allowed to cancel, create a profile for CANCEL in the JESJOBS resource class with ALTER authorization. The only times RACF allows a user to cancel a job are when the user has submitted the job or when the user is defined in the access list of the appropriate CANCEL profile. This authorization to cancel a job also implies that the user can delete the job's SYSOUT data sets. Both the cancel and delete actions are auditable by RACF.

Note: A console operator is able to cancel a jobname with the MVS CANCEL command, which is not controlled by the JESJOBS resource class.

For more information about controlling who can cancel jobs, see *z/OS Security Server RACF Security Administrator's Guide*.

The security administrator can control a TSO/E user's access to SYSOUT data sets by defining profiles in the RACF JESSPOOL resource class. The TSO/E user issues the OUTPUT command to select data from a particular job.

TSO/E installation exit IKJEFF53

TSO/E provides a default exit routine, IKJEFF53, that allows you to tailor the way the CANCEL and OUTPUT commands operate. The exit routine receives control in the following circumstances:

- A user issues the CANCEL command to cancel another user's job.
- A user issues the OUTPUT command to process the output of another user's job.

This exit routine expects the jobname referenced to be in the format "*userid* + one character".

In a multilevel-secure system, you can control access to jobs and data through the JESJOBS and JESSPOOL resource classes. The name of the job is not in the format "*userid* + one character". Therefore, in a multilevel-secure system, replace the IBM-supplied default IKJEFF53 exit routine with the IKJEFF53 exit in SYS1.SAMPLIB. This non-default exit routine allows the use of the JESJOBS and JESSPOOL resource classes to control the jobname restrictions when the classes are active. The non-default exit

routine also is required when only the JESSPOOL resource class is active to ensure that the jobname is in the proper format.

See *z/OS TSO/E Customization* for information on how to install the IKJEFF53 exit routine.

TSO/E auditing

RACF provides the capability to generate an SMF type 80 log record each time the following events occur:

- A user issues a SEND command
- A user issues a LISTBC command

To limit the number of audit records cut, use the RACF SETROPTS LOGOPTIONS command for the SMESSAGE and DIRAUTH resource classes. (Auditing for the DIRAUTH resource class must be done by this method because there are no profiles defined in this class.)

TSO/E restrictions

To ensure that security is not compromised in a multilevel-secure system:

- Remove all user-written exit routines or other modifications that you have made to the system.
- Do not give users TSO/E OPERATOR privilege.
- Do not activate the Information Center Facility. The Information Center Facility does not support security requirements for multilevel security. For more information about the Information Center Facility, see *z/OS TSO/E Customization*.
- Do not use generic TSO system names if you are using system-specific security labels. For more information, see "Generic TSO system names" on page 81.

Checklist for TSO/E setup

Use the following checklist to ensure that you complete all the tasks required to set up TSO/E for multilevel security.

Check off	Item	For more information, see . . .
	Ensure that each TSO/E user is defined to RACF and authorized to use the appropriate security labels.	"TSO/E user identification" on page 80
	Edit SEND parameter of IKJTSO*xx* member of SYS1.PARMLIB: • Set LOGNAME operand to something other than the name of the broadcast data set • Set USEBROD operand to OFF • Set MSGPROTECT operand to ON	"Protect user messages" on page 81
	Define DATASET profiles protecting user logs, with security label SYSHIGH.	"Protect user messages" on page 81
	Define a DATASET profile for the broadcast data set, with security label SYSLOW and UACC(READ).	"Protect user messages" on page 81
	To control which users can send messages to other users, define a profile in the SMESSAGE class for each TSO/E user, and define a generic SMESSAGE ** profile.	"Sending and receiving messages" on page 82
	To control message viewing, activate the DIRAUTH class.	"Sending and receiving messages" on page 82

Check off	Item	For more information, see . . .
	Create profiles in the JESJOBS class to control what jobnames a TSO/E user can submit and cancel.	"Control who can submit and cancel jobs" on page 84
	Create profiles in the JESSPOOL class to authorize users to access data sets created by a job.	"Control access to spool data sets via TSO/E commands" on page 83
	Define security labels for log data sets.	"TRANSMIT and RECEIVE commands" on page 83
	Define security labels for NAMES data sets.	"TRANSMIT and RECEIVE commands" on page 83
	Replace default IKJEFF53 exit routine with the IKJEFF53 exit in SYS1.SAMPLIB.	"TSO/E installation exit IKJEFF53" on page 84
	Limit number of audit records cut for SEND and LISTBC commands.	"TSO/E auditing" on page 85
	Remove all user-written exit routines or other modifications.	"TSO/E restrictions" on page 85
	Ensure that no users have the TSO/E OPERATOR privilege.	"TSO/E restrictions" on page 85
	Ensure that the Information Center Facility is not active.	"TSO/E restrictions" on page 85
	If using security labels on a per-system basis, ensure that generic TSO system names are not in use, or cannot allocate a user to a system on which the user's security label is not active.	"Generic TSO system names" on page 81

VTAM

VTAM is a component of the z/OS Communications Server.

Where to find more information

- *z/OS Communications Server: SNA Network Implementation Guide*
- *z/OS TSO/E Command Reference*
- *z/OS Security Server RACF Security Administrator's Guide*

VTAM passes security information to RACF through the system authorization facility (SAF) to provide access control of information between address spaces.

VTAM provides support for multilevel security as follows:

- VTAM provides control of applications getting access to VTAM services.
- VTAM provides control for sending and receiving cross-address-space TSO/E messages.

Verify application authorization

An access method control block (ACB) allows an application program access to VTAM resources and facilities. VTAM issues a call to RACF for non APF-authorized programs and command processors to verify that the application has the authority to open an ACB and therefore access VTAM resources and facilities. The security administrator identifies to RACF those non APF-authorized application programs

and command processors that need to use the services of VTAM. The RACF resource class used to identify these programs is VTAMAPPL.

In a multilevel-secure system, the VTAMAPPL class must be active, but non APF-authorized programs cannot access VTAM resources. Define a profile (*) in the VTAMAPPL resource class with UACC (NONE):

```
RDEFINE VTAMAPPL * UACC(NONE)
```

Defining this profile restricts ACB opens to APF-authorized programs.

See *z/OS Security Server RACF Security Administrator's Guide* for information about the VTAMAPPL resource class.

VTAM auditing

RACF provides the capability to generate an SMF type 80 log record whenever an authorization check is performed to verify that a non-APF authorized application is allowed to open a VTAM ACB. To require auditing, use the SETROPTS LOGOPTIONS command for the VTAMAPPL resource class:

```
SETROPTS LOGOPTIONS(ALWAYS(VTAMAPPL))
```

See "TSO/E auditing" on page 85 for information about audit records cut when a user receives a message that was sent by the TSO/E SEND command.

Checklist for VTAM setup

Use the following checklist to ensure that you complete all the tasks required to set up VTAM for multilevel security

Check off	Item	For more information, see . . .
	Define a profile that restricts ACB opens to APF-authorized users: `RDEFINE VTAMAPPL * UACC(NONE)`	"Verify application authorization" on page 86
	Activate the VTAMAPPL class: `SETROPTS CLASSACT(VTAMAPPL)`	"Verify application authorization" on page 86
	Require auditing of attempts by non APF-authorized applications to open a VTAM ACB: `SETR LOGOPTIONS(ALWAYS(VTAMAPPL))`	"VTAM auditing" on page 87

z/OS UNIX System Services

Where to find more information

- *z/OS File System Administration*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS UNIX System Services Command Reference*
- *z/OS UNIX System Services Planning*

z/OS UNIX System Services provides the following support for multilevel security:

- z/OS UNIX controls access to files and directories based on security labels in addition to POSIX permissions, access control lists (ACLs) and profiles in the RACF UNIXPRIV class. When the SECLABEL class is active, z/OS UNIX can assign a security label to zFS files and directories when they are created, depending on the security labels of the parent directory and the user.

- The **chlabel** shell command allows a security administrator to assign a security label to a file system object that does not have one.
- z/OS UNIX can assume a security label for a read-only file system that does not already have a security label. This function allows you to use read-only HFS file systems in a multilevel-secure environment. For more information, see “Assumed security labels” on page 20.
- Users can have a different home directory and default program for each security label they are authorized to use.
- z/OS UNIX allows communication between processes only if they have equivalent security labels. Interprocess communication objects are assigned security labels, and can connect only with processes that have equivalent security labels.
- z/OS UNIX allows users to query whether they have write-down mode active, and activate and deactivate it if they have the write-down privilege.
- z/OS UNIX supports the name-hiding function. If the RACF MLNAMES option is active, a request to list the contents of a directory does not return the names of any files to which the user's security label does not give access. A readlink request fails if the user's security label does not give the user READ access to the symbolic link.
- z/OS UNIX commands allow authorized users to display security labels for z/OS UNIX files and directories, and for interprocess communication facilities.
- Socket functions (givesocket, takesocket, sendmsg, and recvmsg) check security labels to ensure that the user has the required authority for the requested function.

z/OS UNIX user IDs

Authorize the following user IDs to use the SYSMULTI security label and assign them a default security label of SYSMULTI:

- The OMVS user ID
- The user ID associated with the zFS started procedure
- The user ID associated with the BPXAS procedure (the procedure used to start the MVS initiator associated with forked/spawned address spaces)

For more information see “Assigning security labels” on page 37.

Support for security labels

For information about z/OS UNIX support for security labels, see “Using security labels with z/OS UNIX System Services” on page 18. For information about using system-specific security labels in a shared file system environment, see “Shared file system environment and system-specific security labels” on page 25.

Home directory and initial program for users

“Assigning a home directory and initial program depending on security label” on page 18 discusses how symbolic links can be used to define a different home directory and initial program for each security label a user is authorized to use. If you have authorized z/OS UNIX users to use multiple security labels, you need to do the following:

- In order to have security labels, a user's home directories must be in a zFS file system. Convert the file system containing the home directories from an HFS file system to a zFS file system, if you have not already done so. You can use the procedure described in “Migrating your HFS version root to a zFS version root with security labels” on page 90.
- Create the new home directories, and move files from the original home directories to the appropriate new directories.

  Example: A user with ID JDOE is defined to have a home directory of /u/jdoe, and is authorized to use the security labels APPLE and PEAR. The security administrator has already converted the file system containing the /u directory to a zFS file system. Either the security administrator or the user JDOE must

create the directories /u/APPLE/jdoe and /u/PEAR/jdoe, and move each file in /u/jdoe to the appropriate new directory.

Tip: Create the new home directories and move the contents of the old home directories to them while the SECLABEL class is inactive. After you activate the SECLABEL class, use the **chlabel** shell command to add security labels to the new directories and moved files.

- Create a symbolic link using the $SYSSECA/ or $SYSSECR/ symbol to indicate that the user's current security label should be substituted into the path.

  Example: Create a symbolic link /u/secsymr that indicates that the user's current security label is to be substituted into the path as a relative directory name:

  ```
  ln -s '$SYSSECR/' /u/secsymr
  ```

- Update the user profiles to specify a home directory or initial program using the symbolic link that you created.

  Example: After you have created the new home directories for the user with ID JDOE, issue the command:

  ```
  ALTUSER JDOE OMVS(HOME("/u/secsymr/jdoe"))
  ```

  If JDOE logs on with a security label of APPLE, the home directory is /u/APPLE/jdoe. If JDOE logs on with a security label of PEAR, the home directory is /u/PEAR/jdoe.

Security labels for z/OS UNIX files, directories, and symbolic links

When the SECLABEL class is active, the system creates security labels for z/OS UNIX files, directories, and symbolic links when the file, directory, or symbolic link is created. For information about how the system determines what security label to assign, see “Security labels for z/OS UNIX files and directories” on page 20. However, if files, directories, or symbolic links are created when the SECLABEL class is not active, the system does not assign them a security label. If you later activate the SECLABEL class, you need to assign a security label to the files, directories, and symbolic links that do not have them. You can use the **chlabel** shell command to do this. The security label of the data set should be consistent with the security label of the mount point.

Assign the security labels listed in the following table to z/OS UNIX files, directories, and symbolic links.

Table 13. Security labels for z/OS UNIX files, directories, and symbolic links

Directory, file, or symbolic link	Security label
/bin and contents	SYSLOW
/dev/console	SYSNONE (see Note 1)
/dev/fd directory	SYSMULTI (see Note 1)
/dev/fdn files	SYSNONE (see Note 1)
/dev/null	SYSNONE (see Note 1)
/dev/random	SYSNONE (see Note 1)
/dev/tty	SYSNONE (see Note 1)
/dev/urandom	SYSNONE (see Note 1)
/dev/zero	SYSNONE (see Note 1)
/etc/ssh	SYSLOW (see Note 2)
/lib and contents	SYSLOW
root	SYSMULTI

Table 13. Security labels for z/OS UNIX files, directories, and symbolic links (continued)

Directory, file, or symbolic link	Security label
root, symbolic links in: • /tmp • /dev • /etc • /var	SYSLOW
/samples	SYSLOW
/SYSTEM	SYSMULTI
/SYSTEM/tmp mountpoint	SYSMULTI
/SYSTEM/dev mountpoint	SYSMULTI
/SYSTEM/etc mountpoint	SYSMULTI
/SYSTEM/var mountpoint	SYSMULTI
/SYSTEM, symbolic links in: • /SYSTEM/tmp • /SYSTEM/dev • /SYSTEM/etc • /SYSTEM/var	SYSLOW
ttys, manager	SYSMULTI (see Note 1)
ttys, subsidiary	SYSMULTI (see Note 1)
/u	SYSMULTI
/u, symbolic link for security label substitution	SYSLOW
/u/*seclabel* mount point directories	*seclabel*
/usr and contents	SYSLOW
/usr/lib/ssh	SYSLOW (see Note 2)
/usr/lpp and contents	SYSLOW
/usr/man and contents	SYSLOW
/var/empty	SYSHIGH (see Note 2)
/var/run	SYSLOW (see Note 2)

Note 1: z/OS UNIX System Services can dynamically create these files and directories when they are first referenced. When the /dev/ directory is properly configured with the SYSMULTI security label, these dynamically created files are assigned the security labels listed. However, if you manually create these files and directories, or if they existed in the /dev/ directory before you activated the SECLABEL class, you should manually assign them the security labels listed.

An automount-managed directory (for example, /u/PEAR) should have a security label of SYSMULTI. The automount-managed file systems should have a security label assigned according to the data in them.

Migrating your HFS version root to a zFS version root with security labels

The file system that contains the binary files and text files delivered by IBM is called the version root (in a sysplex environment) or the root file system (in a non-sysplex environment). When you install z/OS,

you install the version root into an HFS file system. You need to migrate this HFS file system to a zFS file system with security labels assigned to all files and directories.

Steps for migrating your HFS version root to a zFS version root with security labels

Before you begin:

- You need to have your z/OS system installed and running with RACF as the security manager.
- You need to have the root file system (or version file system) mounted at any directory, with no other file system mounted under it. For purposes of illustration, the steps shown use the directory /CloneHFS as the mountpoint , but you can use any directory.

  Guideline: Use a clone (copy) of the root file system, not the production copy.
- You need to have created a user ID with UID 0 and security label SYSMULTI.
- You need to have created a user ID with UID 0 and security label SYSLOW.

  Tip: You can create one user ID with UID 0 and authorization to both the SYSMULTI and SYSLOW security labels.
- The SECLABEL class must be active.
- Data set profiles must be defined that assign a SYSMULTI security label to the zFS root file system you will allocate, and to all other file systems you want to migrate to zFS. You can use generic profiles.

Perform the following steps to migrate your HFS file system to a zFS file system.

1. Log on to a user ID with a UID of 0 and a security label of SYSMULTI.

   _____

2. Create a zFS file system. You need to allocate a VSAM linear data set and format it. See *z/OS File System Administration* for instructions.

   _____

3. Change the DATASET profile that protects the VSAM data set for the zFS file system to have a security label of SYSHIGH.

   _____

4. Mount the zFS file system that you created.

   a. Create a directory to use as a mount point for the zFS file system, or use an existing directory. For example:

   ```
   mkdir MLSzFS
   ```

   For purposes of illustration, in these steps we assume that the directory is MLSzFS, but you can use any directory.

   b. Mount the zFS file system. For example:

   ```
   MOUNT FILESYSTEM('your_zFS_filesystem') TYPE(ZFS) MOUNTPOINT('MLSzFS')
   ```

   _____

5. Create the six directories shown in Table 14 on page 91, with the permission bit mode settings shown. For illustrative purposes the mountpoint is assumed to be /MLSzFS. You can use the TSOMKDIR command, the ISHELL, a REXX exec, or the **mkdir** utility in the shell environment. Figure 1 on page 92 shows the sample commands for the **mkdir** utility. When you are done, the directories all have a security level of SYSMULTI, because your user ID has a security level of SYSMULTI.

Table 14. Directories to create for a zFS file system.

Directory	Permission Bit Mode Settings
/MLSzFS/SYSTEM	7,5,5

Table 14. Directories to create for a zFS file system. (continued)	
Directory	**Permission Bit Mode Settings**
/MLSzFS/SYSTEM/tmp	1,7,7,7
/MLSzFS/SYSTEM/dev	7,5,5
/MLSzFS/SYSTEM/etc	7,5,5
/MLSzFS/SYSTEM/var	1,7,7,7
/MLSzFS/u	7,5,5

```
umask 0000
mkdir -m 755  /MLSzFS/SYSTEM
mkdir -m 1777 /MLSzFS/SYSTEM/tmp
mkdir -m 755  /MLSzFS/SYSTEM/dev
mkdir -m 755  /MLSzFS/SYSTEM/etc
mkdir -m 1777 /MLSzFS/SYSTEM/var
mkdir -m 755  /MLSzFS/u
```

*Figure 1. Sample commands for the **mkdir** utility.*

_____

6. Log off of the user ID with the SYSMULTI user ID.

   _____

7. Log on to a user ID with UID 0 and security label SYSLOW.

   _____

8. Copy the files under the root file system (/CloneHFS) to the zFS file system you created (/MLSzFS) using the **pax** utility. Sample commands to do this are:

   ```
   cd /CloneHFS
   pax -rw -pe ./  /MLSzFS
   ```

   When you are done, the files and directories you copied under /MLSzFS have a security label of SYSLOW.

   _____

9. Unmount the file system from /MLSzFS. Update the BPXPRM*xx* parmlib member to activate this new root.

   _____

When you are done, you have a root zFS with security labels assigned to all files and directories. You can free the root file system.

Disabling cron for general use

`cron` is a clock daemon that runs commands at specified dates and times. It does not check security labels, and should be disabled for general use in a multilevel-secure environment.

Steps for disabling cron for general use

Perform the following steps to prevent a general user from running `cron` jobs, including jobs submitted by `crontab` or batch shell commands.

1. Define a unique security label for `cron` that is not dominated by any general user security label. One way to do this is to define a unique security category that you do not specify on any other security label, and specify this category for the `cron` security label.

Example: If you have already defined your highest security level to be HIGH and activated the SECLABEL class, you could define a unique security label for `cron` named CRONLBL, using a unique category named CRONCAT:

```
RALTER SECDATA CATEGORY ADDMEM(CRONCAT)
RDEFINE SECLABEL CRONLBL SECLEVEL(HIGH) ADDCATEGORY(CRONCAT)
SETROPTS RACLIST(SECLABEL) REFRESH
```

As long as you do not define another security label that specifies the CRONCAT category, no other general user security label will dominate the CRONLBL security label. (Of course, SYSHIGH will always dominate CRONLBL.)

Note: If you want to allow only users running with the SYSHIGH security label to run `cron` jobs, you can use SYSHIGH for `cron` instead of defining a new security label.

---

2. Assign the security label you created in step "1" on page 92 to /usr/spool/cron/ directory.

   Note: This directory might physically reside on the /etc/spool or /var/spool filesystem, with a symlink /usr/spool resolving the name to the /etc/spool or /var/spool directory.

   Example:

   ```
   chlabel CRONLBL /usr/spool/cron/
   ```

---

3. Assign the security label you created in step "1" on page 92 to /usr/spool/cron/crontabs and /usr/spool/cron/atjobs. Doing this prevents general users from creating `crontab` or `at` jobs.

   Example:

   ```
   chlabel CRONLBL /usr/spool/cron/crontabs
   chlabel CROLBL /usr/spool/cron/atjobs
   ```

---

4. If you want to allow an administrator to run `cron` jobs:
   - Give the administrator user ID (with UID 0) authorization to use the security label you created in step "1" on page 92.

     Example: For example, if the administrator's user ID is ADMIN:

     ```
     PERMIT CRONLBL CLASS(SECLABEL) ACCESS(READ) ID(ADMIN)
     ```

   - Start the `cron` daemon while running under the security label you created in step "1" on page 92.

When you are done, only users with the security label that you defined for `cron`, or the SYSHIGH security label, can run `cron` jobs.

z/OS UNIX restrictions

- The `cron` daemon does not check security labels, and should be disabled for general use in a multilevel-secure environment.
- The `pax` and `tar` commands are used to backup and restore files. Neither one backs up or restores security labels for files. In a multilevel-secure environment, users should not use `pax` or `tar` to backup and restore files. Instead, use DFSMSdss.
- The UUCP (UNIX-to-UNIX copy program) group of commands does not support multilevel security.
- Do not define the profile BPX.SAFFASTPATH in the FACILITY class. Defining this profile causes z/OS UNIX to bypass RACF in some cases to improve performance, and you should not bypass RACF in a multilevel-secure environment.

- The hierarchical file system (HFS) does not fully support security labels and multilevel security. You should not mount HFS file systems in read-write mode. If you need to use an existing HFS file system in read-write mode, copy or move it to a zFS file system.

Checklist for z/OS UNIX setup

Use the following checklist to ensure that you complete all the tasks required to set up z/OS UNIX for multilevel security:

- Disable the `cron` daemon for general use.
- Authorize the user ID OMVS to use the SYSMULTI security label, and define its default security label to be SYSMULTI.
- Authorize the user ID associated with the zFS started procedure to use the SYSMULTI security label, and define its default security label to be SYSMULTI.
- Authorize the user ID associated with the BPXAS started procedure to use the SYSMULTI security label, and define its default security label to be SYSMULTI.
- Change your installation's procedures for backing up and restoring files to use DFSMSdss instead of **pax** and **tar**.
- Migrate your HFS version root to a zFS version root with security labels.
- Update user profiles for z/OS UNIX users who are authorized to use more than one security label to specify a different home directory and default program for each security label.
- Assign security labels to z/OS UNIX files and directories that do not have them.

Activating multilevel security

You activate multilevel security by activating the SECLABEL resource class, and then activating the appropriate RACF SETROPTS options. The SETROPTS options that control multilevel security are described in “SETROPTS options that control the use of security labels” on page 26.

Steps for activating multilevel security

Before you begin: You need to have completed the setup described in the preceding parts of this topic, including defining and assigning security labels and setting up your software.

Perform the following steps to activate multilevel security.

1. Activate the SECLABEL class.

   The SECLABEL class should already be active, because you would have needed to activate it to complete the setup of z/OS UNIX. If for some reason it is no longer active, be sure that you activate it before you attempt to activate other SETROPTS options related to multilevel security.

   Activating this class causes the system to use security labels for authorization checks when the resource has a security label. If the resource has a security label, RACF compares the security labels of the user and the resource, and allows the access only if both a mandatory access control check and discretionary access control check are passed. If the resource does not have a security label, RACF does not check the user's security label, and does only a discretionary access control check to determine whether to allow the access.

   You must RACLIST the SECLABEL class as well as activate it. To do this, use the SETROPTS command:

   ```
   SETROPTS CLASSACT(SECLABEL) RACLIST(SECLABEL)
   ```

   Restart any started tasks or jobs that were already active, or you will receive error messages because they do not have a security label associated with them.
2. Activate the MLACTIVE option.

   Activating this option causes the system to require that most resources other than resources related to z/OS UNIX have security labels, and that all users entering the system have security labels. Table

4 on page 28 lists the resources that require security labels when the MLACTIVE option is active. This option should be active in a multilevel-secure environment.

You can activate this option in one of two modes: failures mode and warning mode. In failures mode, if you have not assigned security labels to any users, or to any resources that require them, mandatory access checks involving those users or resources fail. In warning mode, RACF issues a warning message for mandatory access checks involving those users or resources, but allows the access if the discretionary access check succeeds. To verify that you have assigned security labels to all users and resources that require them, you can first activate the option in warning mode:

```
SETROPTS MLACTIVE(WARNING)
```

If RACF issues warning messages, assign any security labels required. When you have run in warning mode without receiving warnings long enough to feel confident that you have assigned security labels to all users and resources that require them, activate the option in failures mode:

```
SETROPTS MLACTIVE(FAILURES)
```

3. Activate the MLFSOBJ option.

   This option controls whether security labels are required for z/OS UNIX files and directories. It should be active in a multilevel-secure environment. Before you activate this option, be sure that you have completed the setup documented in "z/OS UNIX System Services" on page 87, including migrating your HFS version root to a zFS version root with security labels.

   ```
   SETROPTS MLFSOBJ(ACTIVE)
   ```

   If users have problems accessing information in z/OS UNIX files and directories after you activate this option, you can deactivate it, review the information in "z/OS UNIX System Services" on page 87, make corrections, and reactivate it. To deactivate this option:

   ```
   SETROPTS MLFSOBJ(INACTIVE)
   ```

4. Activate the MLIPCOBJ option.

   This option controls whether security labels are required for interprocess communication. It should be active in a multilevel-secure environment.

   If the SECLABEL class is active, security labels are assigned to IPC objects during object creation, and security labels are checked before access is allowed to an IPC object that has a security label. However, as long as the MLIPCOBJ option is not active, an IPC object that does not have a security label can be accessed. If the MLIPCOBJ option is active, IPC objects that do not have security labels can no longer be accessed. Let your system run for a while with the SECLABEL class active before you activate the MLIPCOBJ option, to allow the system to assign security labels to IPC objects as they are created. Then activate the MLIPCOBJ option:

   ```
   SETROPTS MLIPCOBJ(ACTIVE)
   ```

5. Allow or prohibit write-down.

   The MLS and NOMLS options control whether write-down is allowed. When the MLS option is active, users cannot downgrade data by copying it to a lower security label, unless they have been given the write-down privilege and have write-down mode active. When the NOMLS option is active, write-down is allowed. For information about write-down and controlled write-down, see "Preventing declassification of data" on page 12 and "Controlled write-down" on page 13. For information about the MLS and NOMLS options, see "The MLS and NOMLS options" on page 30.

   By default, when a RACF database is initialized NOMLS is in effect.

   If you choose to prohibit write-down, decide whether you want to give some users the write-down privilege. If you do, create the profile IRR.WRITEDOWN.BYUSER in the FACILITY class, and authorize users to this profile to give them the write-down privilege. For example, to give USER8 the write-down privilege by default whenever USER8 enters the system:

```
RDEFINE FACILITY IRR.WRITEDOWN.BYUSER UACC(NONE)
PERMIT IRR.WRITEDOWN.BYUSER CLASS(FACILITY) ACCESS(UPDATE) ID(USER8)
SETROPTS RACLIST(FACILITY) REFRESH
```

Then activate the MLS option. You can activate it in one of two modes: failures and warning. In warning mode, RACF issues a warning for a request to write down if the user does not have write-down mode active, and allows the request. In failures mode, RACF fails a request to write down if the user does not have write-down mode active. Activate warning mode temporarily:

```
SETROPTS MLS(WARNING)
```

Run for a while in warning mode, to determine whether any users who require the write-down privilege do not have it. After you have run for long enough without receiving any errors to feel assured that all users who require the write-down privilege have it, activate failures mode:

```
SETROPTS MLS(FAILURES)
```

6. Activate the name-hiding function, or specify that it is not in effect.

   For information about the name-hiding function, see "The name-hiding function" on page 5. The MLNAMES and NOMLNAMES options control whether the name-hiding function is in effect. These options are described in "The MLNAMES and NOMLNAMES options" on page 29.

   If you choose to activate the name-hiding function, activate the MLNAMES option:

   ```
   SETROPTS MLNAMES
   ```

   If you choose to not have the name-hiding function in effect, activate the NOMLNAMES option:

   ```
   SETROPTS NOMLNAMES
   ```

7. Control whether authorized users can change security labels when the system is not quiesced.

   By default when a RACF database is initialized, the NOMLSTABLE option is in effect, and there are no restrictions on when authorized users can change profiles in the SECLABEL class with the RALTER command or change the SECLABEL field in profiles. If you want to allow security labels to be changed only when any possible users of the security labels are logged off and the RACF command SETROPTS MLQUIET has been issued, set the MLSTABLE option:

   ```
   SETROPTS MLSTABLE
   ```

8. If you choose to use system-specific security labels, activate the SECLBYSYSTEM option.

   The SECLBYSYSTEM and NOSECLBYSYSTEM options control whether security labels are defined on a system image basis. For information on system-specific security labels, see "Using system-specific security labels in a sysplex" on page 24.

   By default when a RACF database is initialized, the NOSECLBYSYSTEM option is in effect, and all security labels are active on all systems in the sysplex. If you have defined system-specific security labels and want them to become active only on the systems for which they are defined, activate the SECLBYSYSTEM option and refresh the SECLABEL class:

   ```
   SETROPTS SECLBYSYSTEM
   SETROPTS RACLIST(SECLABEL) REFRESH
   ```

9. Control which users can change security labels.

   The SECLABELCONTROL and NOSECLABELCONTROL options control whether users who do not have the RACF SPECIAL attribute can make changes to security labels by doing any of the following:

   - Changing a profile in the SECLABEL class with the RALTER command
   - Changing the SECLABEL field of a profile
   - Issuing an ADDSD, ALTDSD, or DELDSD command that causes the security label of a data set to change

By default when a RACF database is initialized, the NOSECLABELCONTROL option is in effect, and any user that has at least READ authority to a profile in the SECLABEL class can use the RALTER command to modify the profile. To restrict this capability to users who have the RACF SPECIAL attribute, activate the SECLABELCONTROL option:

```
SETROPTS SECLABELCONTROL
```

When you are done, you have implemented multilevel security.

Chapter 4. Auditing a multilevel-secure system

Multilevel security requires the creation, maintenance, and protection of an audit trail of accesses to protected objects. The installation can define the security-relevant events that make up this audit trail. Examples of security-relevant events are:

- Identification and authentication of users
- Events that permit a user to access objects
- Deletion of objects
- Privileged actions taken by console operators and security administrators
- User overrides of security labels on hardcopy output.

The system stores audit information for each event audited in an SMF record. The number and type of SMF records that the system generates depend on options that either the security administrator or the system auditor select. These users can use the RACF SMF data unload facility to create a sequential file of the SMF records generated. They can use the data to verify the effectiveness of the installation's security policy, determine whether the installation's security objectives are being met, and identify unexpected security-relevant events.

Where to find more information

- *z/OS MVS System Management Facilities (SMF)*
- *z/OS Security Server RACF Auditor's Guide*
- *z/OS Security Server RACF Command Language Reference*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS Security Server RACF System Programmer's Guide*

Security-relevant events

RACF generates audit records according to the type of event:

- RACF always logs information about certain events.
- RACF can optionally log information about certain events, as specified by the security administrator and the system auditor.

Events always logged

RACF always logs information about events such as unauthorized attempts to access the system, changes to the status of the RACF database, and the issuance of a SETROPTS or RVARY command to change RACF status. In support of multilevel security, RACF also always logs information about the following events:

- A console operator issues a LOGON or LOGOFF command.
- A user attempts to use a security label that is not valid.

Events optionally logged

The security administrator, the auditor, and non-SPECIAL users with appropriate authorization can specify additional security-relevant events for which log records are to be written. The security administrator, who has the RACF SPECIAL attribute, has control over all profiles in the RACF database. Therefore, the security administrator can specify audit requirements in individual data set and general resource profiles. The user who has the AUDITOR attribute is the only user allowed to issue the audit commands that pertain to an entire RACF resource class. Non-SPECIAL users can issue audit commands for their own data sets,

or for group data sets if they have CREATE authorization, and for classes in which they have CLAUTH authorization.

Auditing that the security administrator and other authorized users can specify

The security administrator, who has the SPECIAL attribute, can specify auditing options for a profile, using the AUDIT operand on the following RACF commands:

- ADDSD — Add Data Set Profile
- ALTDSD — Alter Data Set Profile
- RALTER — Alter General Resource Profile
- RDEFINE — Define General Resource Profile

On these commands, the security administrator specifies which type of logging is required for the resources protected by the profile:

ALL
Log both authorized accesses and detected unauthorized access attempts

FAILURES
Log detected unauthorized attempts

NONE
No logging

SUCCESS
Log authorized accesses

Non-SPECIAL users can issue these commands for their own data sets, or for group data sets if they have CREATE authorization, and for classes in which they have CLAUTH authorization.

Auditing that the auditor can specify

The user who has the AUDITOR attribute can do the following:

- Issue the ALTDSD and RALTER commands with the GLOBALAUDIT operand to set global audit options.
- Audit a specific user ID using the UAUDIT option on the ALTUSER command.
- Specify the LOGOPTIONS and SECLABELAUDIT operands on the SETROPTS command to request certain auditing functions.

Logging attempts to access resources in specific classes

The auditor can specify logging of access attempts on a class basis with the LOGOPTIONS option on the SETROPTS command. The class must be active for the logging to be done. The auditor can specify which type of logging is required:

ALWAYS
Log both authorized accesses and detected unauthorized access attempts

NEVER
Suppress all auditing

SUCCESSES
Log authorized access attempts

FAILURES
Log detected unauthorized access attempts

DEFAULT
Logging is controlled by the profile for this resource class

Use the LOGOPTIONS operand with SETROPTS to audit the following types of events.

Table 15. Events to audit using SETROPTS LOGOPTIONS	
RACF class	**Type of event**
DEVICE	Device allocation checking for communications, unit record, or graphics devices
DIRAUTH	Successful and unsuccessful accesses to messages
JESSPOOL	Create, access, or delete a SYSIN or SYSOUT data set
OPERCMDS	Operator commands
PSFMPL	Device access checks for printers for user authorization to bypass security overlays for printer pages
SMESSAGE	Successful and unsuccessful accesses to messages
TEMPDSN	Access or delete a DFP-managed temporary data set. Log only FAILURES, which could indicate that someone is trying to access someone else's data.

Note: SETROPTS LOGOPTIONS does not affect auditing of authorization checks performed with a RACROUTE REQUEST=FASTAUTH request.

The DIRAUTH, JESSPOOL, SMESSAGE, and TEMPDSN resource classes usually do not contain any profiles, so SETROPTS LOGOPTIONS is the only way to audit them.

Logging for resources and users that have a security label assigned

RACF audit records for object access always contain the subject's security label. RACF audit records for object access provide the object's security label when the SECLABELAUDIT option is active and the auditing options for the label's profile in the SECLABEL class require auditing.

Requirement: If you need the security labels of objects that are accessed to be included in your audit records, the auditor must activate the SECLABELAUDIT option and set options that require auditing in the SECLABEL profiles.

The auditor can activate the SECLABELAUDIT option to specify logging of all access attempts to resources that have a security label assigned, and all access attempts by users that have a security label assigned. The SECLABEL profile that defines the security label specifies the auditing that is done. The audit options that are specified in the SECLABEL profile are used in addition to the audit options specified in the profiles for the resource and the user. The additional auditing occurs whenever an attempt is made to access or define a resource protected by a profile, file security packet (FSP), or IPC security packet (ISP) that has a security label specified, or whenever a user running with a security label attempts to access or define a resource. For more information on the SECLABELAUDIT option, see “The SECLABELAUDIT and NOSECLABELAUDIT options” on page 31.

The auditor activates the SECLABELAUDIT option using the SETROPTS command:

```
SETROPTS SECLABELAUDIT
```

The SECLABELAUDIT option provides the ability to selectively audit the actions of users based on a security classification.

SETROPTS AUDIT

The system auditor can use the AUDIT operand on the RACF SETROPTS command to specify the classes for which RACF logs all accesses to the RACF database through RACF commands and DEFINE requests. Valid classes that the auditor can specify are USER, DATASET, and entries in the class descriptor table.

Guideline: Specify `SETROPTS AUDIT(*)`.

Auditing a specific user ID

The auditor can selectively audit the actions of a specific user ID using the UAUDIT operand on the ALTUSER command:

```
ALTUSER userid UAUDIT
```

This option causes RACF to log the following events:

- All RACF commands that the user issues
- All additions, changes, or deletions that the user makes to RACF profiles
- All attempts that the user makes to access RACF-protected resources, except those authorized by global access checking

For more information on how the auditor can use the SETROPTS options for auditing functions, see *z/OS Security Server RACF Auditor's Guide*, *z/OS Security Server RACF Command Language Reference*, and *z/OS Security Server RACF Security Administrator's Guide*.

SMF records

In a multilevel-secure system, security-related SMF records contain audit information from RACF, MVS, and PSF. Each record type contains the date and time of the event, the type of event, the success or failure of the event, and the user associated with the event.

- The SMF type 80 record is the primary audit record for a multilevel-secure system. Depending on the event audited, it contains information about attempts to enter the system, authorized accesses or unauthorized attempts to access protected resources, or authorized or unauthorized attempts to modify profiles in a RACF database.

  The type 80 record also contains security label information, the terminal ID where the event originated, and the text of a console operator command.
- The SMF type 81 record contains information about the initialization of RACF.
- The SMF type 83 record contains information about data sets affected by a change in security label. The type 83 record is created only when one of the RACF commands — ADDSD, ALTDSD, DELDSD — is used to change a security label. The SMF type 83 record contains a list of the data set names affected by the security label change and a link to the corresponding type 80 record that was cut when the command was issued.

MVS also generates SMF records that the auditor can use to monitor the system.

- The SMF type 30 record contains information about the start and termination of a batch job, a TSO/E session, and a started task (subtype 1 - job start, subtype 5 - job termination).

PSF generates the following SMF record:

- The SMF type 6 record is an audit record of the end of a print operation. It contains information about overrides of security labeling on hardcopy output.

SMF records can be written to SMF data sets, or log streams, or both.

- If you use SMF data sets, SMF writes records to the SMF data sets that you allocate. The size of the data that the system can write to SMF data sets is constrained by the VSAM control interval size. As records are created, SMF maintains them in buffers until they are written to DASD. It is possible to lose SMF records when no buffers are available to SMF, or when the allocated SMF data sets fill. To protect against data being lost, you can configure your system to issue operator messages when these conditions are about to occur. For information on how to do this, see “Establish SMF controls” on page 57.

  The SMF data sets are protected by RACF. Because an SMF data set contains different levels of information, assign the SYSHIGH security label to the data sets and permit only authorized users to access the data.
- If you use SMF logging, SMF writes records to the log streams that you set up. The log streams are managed by the system logger. SMF can write much larger chunks of data to the log stream than it can to SMF data sets. This has the potential to make writing SMF records faster. And with SMF logging, operators do not need to switch SMF data sets, nor dump them to archive storage, nor clear them.

Note: If you use SMF logging, SMF does not honor the NOBUFFS(HALT) and LASTDS(HALT) configuration parameters. If you require the SMF data loss prevention provided by these parameters, you should configure SMF to use SMF data sets, not log streams.

See *z/OS MVS System Management Facilities (SMF)* for a description of the record types listed.

Generating audit reports

The system auditor can generate reports to verify that the security policy of the installation is being maintained. RACF provides the SMF data unload facility to provide the data that can be used to create such reports.

The RACF SMF data unload facility (IRRADU00) enables installations to create a sequential file from the security-relevant audit data. The sequential file can be used in several ways: viewed directly, used as input for installation-written programs, and manipulated with sort/merge utilities. It is not intended to be used directly as input to RACF commands. It can also be uploaded to a database manager (for example, Db2) to process complex inquiries and create installation-tailored reports.

For information about how to use the RACF SMF data unload facility, see *z/OS Security Server RACF Auditor's Guide*.

Chapter 5. Operating a system

In a multilevel-secure system, all console operators must be defined to RACF. RACF can then establish an association between the operator and the command that the operator issues. RACF audits that event, and SMF records both the identity of the operator and the text of the command.

Operators use the LOGON command to identify themselves to RACF. In a multilevel-secure system, an operator can issue commands through the master console without logging on only until RACF is fully initialized and able to process logon requests. Until RACF is initialized, the operator cannot issue any commands from a secondary console, with one exception. The VARY MSTCONS command can be issued from a secondary console to establish an alternate master console if operator intervention is required to complete RACF initialization.

Once RACF is initialized, all operators are required to log on with a SYSHIGH security label. The system prompts the operator for a user ID and a password. Optionally, the operator enters a group ID. Regardless of the console in use, an operator is not able to issue commands successfully unless the operator has logged on, except during IPL, as described earlier.

Messages and notices

Two authorization checks can apply to messages sent between a console operator and a user:

1. **Before the message can be sent:** If the RACF SMESSAGE resource class is active the system does a discretionary access check to determine whether the sender has authority to the SMESSAGE profile covering the receiver. If the access check fails, the message is not sent.
2. **Before the message can be received:**
   - **If the sender is not a console operator:** If the RACF DIRAUTH and SECLABEL resource classes are active the system does a mandatory access check to determine whether the receiver's security label dominates the sender's security label. If the access check fails, the message is not received. Because the console operator's security label of SYSHIGH always dominates that of the user, a console operator can always receive a user's message (assuming that the user passed the SMESSAGE check and succeeded in sending the message).
   - **If the sender is a console operator:** No mandatory access check is done. The operator can send a message to any user unless the discretionary access check in the SMESSAGE class prevents the send.

A console operator is always permitted to send a public broadcast notice directed to all users. There is no enforcement of discretionary access control checking or mandatory access control checking when the operator sends a public broadcast notice.

Printed output

The printer operator in a multilevel-secure system is responsible for separating and distributing printer output. End users are not allowed to obtain their output directly from a printer. Instead, the printer operator must verify that the separator pages contain valid identification numbers. These identification numbers are system-generated random numbers that appear on the header and trailer separator pages for each print job. The printer operator is responsible for seeing that the numbers for the header and trailer match before distributing the printed output.

See *PSF for z/OS: Security Guide* for more information about the use of this security procedure.

Dumps and traces

The system assigns a security label to dump and trace data according to the original label of the data.

- A dump data set that contains any system data has a security label of SYSHIGH. Generalized Trace Facility (GTF) data and other data that contain trace data from multiple address spaces also has a security label of SYSHIGH. Therefore, all system programmers who work with these data sets must have a security label of SYSHIGH.
- Other users might have access only to SYSUDUMP, SYSABEND, and SYSMDUMP data sets. Any system information in these data sets is available at the security label of the dumped job.
- In case of a system failure, there might be SMF records in storage waiting to be written to DASD. To recover these records from a system dump, do the following:
  - Allocate a VSAM data set to contain the transferred SMF records from the dump. See *z/OS MVS IPCS Commands* for additional information about the attributes of this preallocated system dump data set.
  - Use the IPCS subcommand SMFDATA to recover the SMF records that remain in the buffers waiting to be written to the SMF data set.

Tape processing

The console operator and tape librarian must assure the physical protection of all tape volumes used in a multilevel-secure system. The following is a list of requirements for tape volumes in a multilevel-secure system:

- All tapes are initialized with standard labels. You can use the utility IEHINITT or the DFSMSrmm utility EDGINERS to do this. Non-labeled or non-standard labeled tapes cannot be used. The console operator must respond to any request for a non-labeled tape or for any action that could change label information by not allowing the request.
- No one should be authorized to use the bypass label processing (BLP) feature. To prevent the use of BLP, define the ICHBLP profile in the FACILITY class, and do not authorize any access to the class.
- All tapes must have a TAPEVOL profile. The DFSMSrmm TPRACF option can enforce this. The TAPEVOL profile can define the tape volume as either a private or a scratch pool volume.
- When the data on a tape is no longer required or when the tape is going to be reassigned for another use, the tape must be erased and reinitialized. The security administrator can then create a new profile for the tape volume. There are two methods you can use to erase the data on a tape:
  - Use a tape management system such as DFSMSrmm. For more information, see “Using DFSMSrmm” on page 49.
  - Degauss the tape. This method does not work for all types of tape.

For additional information on RACF protection of tape volumes, see *z/OS Security Server RACF Security Administrator's Guide*.

Residual temporary data sets on DASD

Certain situations, such as system failure, initiator failure or termination, or automatic restarts, could leave temporary data sets remaining on a DASD. To protect these data sets from unauthorized access, the user who is defined to RACF with the OPERATIONS attribute can scratch residual temporary data sets. This user cannot look at the data sets, thus ensuring the security of the information that the data sets contained prior to their being deleted.

SETROPTS MLQUIET

The SETROPTS MLQUIET command ensures that the security label of a currently opened data set cannot be altered. It is the responsibility of the security administrator and the console operator to ensure that the system is "drained" – that is that all unprivileged jobs are completed and that all users accessing data sets covered by a security label that is being changed are logged off – at the time SETROPTS MLQUIET is issued. To halt all network access, stop all TCP/IP stacks.

Chapter 6. Adding authorized programs to a multilevel-secure system

This topic includes general information on adding authorized programs to a secure system while still maintaining the integrity of the system.

System integrity

An operating system is said to have *system integrity* when it is designed, implemented and maintained to protect itself against unauthorized access, and does so to the extent that security controls specified for that system cannot be compromised. A multilevel-secure trusted computing base ensures system integrity. The trusted computing base has the ability to protect itself against unauthorized user access. An unauthorized program cannot bypass store or fetch protection, bypass password checking, bypass RACF checking, or obtain control in an authorized state.

A change to the trusted computing base could compromise the integrity of the system. The installation must ensure that any authorized programs added to the trusted computing base maintain the same controls or equivalent measures to protect the trusted computing base from unauthorized access. Any installation-written authorized code also must perform the same or equivalent type checking that the trusted computing base uses.

An *authorized program* is any program that executes in PSW key 0-7, in supervisor state, or is authorized by the authorized program facility (APF). See *z/OS MVS Programming: Authorized Assembler Services Guide* for some of the potential system integrity problems for an authorized program, including:

- User-supplied addresses for user storage areas.

  Routines with keys 0-7 must verify that the storage area that they are storing into or fetching from is in fact accessible to the user.
- User-supplied addresses for protected control blocks.

  Routines with keys 0-7 must verify that a control block address is valid and that the control block itself is not fraudulent.
- Resource identification.

  Authorized programs must do validity-checking to ensure that they are using the intended resource.
- SVC routines calling SVC routines.

  Problem programs might have the opportunity to alter data passed to authorized SVC routines.
- Control program and user data accessibility.

  Sensitive system data must be stored in and fetched from protected storage.
- Resource serialization.

  Routines should use locking mechanisms to prevent unauthorized altering of data.

Additionally, you should take care when adding any program that will run under a user ID that has UID(0), or with authority to the FACILITY class resources BPX.DAEMON, BPX.SERVER, or BPX.SUPERUSER, or to resources in the UNIXPRIV class. While such programs do not fit the traditional z/OS definition of an "authorized" program, they have broad authority to access data in the system or to assume other identities. If they have design or programming flaws then use or misuse of such programs could seriously compromise system security or integrity.

Examples of adding products

There might be instances when an installation requires the functions of products that are not part of the trusted computing base. The installation should add a product to the system only if it can insure that doing so will not compromise the integrity of the system.

The following topics are examples of products that an installation might require and modifications that allow the products to be used safely in a multilevel-secure environment.

CICS

Although CICS does not fully support multilevel security, you can use it in a multilevel-secure environment if you take care in the configuration.

First, for each set of related CICS regions (for example, the terminal owning region (TOR), application owning regions (AORs), and file owning regions (FORs)), if possible assign each region in the set the same region user ID. If you cannot do that, ensure that each region user ID has the same security label.

Next, for each set of regions specify a common application name, and protect that application name using a profile in the RACF APPL resource class. Assign the same security label to the APPL profile as the one you assigned to the region user IDs. Do not specify a security label of SYSNONE or SYSMULTI for the CICS region IDs, the APPL profiles that protect access to CICS, or the TERMINAL profiles protecting the 3270 terminals that users use for CICS.

Next, consider how the users' transactions will reach CICS. You have a variety of transaction input choices when you configure CICS, as documented in *CICS External Interfaces Guide*. The following list describes these choices and provides guidance for using them:

- **Input:** Input from a traditional 3270 screen or TN3270 emulator.

  The user is authenticated and bound to the active terminal by the EXEC CICS SIGNON command, which invokes RACROUTE REQUEST=VERIFY. Typically the SIGNON is executed from the CICS-supplied CESN transaction, but a user-written transaction that executes the SIGNON command can also be used. The SIGNON transaction accepts a user ID and password, and optionally a group ID and new password. All transactions subsequently entered from the signed-on terminal are executed with the authority of the associated user ID.

  Guideline: With this configuration, CICS supplies the TERMINAL name as input to the RACROUTE REQUEST=VERIFY. You must configure the TERMINAL profiles with appropriate security labels, and the security label from the TERMINAL definition must match the security label assigned to the APPL name or the signon will fail.
- **Input:** Transaction routing from another CICS address space in the same sysplex, using the CICS MRO (Multi-Region Operation) option, which can be implemented by using Cross-Memory Services (within the same MVS image) or Cross-System Coupling Facility (XCF) between CICSes in different MVS images.

  The user signs on in one CICS system (known as the Terminal-Owning Region (TOR)) and for each transaction a copy of the terminal definition, including its signed-on user ID, is shipped into the target CICS system (known as the Application-Owning Region (AOR)). The user ID is only authenticated when the user signs on in the TOR. Thereafter the user ID is shipped as "Already-Verified". The group ID and port-of-entry are also shipped from the TOR to the AOR, and the signon is replicated in the AOR with a RACROUTE REQUEST=VERIFY,PASSCHK=NO. The connection between the two CICS systems is authorized by checking that the CICS region user IDs are authorized to the DFHAPPL.*applid* profile in the FACILITY class using RACROUTE REQUEST=AUTH. The address space identifying itself as *applid* must have UPDATE authority to the profile, and any address space wishing to connect to *applid* must have READ authority.

  Guideline: You must configure the TERMINAL profiles for the users' terminals with a security label that matches the security label assigned to the CICS APPL name or the signon in the new region will fail.
- **Input:** Transaction routing from another CICS system, either in another MVS address space or on a CICS distributed platform (TX Series), using the APPC (LU6.2) connection protocol.

If the connection is between two z/OS systems, the user ID flows with "Already-Verified" set, and no password is shipped. (The port-of-entry and group ID cannot be shipped, because there are no architected fields for these in the APPC FMH5.) The connection between the two CICS systems can (optionally) be authorized by using profiles in the APPCLU class, obtained by using RACROUTE REQUEST=EXTRACT. This type of connection is specified in CICS with the attribute ATTACHSEC(IDENTIFY) in the connection definition. When CICS receives a user ID and password pair for the first time it uses RACROUTE REQUEST=VERIFY,PASSCHK=YES and caches the user ID and its ACEE. In subsequent uses of the same user ID, the password is validated by a high-performance password validator. If the connection is between a distributed CICS TOR and an MVS-based AOR, the TOR is not trusted to establish an "Already-Verified" connection, so it is required to send a password in the FMH5 for each transaction. This can be required for every transaction (ATTACHSEC(VERIFY) attribute) or just occasionally (ATTACHSEC(PERSISTENT) attribute). When Persistent Verification (PV) is used, CICS maintains "signed-on-to" and "signed-on-from" lists as required by the PV architecture, but it does not use the RACROUTE REQUEST=SIGNON function for this purpose, because the CICS PV support was implemented before this RACF support became available.

Guideline: Do not configure CICS to accept this interface. Its use could inappropriately declassify data, because APPC does not ensure that both ends of the conversation have the same security label.

- **Input:** Transaction routing from a non-CICS system, using the CICS Transaction Gateway and the External Presentation Interface (EPI).

  The EPI is a programmable interface that emulates the interface between a TOR and an AOR so that it can be used from a non-CICS client program. The connection definition within CICS uses ATTACHSEC(VERIFY) so the client must provide a user ID and password. The CICS Transaction Gateway (CTG) is a specialized client program available for Java™ applications (for example, servlets) that communicate with CICS over the EPI. CTG is available on distributed and MVS platforms.

  Guideline: Do not configure CICS to accept CTG or EPI. These interfaces into CICS do not provide sufficient control over the flow of data from one address space to another, and their use could inappropriately declassify data.

- **Input:** Input from IBM MQ using the MQ CICS bridge.

  This method uses MQ to transport data between systems, allowing communication between a CICS region and an application running remotely.

  Guideline: For more detailed information about MQ configuration in general, see “IBM MQ for z/OS” on page 115. Specifically for CICS, ensure that the CICS region ID has a security label that matches the security label that you assign to the MQCONN profile that protects the CICS-to-MQ connection. Also ensure that the "mover" address space assigned to transporting data between the CICS queue and the TCP/IP network has a security label that matches the security label of the MQCONN profile and of the SERVAUTH profile that protects the IP addresses it will communicate with.

- **Input:** Direct connection from the Web using CICS Web Support.

  In the native CICS sockets support, input HTTP connections are represented by a CICS resource known as a TCPIPSERVICE, which has the SSL(NO|YES|CLIENTAUTH) and AUTHENTICATE(NONE | BASIC | CERTIFICATE | AUTOREGISTER | AUTOMATIC) attributes.

  Guideline: Do not use AUTHENTICATE(NONE); rather, choose one of the other AUTHENTICATE options. CICS uses TCP/IP for this interface, and therefore the CICS region ID must have a security label that matches the security zone assigned by the Communications Server to the security zone that contains the IP address. If a user on an inappropriate IP address attempts to communicate with the CICS region, the Communications Server rejects the conversation. CICS then authenticates the user ID via the supplied user ID and password or certificate and the authentication checks that the user has access to the APPL profile. This requires that the user's default security label match that of the APPL profile in order for a successful signon. Thus, CICS, the user's IP address, and the user must all have matching security labels.

- **Input:** Direct connection from any socket-based client using the CICS Sockets Feature.

  This feature, despite its name, is part of the TCP/IP software stack, not part of CICS. It uses a user exit EZACICSE to provide the user ID under which transactions run. It does not support SSL.

Guideline: Do not configure CICS to accept this interface. It does not provide sufficient control over the flow of data from one address space to another, and its use could inappropriately declassify data.

- **Input:** ONC/RPC client.

  This is a CICS-supported interface for doing remote procedure calls over TCP/IP. It was the precursor for the CICS Web Support but is now little-used. It assigns a user ID by means of a user-replaceable module. It does not support SSL.

  Guideline: Do not configure CICS to accept this interface. It does not provide sufficient control over the flow of data from one address space to another, and its use could inappropriately declassify data.

- **Input:** CORBA client over IIOP.

  This is the basis of the CICS support for Enterprise Java Beans. It also uses a TCPIPSERVICE definition to specify the inbound TCP/IP connection, with an attribute of PROTOCOL(IIOP). The SSL(NO | YES | CLIENTAUTH) attribute is fully supported, but the only AUTHENTICATE options are NONE and CERTIFICATE. The latter requires a valid client certificate which maps to a valid user ID, otherwise the connection is rejected. AUTHENTICATE(NONE) obtains the user ID by means of a user-replaceable module.

  Guideline: CICS communicates with an IP address for this interface, and thus you must ensure that the CICS region ID and the SERVAUTH profile protecting the IP address have matching security labels. Specify SSL(CLIENTAUTH) and use certificates to identify the clients, as this provides better security than having the user-replaceable module supply a fixed identity. Ensure that the client user ID has a default security label matching that of the IP address, the CICS region ID, and the APPL profile.

- **Input:** Non-CICS address spaces communicating into CICS.

  Any non-CICS MVS address space can communicate with CICS using the MRO protocol, just as CICS-to-CICS communication can. The protocol is then known as the External CICS Interface (EXCI). This protocol passes the current thread's user ID (the one implied by TCBSENV) into CICS . It is also possible to specify the (unauthenticated) user ID directly on the callable interface to EXCI. This can be optionally controlled by the specification of SURROGCHK=YES option which performs a check on *userid*.DFHEXCI in the SURROGAT class.

  Guideline: Do not configure CICS to accept this interface. It does not provide sufficient control over the flow of data from one address space to another, and its use could inappropriately declassify data.

- **Input:** External Call Interface.

  This is a formal way for non-CICS programs to do distributed program links into CICS, emulating the EXEC CICS LINK command. It can use the LU6.2 (APPC) interface into CICS and is required to provide a user ID and password as previously described for the EPI.

  Guideline: Do not configure CICS to accept this interface. It does not provide sufficient control over the flow of data from one address space to another, and its use could inappropriately declassify data.

- **Input:** ECI from the CICS Transaction Gateway (CTG).

  The CTG also provides Java methods for invoking the ECI. The MVS version of the CTG uses EXCI to implement ECI.

  Guideline: Do not configure CICS to accept this interface. It does not provide sufficient control over the flow of data from one address space to another, and its use could inappropriately declassify data.

- **Input:** ECI over TCP/IP.

  This feature, new in CICS TS 2.2, allows socket-based clients to do distributed program links into CICS without requiring LU6.2. It uses a TCPIPSERVICE with PROTOCOL(ECI). To authenticate the user, specify ATTACHSEC(VERIFY) on the TCPIPSERVICE. The client must provide a user ID and password. To bypass authentication, specify ATTACHSEC(LOCAL).

  Guideline: CICS uses TCP/IP functions for this interface, and the Communications Server ensures that the CICS region ID has a security label matching that of the SERVAUTH profile that protects the IP address of the client. When CICS authenticates the user, RACF requires that the specified user ID has a default security label matching that of the APPL profile. ATTACHSEC(VERIFY), which is the default, should always be used.

Db2

`Where to find more information:` *Multilevel Security and Db2 Row-Level Security Revealed*, SG24-6480 in IMS in IBM Documentation (www.ibm.com/docs/en/ims).

`Requirement:` If you use Db2, you must install Db2 Version 8 or Version 9 to provide support for multilevel security. Earlier releases of Db2 do not support multilevel security and should not be used.

Db2 Version 8 and above allows you to protect selected Db2 objects with security labels. Figure 2 on page 111 shows these Db2 objects and their hierarchy.

- Subsystem or data sharing group
  - Database
    - Table space
      - Table
        - Column
        - Row
  - View
  - Storage group
  - Buffer pool
  - Plan
  - Collection
    - Package
  - Schema
    - Stored procedure, user-defined function
    - Java ARchive (JAR)
    - Distinct type
    - Sequence

Figure 2. Object hierarchy for Db2 objects that support security labels

The support for security labels is provided in two ways:

- For rows within a table, Db2 manages the security labels using SAF.
- For other Db2 objects, the Db2 RACF access control module (DSNXRXAC) uses SAF to provide support for security labels.

The Db2 RACF access control module

The Db2 RACF access control module (DSNXRXAC) is required to support security labels for some Db2 objects. Beginning with Db2 Version 8, this module is shipped with the Db2 UDB for z/OS in *prefix*.SDSNSAMP. For information on installing, customizing, and using DSNXRXAC, see the topics about *RACF Security for Db2* in Db2 (www.ibm.com/docs/en/db2).

Security labels for rows in a table

Many applications require row-level security within the relational database, so that user access can be restricted to a specific set of rows. Db2 Version 8 and above provides security with row-level granularity by allowing a security label to be assigned to each row in a table. To use security labels for the rows in a table, create a column in the table specifying the AS SECURITY LABEL clause. The column contains the security labels for the rows of the table. A table can have only one security label column, and the data type associated with the column must be CHAR(8) NOT NULL WITH DEFAULT. You can specify a security label column using either the CREATE TABLE or ALTER TABLE statement.

When a user attempts to perform an operation on a row, the results depend on the user's security label, the row's security label, the operation, and whether the user is allowed to write down. For detailed information about how these variables interact, see the topic about *Db2 Administration Guide* in Db2 (www.ibm.com/docs/en/db2).

Security labels for other Db2 resources

If the MLACTIVE option is active, profiles in the following RACF resource classes used by Db2 require security labels. Where two classes are listed, one is a member class and the other is its associated grouping class.

- DSNADM (administrative authorities)
- DSNR (access to Db2 subsystems)
- MDSNBP, GDSNBP (buffer pools)
- MDSNCL, GDSNCL (collections)
- MDSNDB, GDSNDB (database)
- MDSNJR, GDSNJR (JAR)
- MDSNPN, GDSNPN (plans)
- MDSNSC, GDSNSC (schema)
- MDSNSG, GDSNSG (storage groups)
- MDSNSM, GDSNSM (system privileges)
- MDSNSP, GDSNSP (stored procedures)
- MDSNTB, GDSNTB (tables, views, indexes)
- MDSNTS, GDSNTS (table spaces)
- MDSNUF, GDSNUF (user-defined functions)

You must specify a security label for every profile in these classes. You are responsible for ensuring that a proper hierarchy of security labels exists – in general the security label of an object higher in the object hierarchy should dominate the security labels of objects lower in the hierarchy. Figure 2 on page 111 shows the object hierarchy. For example, the security label of a table space should dominate the tables within it, and the security label of a database should dominate the tables within it. The system cannot enforce this hierarchy.

Requirement: You must install the Db2 RACF access control module (DSNXRXAC) in order to protect the resources in these classes with security labels. For information on installing, customizing, and using DSNXRXAC, see the topics about *RACF Security for Db2* in Db2 (www.ibm.com/docs/en/db2).

Table 16. Recommended security labels for Db2 profiles

Profiles	**Recommended security label**
Profiles in the DATASET class that protect Db2's underlying VSAM data sets	SYSHIGH, or the highest security label of the data stored within those tables
Profiles in the DSNADM class that protect SYSADM, SYSOPR, and SYSCTRL privileges	SYSHIGH, or the highest security label of the data contained within the specific Db2 subsystem
Profiles in the DSNR class	SYSMULTI

Security labels for Db2 subsystems

Assign a security label of SYSMULTI to Db2 address spaces. SYSMULTI allows an address space to communicate with callers having any security labels

Security labels for data sets holding Db2 data

Assign a security label of SYSHIGH to data sets that hold Db2 data.

Security labels for Db2 users

You must ensure that all Db2 users are authorized to the security labels that they require.

- Authorize all users who perform read operations to use a security label that dominates the data that they need to read.
- For users who perform write operations or a combination of read and write operations on Db2 objects, do one of the following for each Db2 object a user needs to write or read and write:
  - Authorize the user to use a security label that is equivalent to the security label of the Db2 object.
  - Authorize the user to use a security label that dominates the security label of the Db2 object, and give the user the write-down privilege.

Note: The following Db2 privileges allow only read operations:

- 0233 - ANY OF THE TABLE PRIVILEGES (Authority for DESCRIBE TABLE)
- 0267 - DISPLAY (Function, Procedure)
- 0062 - DISPLAY (System privilege)
- 0244 - DISPLAY ARCHIVE
- 0112 - DISPLAY BUFFERPOOL
- 0009 - DISPLAY PROFILE
- 0014 - DISPLAY RLIMIT
- 0099 - DISPLAYDB (Database)
- 0016 - MONITOR1
- 0017 - MONITOR2
- 0054 - REFERENCES
- 0240 - REPORT
- 0050 - SELECT

All other Db2 privileges allow write operations.

Access requirements for Db2 users

If a Db2 resource is protected by a profile in a RACF resource class that requires security labels (see “Security labels for other Db2 resources” on page 112 for a list of those classes), the authorization that a user requires to pass a discretionary access check for the resource depends on whether the MLS option is active:

- If the MLS option is not active, a user needs READ authorization to access the resource.
- If the MLS option is active, and the request is not a write request, a user needs READ authorization to access the resource.
- If the MLS option is active, and the request is a write request, a user needs UPDATE authorization to access the resource.

Review the access lists for the profiles in those classes to be sure that all users have sufficient authority.

Authority checking for users with installation sysadm or installation sysopr authority

In a multilevel-secure environment, the Db2 RACF access control module is not called for users with installation sysadm or installation sysopr authorization. Therefore, no mandatory access checks are performed or audited for these users. However, Db2 does enforce row level checking for these users.

DFSORT

DFSORT is a program that sorts, merges, and copies data sets on MVS operating systems. To use this product in a multilevel-secure environment, you must install the product as non-resident, that is, not in the link-pack area, and remove the SVC IGX00017 from LPALIB. (The SVC might have been renamed.)

Also, to eliminate some warning messages that will be generated as a result of the identified actions, you can set the following SORT options:

- EXCPVR=NONE, which prevents DFSORT from moving into supervisor state when reading and writing SORT data sets.
- SMF=NO, which indicates that SMF records are not to be produced during the execution of DFSORT. (DFSORT uses an SVC routine to produce SMF records.)

Information Management System (IMS)

Although IMS does not fully support multilevel security, you can use it in a multilevel-secure environment if you take care in its configuration.

A typical IMS configuration contains a control region (CR) , several message processing regions (MPRs), and possibly some batch message processing regions (BMPs). You can have multiple such configurations, each representing a related set comprising a control region, message processing regions, and BMPs.

Guidelines:

- Ensure that the user IDs that you assign to the control region and each message processing region within a configuration have the same security label. Do *not* use SYSNONE or SYSMULTI for this security label.
- Ensure that you configure IMS to make use of RACF application group name (AGN) security to control which BMPs can connect to each IMS configuration. Assign the same security label to the RACF profile that protects the application group name, for example in the AIMS class.
- Ensure that each BMP in the configuration runs with the same security label as the application group name (and thus, as the control region and message processing regions).
- When a user at a 3270 terminal signs on to the control region, IMS provides the TERMINAL identity to RACF during the signon process, along with an application name (APPL parameter).
  - Ensure that the TERMINAL profiles that protect terminals used for the control region have a security label matching that of the control region.
  - Ensure that the APPL profile that protects the application name provided by the control region has the same security label as the ID that you assign to the control region.
  - During the user's signon, RACF uses the security label from the TERMINAL profile to assign a default security label to the user's session. Ensure that the user has access to this security label. RACF also verifies that the user has access to the APPL profile, which verifies that the APPL and TERMINAL profiles have the same security label, ensuring that only appropriate users and terminals connect to the control region.
- Do not allow IMS regions to connect to Db2 subsystems that run with multilevel security active, because IMS does not pass sufficient information to Db2 to prevent inappropriate declassification of data. Instead, only allow IMS to connect to specific Db2 subsystems that run with the same security labels as the IMS region IDs.

Interactive System Productivity Facility (ISPF)

ISPF is a dialog manager that provides services to interactive applications. The authorized code in ISPF is executed if you use ISPF when in TSO Session Manager Mode. Therefore, you cannot use that particular facility of ISPF in a multilevel-secure system. To remove the authorized code from ISPF, you must remove two CSECTs — ISPSC93 and ISPSC94 — and then re-linkedit the load modules from which the CSECTs were removed. The load module names are IGC0009C and IGC0009D.

ISPF allocates some data sets dynamically (for example, log, list, and edit recovery). Users who have access to multiple security labels might have problems using ISPF, because by default ISPF does not take the user's current security label into account when allocating these data sets. RACF provides two sample exits in SYS1.SAMPLIB(RACEXITS) to help resolve these problems: ISPFX16 and CHANGENM. To allow proper allocation of ISPF temporary data sets, install these exits.

IBM MQ for z/OS

An IBM MQ configuration generally includes the following:

- One or more queue manager (QM) address spaces, possibly communicating as a queue-sharing group in a sysplex and making use of list structures in the coupling facility. Each queue manager runs under a specified RACF identity.
- One or more applications communicating with the queue manager address spaces to send (put) or receive (get) data from specific MQ queues managed by the queue managers. Each application runs under a specified RACF identity.
- One or more *mover* address spaces. Each mover communicates with one queue manager and handles communication between one or more queues and one or more IP addresses or APPC LU 6.2 addresses. Each mover runs under a specified RACF identity. In addition, for some configurations a mover can accept a message containing a user ID, and the queue manager performs some security checks using that user ID.

Guidelines: Follow these guidelines to minimize the risk of compromising security if you use IBM MQ in a multilevel-secure environment:

- Configure a separate queue manager (or set of queue managers) for each classification of data to be handled, and choose an appropriate security label for that data. Assign a RACF user ID to each queue manager, and assign the security label of the data to the user ID. Thus, each queue manager runs under an ID that has the same security label as the data it controls. Do not specify SYSNONE or SYSMULTI for any MQ-related user or resource profiles.
- The queue manager checks each connection between either an application or a mover using resources in the MQCONN class. Define profiles in the MQCONN class to protect these connections. Each profile should have a security label; assign the same security label that you assigned to the queue manager user ID. Assign the same security label to the RACF user IDs used for the application and mover address spaces. The authorization checking for the connections ensures that the application and mover address spaces operate under RACF user IDs that have matching security labels.
- Do not configure movers that use APPC LU 6.2.
- When configuring a mover to use TCP/IP, the Communications Server verifies that the mover's RACF ID has a security label that matches the SERVAUTH profile that protects the IP addresses that the mover communicates with. This ensures that the mover accepts or sends data only to IP addresses of the same classification as the mover, the queue manager, and the applications that deal with the queues.

Adding other server-based products

When adding server-based products not discussed in this document, you need to determine, from the product's documentation or the vendor, whether and to what extent the server supports multilevel security.

Servers that support multilevel security

When we speak of a server "supporting multilevel security", there are several common possibilities:

1. The server can run with a security label of SYSMULTI and can process work concurrently for users with different security labels and properly separate all the users and data based on their individual security labels. You should not attempt to assign a SYSMULTI security label to a server unless the server documentation or vendor explicitly states that the server can run properly using a SYSMULTI security label.

How to run the server in a multilevel-secure environment: Assign the server's user ID a default security label of SYSMULTI, and issue a PERMIT or CONNECT command to grant the server's user ID access to the SYSMULTI security label.

2. The server can run in a multilevel-secure environment, but must run with a specific security label other than SYSMULTI, and can access only that level of data. However, multiple instances of the server can run simultaneously on the system, each with its own security label. To support users with different security labels you would run multiple copies of the server and each user would choose the appropriate server to talk to based on the security labels of the server and user.

   How to run the server in a multilevel-secure environment: Assign multiple user IDs to the server, one for each instance of the server code that you plan to run. Assign each user ID an appropriate default security label other than SYSMULTI, and issue a PERMIT or CONNECT command to grant the user ID access to its security label. If the server creates any log files in the z/OS UNIX file system, or creates any MVS data sets containing log information, create separate UNIX directories or MVS data sets for each instance of the server to use, each with the same security label as the server user ID's default security label. If the server runs as a started task, create multiple MVS procedures, and a profile in the STARTED class for each one. If the server runs as a z/OS UNIX daemon, run the multiple instances of the daemon as started tasks.

3. The server can run in a multilevel-security environment, but must run with a specific security label other than SYSMULTI, and can access only that level of data. In addition, unlike case “2” on page 116, only one server can run on a system. If you need multiple servers, each must run on a different system.

   How to run the server in a multilevel-secure environment: Same as for case “2” on page 116.

4. Similar to case “3” on page 116, but only one copy of the server can run in a sysplex. This is more limited than cases “2” on page 116 and “3” on page 116, because you can only run one server, and must choose one specific security label for it to support.

   How to run the server in a multilevel-secure environment: Choose the security label that the server is to support. Assign the server's user ID a default security label of the security label you selected, and issue a PERMIT or CONNECT command to grant the server's user ID access to the security label.

Servers that do not support multilevel security

When we speak of a server that "does not support multilevel security", generally we mean that the server cannot run with a security label of SYSMULTI, and even if run with a specific security label other than SYSMULTI, cannot properly support multilevel security. If you have a server like this, you must not run it on a multilevel-secure system.

A server that meets any of these conditions does not support multilevel security:

- It authenticates users with RACROUTE REQUEST=VERIFY (rather than initACEE or a z/OS UNIX function such as pthread_security_np() or _passwd()) and does not specify APPL= on the VERIFY request, and specifies ACEE= to prevent RACF from anchoring the ACEE in the TCB.
- It does not communicate with its users via TCP/IP.
- It communicates with its users via TCP/IP, but does either of the following:
  - Uses RACROUTE REQUEST=VERIFY to authenticate its users and does not supply a SERVAUTH port of entry on the VERIFY request.
  - Uses a z/OS UNIX function such as pthread_security_np() or _passwd() to authenticate its users, but does not first use the __poe() service to associate the authentication request with a specific TCP/IP socket.

Neither the system nor the server can properly control which users, running with which security labels, can use the server. If you run the server, inappropriate declassification of data can occur. Do not run a server in a multilevel-secure environment if it meets any of these conditions.

Chapter 7. The certified configuration for the Common Criteria for z/OS 2.5

A Common Criteria (CC) certified system is one that has been evaluated according to the Common Criteria, an internationally recognized ISO standard (ISO 15408) for the assurance evaluation of IT products, and found to meet a specific set of requirements. Beginning with z/OS Version 1 Release 6, each release of z/OS has been evaluated and certified. For a summary of the certifications awarded for each release, see “History” on page 1.

The z/OS functionality, which is modeled on the Common Criteria for Operating System Protection Profile (OSPP) Version 4.3 (OSPP), dated September 27, 2022, and assurances have been evaluated and certified at an EAL4 level of assurance. The RACF functionality (which is also modeled on the OSPP) and assurances have been evaluated at an EAL5 level of assurance.

The system configuration and environment that the evaluation finds meet these requirements is referred to as the *certified system* or *certified configuration* in this topic. The certification report is published on the OCSI web site (www.ocsi.gov.it/index.php/elenchi-certificazioni/prodotti-certificati.html).

The following sections are intended to state requirements that must be fulfilled by the installation in order to run in a certified configuration. Whereas the previous chapters of this document describe an optional configuration for the system in order to provide multilevel security, this chapter documents requirements for the certified configuration.

The evaluation of z/OS did not cover all z/OS security functions, or all methods of achieving the required level of security. An installation can choose to use security functions that were not evaluated, or to use methods of achieving the required level of security that were not evaluated. If an installation makes this choice, it is no longer running the certified configuration, and must take responsibility for the security characteristics of the system.

The evaluation of z/OS did not cover all resources in the FACILITY class. In general, you can choose to use them without compromising the security of your system. However, you need to use them with care and be aware of the security implications. For example, some of the STGADMIN resources can allow reading of all data, and the BLSACTV.SYSTEM resource can allow viewing other users' data in storage. Define profiles that protect these resources with UACC(NONE) and give access only to highly trusted users.

If you are setting up a z/OS 2.5 system in accordance with the z/OS EAL4 and EAL5 security target, the information this document supersedes the information in other documents in the z/OS library.

Assumptions

Each secure system has areas where its security is based on an assumption, and therefore trust. The security of the certified configuration of z/OS, within the scope of the evaluation, is based on the following assumptions:

- z/OS must be delivered, installed, managed, and operated in a manner that maintains IT security policies and objectives.
- The processing resources of the certified configuration of z/OS are located within controlled access facilities that prevent unauthorized physical access.
- The hardware and software critical to security policy enforcement is protected from unauthorized physical modification.
- One or one competent individuals are assigned to manage the certified configuration of z/OS and the security of the information that it contains.
- The system administrative personnel are not careless, willfully negligent, or hostile, and follow and abide by the instructions that are provided by the z/OS documentation.

- Authorized users possess the necessary authorizations to access at least some of the information that is managed by the certified configuration of z/OS. These users act in a cooperating manner in a benign environment.
- Procedures exist to control and monitor changes to the operating system and the hardware configuration to ensure that all such changes are authorized and appropriate.
- All users of the system properly protect all access credentials, such as passwords or other authentication information.

Additional assumptions for EAL4/EAL5 conformance

If you are setting up a z/OS 2.5 system in accordance with the EAL4 or EAL5 security target, observe the following additional assumptions:

- Any other systems with which the certified configuration of z/OS communicates are under the same management control and operate under the same security policy constraints. The certified systems can be deployed in networked or distributed environments only if the entire network operates under the same constraints and resides within a single management domain.

  There are no security requirements that address the need to trust external systems or the communications links to such systems.
- Any other systems that share hardware resources (such as DASD or cryptographic processors with secret keys) are under the same management control and operate under the same security policy constraints.
- All connections to peripheral devices and other systems reside within the controlled access facilities unless they are protected by the TLS v1.2, TLS v1.3, or SSH protocol. The certified configuration of z/OS addresses only security concerns that are related to the manipulation of itself through its authorized access points. Internal communication paths to access points such as terminals or job entry stations are assumed to be adequately protected.
- The security that is provided by cryptographic functions often depends on the strength of the random number generation functions available on the system. The evaluation of the random number generation functions that are provided by z/OS assumes that the ICSF address space, the OpenSSH server address spaces, and any application address spaces that use System SSL or Application Transparent Transport Layer Security (AT-TLS) are restarted at least once per year.

z/OS security functions

The following security functions were subjects of the evaluation of z/OS. For detailed and authoritative statements about the security functions that are subject to evaluation, see *Security Target for IBM z/OS 2.5*, on the web (after the certificate is issued) at the OCSI web site (www.ocsi.gov.it/index.php/elenchi-certificazioni/prodotti-certificati.html).

Identification and authentication

A user can interact with the certified configuration in any of the ways that are listed in “Identification and authentication” on page 127.

Batch system access is authenticated by a combination of a user ID and password. For jobs submitted by an already authenticated user, no additional authentication is required for jobs that are running under the user’s ID. Interactive access for processes that require authentication uses one of the following means of authentication:

- A RACF user ID and a RACF password, RACF PassTicket, or (for some applications) a RACF password phrase
- An x.509 v3 digital certificate that is presented to a server application that uses System SSL or TCP/IP Application Transparent TLS (AT-TLS) to provide TLS-based client authentication, and then mapped using RACF to a RACF user ID
- SSH public key-based authentication.

Access control
z/OS provides the Resource Access Control Facility (RACF), a component of the Security Server feature, as an external security manager (ESM) that performs access control between users and resources that are protected by the discretionary access control mechanisms. RACF provides the security for standard z/OS resources, for UNIX file system and inter-process communication objects, and for network communication end-points and paths.

Communication security
z/OS provides networking functions with the Communications Server. The Communications Server provides support for network communication using the TCP/IP protocol suite.

z/OS provides the following additional security functions as part of the certified configuration:

- TLS communication
- IPsec communication can be used only in an EAL4 configuration. IPsec is not permitted for use in an OSPP 4.3 or EAL5 configuration.
- OpenSSH functions

Security management
z/OS provides a set of commands and options to manage the security functions of the system through RACF.

Various kinds of users can manage different aspects of system security:

- General z/OS security options are managed by security administrators who are identified through RACF attributes (such as SPECIAL or AUDITOR) applied to their user IDs.
- Management of RACF users and their security attributes is performed by security administrators. Management of RACF groups (and to some extent RACF users) can be delegated to group security administrators.
- Users can change their own passwords and password phrases, their default groups, and their user names, but not their user IDs.
- Auditors manage the parameters of the audit system (a list of audited events, for example) and can analyze the audit trail.
- Security administrators can define what audit records are captured by the system.
- Discretionary access rights to protected resources are managed by the owners of the applicable profiles (or UNIX objects) or by security administrators.

Auditing
The certified configuration of z/OS provides the System Management Facilities (SMF) to collect data that is required for auditing and accounting services. This function collects and records system-related, security-related, and job-related information that an installation can use for auditing compliant with OSPP.

Object reuse (for EAL4 or EAL5 certification only)
The certified configuration of z/OS provides object reuse control for the following objects:

- Memory objects
- z/OS data sets
- z/OS UNIX file system objects

  Note: For zFS files, the administrator must leave the new block security (NBS) option set to the default value (enabled) in the IOEFSPRM file. The NBS option must be enabled on any mount commands, and when attaching a multi-file system aggregate.
- z/OS UNIX IPC objects

Self-protection
The certified configuration of z/OS protects itself from tampering and bypassing of the security functions described by employing specific capabilities of the underlying z/Architecture® and z/OS operating system. These capabilities include Validated Boot for z/OS and SMP/E signed GIMZIP packages.

Cryptographic Support

The IBM z/16 server and z/OS 2.5 operating system provide cryptographic functions that are used by the ICSF subsystem. ICSF uses cryptographic hardware that is provided by the operational environment to provide and support cryptographic functions. For communication and remote access, TLS V1.2, TLS v1.3, and SSH version 2 are supported.

All key material that is used for cryptographic functions that are described in the Security Target occupy nonvolatile memory that is assigned to and is accessible only by the operating system and firmware. The keys are deleted when the power to the memory is removed. Nonvolatile keys are deleted by removing the abstraction that represents the key.

Supported hardware

The certified version of z/OS is running within a logical partition that is provided by a certified version of PR/SM, on the z/Architecture as implemented by the following hardware platform:

- IBM z16® with CPACF DES/TDES Enablement (Feature Code 3863) active, a CryptoExpress8 card configured as a coprocessor, and Firmware Bundle 22.
- Virtual Flash Memory (in support of Validated Boot for z/OS).

The following peripheral devices can be used with the certified system, preserving the security functionality:

- All z/OS supported terminals
- All z/OS supported storage and backup devices, such as:
  - Direct access storage devices (DASD), except RVA devices
  - Tape drives
- Any z/OS supported printer
- All z/OS supported Ethernet network adapters

The peripheral devices can be virtualized if the certified system runs in a logical partition.

For information about how to set up systems, software, and devices, see the product or hardware documentation.

Installation

The certified configuration of z/OS must be installed by a ServerPac installation. For information about installing z/OS, see *z/OS Planning for Installation*.

Documentation for the Certified Software Configuration

Documentation for the certified configuration can be found in the RACF HTTP PDF directory (public.dhe.ibm.com/eserver/zseries/zos/racf/pdf/zOSV2R5_Library-CC_Eval.zip) in the `zOSV2R5_Library-CC_Eval.zip` file.

The signatures for this file are:

```
SHA1:    a9b74d3946e369611a44df602b61f9c01c410261
SHA256:52cfa67733c4e0d2c507e282d7f69f56cc13c7e7a0bbfe2a9507b86d7daaf056
SHA512:7fce717802be413e0ad99615f2fe93d2834e3734cc36e3432a084b36e993046def428515ff8ac8eb4e2d6bf346a27f4385f16a3600c84521ea356cdcf14be0a0
```

The certified software configuration

The certified system consists of:

- z/OS Common Criteria Evaluated Base, which includes:
  - z/OS
  - Documentation APAR OA66552
  - Documentation APAR OA68391

Install z/OS and the PTF according to the directions delivered with the media, and configure them according to the directions in "Software restrictions in the certified configuration" on page 125.

Your installation can choose not to use any of the elements delivered within the ServerPac. However, you must install, configure, and use at least the RACF component of the Security Server optional feature and the ICSF component of Cryptographic Services.

The IEASYSxx parmlib OSPROTECT parameter specifies the operating system mitigation mode for unauthorized programs and users. OSPROTECT must be set to 1 or its equivalent value SYSTEM, which is the default.

Your system must be configured with address space layout randomization (ASLR) enabled for storage access. This setting is enabled through the following DIAGxx statement: ASLR(YES).

The certified software configuration:

- Includes an encrypted RACF database. For information about encrypting the RACF database, see https://www.ibm.com/docs/en/zos/2.5.0?topic=database-migrating-racf-vsam-data-set.
- Uses signed program objects. For the setup required to verify package signatures, see https://www.ibm.com/docs/en/zos/2.5.0?topic=guide-preparing-verify-signatures-gimzip-packages.
- Uses Validated Boot for z/OS. For the driving system and target system requirements for using Validated Boot, see https://www.ibm.com/docs/en/SSLTBW_2.5.0/com.ibm.zos.v2r5.e0zb100/DrReqsForValidatedBoot.htm.

  Note: To validate signatures, you require an IBM z16 with the appropriate microcode level.

Rules: You can add software that is not part of z/OS to your system if you follow these rules:

- The software must have the following characteristics:
  - Cannot run in supervisor state.
  - Cannot run APF-authorized.
  - Cannot run with key 0 through 7.
  - Cannot run with UID(0), or with authority to the FACILITY class resources BPX.DAEMON, BPX.SERVER, or BPX.SUPERUSER, or with authority to UNIXPRIV class resources.
- Do not replace any element in the ServerPac that provides security functions with other third-party products.
- Do not install system exits that run authorized (in supervisor state, APF-authorized, or with key 0 through 7), except for the exits listed in "MVS supplied exit routines" on page 62 and "RACF exit routines" on page 73.
- Do not add your own local checks to IBM Health Checker for z/OS because those checks run authorized. If you need to add your own checks, add them as unauthorized remote checks. For information about writing your own checks, see *IBM Health Checker for z/OS User's Guide*.
- Do not use the RACF authorized caller table (ICHAUTAB) to allow unauthorized programs to issue RACROUTE REQUEST=VERIFY (RACINIT) or RACROUTE REQUEST=LIST (RACLIST).

The following elements and element components cannot be used in the certified configuration because either they violate the security policies on which the certification was based, or they were not evaluated (due to complexity, scheduling, or other reasons). They must not be configured for use, or must be

deactivated, as described in "Restricting software not allowed in the certified configuration after you install" on page 122:

- Apache Server
- BCPii
- Bulk Data Transfer (BDT), all elements:
  - BDT (FMID HBD6602)
  - BDT File-to-File (FMID JBD6201)
  - BDT Systems Network Architecture (SNA) NJE (FMID JBD6202)
- DFS Server Message Block (SMB), from the element zFS File System
- Infoprint Server (FMIDs HMOS705, HNET7C0, HOPI7B0)
- IPSec:
  - Permitted in an EAL4 configuration.
  - Not permitted in an OSPP configuration.
  - Not permitted in an EAL5 configuration.
- JES3 (FMID HJS77B0)
- Kerberos
- LDAP
- NFS
- PKI Services
- SUDO

In addition, the following components cannot be used in the certified configuration:

- APPC/MVS
- The DFSMS Object Access Method for content management type applications
- The RACF remote sharing facility in remote mode. See "RACF" on page 125.
- The multi-level security environment.
- JES2 NJE communication through TCP/IP. JES2 NJE must use SNA or BSC in the certified configuration.
- JES2 Execution Batch Monitor (XBM) facility

Restricting software not allowed in the certified configuration after you install

About this task

When you reach the customization step of the ServerPac installation of z/OS, your system includes software that you cannot use in a certified configuration. That software includes:

- Entire data sets installed by the ServerPac
- Specific load modules in shared data sets (LINKLIB, CSSLIB, LPALIB, MIGLIB, SIEALNKE, and SIEAMIGE)

You must complete the tasks documented in this section to meet the requirements of the certified configuration.

Before you begin: You must complete the ServerPac installation of z/OS up to the customization step.

Subtask	Associated instructions (see. . .)
Restrict specified data sets	"Restrict specified installed data sets" on page 123

Subtask	Associated instructions (see. . .)
Restrict specified load modules in shared data sets	“Restrict specified load modules in shared data sets” on page 124
Restrict specified load modules installed into the HFS	“Restrict specified load modules installed into the HFS” on page 124
Disable APPC/MVS	“Disable APPC/MVS programs” on page 124
Restrict the IEHINITT utility	“Restrict the IEHINITT utility” on page 124
Restrict the IRRSMO00 callable service	“Restrict the IRRSMO00 callable service” on page 125

Restrict specified installed data sets

Table 17 on page 123 lists the data sets installed by the ServerPac that you cannot use in the certified configuration. You must restrict these data sets from being used.

Table 17. Installed data sets that you must restrict in the certified configuration

Dataset
SBDTCMD SBDTLIB SBDTLINK SBDTLPA SIATLIB SIATLINK SIATLPA SIATMIG

Steps for restricting installed data sets that cannot be used in the certified configuration

About this task

Before you begin: You need to complete the ServerPac installation of z/OS up to the customization step.

Perform the following steps to ensure that the restricted data sets are not used on your system.

Procedure

1. Ensure that none of the restricted data sets are in your link list or APF list.

   The ServerPac supplies a link list and APF list. If you are using the supplied lists, remove the specified data sets from them. If you are using your own lists, ensure that the specified data sets are not included in them. The lists are in either the PROG*xx* parmlib member, or the LINKLST*xx* and IEAAPF*xx* parmlib members, depending on which members your installation uses.

   _____

2. Ensure that none of the restricted data sets are referenced in your IEALPA*xx* parmlib member.

   _____

Results

When you are done, you have ensured that the data sets listed in Table 17 on page 123 are not used on your system.

Restrict specified load modules in shared data sets

Some software that you cannot use in the certified configuration is installed in data sets that also include allowed software. Use RACF program control to restrict the load modules in the shared data sets that cannot be used. For example, if RACF program control is already active on your system, issue the following commands:

```
RDEFINE PROGRAM EUV* ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM GLDSLAPD ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM ANF* ADDMEM('SYS1.LINKLIB') UACC(NONE)
RDEFINE PROGRAM AOP* ADDMEM('SYS1.LINKLIB') UACC(NONE)
RDEFINE PROGRAM API* ADDMEM('SYS1.LINKLIB') UACC(NONE)
RDEFINE PROGRAM IOEC* ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOED* ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEFDLL ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEFMPRT ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEFMTU1 ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEFMTU2 ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEFSKN ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEG* ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM IOEP* ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM DFSCNTL ADDMEM('SYS1.SIEALNKE') UACC(NONE)
RDEFINE PROGRAM DFSKERN ADDMEM('SYS1.SIEALNKE') UACC(NONE)
PERMIT IOEFSKN CLASS(PROGRAM) ID(ZFS) ACCESS(READ)
SETROPTS WHEN(PROGRAM) REFRESH
```

Do not add any users to the access lists for the PROGRAM class profiles that protect these programs.

Restrict specified load modules installed into the HFS

Some DFS/SMB modules that you cannot use in the configuration are installed into the HFS. To disable them, from a certifiedUNIX shell environment (for example, via telnet to z/OS UNIX System Services, or via the TSO/E OMVS command, or via BPXBATCH) issue the following commands:

```
chmod 600 /usr/lpp/dfs/global/bin/*
chmod 600 /usr/lpp/dfs/global/bin/IBM/IOEHM*
chmod 755 /usr/lpp/dfs/global/bin/zfsadm
chmod 1755 /usr/lpp/dfs/global/bin/IOEZADM
```

Disable APPC/MVS programs

Use RACF program control to disable the APPC programs ATBINMIG, ATBSDEPE, ATBSDFMU, ATBSDFCS, and ATBSDFM1, and ASCH (transaction scheduling) programs. Do not add any users to the access lists for the PROGRAM class profiles protecting these programs. For example, if RACF program control is already active on your system, you could issue the following commands:

```
RDEFINE PROGRAM ATB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKLIB') UACC(NONE)
RDEFINE PROGRAM ASB* ADDMEM('SYS1.MIGLIB' 'SYS1.LINKILIB') UACC(NONE)
SETROPTS WHEN(PROGRAM) REFRESH
```

Do not start the APPC or ASCH address spaces.

Restrict the IEHINITT utility

RACF does not perform authorization checking for tape volumes that the IEHINITT utility accesses when it issues the OPEN macro instruction. Use RACF program control to restrict the use of the IEHINITT utility to only authorized administrators. For example, if RACF program control is already active on your system, you could issue the following commands:

```
RDEFINE PROGRAM IEHINITT ADDMEM('SYS1.LINKLIB'//NOPADCHK) UACC(NONE)
SETROPTS WHEN(PROGRAM) REFRESH
```

Then, add any administrators who need to use IEHINITT to the access list for the IEHINITT profile.

Guideline: If your installation uses DFSMSrmm, administrators should use EDGINERS instead of IEHINITT.

Restrict the IRRSMO00 callable service

The use of the IRRSMO00 (Security Management Interface) must be restricted. To disable the IRRSMO00 callable service, define a discrete profile of the name IRR.IRRSMO00.DISABLE.XML in the XFACILIT class. This is a profile existence check only. Profile attributes such as the UACC and access list are not relevant.

The IRR.IRRSMO00.DISABLE.XML profile can be defined with these commands:

```
RDEFINE XFACILIT IRR.IRRSMO00.DISABLE.XML  UACC(NONE)
SETR RACLIST(XFACILIT) REFRESH
```

Software restrictions in the certified configuration

Many components of z/OS can be used in the certified configuration with restrictions. This section describes those restrictions and additional configuration that might be required.

Digital Certificate Access Server (DCAS)

The Digital Certificate Access Server (DCAS) is a host-based server that provides some distributed z/OS security services. The most common service is PassTicket generation. DCAS typically works with SSL-authenticated clients that provide logon services on behalf of users (typically workstation users) who want to log on to host applications. Together they can allow users to log on to host applications without having to know their passwords, and possibly even their user IDs. On the host, DCAS works with RACF to provide this function.

Rules: The following rules apply when DCAS is used in the certified configuration:

- DCAS configuration options must specify CLIENTAUTH LOCAL2.
- DCAS configuration options must specify SERVERTYPE CERTTYPE. SERVERTYPE ALLTYPES and SERVERTYPE USERIDTYPE must not be used.
- Network applications that use DCAS must be controlled by using the resource EZA.DCAS.*system-name* in the SERVAUTH class.

OpenSSH functions

The SSHD daemon can be used for OpenSSH functions.

Restrictions: The following restrictions apply to the SSHD daemon in the certified configuration:

- It must be configured to use protocol version 2 and one of the AES-based encryption suites.
- It must be configured to allow only password-based (including password phrase) authentication of users or public-key-based authentication of users with the public keys stored in RACF key rings. Host-based and public-key-based user authentication with the keys stored elsewhere cannot be used.
- It must be configured with privilege separation enabled. The goal of privilege separation is to prevent privilege escalation by containing any programming errors within the unprivileged processes. When running with privilege separation enabled, SSHD separates privileges by creating an unprivileged child process to deal with incoming network traffic. After successful authentication, this unprivileged process ends, and another process is created that has the privileges of the authenticated user. This user-privileged process handles daemon operations that do not require UID(0) privileges.
- It must be configured with the **ForwardAgent** option set to OFF, which is the default.
- It must be configured with the **AuthorizedKeysCommand** and **AuthorizedPrincipalsCommand** settings disabled.

RACF

Do not use the RACF remote sharing facility (RRSF) in remote mode. If you use RRSF in local mode, ensure that command direction cannot be used by taking one of the following actions:

- Ensure that the RRFSFDATA class is not active.

- Define the profile DIRECT.* in the RRSFDATA class with UACC(NONE) and no users in the access list.

Do not use multifactor authentication. You can disable the use of multifactor authentication by making the MFADEF class inactive.

Data set key protection

A data set encryption key must be defined as a protected key. For more information about protected keys, see *z/OS Cryptographic Services ICSF Administrator's Guide*.

A data set encryption key is considered destroyed with the destruction of the master key. The destruction of the master key results in the immediate invalidation of every key that is encrypted by the master key. As a result, the data set encryption key is no longer usable.

Transport Layer Security (TLS) processing

Transport Layer Security (TLS) can provide communications security in the certified configuration. TLS can be used directly by applications, or through the AT-TLS functions provided by the Communications Server.

Rules: The following rules apply to TLS processing in the certified configuration:

- For OSPP 4.3 compliance, the certified configuration can use the TLSv1.2 protocol. For EAL4 or EAL5 compliance, the configuration can use either TLSv1.2 or TLSv1.3.
- x.509 client and server certificates must be Elliptic Curve (ECC) NIST Curve 384 or 521, with SHA-384 or 512 signatures. These certificates must be managed by RACDCERT and their private keys must be stored in an ICSF PKA key data set (PKDS).
- Any application that uses TLS with client or server authentication must be configured for revocation checking through OCSP responses.
- Must use one of the cipher suites that are listed in "System SSL" on page 140.
- Any application that performs public key authentication by using digital certificates over TLS must be configured to use a RACF key ring. The key ring contains the application's certificate and private key and the supported Certificate Authority (CA) certificates that can be used to provide the certificates that the application will support. The use of gskkyman for this purpose is not part of the application evaluated configuration.
- Applications cannot enable TLS session renegotiations.
- Applications must disable session resumption.
- TN3270 for TSO/E.

Configuration options for the evaluated configuration

To use the ciphers that have been assessed as part of the evaluation for SSHD, the following settings in the configuration files are required that deviate from the default configuration:

sshd_config and ssh_config:

Ciphers:
Allow only the following ciphers: aes256-cbc, aes256-ctr, and AEAD_AES_256_GCM.

MACs:
Allow only the following functions: hmac-sha2-256, hmac-sha2-512, and AEAD_AES_256_GCM

sshd_config and ssh_config:

CiphersSource:
Needs to be set to ICSF to use ciphers implemented by CPACF

MACsSource:
Needs to be set to ICSF to use hash functions implemented by CPACF

moduli:
In each entry the Type field should be set to 5 and the Tests field should be set to 4 or 8. If the Test field is set to 4, the Tries field should be also set to 4.

For ECDSA key generation, use the **ssh-keygen -t ecdsa** command.

Advice for Users of the ICSF PKCS#11 Functions

Key objects should not be allowed to be used for wrapping/unwrapping as well as encryption/decryption. Therefore the key object attributes CKA_WRAP/CKA_UNWRAP should not be assigned together with the key object attributes CKA_ENCRYPT/CKA_DECRYPT. Access control services shall be used to protect access to key tokens and to PKCS#11 functions. For details of the access control services and the RACF profiles used see Controlling token access and key policy in *z/OS Cryptographic Services ICSF Writing PKCS #11 Applications*.

Boundary checking is enabled

In the certified system, applications that are written in a language for which the compiler supports boundary checking, such as a stack protection mechanism, must enable boundary checking. For example, if application code is written in C or C++ and the IBM C/C++ compiler is used, the STACKPROTECT option must be specified.

System configuration

The following sections describe the requirements for the certified system's configuration. Note that the following sections name optional parameters and controls that can be set in conformance with the certified configuration. The installation must fulfill the mandatory statements to maintain a minimum level of security that is required by the certified configuration of z/OS. The installation can choose between non-mandatory statements to implement a security policy that matches its own security requirements.

Identification and authentication

The evaluation covers the following ways in which users can interact with z/OS:

- As a TSO/E user
- As an operator at a console
- By submitting JES2 jobs
- As a UNIX user, including access through the UNIX shell or as a client of a UNIX-based server such as SSH
- As a Communications Server Policy Agent or Load Balancing Advisor client

Users can be identified and authenticated through the following methods:

- An alphanumeric RACF user ID and a password or password phrase
- An x.509 v3 digital certificate that is presented to a server application that uses System SSL or TCP/IP Application Transparent TLS (AT-TLS) to provide TLS-based client authentication. This certificate is mapped by the server application or by AT-TLS to a RACF user ID. For a list of the certified server software that supports this function, see “Authentication through client digital certificates” on page 130.
- An x.509 v3 digital certificate that is presented to a client application that uses System SSL or TCP/IP AT-TLS to provide TLS-based server authentication requires the server domain name to match either the server certificate Subject Alternate Name (SAN) DNS value or the Common Name within the subject distinguished name (DN).
- SSH public key-based authentication.

Any other authentication mechanism has not been subject to evaluation and violates the certified configuration of z/OS.

Rules: Follow these rules for identification:

- All human users of the certified configuration of z/OS must be assigned a unique user identifier (user ID).

- Operator consoles must be configured to require that operators log on. Specify that a logon is required for all consoles in the CONSOL*xx* member of SYS1.PARMLIB. For information about the CONSOL*xx* member, see *z/OS MVS Initialization and Tuning Reference*.

  Note: The Hardware Management Console (HMC) and support element console both allow entry of z/OS operator commands, but neither supports the MVS LOGON command. Therefore, there is no operator accountability when an operator uses these consoles.

  Guidelines: Take extra care to protect these consoles:

  – Use physical security (for example, place them in a locked room)
  – Limit distribution of passwords for these consoles
  – Use these consoles for z/OS operation only in an emergency

  For more information about the Hardware Management Console and the support element, see *Hardware Management Console Operations Guide*, GC38-0470.

For information about identification and authentication, see the following documents:

- *z/OS OpenSSH User's Guide*
- *z/OS Common Information Model User's Guide*
- *z/OS Communications Server: IP Configuration Guide*
- *z/OS Communications Server: IP Configuration Reference*
- *z/OS Integrated Security Services Network Authentication Service Administration*
- *z/OS JES2 Initialization and Tuning Guide*
- *z/OS Security Server RACF Security Administrator's Guide*
- *z/OS TSO/E Customization*
- *z/OS UNIX System Services Planning*
- *z/OS Network File System Guide and Reference*
- *z/OS Cryptographic Services System SSL Programming*

Configuring the session timeout value

To conform to the certified configuration of z/OS, the installation must establish a means for enabling and disabling a timeout value for periods of inactivity, including:

- TSO/E user address space inactivity
- Job wait time

In the SMFPRMxx parmlib member, the JWT parameter specifies the maximum amount of time that a job or TSO/E user address space is allowed to wait continuously. By default, JWT is set to 10 minutes. Evaluate this setting based on your installation's security policy.

For more information, see the description of SMFPRMxx in *z/OS MVS Initialization and Tuning Reference*.

Passwords and password phrases

To conform to the certified configuration of z/OS, the following password policy must be configured:

- 6 characters in length, at minimum
- No more than 5 failed attempts before revocation
- At least 1 numeric character, not in the first or last position

This policy can be expressed by using the following SETROPTS statement:

```
SETROPTS PASSWORD(REVOKE(5)
   RULE1(LENGTH(6:8) ALPHA(1,6) ALPHANUM(2:5))
   RULE2(LENGTH(7) ALPHA(1,7) ALPHANUM(2:6))
   RULE3(LENGTH(8) ALPHA(1,8) ALPHANUM(2:7)))
```

In addition, the evaluation covered the MIXEDCASE suboption, which specifies whether lowercase characters are allowed in the password.

For EAL4 or EAL5 certification only: Additional suboptions related to the PASSWORD option of SETROPTS are covered by the EAL4 or EAL5 evaluation, as follows:

- INTERVAL, which requires the user to change the password after the specified time period
- HISTORY, which tracks the specified numbers of recent passwords and prevents their reuse
- MINCHANGE, which specifies the minimum time before a user can change a password

An installation can choose to apply these additional settings if the basic requirements for the password, as stated, are met.

Rules: To help ensure that passwords are secure, follow these rules:

- Administrators that use ADDUSER to define a new user who will have a password must specify an appropriate password by using the PASSWORD operand, rather than allowing the password to default to the user's default group.
- Administrators changing a user's password by using ALTUSER must specify an appropriate password by using the PASSWORD operand.
- Administrators must not change a user's password by using the PASSWORD command.
- On a system that allows mixed-case passwords (SETROPTS PASSWORD(MIXEDCASE) is specified), administrators creating new USER profiles through the ADDUSER command should ensure that each user's initial password contains at least one lowercase character. If the administrator specifies an initial password that uses all uppercase characters, the user can log on using lowercase or mixed-case variants of the password until the user changes the password to one containing one or more lowercase characters. Administrators should ensure that all users with the CLAUTH(USER) attribute understand this rule.

Also, to conform with the certified configuration administrators should assign expired passwords to users, which is the RACF default, rather than using the NOEXPIRED operand to assign nonexpired passwords. This practice helps to ensure accountability for subsequent actions that are taken by those users because after the initial logon only the user (and not the administrator) knows the password.

In the certified configuration, users can also authenticate by using a password phrase, for applications that support it.

For information about passwords, password policies, and password phrases, see *z/OS Security Server RACF Security Administrator's Guide*.

PassTickets

In the certified configuration, PassTickets can be processed in the following ways:

- Key 0 callers can use an internal service located via the RCVT (RCVTPTGN) to generate PassTickets.
- Callers with appropriate authorization can use an external service invoked by R_ticketserv() or R_gensec() to generate and evaluate PassTickets.
- Java applications can use a JNI interface to R_ticketserv() and R_gensec() to generate and evaluate PassTickets.
- The Express Logon Facility (ELF) provided by the Communications Server allows a user to present an x.509 v3 digital certificate to the TN3270 server, which then maps the certificate to a RACF user ID and signs the user on to the host application using the mapped user ID and a PassTicket computed via the RCVTPTGN service.
- Applications can use the Digital Certificate Application Server (DCAS) provided by the Communications Server to generate a PassTicket for a specified user ID and application name, or to map a digital certificate for the server's client to a RACF user ID, and generate a PassTicket for that user and an application name.

  Restriction: The user ID requesting DCAS services must be authorized to the resource EZA.DCAS.*system_name* in the SERVAUTH class.

Authentication through client digital certificates

The following applications can accept client certificates and map them to RACF user IDs, as part of the client authentication process:

- SSL-aware applications
- The AT-TLS functions of the Communications Server

Rules:

- Applications that accept client certificates as part of the client authentication process must store the application private key and any needed certificate authority (CA) certificates and keys in the following way:
  - Using RACF key rings

Guideline: RACF certificate name filtering can be used to map multiple certificates to a single RACF user ID. In addition, a certificate can contain a host-ID mapping extension that assigns a RACF user ID. If you use these functions, take care that you do not lose user accountability if multiple certificates specify or map to the same user ID. Such "multiple user" mappings can be used only for accessing "public" data or other data not requiring user accountability.

The following application supports client authentication through digital certificates when using SSL/TLS sessions in the certified configuration:

- TN3270, using the Express Logon Feature (ELF) and the Digital Certificate Application Server (DCAS)

Started procedures

Rule: Started procedures must have RACF user IDs and group names.

To associate the names of started procedures with specific RACF user IDs and group names, an installation can use the following methods:

- The STARTED class (the preferred method).
- The started procedures table (ICHRIN03).

Guideline: Assign a protected user ID to a started procedure, unless other usages of the assigned user ID require a password. Protected user IDs are user IDs that have the NOPASSWORD, NOOIDCARD, and NOPHRASE attributes.

For information about started procedures and their security attributes, see *z/OS Security Server RACF Security Administrator's Guide*.

z/OS UNIX superuser privileges

An installation can choose to grant users the ability to obtain z/OS UNIX superuser privileges in several ways:

- Give the user a subset of superuser privileges by granting access to profiles in the UNIXPRIV class.
- Give the user the ability to become a superuser by granting access to the BPX.SUPERUSER profile in the FACILITY class.
- Assign the user UID(0).

Guideline: Assign UID(0) to a user only if you can't use one of the other methods.

For more information about superuser privileges and the methods of granting them, see *z/OS UNIX System Services Planning*.

Surrogate authority

A security administrator can choose to give a user *surrogate authority* over another user using the SURROGAT resource class. Surrogate authority allows the surrogate user to submit a job under the user

ID of the other user without specifying the other user's password, and to use the z/OS UNIX su command to switch to the other user's ID without specifying the other user's password.

For more information about surrogate authority, see *z/OS Security Server RACF Security Administrator's Guide*.

Access control

This topic describes the access control mechanisms as well as the objects that are subject to access control within the certified configuration of z/OS. Other access control mechanisms and other objects that are subject to access control were not evaluated.

The installation must be aware that the protection of the z/OS system must match the computing requirements of the system; therefore the installation should prepare a security plan that covers how the computing resources of z/OS are to be protected and how that protection matches the computing requirements of the system's users.

Data sets

Rule: The installation must protect all data sets with RACF. To operate in accordance with the certified configuration, the RACF SETROPTS option PROTECTALL(FAILURES) must be set to enforce this requirement.

Data set encryption is performed using AES in XTS mode with 256-bit keys.

An installation can choose to implement data set protection profiles under the standard RACF naming conventions, create a RACF group for each high-level qualifier that is not a user ID, and permit users to protect any data set that has that high-level qualifier by giving them CREATE authority in that group. Generally, though, administrators (with the SPECIAL or group-SPECIAL attribute) protect data sets. Users normally have USE authority in the group, and can create new data sets if they have ALTER authority through a generic profile, but do not have the ability to protect data sets by creating or changing profiles.

An installation can use the RACF naming convention table to set up and enforce a data set naming convention other than that used by RACF. The table can do the following:

- Supply a qualifier to be used as the high-level qualifier for authorization checking
- Convert data set names to RACF naming convention form for RACF use
- Convert names in RACF form to the installation's format for external display
- Enforce a naming convention by not allowing the definition of data sets that do not conform to an installation's rules
- Reduce RACF overhead by determining whether a data set is a user or group data set

An installation can create a naming convention table (module ICHNCV00), which RACF uses to check and modify (internally to RACF) the data set name in all commands and macros that process data set names. An installation can use the table to selectively rearrange data set names to fit the RACF convention without actually changing those names.

If an installation needs to protect data sets that have names consisting of a single qualifier, the installation can RACF-protect those data sets by issuing the SETROPTS command with the PREFIX operand.

An installation can also protect data sets that have names consisting of a single qualifier by using a generic profile *qualifier*.**, and activating the RACF EGN option with the SETROPTS command. This method of protection was not evaluated, but an installation can choose to use it.

An installation can choose to grant access to data sets by using conditions specified as follows

- WHEN(CONSOLE(*console-id* ...))
- WHEN(JESINPUT(*device-name* ...))
- WHEN(PROGRAM*(program-name* ...))
- WHEN(TERMINAL(*terminal-id* ...))

- WHEN(SERVAUTH(*SERVAUTH_profile name ...*))

The certified configuration of z/OS imposes no further restrictions on granting and denying user access on data sets. The installation can apply any policy it chooses.

For more information about protecting data sets, see *z/OS Security Server RACF Security Administrator's Guide*, and Chapter 3, “Establishing multilevel security,” on page 33.

DASD volumes

DASD volumes might or might not be managed by SMS.

SMS-managed DASD volumes

Use of the storage administration profiles in the FACILITY class was not evaluated, but you can choose to use them without compromising the security of your system. However you need to use them with care and be aware of the security implications. Define profiles protecting these resources with UACC(NONE) and give access only to highly trusted users.

For SMS-managed DASD volumes, the security administrator can define profiles in the FACILITY class with the high-level qualifier STGADMIN to control authorization to storage administrator and user commands. For information about the storage administration profiles in the FACILITY class, see *z/OS DFSMSdss Storage Administration* or *z/OS DFSMSdfp Storage Administration*.

DASD volumes that are not managed by SMS

For DASD volumes that are not managed by SMS, the security administrator can define the volumes to RACF using profiles in the DASDVOL class, and authorize users to perform maintenance operations (such as dump, restore, scratch, and rename) without having access to the data set profiles protecting the data sets on the volume. A user who does not have the necessary DASDVOL authority must have the necessary authority in the DATASET class to each of the data sets on the volume.

For more information about protecting volumes with the DASDVOL class, see *z/OS Security Server RACF Security Administrator's Guide*, and “Protecting DASD volumes” on page 44.

Tape volumes

Profiles in the TAPEVOL class protect tape volumes in the following circumstances:

- When the RACF TAPEVOL class is active and the IEHINITT utility is used to initialize a tape volume
- When the RACF TAPEVOL class is active, and SETROPTS NOTAPEDSN is in effect, and TAPEAUTHDSN=NO is specified in SYS1.PARMLIB(DEVSUP*xx*), and a user accesses data on the tape.

Special considerations for data on tape

A data file on tape can be protected in various ways, depending on RACF options and system options:

- The TAPEVOL class is active, and SETROPTS NOTAPEDSN is in effect, and TAPEAUTHDSN=NO is specified in SYS1.PARMLIB(DEVSUP*xx*):

  In this mode, the data is protected by the TAPEVOL profile for the tape, or is unprotected if no profile exists.
- The TAPEVOL class is inactive, and SETROPTS TAPEDSN is in effect, and TAPEAUTHDSN=NO is specified in SYS1.PARMLIB(DEVSUP*xx*):

  In this mode, the data is protected by the DATASET profile for the data set. However, protection might be ineffective for data sets with names longer than 17 characters, and the physical tape volume labels record only the last 17 characters of a data set name. Use this mode only if an active tape management system (DFSMSrmm) is keeping track of tape contents, and will reject the tape volume request if the data set name does not match the name that is specified by the user.
- The TAPEVOL class is active, and SETROPTS TAPEDSN is in effect, and TAPEAUTHDSN=NO is specified in SYS1.PARMLIB(DEVSUP*xx*), and TAPEVOL profiles contain RACF TVTOCs:

In this mode RACF verifies that the user specified the correct data set name. Then, security for the data set is provided by the DATASET profile for the data set.

- TAPEAUTHDSN=YES is specified in SYS1.PARMLIB(DEVSUP*xx*):

  In this mode, the system checks access based on the data set name specified by the user, regardless of the SETROPTS tape-related options in effect.

- TAPEAUTHF1=YES is specified in SYS1.PARMLIB(DEVSUP*xx*), and either SETROPTS TAPEDSN is in effect or TAPEAUTHDSN=YES is specified in SYS1.PARMLIB(DEVSUP*xx*):

  In this mode, in addition to the access check for the data set name specified by the user, the system performs an additional check for the first data set on the tape. This mode requires an active tape management system (DFSMSrmm), which provides the data set name for the first file on the tape.

For information about setting up and activating DFSMSrmm, see "Using DFSMSrmm" on page 49.

Devices

A user authorized to define profiles in the DEVICES class can use this class to control which users can allocate unit record devices, teleprocessing or communications devices, and graphics devices.

For more information about protecting devices, see *z/OS Security Server RACF Security Administrator's Guide*, and "Unit record, communication, and graphic devices" on page 58.

Terminals

The security administrator can protect terminals by defining profiles in the TERMINAL or GTERMINL class, and can use the TERMINAL operand on the SETROPTS command to define user access to terminals that are not protected by a profile in one of those classes.

The security administrator can also control access to terminals for groups of users. If the option NOTERMUACC is specified in a group profile, users within the group can only use terminals to which they are specifically authorized in the access list for the TERMINAL profile that protects the terminal.

The security administrator can restrict the use of a terminal to specific days and a time period within those days using the DAY and TIME suboptions on the WHEN option on the RDEFINE and RALTER commands.

For more information about protecting terminals, see *z/OS Security Server RACF Security Administrator's Guide*, and Chapter 3, "Establishing multilevel security," on page 33.

Operator commands

The security administrator can protect operator commands by defining profiles in the OPERCMDS class. Resources in this class are in the form *subsystem-name.command-name.operand* where *subsystem-name* is the name of the processing environment of the command (for example, JES2, RACF, MVS). *operand* is optional, depending on the command.

For more information about protecting operator commands, see *z/OS MVS Planning: Operations*, *z/OS JES2 Initialization and Tuning Guide*, and Chapter 3, "Establishing multilevel security," on page 33.

Programs

The RACF program control function can be used to restrict the ability of users to execute programs that reside in z/OS partitioned data sets or libraries. For information about program control, see *z/OS Security Server RACF Security Administrator's Guide*.

Rules: Follow these rules for authorized programs:

- The installation must protect all authorized program libraries from update or alter access by users other than the system administrators.
- The installation must protect the system configuration library from any modification by users other than the system administrators.

Consoles

The security administrator can control access to consoles by defining profiles in the CONSOLE class and activating the class. For more information about protecting consoles, see *z/OS Security Server RACF Security Administrator's Guide*.

z/OS UNIX file system objects

z/OS file system objects are always subject to discretionary access control. An installation can tailor the protection of individual z/OS UNIX file system objects by employing the following access control mechanisms:

- UNIX permission bits
- Access control list entries

The installation can control additional privileges and restrictions by defining profiles in the UNIXPRIV class protecting the resource SUPERUSER.FILESYS.ACLOVERRIDE. A user who has authority to this profile can override the access control defined by the access control lists for z/OS UNIX file system objects.

For more information about protecting z/OS UNIX resources, see *z/OS UNIX System Services Planning* and "z/OS UNIX System Services" on page 87.

z/OS UNIX IPC objects

z/OS UNIX IPC objects are always subject to discretionary access control. The permission bits associated with the IPC object define the discretionary access to those objects. The permission bits are determined by the creator of the IPC object and are saved in memory by the z/OS UNIX kernel.

For more information about protecting z/OS UNIX IPC objects, see *z/OS UNIX System Services Planning* and "IPC objects" on page 23.

Global resource serialization services (EAL4 or EAL5 only)

Evaluated only for EAL4 or EAL5 configurations, it is possible for the global resource serialization ENQ and GQSCAN services and the corresponding 64-bit services ISGENQ and ISGQUERY to be used as covert communication mechanisms to declassify data. These services can be issued by unauthorized callers. Both ENQ and ISGENQ take character data as input in serializing abstract resources. The GQSCAN and ISGQUERY macros enable programs to scan for resource requests across the global resource serialization complex. Therefore ENQ and ISGENQ are potential transmit mechanisms where GQSCAN and ISGQUERY could be used for receiving, and the abstract resource names could be the data.

The DISPLAY GRS system command can internally issue a GQSCAN. Because this command runs authorized, global resource serialization processing does not check the FACILITY class profile for authorization to issue this GQSCAN. In the certified configuration, the installation must protect the DISPLAY GRS operator command and the consoles from which it can be issued. For information about protecting operator commands and consoles, see *z/OS Security Server RACF Security Administrator's Guide*.

For more information on global resource serialization, see *z/OS MVS Planning: Global Resource Serialization*.

Common Information Model data (EAL4 or EAL5 only)

Evaluated only for EAL4 or EAL5 configurations, access to the CIM server is controlled by the profile CIMSERV in the RACF resource class WBEM. For information about setting up security for CIM, see *z/OS Common Information Model User's Guide*.

RACF resource classes

The certified configuration covered the use of the RACF resource classes listed in Table 18 on page 135. The installation can use these classes to implement protection of the respective objects.

The use of all other RACF classes was not subject to evaluation. However, the installation can choose to use additional classes.

Table 18. RACF resource classes in the Common Criteria certified configuration

Class	Function
CFIELD	Defines the installation's custom fields.
CONSOLE	Controls access to MCS or SMCS consoles. Also controls conditional access to other resources for commands originating from an operator console.
CRYPTOZ	Controls the use of PKCS #11 tokens.
DASDVOL	Controls access to DASD volumes for maintenance operations.
DEVICES	Controls access to unit record devices, teleprocessing or communication devices, and graphic devices.
DIGTCERT	Used to register x.509 v3 digital certificates in the RACF database.
DIGTCRIT	Used to define additional mapping criteria for the interpretation of x.509 v3 digital certificates presented by clients when the certificates are not specifically registered in the RACF database, and to assign a RACF user ID to the client's session as part of the client authentication process.
DIGTNMAP	Used to define the primary mapping rules for the interpretation of x.509 v3 digital certificates presented by clients when the certificates are not specifically registered in the RACF database, and to assign a RACF user ID to the client's session as part of the client authentication process.
DIGTRING	Implements key rings for servers or users in the RACF database, holding information about allowable Certificate Authority (CA) certificates and private keys for locally defined personal certificates and local signing certificates.
FACILITY	Used by various components of z/OS to manage specific privileges that could be assigned to users so that they do not need the SPECIAL attribute or the z/OS UNIX superuser privilege. Only a few profiles in this class are relevant for the evaluation.
FSACCESS	Controls access to z/OS UNIX file systems.
GDASDVOL	Grouping class for DASDVOL
GLOBAL	Defines the entries in the global access checking table.
GTERMINL	Resource group class for TERMINAL class.
GXFACILI	Resource group class for the XFACILIT class.
JESINPUT	Port of entry class to control which JES2 input devices a user can use to submit batch work to the system.
JESJOBS	Controls the submission and cancellation of jobs by job name.
JESSPOOL	Controls access to job data sets on the JES spool (that is, SYSIN and SYSOUT data sets).
KERBLINK	Used to map user identities of local and foreign user IDs.
LOGSTRM	Controls access to system logger resources, such as log streams and the coupling facility structures associated with them.

Table 18. RACF resource classes in the Common Criteria certified configuration (continued)	
Class	**Function**
NODES	Controls the following on MVS systems: • Whether jobs are allowed to enter the system from other JES2 nodes • Whether jobs that enter the system from other nodes have to pass user identification and password verification checks
OPERCMDS	Controls who can issue operator commands.
PROGRAM	Controls access to programs (load modules).
PTKTDATA	Used to configure PassTicket processing.
RDATALIB	Used to perform authorization checking for the R_datalib callable service.
SDSF	Controls the use of authorized commands in the System Display and Search Facility (SDSF).
SERVAUTH	Controls a client's authorization to use a server or to use resources managed by the server.
SERVER	Controls the validity of servers for the application environment.
SMESSAGE	Controls to which users a user can send messages (TSO only).
STARTED	Assigns an identity to a started task during the processing of an MVS START command. An alternative to the started procedures table (ICHRIN03).
TAPEVOL	Controls access to tape volumes.
TERMINAL	Controls access to terminals (TSO/E).
TSOPROC	TSO logon procedures.
UNIXPRIV	Used to grant z/OS UNIX privileges.
VTAMAPPL	Controls who can open ACBs from non-APF authorized programs. This prevents programs from counterfeiting login screens.
WRITER	Controls the user of JES2 printers and outbound NJE processing.
XFACILIT	Similar to the FACILITY class, but supporting longer resource and profile names (up to 246 characters, while the FACILITY class supports up to 39 characters).

RACF options

Rules: To conform with the certified configuration, the security administrator must configure some RACF options (using the RACF SETROPTS command):

- The security administrator must configure RACF with the following options:
  - CATDSNS(FAILURES)
  - NOCOMPATMODE
  - ERASE(ALL)
  - GENERIC(*)
  - PROTECTALL(FAILURES)
  - CLASSACT(TEMPDSN)
  - JES(BATCHALLRACF)

– PASSWORD (see “Passwords and password phrases” on page 128 for information about how to specify this option)

All other RACF options are optional.

For information about RACF options, see *z/OS Security Server RACF Security Administrator's Guide*. For information about the SETROPTS command, see *z/OS Security Server RACF Command Language Reference*.

Auditing

The evaluation covered various audit settings and various events generated by the system. The installation can choose to apply auditing for protected resources by applying the settings and choosing events described in the following sections.

For more information about auditing, see Chapter 4, “Auditing a multilevel-secure system,” on page 99 and *z/OS Security Server RACF Auditor's Guide*.

Protecting audit data

Rules: In the certified configuration you must follow these rules for protecting audit data:

- The installation must define at least two SMF data sets.
- The installation must protect the SMF and dump data sets holding the audit trail and grant access only to authorized users.

Guideline: To ensure full accountability, configure SMF to enter a system wait state when no more audit records can be written. Do this by specifying the following options in the SMFPRM*xx* member of SYS1.PARMLIB

- NOBUFFS(HALT)
- LASTDS(HALT)

For information about coding these options, see *z/OS MVS Initialization and Tuning Reference*.

To recover data in the SMF buffers that have not been written to disk when SMF enters the system wait state, take a system dump that includes the SMF address space. You can use the Interactive Problem Control System (IPCS) subcommand SMFDATA to read the dump, extract the data in the SMF buffers, and write it to an SMF data set. For information on the SMFDATA command, see *z/OS MVS IPCS Commands*.

Audit settings

A user with the AUDITOR attribute can further choose to tailor the auditing using the following RACF options:

- AUDIT or NOAUDIT (for each profile class)
- CMDVIOL or NOCMDVIOL
- LOGOPTIONS (for each profile class)
- OPERAUDIT or NOOPERAUDIT
- SAUDIT or NOSAUDIT

The auditor sets these options using the SETROPTS command.

Audit configuration can be delegated at the group level by giving the group-AUDITOR attribute to a user.

The installation can choose to include auditing for the RACF classes listed in Table 18 on page 135, or for other classes that the installation chooses to use. To do this, issue the SETROPTS LOGOPTIONS command with the appropriate options.

The installation can choose to audit changes in the protection of all or individual classes by issuing the SETROPTS AUDIT command with the appropriate options.

The installation can choose to assign either the trusted or privileged attribute to a started procedure. The trusted attribute allows auditing of resources accessed by the started procedure (though by default no such auditing occurs). In contrast, the privileged attribute bypasses auditing of resource access for the started procedure.

Capturing and processing security-relevant audit events

The following audit events were covered by the evaluation. The installation can choose to include them in the audit trail.

- Start-up and shutdown of the audit functions

  RACF creates a type 81 SMF record upon initialization.

  For information about auditing this event, see Chapter 4, "Auditing a multilevel-secure system," on page 99, *z/OS Security Server RACF Auditor's Guide*, and *z/OS MVS System Management Facilities (SMF)*.
- Reading of information from the audit records; unsuccessful attempts to read information from the audit records

  The installation can obtain these audit records by adding AUDIT(ALL(READ)) to the RACF data set profile that protects the SMF data sets.
- All modifications to the audit configuration that occur while the audit collection functions are operating

  The installation can choose to enable auditing on the following resources for operator commands that might affect the audit configuration:

  – MVS.SETSMF.SMF
  – MVS.SET.SMF
  – MVS.SWITCH.SMF

  The installation can further choose to apply additional access and audit controls on the SMFPRMxx parmlib members that hold the SMF configuration statements.

  The installation should be aware that the auditor role is by definition in control of the system audit settings and processing. The installation should apply the appropriate organizational or other controls, if accountability of the auditor's actions is desired.
- All requests to perform an operation on an object covered by the security policy

  The installation can obtain these audit records by enabling LOGOPTIONS for the resource class that protects the object.

  See also "Audit settings" on page 137.
- Rejection or acceptance by z/OS security functions of any tested secret; all use of the authentication mechanism; all use of the user identification mechanism, including the identity that is provided during successful attempts; success and failure of binding user security attributes to a subject (for example, success and failure to create a subject)

  Failure is indicated by type 80 SMF records. Success is indicated by the following types of SMF records:

  – SMF type 30 records for batch jobs and TSO
  – SMF type 30 and SMF type 80 for rlogin and telnet
  – SMF type 80 for operator logon

  The installation can choose to adapt the SMF settings accordingly.

  For more information about these audit records, see *z/OS MVS System Management Facilities (SMF)* and *z/OS Security Server RACF Macros and Interfaces*.
- All modifications of the values of security attributes; modifications of the default setting of permissive or restrictive rules; all modifications of the initial value of security attributes; all modifications to the values of data that is related to z/OS security functions; all attempts to revoke security attributes; modifications to the group of users that are part of a role.

The installation can obtain these audit records by activating the SAUDIT option of RACF.

- Every use of the rights of a role

  The installation can obtain these audit records by enabling the SAUDIT and OPERAUDIT RACF options.
- User and Group management events. Successful and unsuccessful add, delete, modify, disable, enable, and credential changes.

Additional audit events (EAL4 or EAL5 only)

The following audit events are included for EAL4 or EAL5 evaluation purposes. The installation can choose to include them in the audit trail.

- Actions taken due to audit trail storage shortage

  The installation can instruct the operators on how to respond to messages that indicate that the SMF data sets are becoming full. The operators can choose to run the SMF dump program in order to preserve the audit records.
- Actions taken due to the audit storage failure

  See "Audit settings" on page 137 for SMF configuration statements that cause the system to enter a wait state when no more audit records can be written.

  For more information about handling this event, see *z/OS MVS System Management Facilities (SMF)* and Chapter 4, "Auditing a multilevel-secure system," on page 99.
- All attempts to export information

  This event is recorded by SMF type 80 records for DEFINE events (event code 7) associated with TAPEVOL profiles, combined with SMF type 80 records for ACCESS events (event code 2) for the labeled data that is being exported.
- Attempts to override human-readable output marking

  The installation can choose to apply audit controls to the PSF.DPAGELBL resource in the PSFMPL class to record that specific event.

  For more information about auditing this event, see *z/OS Security Server RACF Auditor's Guide* and "Auditing PSF" on page 67.
- All attempts to import user data, including any security attributes

  This event is recorded by SMF type 80, subtype 2 records associated with TAPEVOL profiles.
- Execution of the test of the underlying machine and the results of the test

  The certified configuration provides no automated audit capabilities for the test that is described in "Abstract machine testing" on page 144. The installation can choose to record the execution of the test together with the results by using appropriate means.
- Changes to the time

  The installation can obtain these audit records by specifying audit options for the profile that protects the RACF resource MVS.SET.TIMEDATE.

Roles

The installation can choose to grant a user ID, or protected ID, any of the following attributes, combination of attributes, or no attributes at all:

- SPECIAL
- Group-SPECIAL
- AUDITOR
- Group-AUDITOR
- OPERATIONS

- ROAUDITOR

For example, the installation can choose to grant the AUDITOR privilege only to users not having the SPECIAL privilege. Likewise, the installation should keep user privileges such as the SPECIAL or OPERATIONS user attributes under control and assign these attributes based on operational needs.

Secure communication

The certified configuration of z/OS provides means of secure communication between systems sharing the same security policy.

Support for network communication is provided by the following components of z/OS:

- The Communications Server provides support for network communication using the TCP/IP protocol suite. It supports both IPv4 and IPv6 for IP.
- The System SSL component provides SSL/TLS functions.

The cryptographic operations performed by the Communications Server and System SSL make use of the hardware cryptography when appropriate. Therefore, ICSF must be active and a CryptoExpress 8 card must be operational.

z/OS Communications Server

z/OS Communications Server provides the following security functions, which the installation can use to protect communications in a network:

- Access control for the IP stack, ports, and port ranges.
- Application Transparent Transport Layer Security (AT-TLS), which uses Secure Sockets Layer (SSL) and Transport Layer Security (TLS) to set up a trusted channel to another trusted IT product through a potentially insecure network, in a way transparent to the application. The selectable algorithms can be limited by configuring a subset of allowable algorithms at the server. Servers can support encryption using AES with 256-bit key length. AT-TLS supports all cipher suites that are supported by System SSL.

  Rule: Remote terminal access to z/OS, when protected by SSL/TLS, must use the TLSv1.2 or TLS v1.3 protocol. Accordingly, the installation should provide users with guidance on how to configure their TN3270 terminal emulator programs.

- **Valid for EAL4 only:** IPsec security associations. z/OS Communications Server can be configured to establish IPsec security associations at the IP layer. Packets that are transmitted between security association endpoints can be encrypted, authenticated, or both. The required cryptographic operations are performed by using configured algorithms. The installation can choose the following approaches:
  - Either pre-shared secrets or certificate-based peer authentication can be used. If the installation chooses to use pre-shared secrets for authentication of the communication partners, they must conform to the rules outlined in “Passwords and password phrases” on page 128.
  - Key exchange and perfect forward secrecy can be implemented by using Diffie-Hellman group 20 (384-bit Random ECP) and DH Group 21 (521 bit).
  - Encryption of network data can be implemented through AES-CBC or AES-GCM with 256-bit keys (only available using the ESP protocol).
  - The authenticity and integrity of the network data can be ensured by using HMAC-SHA2-256-128, HMAC-SHA2-384-192, HMAC-SHA2-512-256, or AES-XCBC-MAC-96.

For information about using and setting up SSL, see *z/OS Communications Server: IP Configuration Guide*.

System SSL

The System SSL component of the z/OS Cryptographic Services element provides support for the System Sockets Layer (SSL) and Transport Layer Security (TLS) protocols. This support is required by applications that must use SSL directly through System SSL or by taking advantage of the AT-TLS functions of the Communication Server. SSL/TLS can be used to set up a trusted channel to another system through a potentially insecure network.

In the certified configuration, TLS processing must be configured as described in the following list. A variable within parentheses identifies the attribute or environment variable that controls the setting.

For an OSPP 4.3 evaluation, the certified configuration can use the TLSv1.2 protocol. For an EAL4 or EAL5 evaluation, the certificated configuration can use either TLSv1.2 or TLSv1.3.

- For TLS V1.2, the following is allowed in the evaluated configuration:
  - Ciphers (GSK_V3_CIPHER_SPECS_EXPANDED):
    - C024 - TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
    - C02C - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
  - Supported elliptic curve groups (GSK_CLIENT_ECURVE_LIST, GSK_SERVER_ALLOWED_KEX_ECURVES):
    - 0024 - secp384r1
    - 0025 - secp521r1
  - Digital signatures (GSK_TLS_SIG_ALG_PAIRS, GSK_TLS_CERT_SIG_ALG_PAIRS):
    - 0403 - SHA-256 with ECDSA
    - 0503 - SHA-384 with ECDSA
    - 0603 - SHA-512 with ECDSA
- For TLS V1.3, the following is allowed in the evaluated configurations:
  - Ciphers (GSK_V3_CIPHER_SPECS_EXPANDED):
    - 1302 - TLS_AES_256_GCM_SHA256
  - Supported elliptic curve groups (GSK_CLIENT_ECURVE_LIST):
    - 0024 - secp384r1
    - 0025 - secp521r1
  - Supported key shares (GSK_SERVER_TLS_KEY_SHARES, GSK_CLIENT_TLS_KEY_SHARES):
    - 0024 - secp384r1
    - 0025 - secp521r1
  - Digital signatures (GSK_TLS_SIG_ALG_PAIRS, GSK_TLS_CERT_SIG_ALG_PAIRS):
    - 0403 - SHA-256 with ECDSA
    - 0503 - SHA-384 with ECDSA
    - 0603 - SHA-512 with ECDSA
- Certificate processing:
  - x.509 client and server certificates must be Elliptic Curve (ECC) NIST Curve 384 or 521, with SHA-384 or 512 signatures. These certificates must be managed by RACDCERT (connected to a RACF key ring) and with private keys that are stored in the ICSF public key data set (PKDS). The certificate to be used during the handshake is identified by its label (GSK_KEY_LABEL), or is the default certificate for the RACF key ring. The RACF key ring must also contain any certificate authority certificates that are needed to validate the partner certificate when requested. Use GSK_KEYRING_FILE to identify the RACF key ring.
  - Any certificates in use must conform to RFC 5280 (GSK_CERT_VALIDATION_MODE=5280).
  - The client certificate that contains the extended key usage extension must specify the client authentication capability.
  - The server certificate that contains the extended key usage extension must specify the server authentication capability.
  - Any application that uses TLS with client or server authentication must be configured for revocation checking through OCSP responses. For information about configuring OCSP, see the topic “Enabling OCSP support” in *z/OS Cryptographic Services System SSL Programming*.

  - The client application must enable server domain name validation (GSK_REFERENCE_ID_DNS). For information about setting up and using server domain name validation with System SSL:
    - For z/OS 2.5, see APAR OA63164.
    - For z/OS 3.1 and later, see *z/OS Cryptographic Services System SSL Programming*.
  - The server application that requests client authentication must require a certificate from the client (GSK_CLIENT_AUTH_NOCERT_ALERT=ON) and use RACF to map the certificate to a RACF user ID. For more information about user ID mapping, see the topic Using a hostIdMappings extension in *z/OS Security Server RACF Security Administrator's Guide*.
- Disable server renegotiation (GSK_RENEGOTIATION=DISABLED).
- Disable session resumption by setting the session timeout to 0 (GSK_V3_SESSION_TIMEOUT=0) and disabling server session tickets (GSK_SESSION_TICKET_SERVER_ENABLE=OFF).

IP network applications

In the certified configuration, z/OS provides the following privileged network applications:

- TN3270 (from the Communications Server)
- OpenSSH

The administrator must ensure that no additional privileged network services are accessible over the network. The certified configuration allows the use of nonprivileged network applications.

Rules: In addition to the configuration information in *z/OS Communications Server: IP Configuration Guide*, follow this rule:

- Do not configure the FTP daemon to allow anonymous access.

OpenSSH

The SSH protocol can be used to set up a trusted channel to another system through a potentially insecure network. z/OS provides an SSHD daemon that supports the SSHv2 protocol, and the following commands, to allow remote users to perform work on a z/OS system:

- ssh, to establish a z/OS UNIX shell environment
- scp, to perform remote file copying operations
- sftp, to perform file transfer operations

Within the evaluated configuration, SSH supports encryption that uses:

- AES-CTR
- AES-CBC and AES-GCM with 256-bit key length
- hmac-sha2-256
- hmac-sha2-512
- AEAD-based MAC.

An SSH trusted channel should be rekeyed after no more than 2**28 bytes are transferred or the one-hour connection time threshold is met. The SSH rekeying is controlled through RekeyLimit. The first RekeyLimit value is the number of bytes that can be transferred before a rekeying is performed. The second RekeyLimit value is the maximum number of seconds that can pass before the rekeying is performed.

Within the evaluated configuration, SSH supports the use of ecdsa-sha2-nistp384 and ecdsa-sha2-nistp521 for server authentication and host-based authentication.

Key exchange protocol

The key exchange method specifies how one-time session keys are generated for encryption and for authentication, and how the server authentication is done. The Diffie-Hellman Key Exchange is a method for exchanging secret keys over a non-secure medium without exposing the keys.

The key exchange methods in OpenSSH allowed in the evaluated configuration are:

- ecdh-sha2-nistp384
- ecdh-sha2-nistp521

Host-based firewall

An installation should minimize the number of open ports on the system.

At a minimum, the evaluated configuration requires open ports for the following functions:

- SSH (for example, port 22)
- Telnet over TLS (a client-specified encrypted port)

An installation's applications might require additional ports to be opened.

For information about configuring a firewall for z/OS, see the IP Security chapter of *z/OS Communications Server: IP Configuration Guide*.

Import and export of data to tape or disk

When a z/OS data set that contains a zFS file system is exported (dumped and restored by DFSMSdss), the security information that is associated with the files and directories in the zFS file system are exported. This happens because the security information is included as extended attributes in the nodes of the file system.

Steps for exporting data that has security information

About this task

Perform the following steps to export data that has security information to tape or disk. You must be familiar with DFSMSdss storage administration and zFS file systems, and be authorized to run the ADRDSSU program.

Procedure

1. Place the data to be exported in a zFS data set.

   _____
2. Ensure that you have READ access to the zFS data set through its DATASET profile.

   _____
3. Ensure that the DATASET profile for the zFS data set specifies AUDIT(ALL)

   _____
4. Specify PROTECT=YES on the output DD for ADRDSSU to ensure the creation of a TAPEVOL profile. Or, specify a previously created profile using the VOL=SER= parameter.

   _____
5. Dump the zFS data set by using DFSMSdss (program ADRDSSU). Do not specify the ADMIN keyword. By omitting the ADMIN keyword, you force ADRDSSU to perform a RACF authorization check against the zFS data set, which helps to ensure proper auditing of the export operation.

   _____

Results

You exported data with security information to a tape or disk.

System time

Rule: The installation must ensure that the system time is correct.

For example, the system programmer can issue the **SET TIME** command to set the time correctly. For more information, see the description of the SET TIME command in *z/OS MVS System Commands*.

Residual data

Residual data sets on DASD

Rule: For an EAL4 or EAL5 configuration, the installation must ensure that the RACF ERASE(ALL) option is active.

Residual data on tape

The installation can ensure that tapes in the scratch pool are properly erased when they are no longer required. For information about ensuring that tapes are erased before they are scratched, see "Protecting data on tape" on page 45 and *z/OS DFSMSrmm Implementation and Customization Guide*.

Rule: Configure DFSMSrmm to force tapes to be erased before they are returned to the scratch pool. To do so, use the DFSMSrmm SECCLS parmlib option for parmlib member EDGRMMxx, specify the data set masks to be used, and specify the erase option. For more information, see "Using DFSMSrmm" on page 49.

Abstract machine testing

All z/OS machines undergo testing before they are shipped, and they perform consistency checks and other self-testing during operation. But if you think there might be a problem with a machine, and want to verify that the underlying hardware operates in conformance with requirements of the z/Architecture and thus with the requirements that the certified configuration imposes on the hardware, you can request that IBM service personnel perform a conformance and function test.

Operating system updates

Software updates and maintenance ("service") for z/OS is provided by IBM on a continual basis as SMP/E installable PTFs.

The installation uses signed code packages for all software deliverables from IBM and other vendors, including portable software instances for z/OSMF Software Management, ServerPac deliverables, and PTFs for service updates. SMP/E provides a mechanism for validating the signatures that are associated with signed code packages. With this capability, an installation can ensure that the software originated from IBM and was not altered in transit. Signed code packages are provided through GIMZIP and GIMUNZIP code package signing and validation. They are validated by using the software vendor's public key, and are then input for the build process for Validated Boot for z/OS.

SMP/E and z/OSMF provide capabilities to verify the signatures of the GIMZIP packages for these product and service offerings. However, verifying signatures requires preparation and one-time setup. For more information, see *z/OS SMP/E User's Guide*.

Banner message (logon panel)

In z/OS, VTAM® uses a UNIX System Services table (USSTAB) to display the logon panel for TSO/E users. The logon page is defined by the USSTAB macro.

The installation must define the logon page by defining the USSTAB structure for it. Then, the installation must associate the USSTAB with each of the device addresses that are defined in VTAM.

For more information about defining the USSTAB, see *z/OS Communications Server: SNA Resource Definition Reference*.

Appendix A. Accessibility

Accessible publications for this product are offered through IBM Documentation for z/OS (www.ibm.com/docs/en/zos).

If you experience difficulty with the accessibility of any z/OS documentation see How to Send Feedback to IBM to leave documentation feedback.

Notices

This information was developed for products and services that are offered in the USA or elsewhere.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

This information could include missing, incorrect, or broken hyperlinks. Hyperlinks are maintained in only the HTML plug-in output for IBM Documentation. Use of hyperlinks in other output formats of this information is at your own risk.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Site Counsel
2455 South Road

Poughkeepsie, NY 12601-5400
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or

reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM Online Privacy Statement

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session cookies that collect each user's name, email address, phone number, or other personally identifiable information for purposes of enhanced user usability and single sign-on configuration. These cookies can be disabled, but disabling them will also eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at ibm.com®/privacy and IBM's Online Privacy Statement at ibm.com/privacy/details in the section entitled "Cookies, Web Beacons and Other Technologies," and the "IBM Software Products and Software-as-a-Service Privacy Statement" at ibm.com/software/info/product-privacy.

Policy for unsupported hardware

Various z/OS elements, such as DFSMSdfp, JES2, and MVS, contain code that supports specific hardware servers or devices. In some cases, this device-related element support remains in the product even after the hardware devices pass their announced End of Service date. z/OS may continue to service element code; however, it will not provide service related to unsupported hardware devices. Software problems related to these devices will not be accepted for service, and current service activity will cease if a problem is determined to be associated with out-of-support devices. In such cases, fixes will not be issued.

Minimum supported hardware

The minimum supported hardware for z/OS releases identified in z/OS announcements can subsequently change when service for particular servers or devices is withdrawn. Likewise, the levels of other software products supported on a particular release of z/OS are subject to the service support lifecycle of those

products. Therefore, z/OS and its product publications (for example, panels, samples, messages, and product documentation) can include references to hardware and software that is no longer supported.

- For information about software support lifecycle, see: IBM Lifecycle Support for z/OS (www.ibm.com/software/support/systemsz/lifecycle)
- For information about currently-supported IBM hardware, contact your IBM representative.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at Copyright and Trademark information (www.ibm.com/legal/copytrade.shtml).

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Glossary

This glossary defines technical terms and abbreviations used in z/OS documentation related to multilevel security. If you do not find the term you are looking for, refer to the index of the appropriate z/OS document.

access
A specific type of interaction between a subject and an object that results in the flow of information from one to the other.

access authority
An authority that relates to a request for a type of access to protected resources. In RACF, the access authorities are NONE, READ, UPDATE, ALTER, and EXECUTE.

access control list
A list within a profile of all users authorized to access the resource (or resources) protected by the profile, and their access authorities.

access list
See access control list.

authority
The right to access objects, resources, or functions.

authorization checking
The action of determining whether a user is permitted access to a RACF-protected resource.

authorized program
A program that runs under one or more of the following conditions:

- With, or with a capability to obtain, a system protection key (PSW protection keys 0-7)
- In supervisor state
- With authorized program facility (APF) authorization

category
An installation-defined name corresponding to a department or area within an organization with similar security requirements.

certification
The technical evaluation of a system's security features, made as part of and in support of the approval/accreditation process, that establishes the extent to which a particular computer system's design and implementation meet a set of specified security requirements.

character special file
A z/OS UNIX file that provides access to an input/output device.

classification
Assigning an entity to a class based on security requirements.

clearance
The trustworthiness that can be placed in a user of the system. It does not imply need to know, but rather is the authorization required to receive information.

DAC
discretionary access control

data integrity
The state that exists when computerized data is the same as that in the source documents and has not been exposed to accidental or malicious alteration or destruction.

data set profile
A profile that provides RACF protection for one or more data sets. The information in the profile can include the data set profile name, profile owner, universal access authority, access list, and other data.

discretionary access control
A means of restricting access to objects based on the identity of a subject and/or the groups to which the subject belongs. The controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject.

disjoint security labels
Two security labels are disjoint when each of them has at least one category that the other does not have. Neither of the labels dominates the other.

dominate
One security label dominates a second security label when (1) the security level that defines the first is equal to or greater than the security level that defines the second and (2) the set of security categories that defines the first includes the set of security categories that defines the second.

DSMON
Data security monitor auditing program, which is part of RACF. It is a batch program that generates a set of reports showing the current status of system security controls.

equal mandatory access check
A mandatory access check in which the security label of the user must be equivalent to the security label of the resource in order for the user to be granted access to the resource.

equivalence
Two security labels that contain the same security level and the same set of security categories are considered to be equivalent. Each of the security labels dominates and is dominated by the other.

erase-on-scratch
The physical overwriting of data on a DASD data set when the data set is deleted (scratched) or when space is released.

global access checking
The ability to allow an installation to establish an in-storage table of default values for authorization levels for selected resources.

group
A collection of RACF users who can share access authorities for protected resources.

identification label
Text, graphics, or a combination of text and graphics containing security information that is printed on each output page in a multilevel-secure environment. An identification label is associated with a security label, and the security label of the data printed determines which identification label is printed.

incompatible security labels
See *disjoint security labels*.

MAC
mandatory access control

mandatory access control
A means of restricting access to objects based on the sensitivity (as represented by a label) of the information contained in the objects and the formal authorization (clearance) of subjects to access information of such sensitivity.

multilevel device
A device that is used in a manner that permits it to simultaneously process data of two or more security levels without risk of compromise. To accomplish this, sensitivity levels are normally stored on the same physical medium and in the same form (machine-readable, human-readable) as the data being processed. Contrast with single-level device.

multilevel security
A security policy that allows the classification of data and users based on a system of hierarchical security levels (for example: unclassified, secret, top secret) combined with a system of non-hierarchical security categories (for example: Project A, Project B, Project C). The system imposes mandatory access controls restricting which users can access data based on a comparison of the classification of the users and the data. In order to access data, a user must have a security level greater than or equal to that of the data, and be authorized to all of the categories assigned to the

data. The mandatory access controls exist in addition to any discretionary access controls (such as access lists) that users can manipulate, and a user must pass both the mandatory controls and any discretionary controls in order to access the data protected by those controls.

name-hiding function
A function that restricts the display of the names of data sets, files, and directories to those that the user is authorized to read.

object
A passive entity that contains or receives information. Access to an object potentially implies access to the information it contains. Examples of objects are: records, blocks, pages, segments, files, directories, and programs, as well as bits, bytes, words, fields, processors, video displays, keyboards, clocks, printers, and network nodes.

object reuse
The reassignment to some subject of a medium (page frame, disk track, magnetic tape) that contained one or more objects. To be securely reassigned, such media must contain no residual data from any previous objects.

password
A private character string that is used to authenticate an identity.

profile
Data that describes the significant characteristics of a user, a group of users, or one or more computer resources.

RACF
Resource Access Control Facility. A component of the Security Server feature of z/OS that provides security functions.

RACF database
A collection of interrelated or independent data items stored together without unnecessary redundancy, to serve Resource Access Control Facility (RACF).

RACF report writer
A RACF function that prints out RACF SMF records and produces reports on system use and resource use from information found in the records.

read
A fundamental operation that results only in the flow of information from an object to a subject.

read access
Permission to read an object.

reverse mandatory access check
A mandatory access check in which the security label of the resource must dominate the security label of the user in order for the user to be granted access to the resource.

root file system
In a non-sysplex environment, the z/OS UNIX file system that contains the binary files and text files delivered by IBM. The root file system is called the version root in a sysplex environment.

seclabel
See *security label*.

security
The protection of sensitive information from unauthorized disclosure, destruction, or alteration. It is intended to prevent subjects from gaining access to objects in ways other than those for which they are authorized.

security category
A special designation for data at a given security level that indicates that only people properly briefed and cleared can receive permission for access to the information.

security label
A name that represents the combination of a hierarchical level of classification (security level) and a set of nonhierarchical categories (security category). Security labels are used in the trusted computing base as the basis for mandatory access control decisions.

seclevel
See *security level*.

security administrator
The user with the RACF SPECIAL attribute.

security level
A hierarchical designation for data that represents the sensitivity of the information.

security policy
The set of laws, rules, and practices that regulate how an organization manages, protects, and distributes sensitive information.

sensitive information
Information that, as determined by a competent authority, must be protected because its unauthorized disclosure, alteration, loss, or destruction will at least cause perceivable damage to someone or something.

single-level device
A device that is used to process data of a single security level at any one time. Because the device need not be trusted to separate data of different security levels, security labels do not have to be stored with the data being processed. Contrast with multilevel device.

storage object
An object that supports both read and write accesses.

subject
An active entity, generally a person, process, or device, that causes information to flow among objects or changes the system state.

symbolic link
A type of z/OS UNIX file system entry that contains the pathname of and acts as a pointer to another file or directory.

SYSHIGH
A security label that the system creates. It is the highest security label in a system. It is defined as the highest security level in the system and includes all security categories.

SYSLOW
A security label that the system creates. It is the lowest security label in a system. It is defined as the lowest security level in the system and has no security categories associated with it.

SYSMULTI
A security label that the system creates. It is considered to be equivalent to any defined security label.

SYSNONE
A security label that the system creates. When SYSNONE is assigned to a resource, the security label of the user accessing the resource is treated as if it is equal to SYSNONE.

system-specific security label
A security label that is active on one or more of the systems in a sysplex, but not on all of them.

tranquility
Keeping the security classification of an object constant while it is in use; keeping the security classification of a subject constant while it is active.

temporary data set
An uncataloged data set that is meant to exist only for the life of the job. DFSMSdfp creates and deletes temporary data sets, which include VIO and MVS data sets, but not JES SYSIN and SYSOUT data sets.

trusted computer system
A system that employs sufficient hardware and software integrity measures to allow its users to process different levels of sensitive or classified information simultaneously without compromising the security of each user.

trusted computing base
The totality of protection mechanisms within a computer system, including hardware, microcode, and software, the combination of which is responsible for enforcing a security policy. A trusted computing

base creates a basic protection environment and provides additional user services required for a trusted computer system.

trusted software
The software portion of a trusted computing base.

UACC (universal access authority)
The default access authority that applies to a resource if the user or group is not specifically permitted access to the resource. The universal access authority can be any of the access authorities.

user
See *subject*.

version root
In a sysplex environment, the z/OS UNIX file system that contains the binary files and text files delivered by IBM. The version root is called the root file system in a non-sysplex environment.

write
A fundamental operation that results only in the flow of information from a subject to an object.

write access
Permission to write an object.

write-down
The action of an address space creating output data at a lower labelled classification than that at which the address space is executing.

write-down mode
A user mode in which the user can write data to an object with a lower security label than the user's current security label on a system where write-down is normally disallowed because the RACF MLS(FAILURES) option is in effect.

write-down privilege
A user privilege that allows the user to activate and deactivate write-down mode.

zFS
z/OS File System. A z/OS UNIX file system that can be used in addition to the hierarchical file system (HFS). z/OS provides support for zFS in its zFS File System element.

Index

Special Characters

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X

Z

Product Number: 5650-ZOS

GA32-0891-50