IBM eServer™

Multilevel Security

New England
RACF Users Group
October 2004

George Markouizos
RACF Development
Telephone: (845) 435-8563
e-mail: gmarkou@us.ibm.com

Table of Contents

Trademarks

- The following are trademarks or registered trademarks of the International Business Machines Corporation:
  - CICS
  - DB2
  - MVS
  - MVS/ESA
  - OS/390
  - RACF
  - S/390
  - z/OS
- UNIX is a registered trademark of The Open Group in the United States and other countries.

Why Multilevel Security?

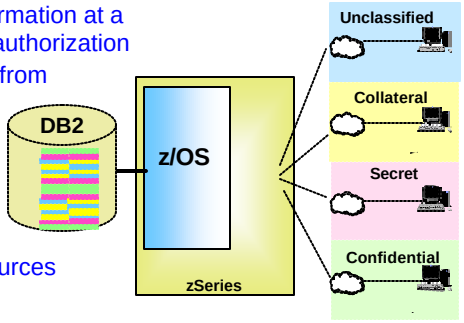
- Highly secure data
- Shared between people/organizations with different "need to know".
  - Multilevel Security provides a way to segregate users and their data from other users and their data regardless of access lists, UACC, etc.
- Must be
  - Manageable, Affordable, Resilient, Highly available
- Valuable to government agencies
  - Use of functions like name-hiding, write-down, *-property (no write-down)
- Valuable to commercial clients (i.e. service bureau)
  - Can be set up using a small set of SECLABELs and few SETROPTS options (MLACTIVE and SECLABELCONTROL)

    Example: MVS system with HTTP Server
    - Assign a “low” SECLABEL to external customers so they can access “external” data
    - Assign a “high” SECLABEL to employees so they can access both “internal” and “external” data

IBM eServer™

What is Multilevel Security?

- A secure computing environment with two goals:
  - Controls to prevent unauthorized individuals from accessing information at a higher classification than their authorization
  - Controls to prevent individuals from declassifying information
- Controls
  - Classifies data using
    - Security Levels
    - Security categories
  - System controls access to resources
    - Labels all resources
    - Enforces accountability
    - Prevents 'declassifying' data
    - Does not allow reuse of data objects until purged



Multilevel Security on zSeries

•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 nonhierarchical 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.

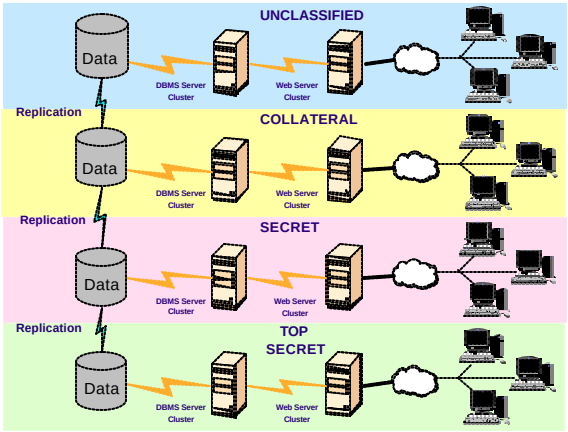
•Characteristics of a multilevel-secure system include the following:

- •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).

What is the problem?

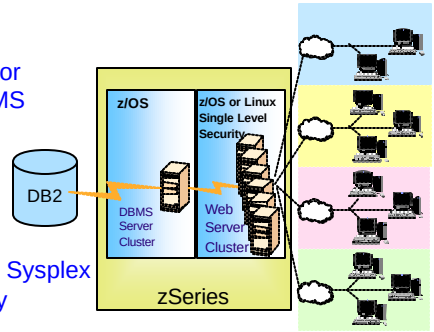
Without Multilevel Security

- Separate servers required for each security compartment
- Data is replicated between database servers
- Costly to implement and manage
- Infrastructure could be multiplied many times for each application or managing organization



The solution! - Multilevel Security on zSeries

- Proven mainframe reliability and quality of service
  - Self-optimizing
    - Managing to business priorities: z/OS Workload Manager
    - Managing resources: Intelligent Resource Director
    - Managing storage: z/OS DFSMS
  - Robust z/OS security:
    - RACF & PKI Services
    - Intrusion Detection Services
    - Address Space Isolation
  - zSeries cryptography
  - Scale and high availability: Parallel Sysplex
  - Business recovery: GDPS (Globally Dispersed Parallel Sysplex)
  - Server consolidation:
    - Linux for zSeries or z/OS for Web Serving
    - Secure Partitions: LPARS certified at Common Criteria EAL5



Multilevel Security on zSeries

- GA (General Availability):
  - z/OS 1.5 - March, 2004
  - DB2 v8 - March, 2004

Commercial Exploitation

- Application servers shared across multiple customer constituencies
  - Labeling allows data to be compartmentalized or isolated from other customers
    - Data protected for competitive, privacy and integrity reasons
  - Labeling of data and application identities provides the means for both the aggregation and compartmentalization of data.
- zSeries is able to host large databases on behalf of transaction programs or other application servers
  - Facilitates some database aggregation that reduces execution costs
- Database on demand capability
  - Functionality is provided by strength of security on z/OS and within DB2
  - DB2 z/OS V8 row-level security and z/OS V1.5 with RACF provide the operating system and security services that make database on demand capability whole

Whenever you move or replicate data to provide a new security container or isolation point, it's an opportunity to consider labeling the data and, in turn, saving on processor, network, storage and administrative expenses.

DB2 UDB for z/OS Version 8 provides row-level security.
A relational database is built with a collection of columns and rows. You can apply security labels to the columns and rows and provide a level of privacy or protection that isolates elements of the database from users running with differing security labels. It is the same form of labeling used by several government agencies that you can leverage to provide a commercial version of database compartmentalization that meets the needs of on demand computing.

Application serving on demand

- Service business acquires a collection of servers and hosts a specific application and its associated data on that server infrastructure.
- Sells subscriptions for the application to other businesses
  - Subscribers need their data isolated from other businesses
- Make a subset of the information available by aggregating data using labels. Isolate sensitive data.
- Saves server and network costs associated with replicating data across business units.

Outsourcer running an application practice
Common DB schema across customers
Seclabel="customer_name"

DB2_SECURITY_ LABEL_EXT	COL1	COL2	COL3
Customer A			
Customer B			
Customer A			
Customer B			
Customer C			
Customer D			
Customer E			
Customer A			
Customer B			
Customer D			
Customer E			

Figure 2 - Using security labels for application serving on demand

Businesses may choose to subscribe to the application service because they cannot afford or don't have the skills to host their own version of the application's computing infrastructure. At the same time, they want their data isolated from other subscribers.

By associating labels with each subscriber, the database can be consolidated across multiple buyers while retaining the appropriate levels of compartmentalization. This allows the services business to save disk and management resources by consolidating the needs of multiple buyers into fewer databases. In addition, if a specific buyer asks for extensions to the database schema, this can be accomplished simply and easily – without migrating the data, taking down the application, or compromising the compartmentalization of the data.

This can be accomplished with an online DB2 schema alteration and an online database reorg.

Financial services on demand

- Government regulations may inhibit one business unit from seeing personal, consumer information associated with another business unit.
- Subset of information may be valuable for data mining
  - Identifying trends
  - Developing new services
- Make a subset of the information available by aggregating data using labels. Isolate sensitive data.
- Saves server and network costs associated with replicating data across business units.

Large company managing HR for subsidiaries
"corporate phone book"
Seclabel="subsidiary_name"

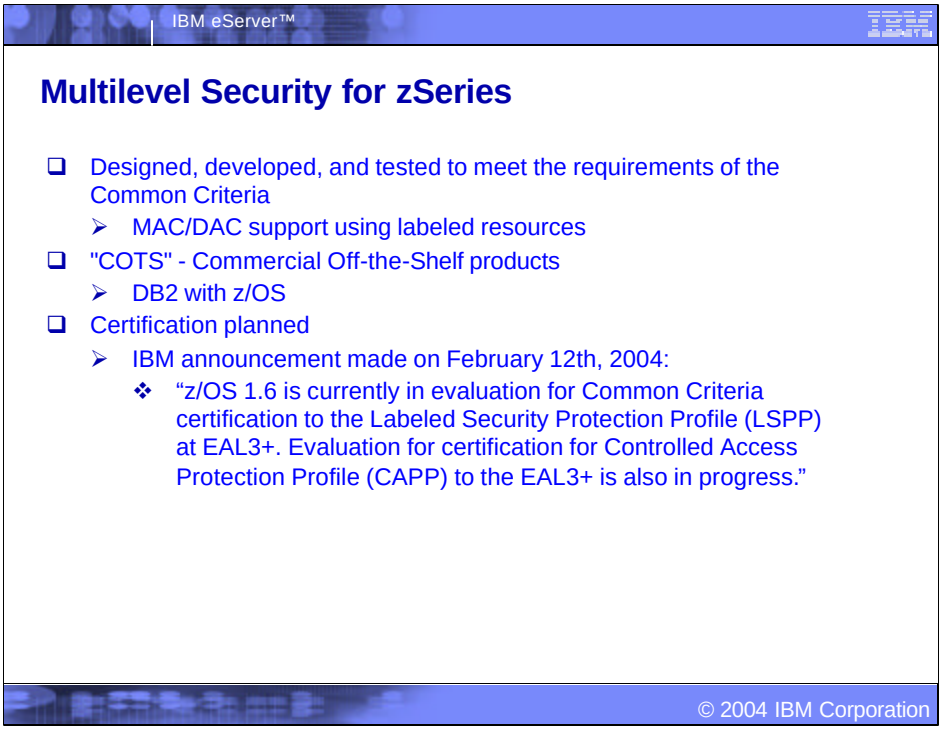
DB2_SECURITY_LABEL_EXT	COL1	COL2	COL3
Subsidiary 1			
Subsidiary 2			
Subsidiary 3			
Subsidiary 1			
Subsidiary 4			
Subsidiary 2			
Subsidiary 3			
Subsidiary 4			
Subsidiary 1			
Subsidiary 2			
Subsidiary 4			

Figure 3 - Using security labels for financial services on demand

Commercial on demand services summary

- Security labeling of data and application identities
  - Provides both compartmentalization and aggregation of data
- Need to replicate or move data to provide a new security container or isolation point?
  - Consider labeling
    - Save on processor, network, storage and administrative expense
- Examine database organization and flow of data between application servers
  - Labeling may provide additional security and deployment savings to your business

Source for this slide and the previous examples:
'Database on demand' by Jim Porell
August 2003 z/OS HOT TOPICS Neswletter, Issue 9



IBM announcement made on February 12th, 2004:

- http://www-1.ibm.com/press/PressServletForm.wss

- Mandatory Access Control (MAC) – 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 (with RACF SPECIAL authority) defines the sensitivity of each object by means of a security label and controls each subject's access to information by specifying which security labels the subject can use.

- Discretionary Access Control (DAC) – The principle of restricting access to objects based on the identity of the subject (the user or the group to which the user belongs). DAC is implemented using access control lists. The security administrator defines a profile for each object and updates the access list of 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.

z/OS V1R5 Multilevel Security enhancements

- New special system-defined SECLABEL
  - **SYSMULTI**
    - Used in cases where any classification of data could be "processed".
    - Compares as "equivalent" to any other defined SECLABEL for MAC decisions.
    - Intended for
      - daemons and servers that can accept connections from users running at different classification levels (SECLABELs) and properly mediate data access
      - UNIX directories (often, not always, root in a file system) that can have subdirectories of different SECLABELs.
    - Generally should not be assigned to real users, nor to a server that is not designed to handle multiple SECLABELs.

SECLABELs and MAC checking

- Three types of MAC checking
  - MAC
    - User's current SECLABEL dominates Resource's SECLABEL
  - RVRSMAC (Reverse MAC)
    - Resource's SECLABEL dominates User's current SECLABEL
  - EQUALMAC (Equal MAC)
    - User's current SECLABEL is equivalent to the Resource's SECLABEL.
- New operand EQUALMAC= added on the ICHERCDE macro
  - EQUALMAC=YES
    - The class requires SECLABEL equivalence

Equivalence - Either the SECLABEL names are identical, or two different SECLABEL names that are defined with identical security level and categories. When determining equivalence, the SECLABEL SYSMULTI is considered equivalent to any SECLABEL

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.

SECLABELs for z/OS UNIX Processes and Sockets

- Prior to z/OS V1R5, TSO/E users:
  - Have the ability to select their current SECLABEL by specifying it on the logon panel, or they can use their default.
  - The value they enter is saved in the TSO segment and used as the default the next time they log on.
- This function has been modified to:
  - Handle workstations (allowing for both reading and writing)
  - Support the z/OS UNIX environment where a user may enter the system from a remote IP address using an application such as rlogin.
  - Associate SECLABELs to IP addresses.

Prior to z/OS V1R5, when users enter the system through TSO/E they have the ability to select their current SECLABEL by specifying it on the logon panel, or they can use their default. The value they enter is saved in the TSO segment and used as the default the next time they log on. The SECLABEL of the terminal they are logging on to must dominate the user's SECLABEL.

This function has been modified to handle workstations (allowing for both reading and writing), the usage and characteristics of the TERMINAL class have changed to require SECLABEL equivalence and to supply the SECLABEL if none specified. The function has also been extended to support the z/OS UNIX environment where a user may enter the system from a remote IP address using an application such as rlogin.

The extension of SECLABELs to z/OS UNIX entails associating SECLABELs to IP addresses. Since the TERMINAL class cannot handle IP V6 addresses (due to their length), the usage and characteristics of the SERVAUTH class, which currently is used by the z/OS Communications Server (TCP/IP) to check server access authorization, have been changed so that IP V6 addresses can be accommodated.

SECLABELs for z/OS UNIX Processes and Sockets

- SERVAUTH class usage and characteristics have been enhanced to accommodate IP V6 addresses.
- New parameters have been added to InitACEE to allow the SECLABEL and SERVAUTH values to be passed.
- Corresponding changes have been made to allow applications to pass these values through UNIX System Services to InitACEE. These changes accommodate applications willing to change their code to allow the specification of a SECLABEL by the user.
  - New z/OS UNIX callable service, _poe, to set the port of entry for use by servers. Can set TERMINAL or SERVAUTH
  - z/OS UNIX Kernel will provide the server SECLABEL on the User Authentication call

The expanded usage of the SERVAUTH class requires a number of changes to its characteristics. Profiles in this class will require a SECLABEL if the MLACTIVE option is active. SECLABELs will be checked for equivalence rather than following the normal MAC authorization rules. Because it is a port of entry, changes to the SERVAUTH class can cause information to be removed from VLF, just as they can for TERMINAL and APPCPORT.

When the existing MLACTIVE option is set additional checking will be done by RACROUTE REQUEST=VERIFY and InitACEE to ensure that server applications do not allow users running with different security levels in the same server address space. When anchoring an client ACEE, where the server address space has an ACEE, the SECLABELs associated with each ACEE will be checked for equivalence. If they are not equivalent, the request will be failed with message ICH408I. Other products anchoring client ACEEs should also ensure equivalence. Note that assigning SYSMULTI to the server address space indicates that the server allows clients at multiple security levels since SYSMULTI is equivalent to any defined SECLABEL.

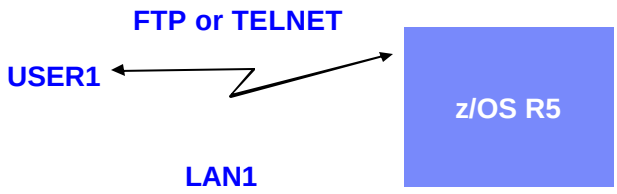
For applications that do not allow the specification of a SECLABEL, a SECLABEL for the user must be derived from the user's port of entry; a resource in the SERVAUTH or TERMINAL class. TCP/IP, z/OS UNIX and RACF will work together to determine the SECLABEL associated with the IP address when the SECLABEL class is active, and associate it with the user's security environment if the user is authorized to use it.

SECLABELs for z/OS UNIX Processes and Sockets

- Administrator can define IP subnetworks via RACF profiles
  - SERVAUTH class
  - Any granularity desired, down to individual IP address if needed
- SERVAUTH profile contains SECLABEL for that subnetwork
  - Installation is responsible for network topology and protection of network links
    - IPSEC (VPN) can also be used to help this
- TCP/IP stack ensures that application on host can only send/receive packets if application and IP address have equivalent SECLABEL
  - Support for servers or daemons that understand MLS (FTP, TELNET, INET)
    - Assign SYSMULTI SECLABEL to server/daemon
    - Can then communicate with any of the subnetworks

SECLABELs for z/OS UNIX Processes and Sockets

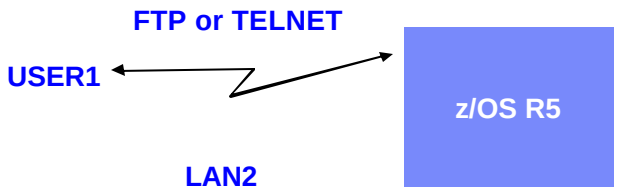
- Consider USER1 with access to
  - SECLABELs A and B
  - Workstations on three LANs
    - LAN1 defined with SECLABEL A
    - LAN2 defined with SECLABEL B
    - LAN3 defined with SECLABEL C



The user's session will run with SECLABEL A

SECLABELs for z/OS UNIX Processes and Sockets

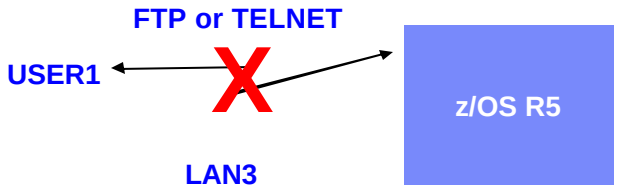
- Consider USER1 with access to
  - SECLABELs A and B
  - Workstations on three LANs
    - LAN1 defined with SECLABEL A
    - LAN2 defined with SECLABEL B
    - LAN3 defined with SECLABEL C



The user's session will run with SECLABEL B

SECLABELs for z/OS UNIX Processes and Sockets

- Consider USER1 with access to
  - SECLABELs A and B
  - Workstations on three LANs
    - LAN1 defined with SECLABEL A
    - LAN2 defined with SECLABEL B
    - LAN3 defined with SECLABEL C



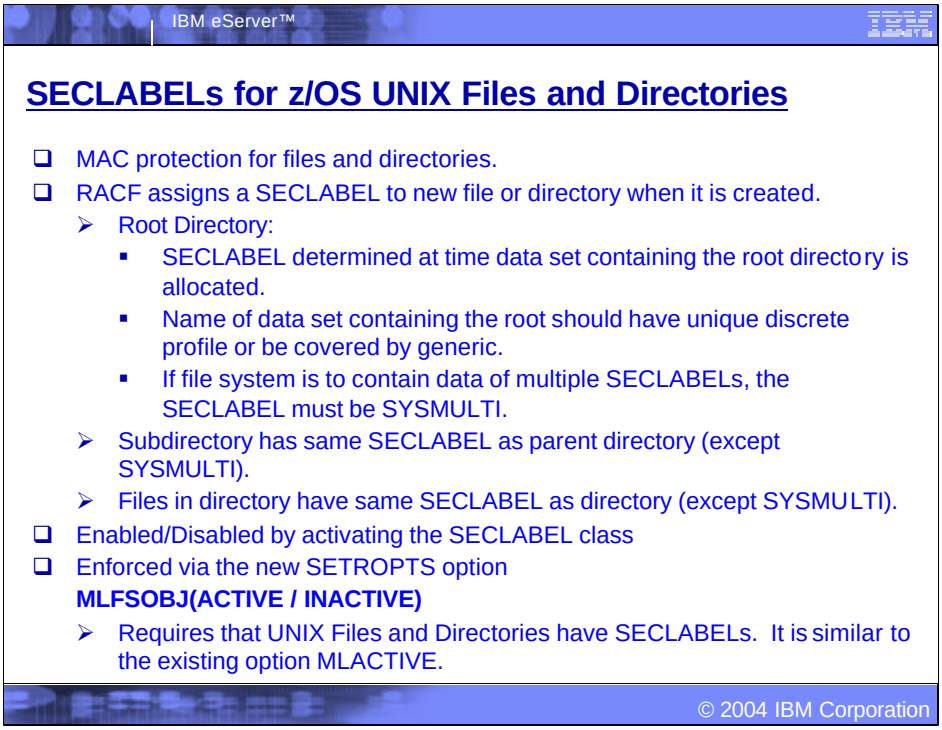
The user's session will fail,
since the user cannot use SECLABEL C

SECLABELs for z/OS UNIX Processes and Sockets

- Program access to SERVAUTH (enhancements to WHEN(PROGRAM) Conditional Access to the SERVAUTH class)
  - Allow appropriate use of PING and TRACEROUTE by a network administrator when multilevel security is enabled
    - Communications Server (TCP/IP) has the ability to restrict access to SERVAUTH resources to users running certain programs
- Allowed **ONLY** in a "**clean environment**" (like PADS – Program Access to Data Sets)
  - All programs previously loaded must be program-controlled
  - Uncontrolled programs cannot be loaded into the environment after access has been granted to the SERVAUTH based on the program name

To allow appropriate use of the "ping" and "traceroute" commands by a network administrator, when running with MLS functions enabled, Comm Server needs the ability to restrict access to SERVAUTH resources to users running certain programs. This can be done by allowing the specification of WHEN(PROGRAM(some_name)) for SERVAUTH general resource profiles.

To support this Comm Server requirement the PERMIT command has been changed to allow WHEN(PROGRAM) to be specified for the SERVAUTH class. Additionally, since Comm Server uses RACROUTE REQUEST=FASTAUTH rather than REQUEST=AUTH, WHEN(PROGRAM(...)) support has been added only to FASTAUTH processing. As with Program Access to Data Set, Program Access to SERVAUTH is only allowed in a clean environment; all programs previously loaded must be program-controlled, and no uncontrolled programs may be loaded into the environment after access is granted to the SERVAUTH based on the program name. Unlike Program Access to Data Set, PADCHK/NOPADCHK specification in the PROGRAM class has no meaning: only the current program's or first program's authority is checked.



The SECLABEL for the FSP of the root within each z/OS UNIX file system data set will be determined at the time the data set is allocated, via the standard data set SECLABEL association, and will be set by make_root_FSP when the SECLABEL class is active. The SECLABEL will either be an installation-defined SECLABEL or a special SECLABEL indicating that the file system may contain 'multilevel' contents. The special SECLABEL, SYSMULTI, will allow for file systems that contain data of multiple SECLABELs.

Note The name of the data set containing the root should not match more than 1 discrete profile since the volume serial number is not available to make_root_FSP.

SECLABELs for z/OS UNIX Files and Directories

- SECLABEL cannot be changed.
  - Use the z/OS UNIX command, **chlabel**, to set one.
    - **R_setfsecl** - New callable service invoked by **chlabel** to set the SECLABEL of files or directories.
  - Copy the file to a directory with the appropriate SECLABEL to change it (subject to dominance and write-down).

- **R_setfsecl**
  - Users with **RACF SPECIAL** can set the SECLABEL of a file or directory if the FSP has a SECLABEL that is blank or zero

SECLABELs for z/OS UNIX Files and Directories

Since the zSeries file system (zFS) and the hierarchical file system (HFS) can both participate in shared sysplexes, note:

- The zSeries file system (zFS) supports security labels:
  - Symbolic links are protected by security labels.
  - Hard links are protected by security labels.
  - If a z/OS UNIX file, directory, or symbolic link was created in a zFS file system without being assigned a security label, the security administrator can assign a security label to it using the **chlabel** shell command (*).
- The hierarchical file system (HFS) does not fully support security labels.
  - If you want to use an HFS file system in read-write mode, and use security labels in the file system, you must copy or move it to a zFS file system.
  - The HFS file system does not support the name-hiding function.

- (*) That could happen for existing files and directories (created before enabling SECLABELs).
- For more information, see publication GA22-7509-00 - **Planning for Multilevel Security**

SECLABELs for z/OS UNIX Interprocess Communications

- MAC protection for
  - IPC Objects (shared memory, message queues, semaphores)
  - Sigqueue
  - Pipes
  - UNIX Sockets
  - PTrace
- Communication can only occur between processes with equivalent SECLABELs (a.k.a. EQUALMAC).
  - With limited exceptions:
    - The resource or the accessor SECLABEL is SYSMULTI.
- SECLABEL cannot be changed later.
- Enabled/Disabled by activating the SECLABEL class
- Enforced via the new SETROPTS option
  MLIPCOBJ(ACTIVE / INACTIVE)
  - Requires that UNIX IPC Objects have SECLABELs. It is similar to the existing option MLACTIVE.

▪When creating an IPC security packet (ISP), if the SECLABEL class is active, RACF will copy the process SECLABEL, if one exists, into the ISP. Later, when checking access to the IPC for a subsequent connection, RACF will reject the request if the current process does not have a SECLABEL or the SECLABEL does not match. Once a SECLABEL has been assigned to an IPC object, there is no way to change it.

•Access checking for IPC objects will be treated as EQUALMAC checking, meaning that SECLABELs must be equivalent, unless the resource's SECLABEL is SYSMULTI, or the accessor's SECLABEL is SYSMULTI. While most classes require the specification of EQUALMAC=YES in the class descriptor table IPC access checking will not rely on this.

SECLABEL By System

- Allows customer to share a RACF database between systems and isolate use of specified SECLABELs to specified systems
- Specified by a member list on a SECLABEL profile
  - No members listed
    - Usable anywhere
  - Members listed
    - Usable only on one of those systems
- Not applicable to the SECLABELs provided by RACF, e.g.
  - SYSHIGH, SYSLOW, SYSNONE, SYSMULTI
- Enabled/Disabled via the new SETROPTS option **SECLBYSYS/NOSECLBYSYS**

In a SYSPLEX where many SYSPLEX members share the same RACF database, the ability to limit the use of SECLABELs to certain members may be a desirable function. This allows one member of the SYSPLEX to run work at SECLABEL A, while another handles SECLABEL B, keeping work separated based on security classification, while still sharing the RACF database. The specification of ADDMEM(SYSID) to the SECLABEL definition will allow SECLABELs to be activated on a per-system basis. Activation occurs when a SETR RACLIST(SECLABEL) REFRESH is issued after activating a new SETR option, SECLBYSYSTEM. To ensure jobs are submitted and execute on systems with the correct SECLABEL, JES will determine which SECLABELs are active on each system.

SECLABEL By System

Example:

- SECLABELs A, B, and C
- Systems SYS1 and SYS2

- Administrator could define them as follows:
  - RDEF SECLABEL (A,B) … ADDMEM(SYS1)
  - RDEF SECLABEL C … ADDMEM(SYS2)

Then

- Any attempt to access system SYS1 using SECLABEL C, or any attempt from SYS1 to access resources with SECLABEL C would fail
- Any attempt to access system SYS2 using SECLABEL A or B, or any attempt from SYS2 to access resources with SECLABEL A or B, would fail.

- RLIST, LISTDSD, SEARCH
  - List ONLY those profiles that have Active SECLABELs

SECLABEL By System

- New operand SIGNAL= added on the ICHERCDE macro
- Enhancements to SETROPTS processing for SECLABEL By System:
  - New ENF Signal is sent to listeners for those CDT classes that have SIGNAL=YES for
    - SETR RACLIST
    - SETR RACLIST REFRESH
    - SETR NORACLIST
- For the SECLABEL class, allows JES to keep a current list of active SECLABELs by listening for this signal.

Write-Down By User Privilege

Allows the Security Administrator to authorize specific users to Write-Down when SETR MLS is in effect.

- **R_writepriv**
  - New callable service to allow users to dynamically enable, disable, and reset Write-Down.
- **RACPRIV**
  - New RACF command to provide TSO/E users an interface to the callable service.
- **IRR.WRITEDOWN.BYUSER**
  - New RACF profile in the FACILITY class, used in the administration of the Write-Down privilege.
- **writedown**
  - New command for z/OS UNIX users.

- Write-Down privilege can be assigned to individual users that allows them to select the ability to write-down when SETR MLS is in effect. Users who have been assigned this privilege can enable and disable their ability to write-down.
- The write-down by user privilege can be activated and deactivated on a system through the creation of the FACILITY class profile IRR.WRITEDOWN.BYUSER.
  - Users with READ access can request Write-Down privilege
  - Users with UPDATE access can request Write-Down privilege and have it by default
- It is only checked if SETR MLS(FAILURES) or SETR MLS(WARNING) has been activated.
- Authority checking has been modified in RACROUTE REQUEST=AUTH and RACROUTE REQUEST=DEFINE.
- VERIFY/VERIFYX has also been modified.
  - A bit in the UTOKEN is set to indicate whether or not the user has the write-down privilege enabled. When the UTOKEN is built, the write-down privilege will be indicated if the user has it enabled by default.
- The new callable service, IRRSWP00, will allow users who have the write-down privilege to enable it, disable it, or reset it to the default setting, resulting in the resetting of the UTOKEN indicator. They can also query the current setting. The callable service provides the auditing.

Name Hiding

Allows installations to prevent users from discovering data set names, file names, and directory names that they didn't already know.

- Enabled/Disabled via the new SETROPTS option **MLNAMES/NOMLNAMES**
- Needed only if
  - The dataset names contain sensitive data
  - The file names contain sensitive data
- Should not be enabled, unless necessary, because it can cause performance degradation.

- Prior to z/OS V1R5:
  - Users with the ability to list the VTOC of a volume or list the entries in a catalog can see the names of many data sets.
  - z/OS UNIX users issuing the **ls** command against a SYSMULTI directory can see the names of subdirectories and files with differing security.
- This new function will prevent these from happening.
- Note that SECLABELs are not required to use some parts of this support. Data set names will be hidden if a user does not have at least read access to a DATASET profile based on discretionary access checking. For files and directories, however, this option results only in a check of security labels, which will always be considered successful if the SECLABEL class is inactive.

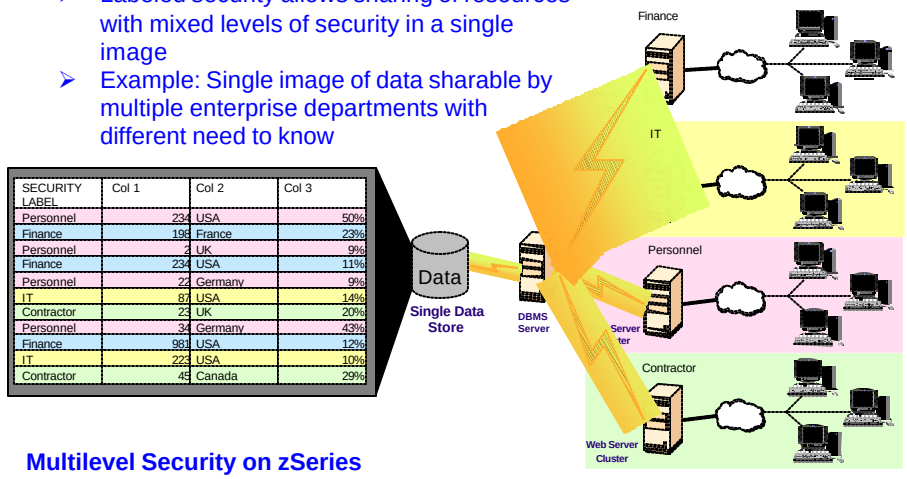
Miscellaneous Enhancements

- **SECLABEL support for FASTAUTH**
  - FASTAUTH was modified to provide support for SECLABELs
- Auditing
  - Two new Event Codes.
  - New Event Code Qualifiers and Relocate sections added to a number of events.
- **Enhancements to RACF Utilities**
  - In addition to the changes in UT200 and DB Unload for SERVAUTH, the following utilities have been enhanced:
    - SMF Unload
    - SAF Trace

Multilevel Security on z/OS V1R5 and DB2 V8

- Multilevel Security on z/OS V1R5 with DB2 V8
  - Labeled security allows sharing of resources with mixed levels of security in a single image
  - Example: Single image of data sharable by multiple enterprise departments with different need to know

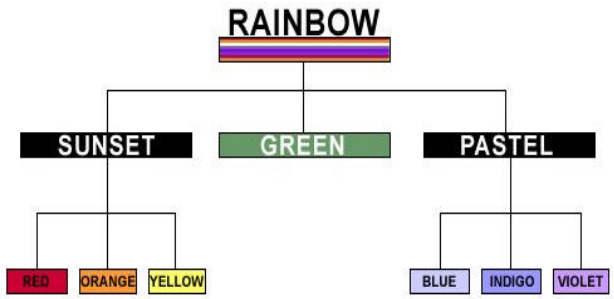
SECURITY LABEL	Col 1	Col 2	Col 3
Personnel	234	USA	50%
Finance	196	France	23%
Personnel	2	UK	9%
Finance	234	USA	11%
Personnel	22	Germany	9%
IT	87	USA	14%
Contractor	23	UK	20%
Personnel	34	Germany	43%
Finance	981	USA	12%
IT	223	USA	10%
Contractor	45	Canada	29%



Multilevel Security on zSeries

- GA (General Availability):
  - z/OS 1.5 - March, 2004
  - DB2 v8 - March, 2004

SECLABEL Hierarchy

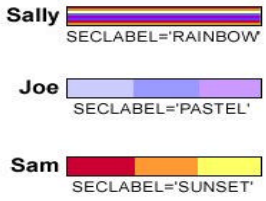


With the hierarchy established in the security manager layer, the system would understand that users with authority to read RAINBOW can read anything. Someone with authority to read PASTEL information can read any row associated with BLUE, INDIGO, VIOLET, or PASTEL. Someone with SUNSET can read SUNSET, RED, ORANGE, YELLOW. This is a lot more powerful than just having an exact match on SECLABEL (i.e., user's label must exactly match the data's label), since it has the notion of "groups" (in this case, sets of categories) that make security administration easier to manage.

See http://www7b.boulder.ibm.com/dmdd/library/techarticle/0209cotner/0209cotner.html

Multilevel Security and DB2

Row Granularity Multilevel Security



DB2_SECURITY_LABEL_EXT	COL1	COL2	COL2
RAINBOW	56	7	76
RAINBOW	24	56	65
RAINBOW	42	6	45
BLUE	3	456	7
INDIGO	113	456	56
VIOLET	3	456	4
BLUE	4	4556	7
RED	4	76	567
ORANGE	33	7	567
RED	5455	76	567
YELLOW	999	65	45

- Table column defined AS SECURITY LABEL
- Check for each new SECLABEL value accessed
- Mandatory access control: run time user to data

Row-level security for applications that need more granular security or mandatory access control. For example, an organization may want a hierarchy in which employees can see their own payroll data, a first line manager can see his or her payroll information and all of the employees reporting to that manager, and so on. Security schemes often include a security hierarchy and non-hierarchical categories.

You can add a column that acts as the security label (SECLABEL) with a column defined AS SECURITY LABEL: Each row value has a specific SECLABEL. The SECLABELs are defined and provided by RACF for a user, then saved in rows for INSERT, UPDATE, LOAD, …

When rows are accessed, DB2 checks for each new SECLABEL value accessed. If access is allowed, then, normal access. If access is not allowed, data is not returned. This is runtime user SECLABEL to data checking, in addition to grant and permit controls.

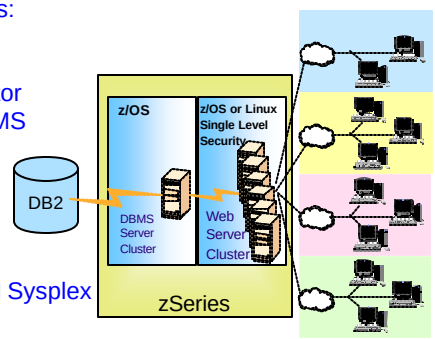
Multilevel Security and DB2

- **Multilevel Security with Row Level Granularity**
  - Use RACF for MAC
    - Use SECLABELs
    - Key advantage is consistent, integrated security
  - Table has a column defined as a security label
    - Each row value has a specific security label
    - Get user security label from RACF
    - Save in rows for INSERT, UPDATE, LOAD, ...
  - Compare SECLABEL in row to SECLABEL for the DB2 users
    - If access is allowed, then normal access
    - If access is not allowed, data not returned
  - Runtime user to data checking
  - Seclabel values are cached to minimize processing time

IBM eServer™

Recap - Multilevel Security on zSeries with DB2 V8

- Proven mainframe reliability and quality of service
  - Self-optimizing
    - Managing to business priorities: z/OS Workload Manager
    - Managing resources: Intelligent Resource Director
    - Managing storage: z/OS DFSMS
  - Robust z/OS security:
    - RACF & PKI Services
    - Intrusion Detection Services
    - Address Space Isolation
  - zSeries cryptography
  - Scale and high availability: Parallel Sysplex
  - Business recovery: GDPS (Globally Dispersed Parallel Sysplex)
  - Server consolidation:
    - Linux for zSeries or z/OS for Web Serving
    - Secure Partitions: LPARS certified at Common Criteria EAL5



Multilevel Security on zSeries

- GA (General Availability):
  - z/OS 1.5 - March, 2004
  - DB2 v8 - 1Q, 2004

Establishing a Multilevel Security environment on zSeries

- **Requirements**
  - Hardware
    - z/OS V1R5 is support on the following servers:
      - z990, z900, z800, G5, G6, MP3000 servers or equivalent
    - DB2 v8 is supported on the following servers:
      - z990, z900, z800 or equivalent
  - Software
    - z/OS V1R5
      - RACF
      - For System specific security labels -JES2
      - Print Services Facility (PSF)
  - Sysplex (shared DASD)
    - All systems must be at z/OS V1R5 or higher
    - All systems must share the RACF database
    - All systems in the GRS complex must be the same as those in the RACF Database
    - JES complex must be the same
- **NOTE:** Infoprint Server and BDT do not support Multilevel Security

z/OS V1R6 Multilevel Security Audit Enhancements

- **Multilevel Security Auditing** (SECLABELAUDIT) enhancements
  - Extends the auditing function of RACF
  - Meets requirements for evaluation of z/OS V1R6 to the Common Criteria for certification to the
    - Labeled Security Protection Profile (LSPP) at Evaluated Assurance Level (EAL) 3+.
    - Controlled Access Protection Profile (CAPP) at Evaluated Assurance Level (EAL) 3+.

z/OS V1R6 Multilevel Security Audit Enhancements

- **What is SECLABELAUDIT**
  - Provides additional auditing of access attempts to protected res ources based on the auditing option in the profile of the security label associated with the resource.
  - Enabled/Disabled by:
    - Activating/Deactivating the SECLABEL class
    - Enabling/Disabling the SETROPTS SECLABELAUDIT option **SETR SECLABELAUDIT/NOSECLABELAUDIT**

z/OS V1R6 Multilevel Security Audit Enhancements

- **Overview of Multilevel Security Auditing**
  - Auditing based on SETROPTS SECLABELAUDIT has been changed such that:
    - Auditing is done based on the security label of the user if it is different than the resource's security label and the resource's security label did not request auditing.
  - This support has been extended to existing RACF Services as well as z/OS Unix System Services (callable services).
  - Enabled/Disabled by:
    - Activating/Deactivating the SECLABEL class
    - Activating/Deactivating the existing SETROPTS option - **SECLABELAUDIT/NOSECLABELAUDIT**

Affected Components

- **RACF Services**
  - **RACROUTE REQUEST=AUTH**
  - **RACROUTE REQUEST=FASTAUTH**
  - **RACROUTE REQUEST=DEFINE**
- **z/OS UNIX System Services (callable services)**
  - **Check Access**
  - **Check IPC Access**
  - **Check Owner Two Files**
  - **Check Process Owner**
  - **Make FSP**
  - **Make ISP**
  - **R_PTRACE**
  - **R_AUDIT**

References

- Security Server (RACF) publications:
  - RACF Command Language Reference (SC28-1919)
  - RACF Security Administrator's Guide (SC28-1915)
  - RACF Callable Services Guide (SC28-1921)
- z/OS publications:
  - Planning for Multilevel Security (GA22-7509-00)
- RACF web site:
  http://www.ibm.com/servers/eserver/zseries/zos/racf
- DB2 web site:
  http://www.ibm.com/software/db2zos
  - Related publications / presentations:
    http://www.ibm.com/software/db2zos/db2zosv8.html
    http://www.ibm.com/software/db2zos/presentations.html
    http://www.ibm.com/software/db2zos/support.html