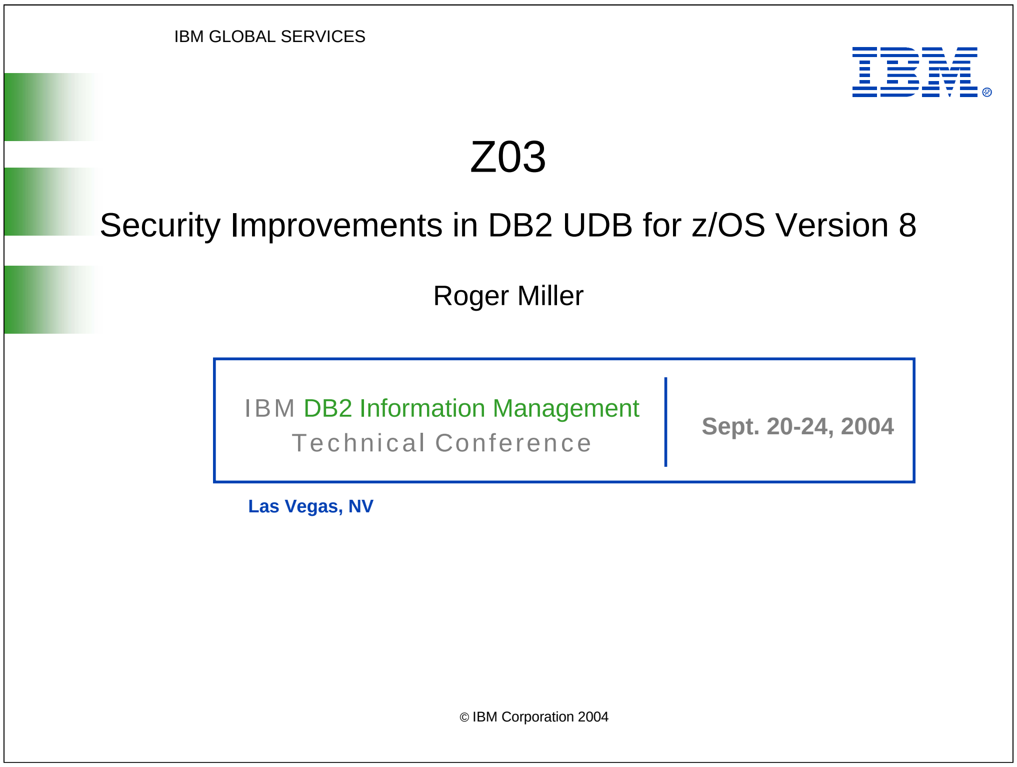


IBM GLOBAL SERVICES

Z03

Security Improvements in DB2 UDB for z/OS Version 8

Roger Miller

IBM DB2 Information Management
Technical Conference

Sept. 20-24, 2004

Las Vegas, NV

Security has changed very substantially in DB2 ® UDB for z/OS Version 8 (DB2 V8), with new options high security with multilevel security and row level security. Additional flexibility in security is provided for e-business and web applications. New options for tighter security, more granularity, and more information for additional flexibility in applications and SQL come in V8. Other new V8 features, such as encryption, improve security as well. This talk has a primary focus on news in security, privacy and auditing. Security has become much more important in the past few years. This presentation will discuss the latest changes.

DB2 V8 became generally available March 2004. More information is available on the web and at conferences. Many books and other information are provided.

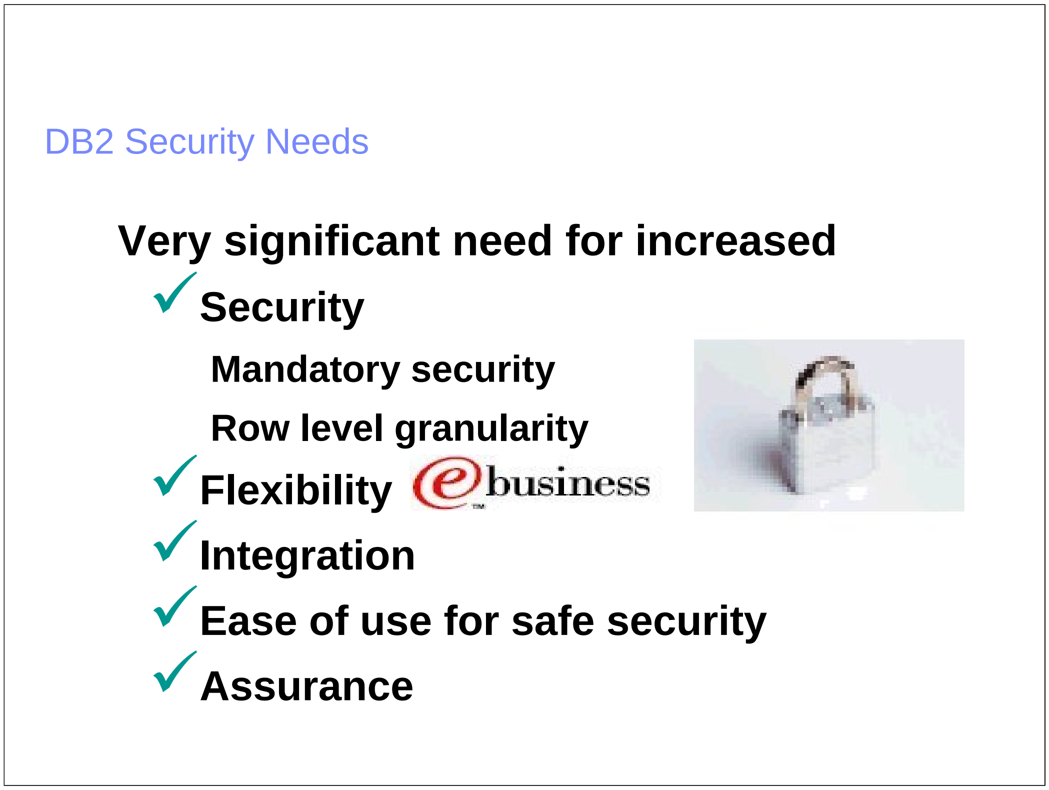
Agenda

- Multilevel Security with Row Level Granularity
- Multilevel Security for Access Control
- Special Registers and Session Variables
- Encryption
- Other authorization changes
  - Return Secondary Authorization Information
  - Materialized Query Tables
  - Sequences
- Summary

This is the outline for our discussion, with most of the information on multilevel security with row level granularity. Customers who use RACF access control can also use multilevel security with their access control. Customers who need more flexible security can use the new session variables to provide secure information to views, triggers, stored procedures and UDFs. Special registers may help.

Encryption has been used with DB2 for some time, but V8 adds new encryption options. The ability to return secondary authorization ids to applications was added with a V6 and V7 APAR. New objects in DB2 V8 - materialized query tables (MQTs) and sequences - have new authorization.

Other V8 changes have implications for security: long names, online schema evolution ...



DB2 Security Needs

Very significant need for increased

- ✓ **Security**
  - **Mandatory security**
  - **Row level granularity**
- ✓ **Flexibility** e business
- ✓ **Integration**
- ✓ **Ease of use for safe security**
- ✓ **Assurance**

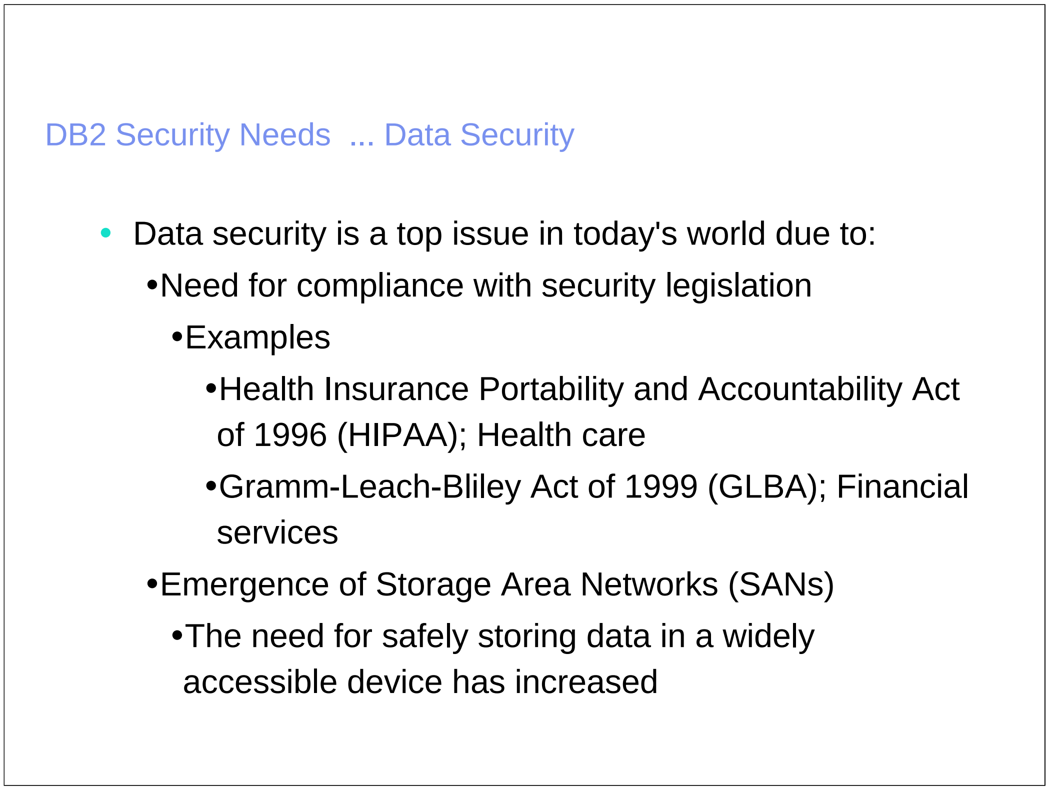
Everyone seems to be more aware of security today. Improving integration and making security more robust and easier to manage are very important.

Customers asked for a wide range of enhancements for security. New options for tighter security, more granularity, and more information for additional flexibility in applications and SQL are needed.

Improved integration with the hardware and software platform provide a better end-to-end solution.

The design point for DB2 security is being easy to use for safe security, with solid system security and some security that is more flexible, that we'll call application security.

Finally, it will be helpful to see what assurance can be provided, such as certification.



DB2 Security Needs ... Data Security

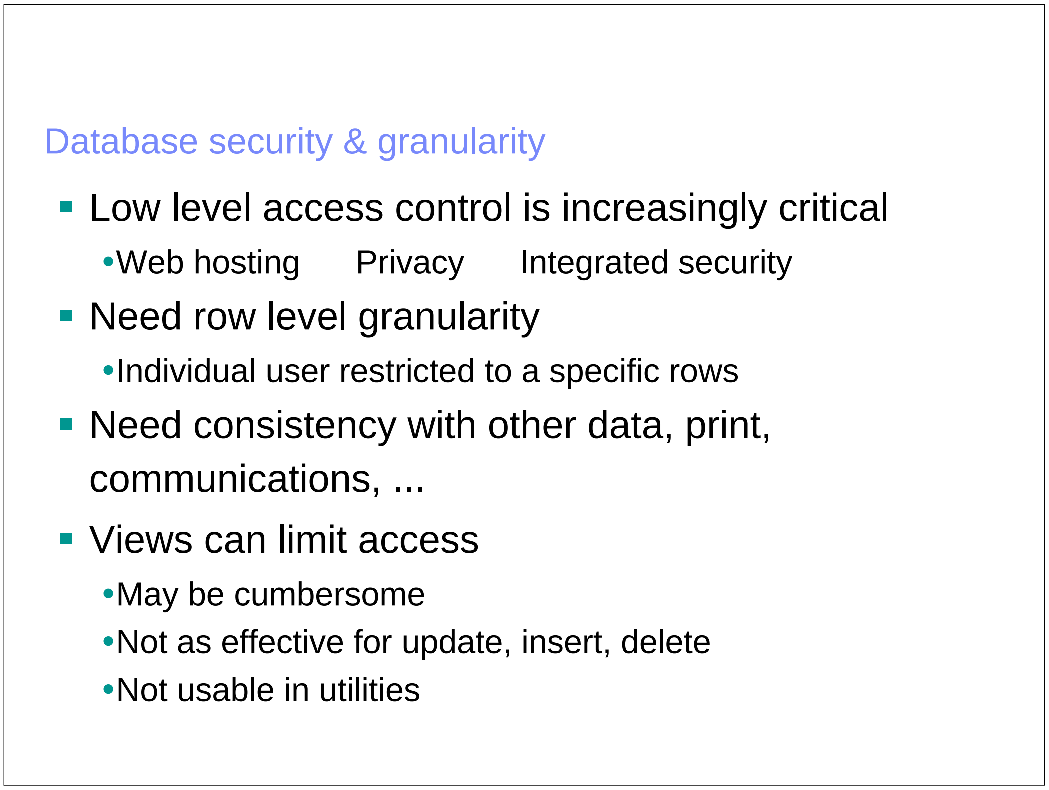
- Data security is a top issue in today's world due to:
  - Need for compliance with security legislation
    - Examples
      - Health Insurance Portability and Accountability Act of 1996 (HIPAA); Health care
      - Gramm-Leach-Bliley Act of 1999 (GLBA); Financial services
  - Emergence of Storage Area Networks (SANs)
    - The need for safely storing data in a widely accessible device has increased

Data security is a top issue for many people. There are a couple reasons for this. One reason is that current world events have increased our need for security. Another reason is that security legislation has been enacted requiring increased security.

Here are two U.S. examples of security legislation. The first one is the Health Insurance Portability and Accountability Act (HIPAA) of 1996, which affects health care. The second, which has existed for a while, relates to the finance industry and is known as the Graham-Leach-Bliley Act of 1999. The HIPAA Act has a deadline of April 15, 2003. This act requires health care companies to protect what is called personally identifiable information (PII).

Data Encryption for IMS and DB2 databases helps provide the security protection that the customer needs.

Another reason why data security has become a top issue concerns storage area networks (SANs). The model for a storage area network is one in which a pool of disk space is used by many different systems and is on the network. The network could be a company's intranet, or it could even be the internet. By having such modularity (much like grid computing) and plugging more storage into the network, a possible security exposure is presented; this is because now different systems, different applications, and different platforms are all accessing the same hardware devices that have data on them. And some of that data may be highly sensitive.

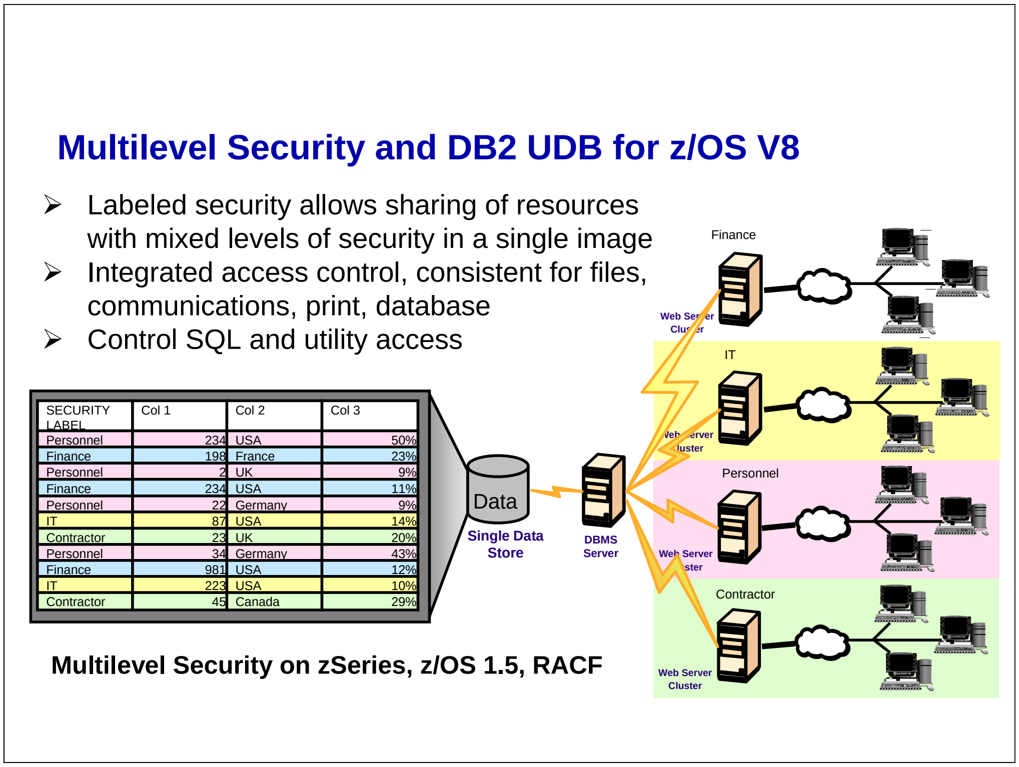


Database security & granularity

- Low level access control is increasingly critical
  - Web hosting Privacy Integrated security
- Need row level granularity
  - Individual user restricted to a specific rows
- Need consistency with other data, print, communications, ...
- Views can limit access
  - May be cumbersome
  - Not as effective for update, insert, delete
  - Not usable in utilities

Low level access control is increasingly critical. Example: web hosting company to store multiple customers' data into a single subsystem, database or table. Security & laws on privacy demand row level security.

Many customers need to extend the granularity from table level to row level, so that an individual user is restricted to a specific set of rows. Views can limit access to selected rows & columns, but they may be cumbersome to construct with desired level of granularity. Views are not very effective for update, insert, delete & utilities. Database constraints, triggers & UDFs, and stored procedures are often needed for update control.



Multilevel Security and DB2 UDB for z/OS V8

- ➢ Labeled security allows sharing of resources with mixed levels of security in a single image
- ➢ Integrated access control, consistent for files, communications, print, database
- ➢ Control SQL and utility access

SECURITY LABEL	Col 1	Col 2	Col 3
Personnel	234	USA	50%
Finance	198	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%

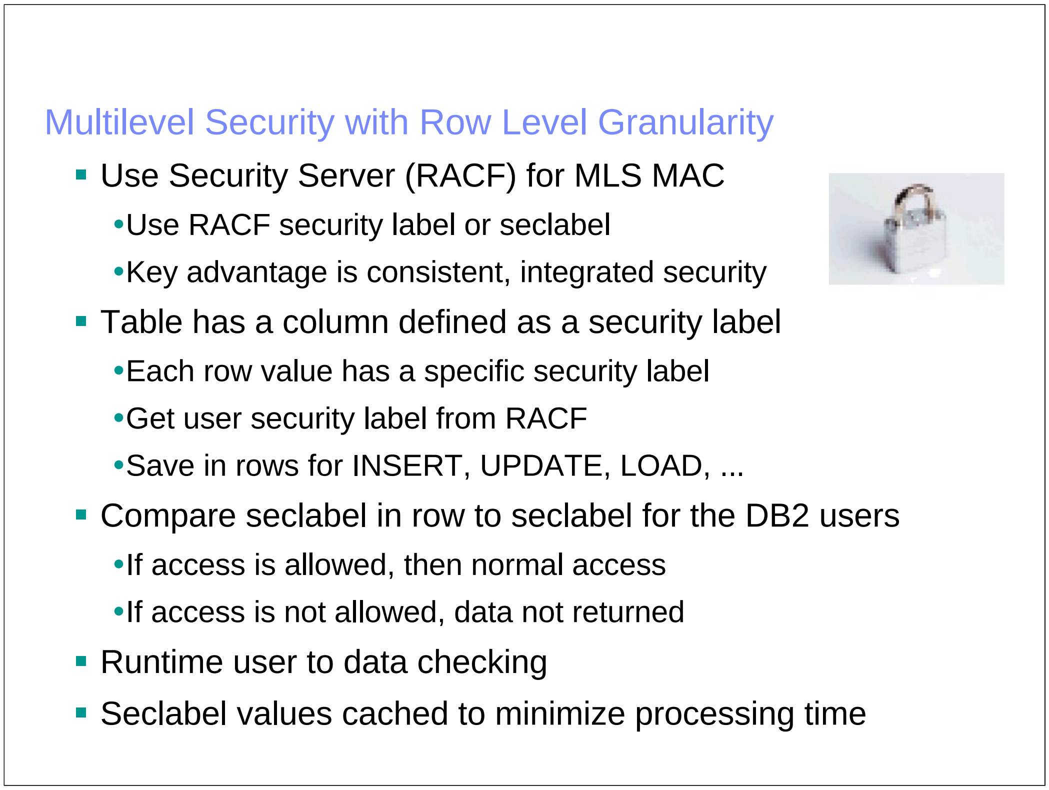
Multilevel Security on zSeries, z/OS 1.5, RACF

z/OS 1.5 and RACF 1.5 or Security Server add another type of security, called multilevel security, labeled security or mandatory access control (MAC) to our capabilities. The only option in the past with a high degree of separation has been physical separation. In the database world that might mean another machine or LPAR or perhaps another subsystem, another database or another table. With multilevel security, we still have a high degree of security even with data in the same table.

Access control is consistent across many types of resources using RACF, so that multilevel controls apply for data sets, for communications, for print and for database access – both objects and now with row level granularity. The DB2 controls are for both SQL access and for utility access.

For an more on multilevel security, see **Planning for Multilevel Security (GA22-7509)**

http://publibz.boulder.ibm.com/epubs/pdf/e0z2e100.pdf

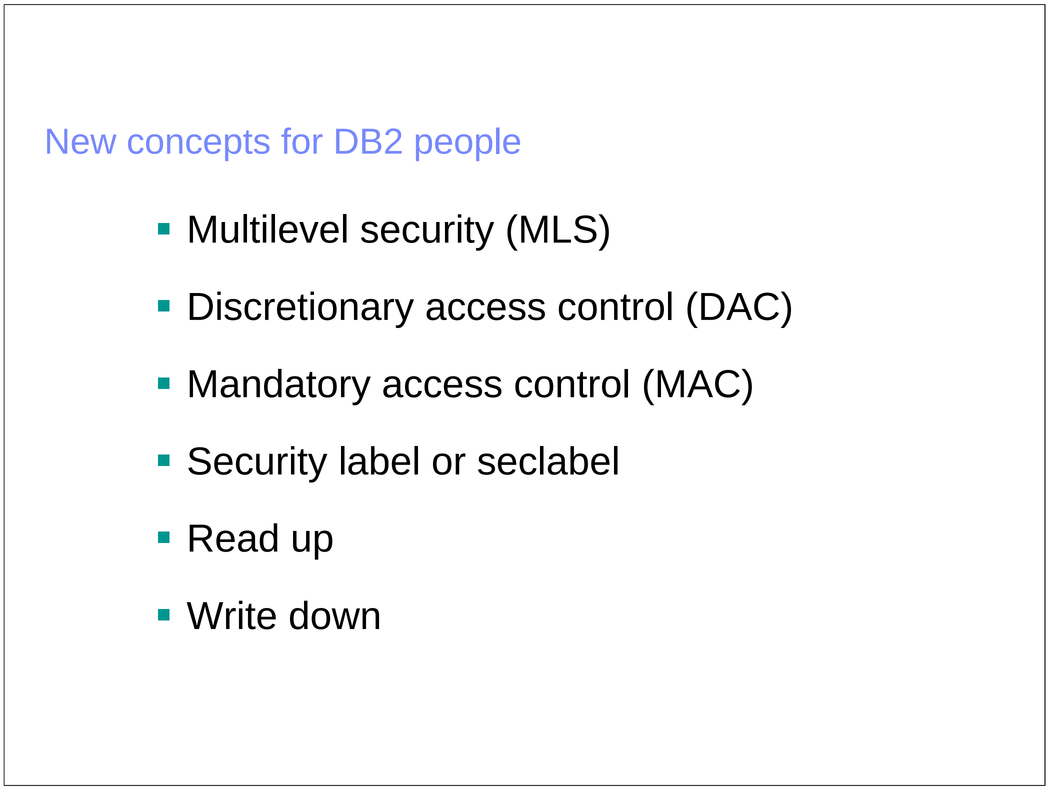


Multilevel Security with Row Level Granularity

- Use Security Server (RACF) for MLS MAC
  - Use RACF security label or seclabel
  - 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 cached to minimize processing time

We have had many requests for row-level security for applications that need mandatory access control (MAC) and more granular security schemes. The approach is to increase the granularity and to use a MAC technique using the RACF security label or seclabel. The key advantage of this design is security that uses a single control point and is consistent for database access, printers, displays, databases and network access. When the table is created, a security label column can be defined or a security label column can be added to an existing table that does not have one. Seclabels are obtained from RACF, and used for the access checking.

Checking is always for the current user to the data.

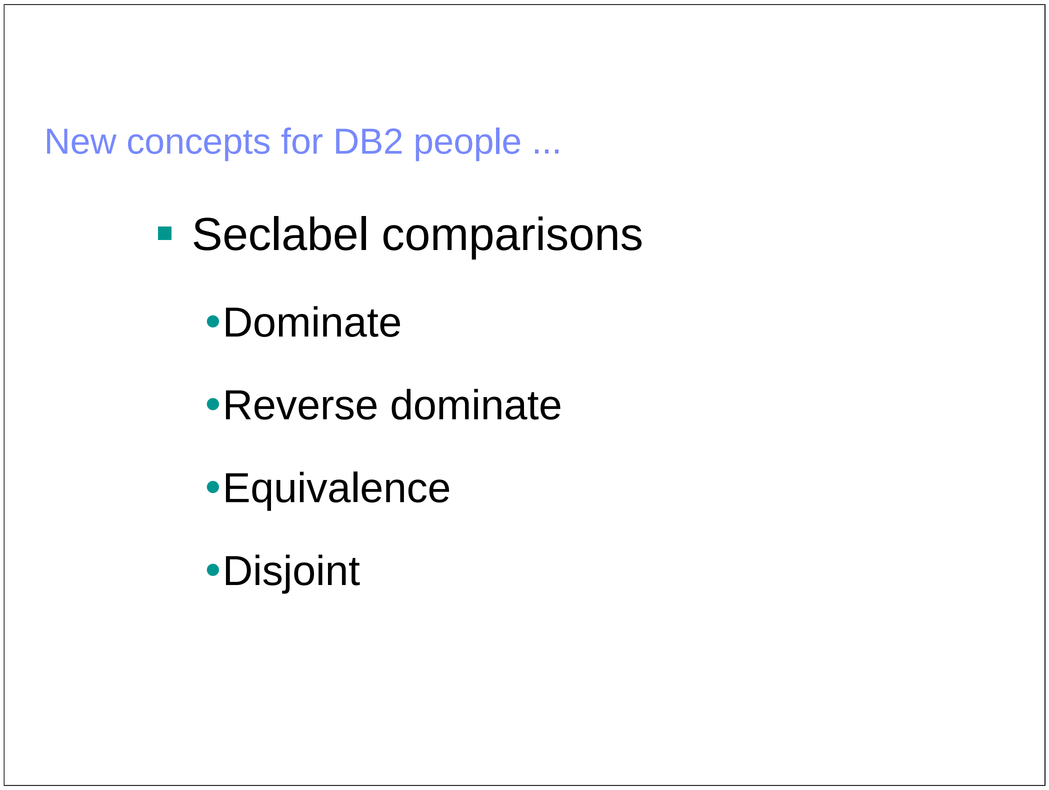


New concepts for DB2 people

- Multilevel security (MLS)
- Discretionary access control (DAC)
- Mandatory access control (MAC)
- Security label or seclabel
- Read up
- Write down

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 (MLS) 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. 1. Controls must prevent unauthorized individuals from accessing information at a higher classification than their authorization (read up). 2. Controls must prevent individuals from declassifying information (write down).



New concepts for DB2 people ...

- Seclabel comparisons
  - Dominate
  - Reverse dominate
  - Equivalence
  - Disjoint

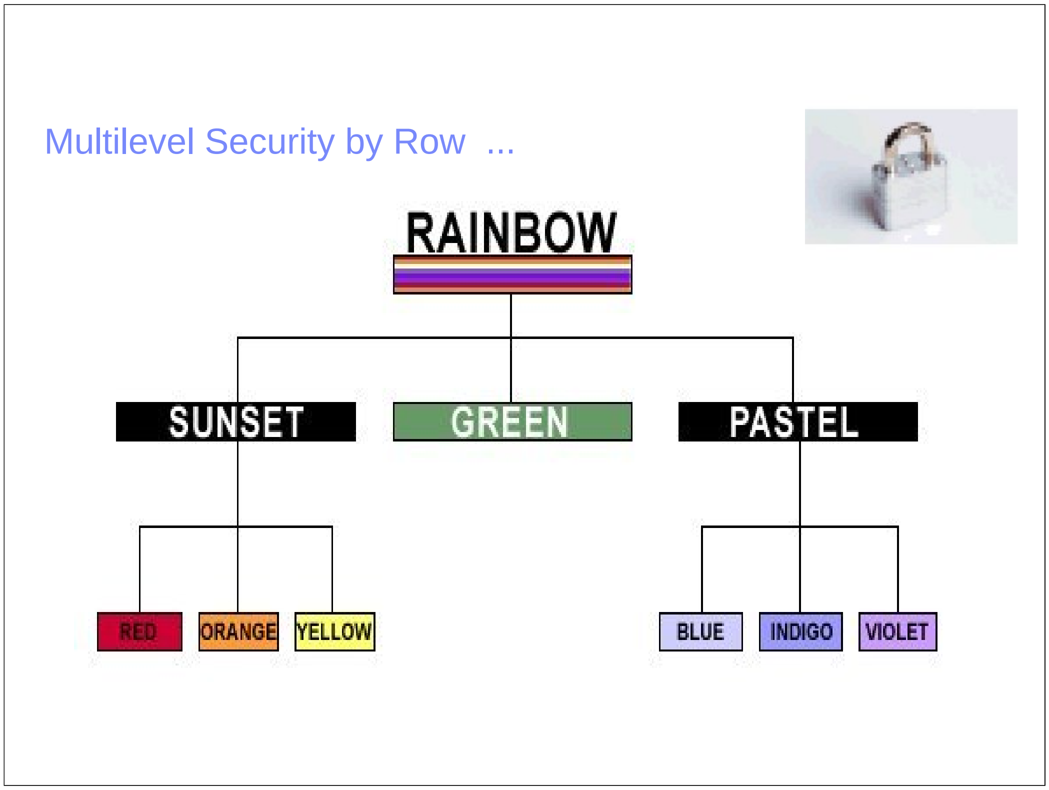
The terms for comparing seclabels differ from relational algebra, and the combination of hierarchies and non-hierarchical categories means that the result of a comparison for valid seclabels has four possible values. These are my explanations, and Null is my name.

Dominate: "greater than or equal to"
Reverse dominate: "less than or equal to"

Equivalence: "equal to" Equivalent means the seclabels are the same or have the same level and set of categories. One way to check is both dominance and reverse dominance are true.

Disjoint: None of the above, what we might call null in database terminology

The new concepts are described in the book Planning for Multilevel Security, available in the general books on the z/OS Library web pages.



Multilevel Security by Row ...

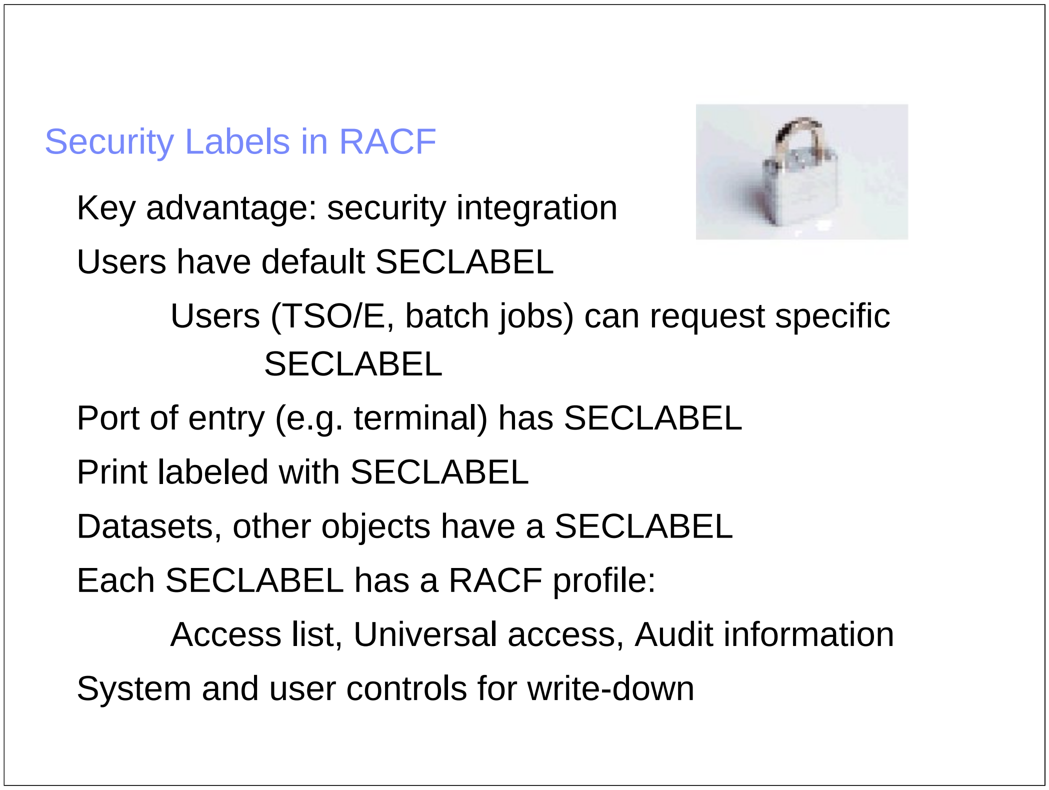
RAINBOW

SUNSET GREEN PASTEL

RED ORANGE YELLOW BLUE INDIGO VIOLET

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

With this additional capability, we'll be able to implement



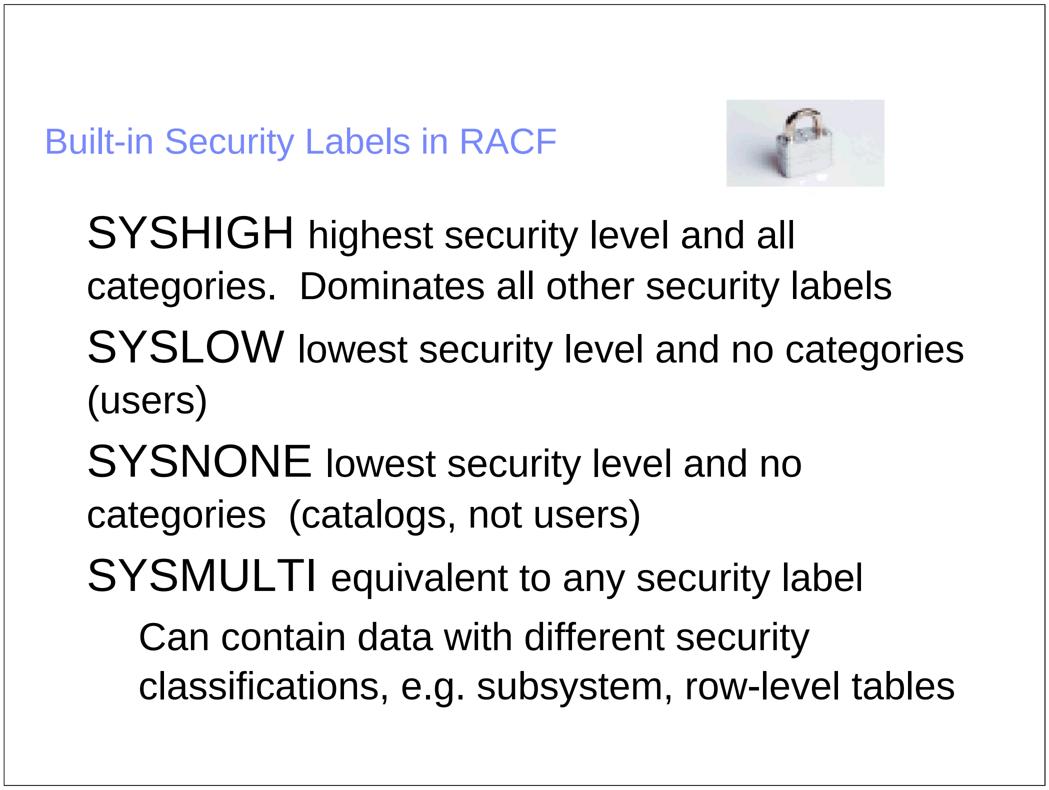
Security Labels in RACF

- Key advantage: security integration
- Users have default SECLABEL
  - Users (TSO/E, batch jobs) can request specific SECLABEL
- Port of entry (e.g. terminal) has SECLABEL
- Print labeled with SECLABEL
- Datasets, other objects have a SECLABEL
- Each SECLABEL has a RACF profile:
  - Access list, Universal access, Audit information
- System and user controls for write-down

The SECLABELs and their relationships are defined within RACF. The key advantage is that security is integrated across the platform, with the same security for files, print, and DB2. SECLABEL can be assigned to users, ports of entry, systems, data sets and other objects. There are controls to require users and objects to have SECLABELs, and to ensure consistency. Printer support will label the output with the SECLABEL.

RACF also has controls for the ability to write down at both the system and the user level. Users who have the ability to write down can enable, disable and reset the ability to write down.

Customers who are interested in this topic will need to read Planning for Multilevel Security.



Built-in Security Labels in RACF

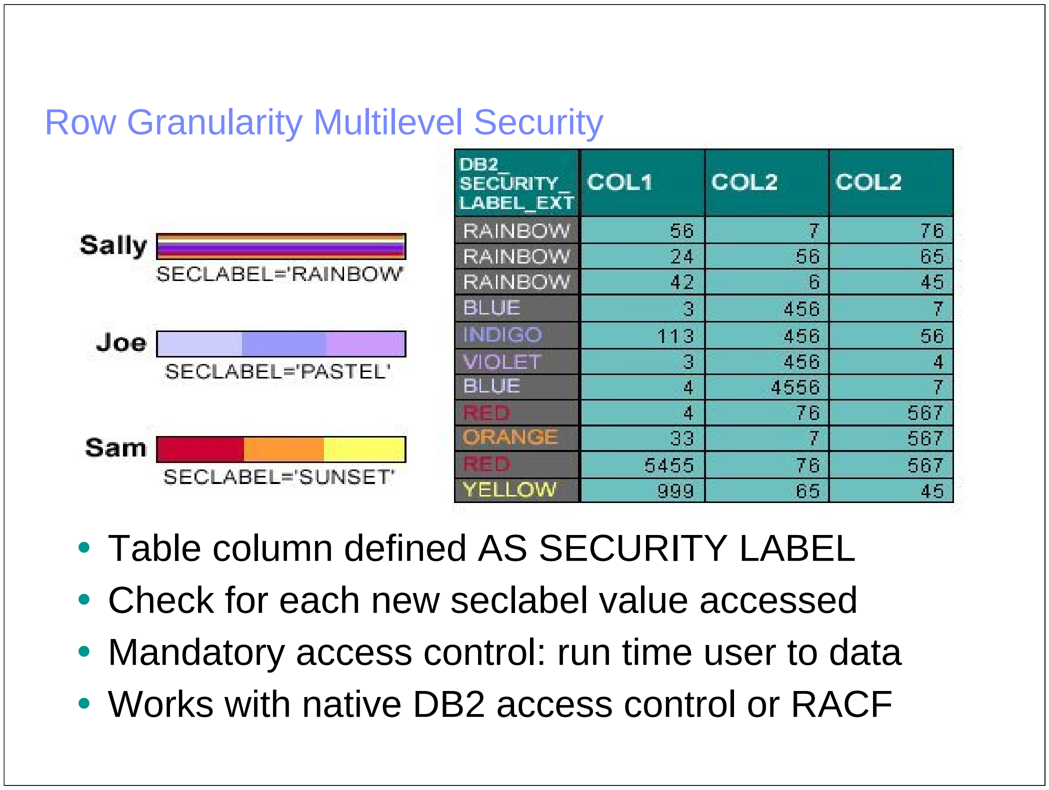
- SYSHIGH highest security level and all categories. Dominates all other security labels
- SYSLOW lowest security level and no categories (users)
- SYSNONE lowest security level and no categories (catalogs, not users)
- SYSMULTI equivalent to any security label
  - Can contain data with different security classifications, e.g. subsystem, row-level tables

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. 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.

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.

SYSMULTI: This label is considered to be equivalent to any defined security label. It is intended for use by multilevel security servers and tables that have multilevel data. See Planning for Multilevel Security for more.



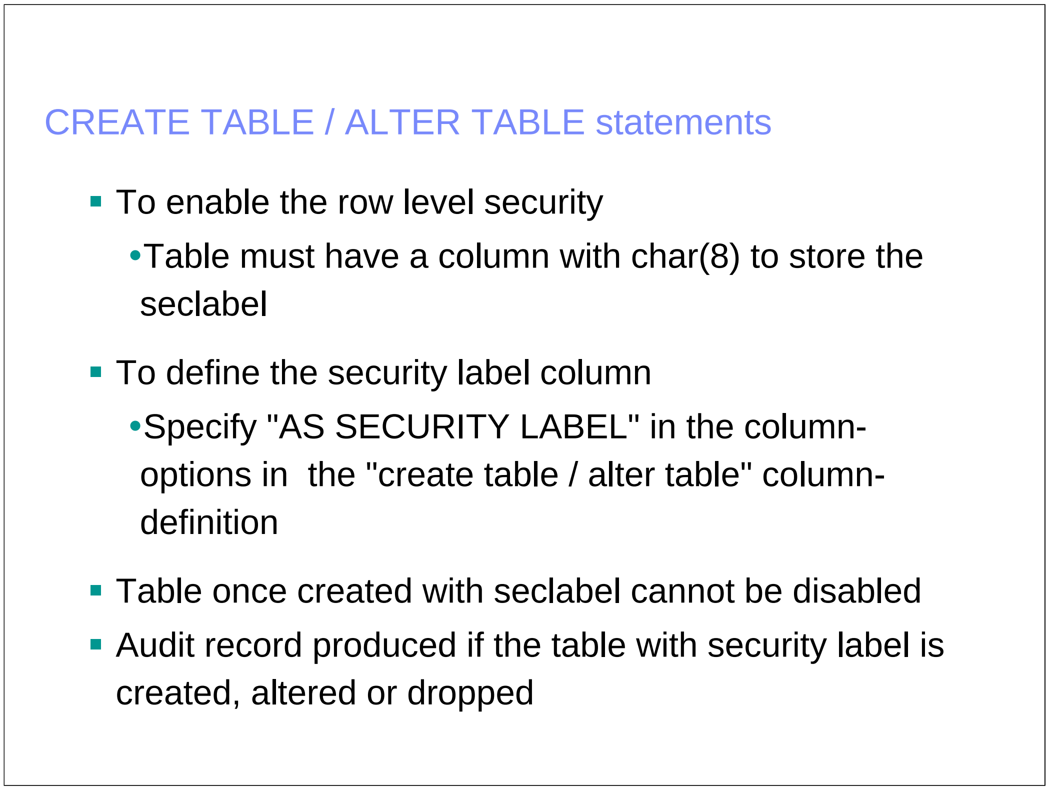
Row Granularity Multilevel Security

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
- Works with native DB2 access control or RACF

Customers asked for 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.

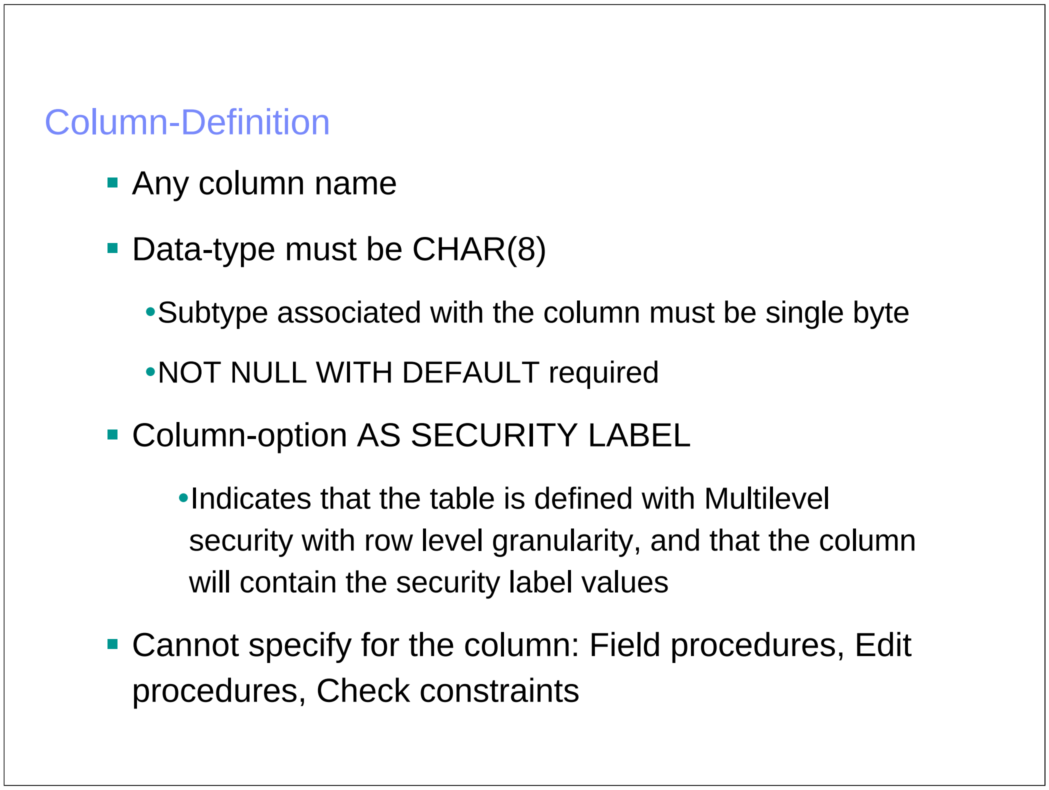
You can add a column that acts as the security label (seclabel), with a column defined AS SECURITY LABEL: Each row value has a specific seclabel. The seclabels are defined and provided by RACF for a user, then saved in rows for INSERT, UPDATE, LOAD, ... When rows are accessed, we check for each new seclabel 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 (MLS) requires z/OS V1R5 & Security Server (RACF).



CREATE TABLE / ALTER TABLE statements

- To enable the row level security
  - Table must have a column with char(8) to store the seclabel
- To define the security label column
  - Specify "AS SECURITY LABEL" in the column-options in the "create table / alter table" column-definition
- Table once created with seclabel cannot be disabled
- Audit record produced if the table with security label is created, altered or dropped

When you CREATE a table or ALTER it, you can decide to implement row-level security by including or adding a column that is specified AS SECURITY LABEL. The audit record is IFCID 0142.
The only technique to disable this security is to drop the table, table space or database.

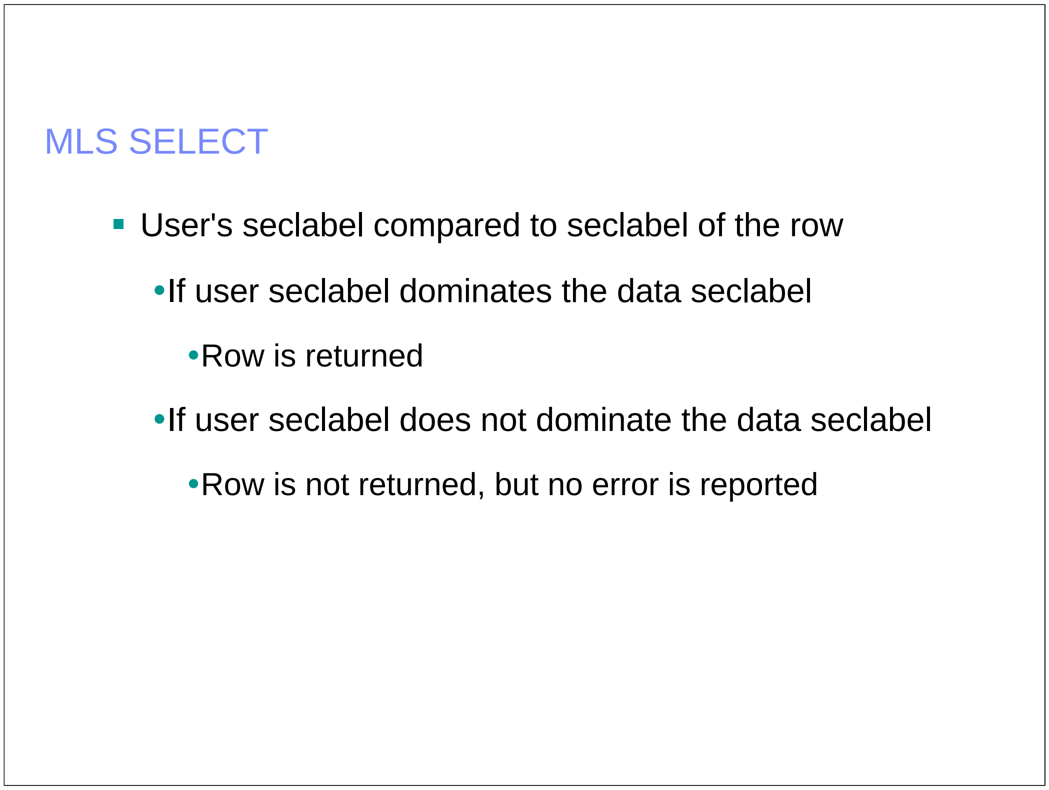


Column-Definition

- Any column name
- Data-type must be CHAR(8)
  - Subtype associated with the column must be single byte
  - NOT NULL WITH DEFAULT required
- Column-option AS SECURITY LABEL
  - Indicates that the table is defined with Multilevel security with row level granularity, and that the column will contain the security label values
- Cannot specify for the column: Field procedures, Edit procedures, Check constraints

Any column name can be the security label, but the same column name cannot be used more than once in the a table. Only one security label column is allowed in a table.

The security label column must be data type single byte character, char(8), NOT NULL WITH DEFAULT. This column cannot have field procedures, edit procedures or check constraints.



MLS SELECT

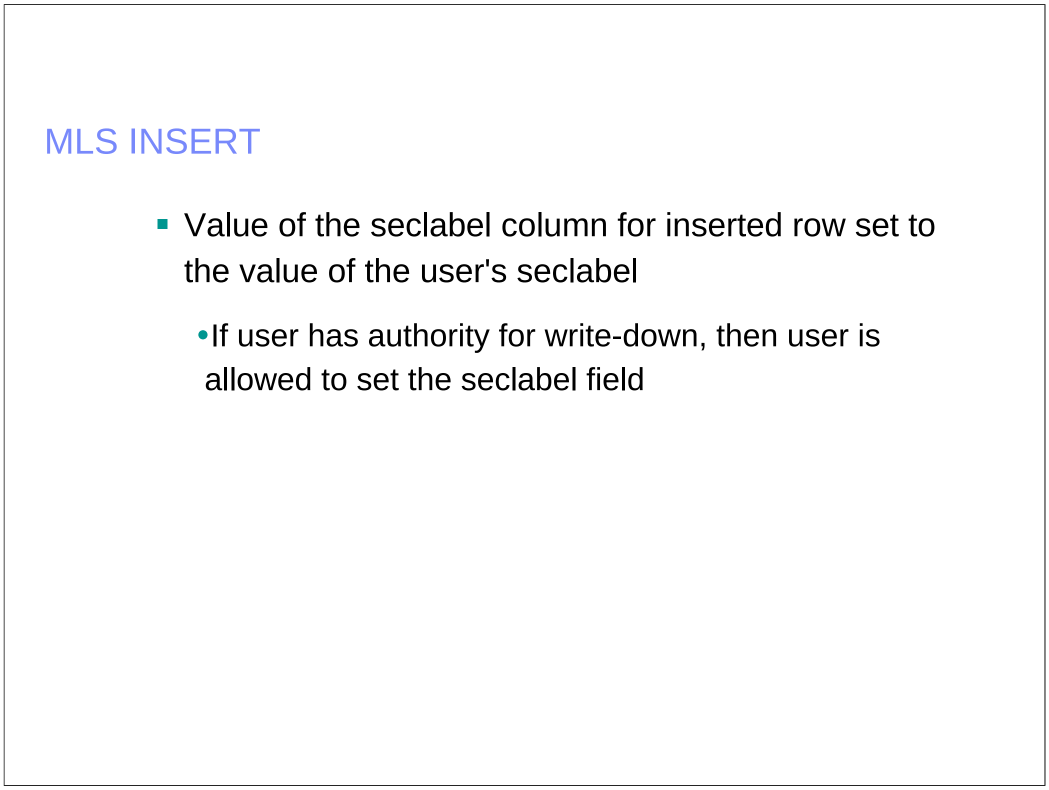
- User's seclabel compared to seclabel of the row
  - If user seclabel dominates the data seclabel
    - Row is returned
  - If user seclabel does not dominate the data seclabel
    - Row is not returned, but no error is reported

The security rule for select is that your current security label must dominate the security label of all the rows read. If your security label does not dominate the label of the data row, then that row is not returned.
The user's seclabel is compared to the data seclabel of the row to be selected.

If user seclabel dominates the data seclabel then the row is returned

If user seclabel does not dominate the data seclabel, then the row is not included in data returned, but no error is reported

User must be identified to Security Server with a valid seclabel. If not, authorization error and audit record produced (IFCID 0140).

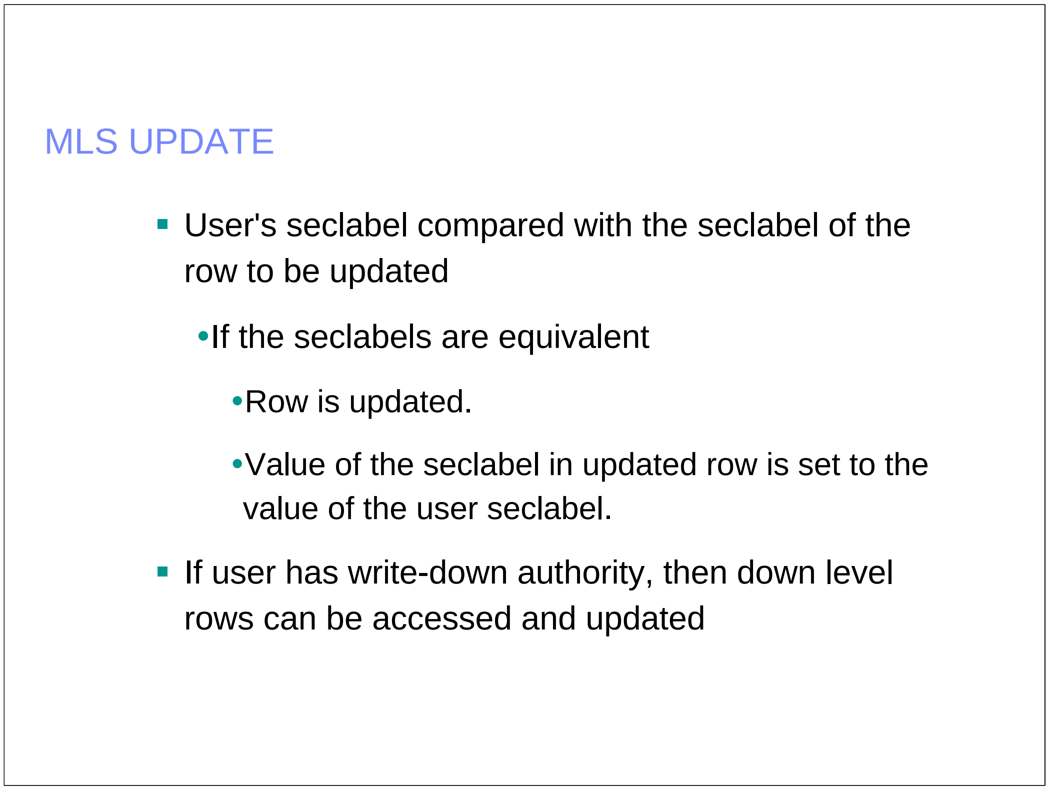


MLS INSERT

- Value of the seclabel column for inserted row set to the value of the user's seclabel
  - If user has authority for write-down, then user is allowed to set the seclabel field

The access rules for INSERT do not require checking, since there is no current row, but we will save the user's current seclabel as we insert a row.

If a user does not have the write-down privilege, then the seclabel of inserted rows will be exactly the current seclabel. If the user does have the write-down privilege, then he or she can set the value of the seclabel column to a seclabel to allow dominance or reverse dominance (writing up or down level), but not disjoint labels.



MLS UPDATE

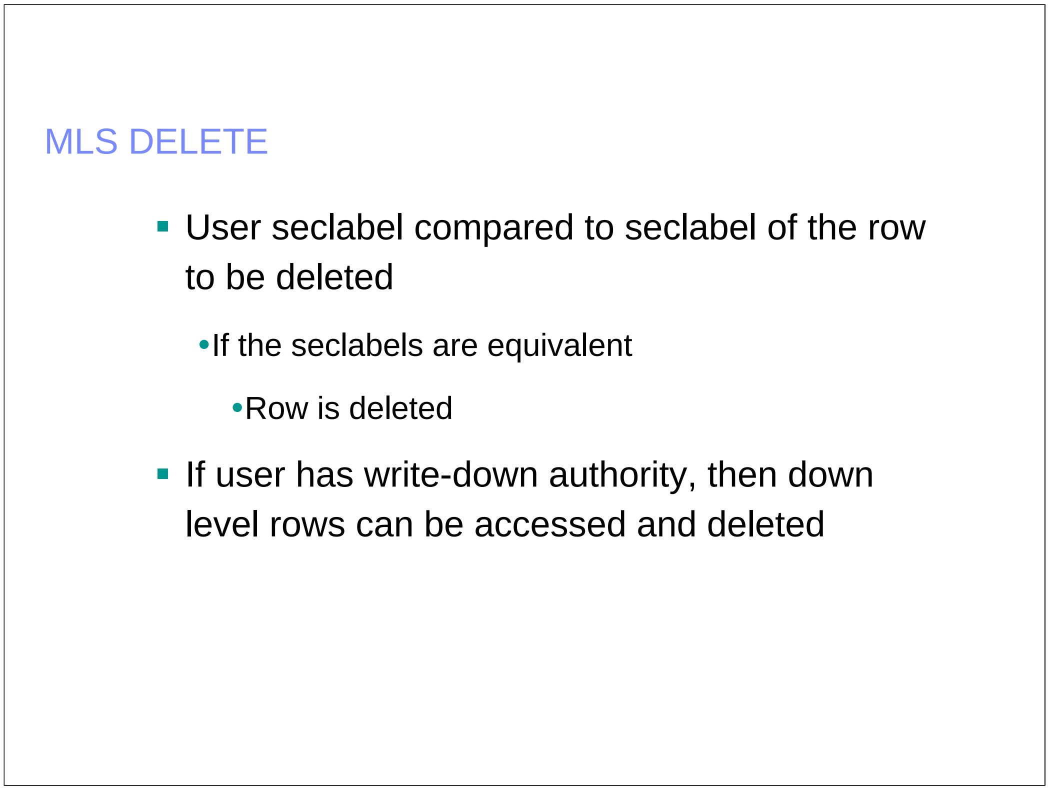
- User's seclabel compared with the seclabel of the row to be updated
  - If the seclabels are equivalent
    - Row is updated.
    - Value of the seclabel in updated row is set to the value of the user seclabel.
- If user has write-down authority, then down level rows can be accessed and updated

The rules for update are similar, with the SELECT rules for access to the data and setting the seclabel like INSERT. Update requires equivalence for users who are not allowed to write down.
User's seclabel is compared with the seclabel of the row to be updated

If the seclabels are equivalent then the row is updated. The value of the original data seclabel in the updated row is set to the value of the user seclabel.

If the user has write-down authority, then down level rows can be accessed and updated to a seclabel that has dominance or reverse dominance. Updating through a view with check option and a predicate specifying the seclabel is an exception, which can be used to force the seclabel to the user's current seclabel.

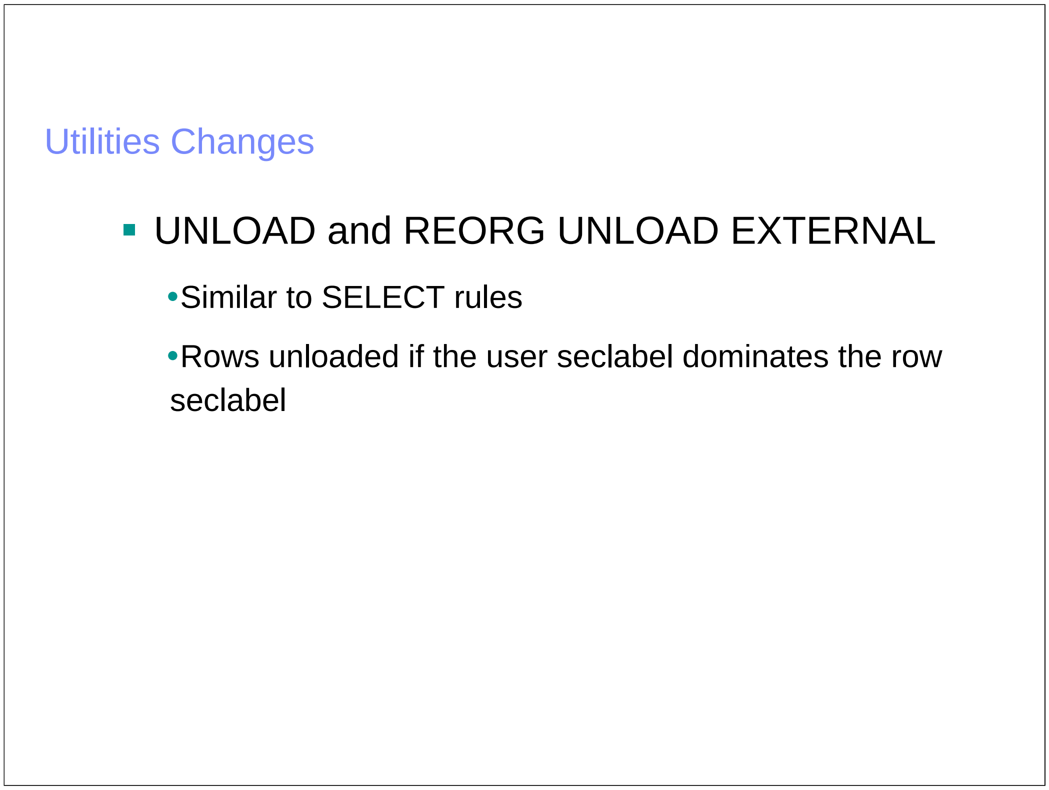
User must be identified to Security Server with a valid seclabel. If not, an authorization error and audit record are produced.



MLS DELETE

- User seclabel compared to seclabel of the row to be deleted
  - If the seclabels are equivalent
    - Row is deleted
- If user has write-down authority, then down level rows can be accessed and deleted

For DELETE, the seclabels must be equivalent, but we will not be recording the seclabel in the row, since the row is being deleted.
Again, a user who has write down authority can access and delete down-level (dominance) rows, but not up-level (reverse dominance) rows.

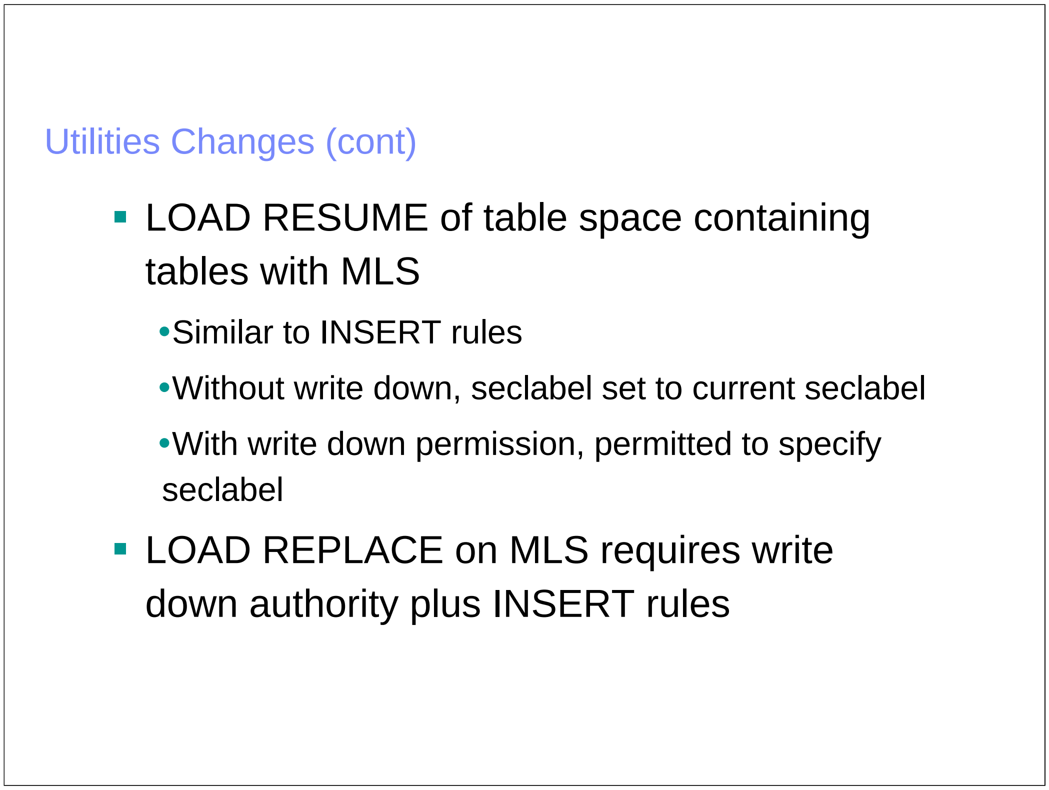


Utilities Changes

- UNLOAD and REORG UNLOAD EXTERNAL
  - Similar to SELECT rules
  - Rows unloaded if the user seclabel dominates the row seclabel

UNLOAD and REORG UNLOAD EXTERNAL use rules similar to SELECT. These utilitities read information, like the SELECT statement, so the authorization is similar to SELECT rules. Users must be identified to RACF and have a valid seclabel.

Rows can only be unloaded if the user seclabel dominates the data seclabel. No error returns if this is not true, but the row is not unloaded.



Utilities Changes (cont)

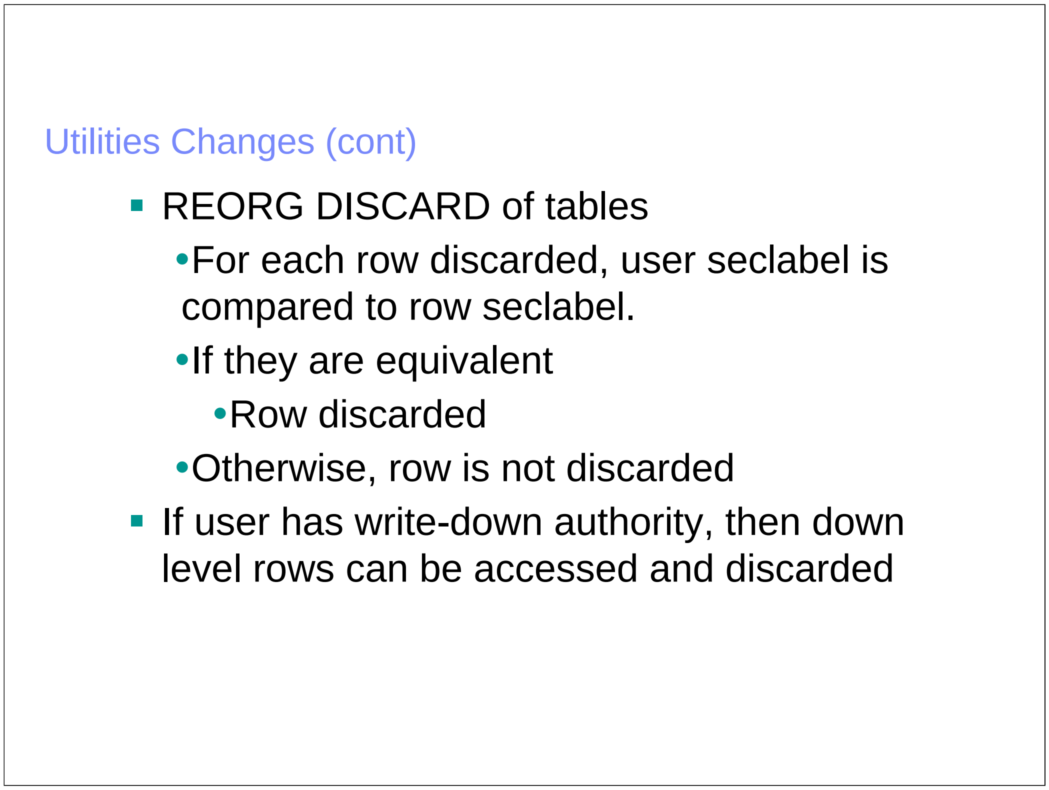
- LOAD RESUME of table space containing tables with MLS
  - Similar to INSERT rules
  - Without write down, seclabel set to current seclabel
  - With write down permission, permitted to specify seclabel
- LOAD REPLACE on MLS requires write down authority plus INSERT rules

LOAD RESUME is like INSERT for MLS rules. LOAD RESUME of a table space containing tables with MLS with row granularity is very similar to the rules for INSERT: Without write down, the row seclabel is set to user's current seclabel.

With write down permission, the user is permitted to specify a valid seclabel (dominance or reverse dominance), but not a disjoint one.

LOAD REPLACE on MLS row table requires write down authority.

LOAD REPLACE deletes all rows, so write down authority is required, but the user seclabel does not need to dominate all rows in the table. Then the insert does allow the user to set seclabel to dominance or reverse dominance.



Utilities Changes (cont)

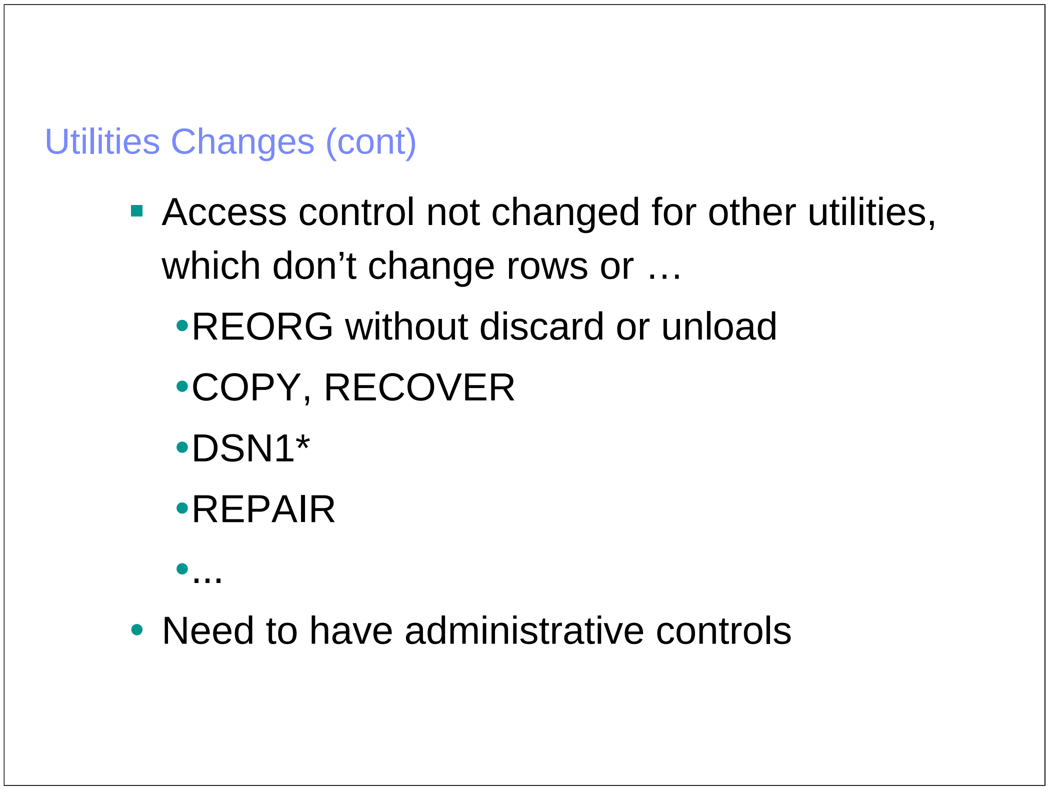
- REORG DISCARD of tables
  - For each row discarded, user seclabel is compared to row seclabel.
  - If they are equivalent
    - Row discarded
  - Otherwise, row is not discarded
- If user has write-down authority, then down level rows can be accessed and discarded

REORG DISCARD is similar to DELETE in function as well as in authorization rules. For each row to be discarded from those tables, if the row qualifies to be discarded, the user seclabel is compared to the data seclabel.

If the seclabels are equivalent, then the row is discarded.
Otherwise, row is not discarded

User must be identified to RACF and have a valid seclabel.

If the user has write-down authority, then the seclabel comparison is different. Rows that are dominated by the current user can be accessed and discarded.

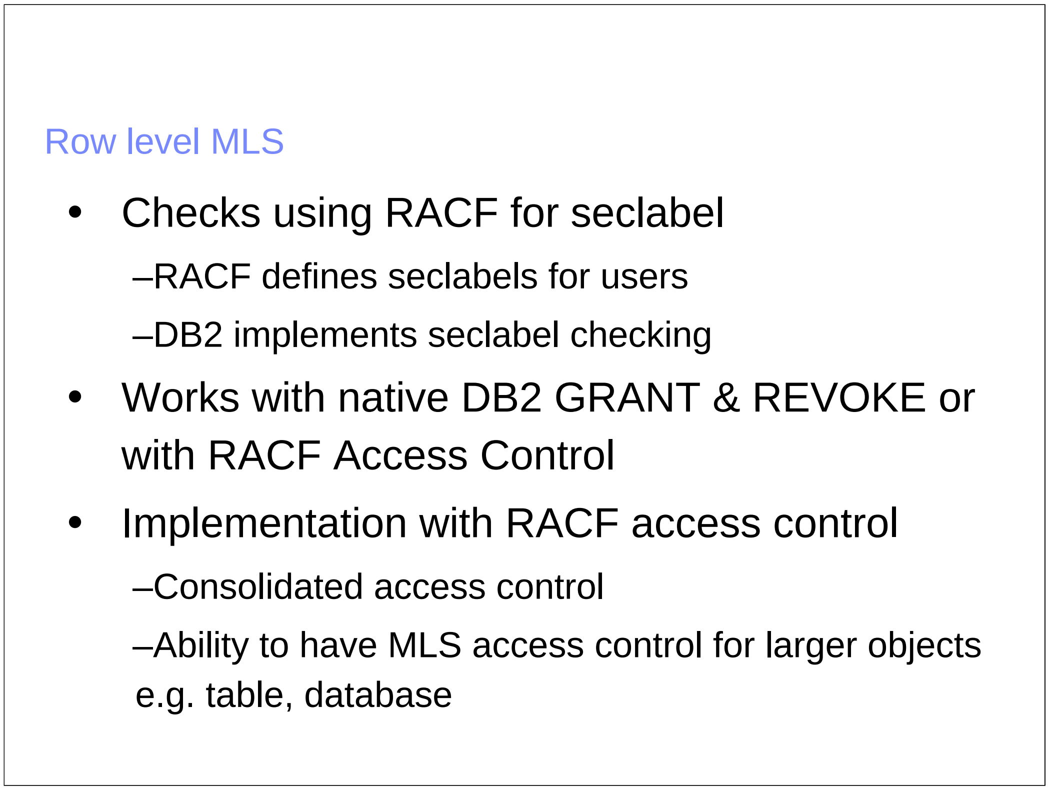


Utilities Changes (cont)

- Access control not changed for other utilities, which don’t change rows or …
  - REORG without discard or unload
  - COPY, RECOVER
  - DSN1*
  - REPAIR
  - ...
- Need to have administrative controls

The utilities which insert and delete data have the new multilevel secirity access controls, but other utilities are not changed. An administrator will generally be running these who has access to all of the data. Data sets for copies and work files need to be protected. DSN1* utilities require access control for the data sets.

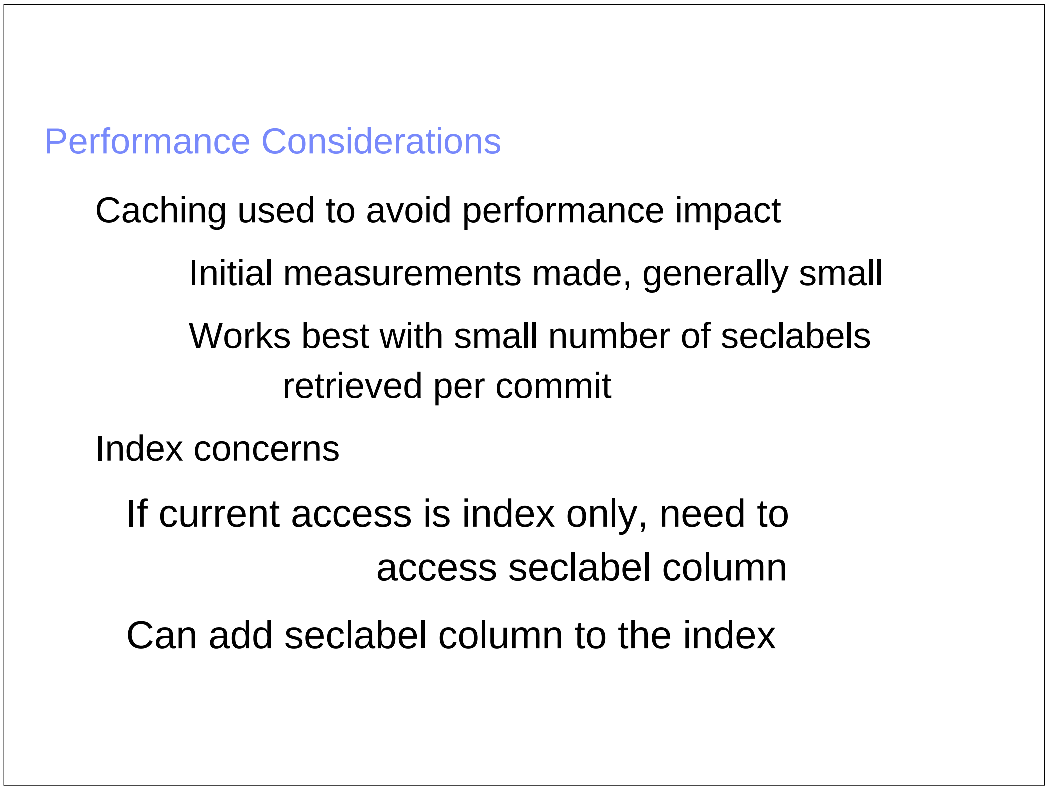
Use RACF controls for the data sets at the highest level of data within the data set. For better control of the other utilities, use RACF access control.



Row level MLS

- Checks using RACF for seclabel
  - –RACF defines seclabels for users
  - –DB2 implements seclabel checking
- Works with native DB2 GRANT & REVOKE or with RACF Access Control
- Implementation with RACF access control
  - –Consolidated access control
  - –Ability to have MLS access control for larger objects e.g. table, database

While row level MLS will use RACF to access the seclabels and to compare them, it does not depend upon using RACF access control (PERMIT). You can use row level MLS with native DB2 GRANT and REVOKE or with RACF PERMITs. Still, the integration is better with RACF access control, since that allows you to have a single consolidated source for the access control and to have MLS for objects like tables and databases. Row-level access controls can be used with native DB2 access controls or with RACF access controls. Use RACF access control for the data sets in any case.

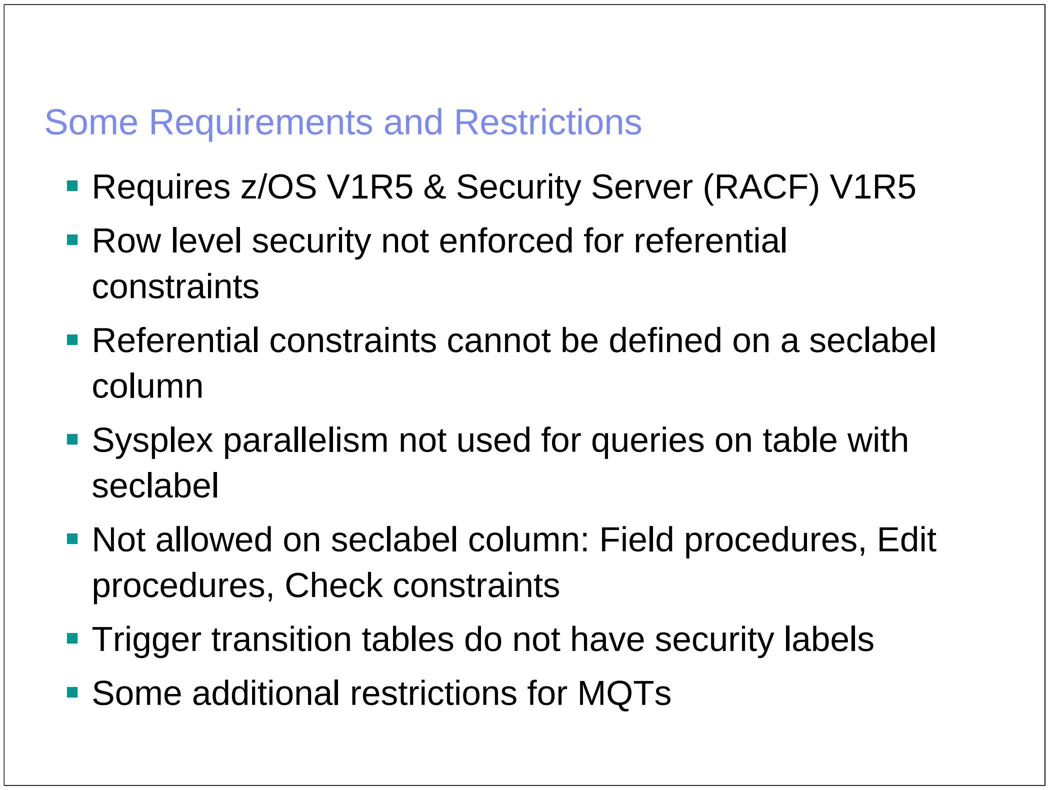


Performance Considerations

- Caching used to avoid performance impact
  - Initial measurements made, generally small
  - Works best with small number of seclabels retrieved per commit
- Index concerns
  - If current access is index only, need to access seclabel column
  - Can add seclabel column to the index

Caching is used to avoid extra calls to RACF. While the impact is not measured yet, the initial measurements of impact are small. The caching would work best if there are a relatively small number of seclabels to be checked compared with the number of rows accessed.

The seclabel column will need to be accessed always. If the current access is index-only, then adding this column would change to access the data as well. If index-only access is needed, then you should add the seclabel column to the index. This would affect uniqueness, but this change may be useful to avoid inference issues.

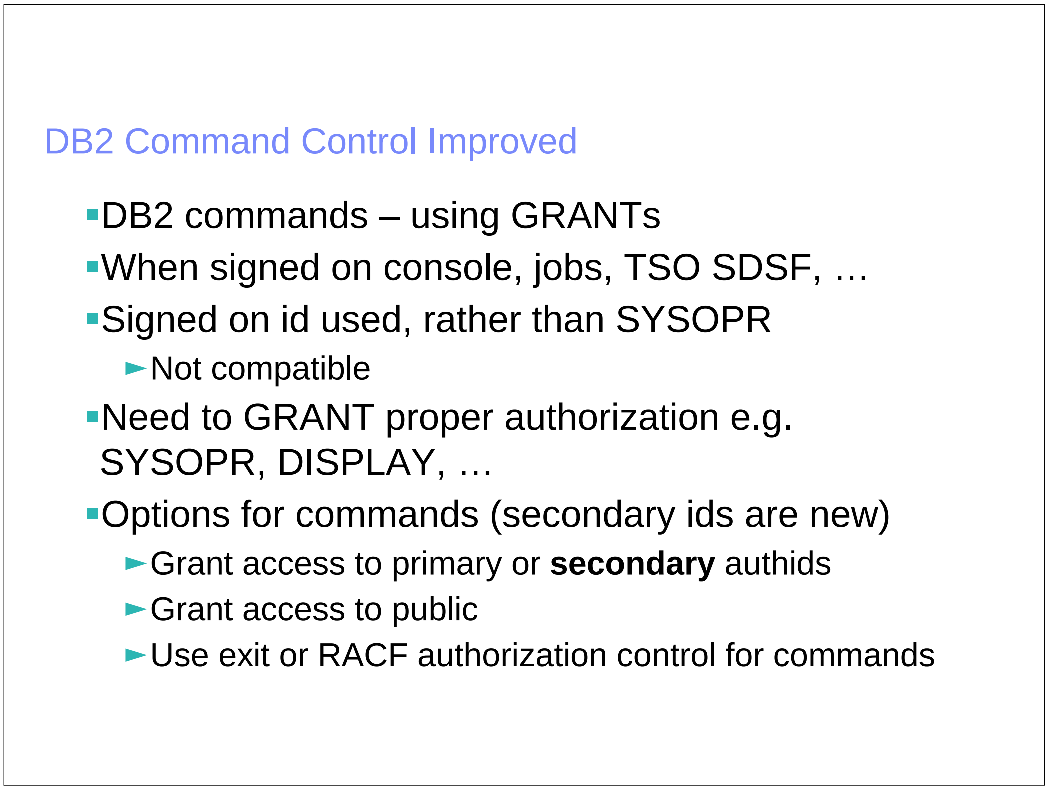


Some Requirements and Restrictions

- Requires z/OS V1R5 & Security Server (RACF) V1R5
- Row level security not enforced for referential constraints
- Referential constraints cannot be defined on a seclabel column
- Sysplex parallelism not used for queries on table with seclabel
- Not allowed on seclabel column: Field procedures, Edit procedures, Check constraints
- Trigger transition tables do not have security labels
- Some additional restrictions for MQTs

Note that there are requirements for z/OS V1R5 and the Security Server (RACF) V1R5 (or equivalent function).
There are some additional restrictions: Row level security is not enforced for referential constraint checking.

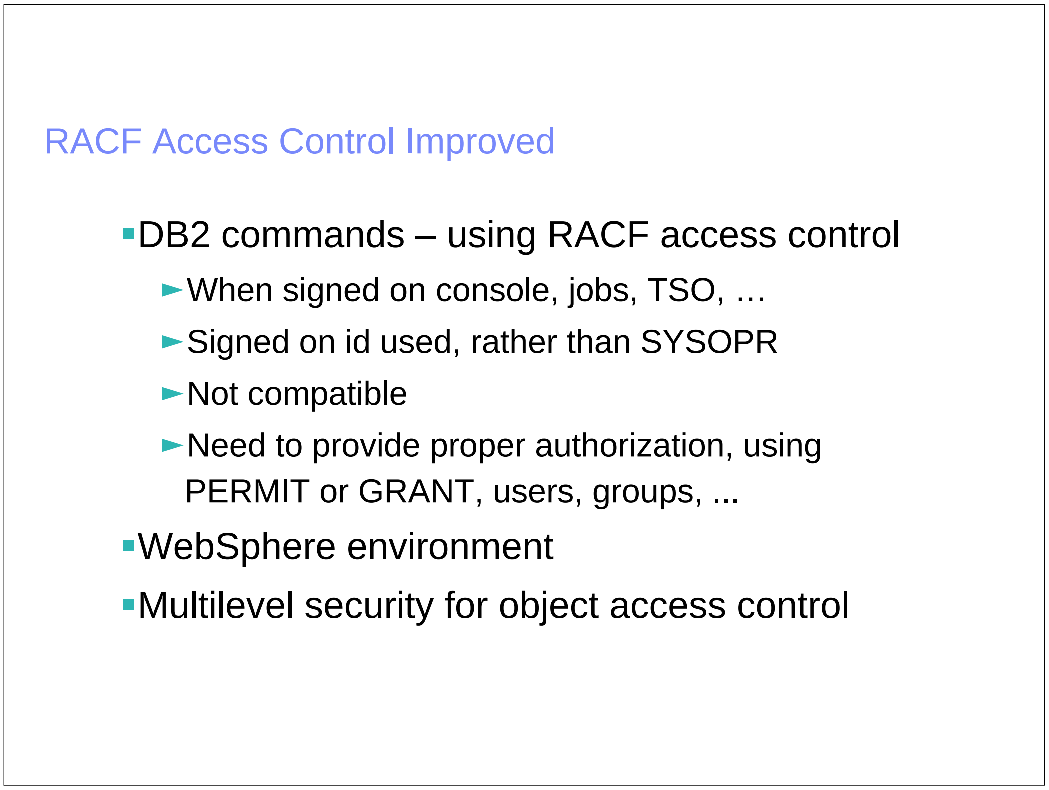
Referential constraints cannot be defined on a security label column. Sysplex parallelism is not used for queries that access a table with a security label column. Field procedures and edit procedures are not allowed on a security label column. Trigger transition tables do not have security labels.



DB2 Command Control Improved

- DB2 commands – using GRANTs
- When signed on console, jobs, TSO SDSF, …
- Signed on id used, rather than SYSOPR
  - Not compatible
- Need to GRANT proper authorization e.g. SYSOPR, DISPLAY, …
- Options for commands (secondary ids are new)
  - Grant access to primary or **secondary** authids
  - Grant access to public
  - Use exit or RACF authorization control for commands

Access control is improved for DB2 commands, whether or not RACF access control is used. While there is an improvement, the change is not completely compatible, and customers need to be sure that the appropriate authorization is GRANTed. Grants could only use the primary authorization id for commands before, but the ability to use secondary ids is added in V8. Some alternatives would be granting access to the individual ids, granting access to public or using the Access Control Authorization Exit or RACF control of commands (see next page).



RACF Access Control Improved

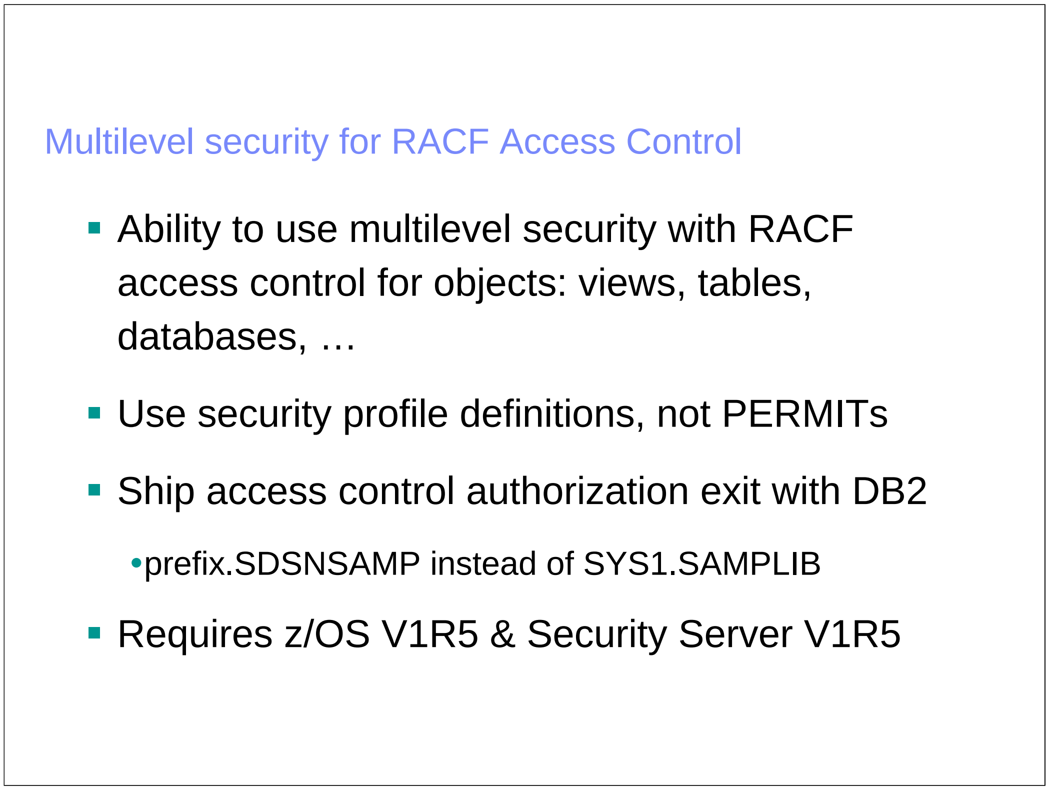
- DB2 commands – using RACF access control
  - When signed on console, jobs, TSO, …
  - Signed on id used, rather than SYSOPR
  - Not compatible
  - Need to provide proper authorization, using PERMIT or GRANT, users, groups, ...
- WebSphere environment
- Multilevel security for object access control

RACF access control has been improved in several ways. DB2 operator commands are able to use RACF access control for the first time. If the DB2 command is issued in an environment that has an ACEE, then RACF access control can be used. Signed on consoles do have an ACEE, but others do not. Jobs and TSO environments generally have an ACEE.

Access control is improved for DB2 commands, whether or not RACF access control is used. RACF has not been able to control access for DB2 commands in the past. While there is an improvement, the change is not completely compatible, and customers need to be sure that the appropriate authorization is GRANTed or PERMITted. Authorization with RACF is improved for the WebSphere environment and for use of multilevel security.

The WebSphere environment is better-managed, with more robust handling of the ACEE in that environment.

RACF access control is enhanced with the ability to have multilevel security. Having MLS at a table level and a database level provides complementary function to the new row level MLS.

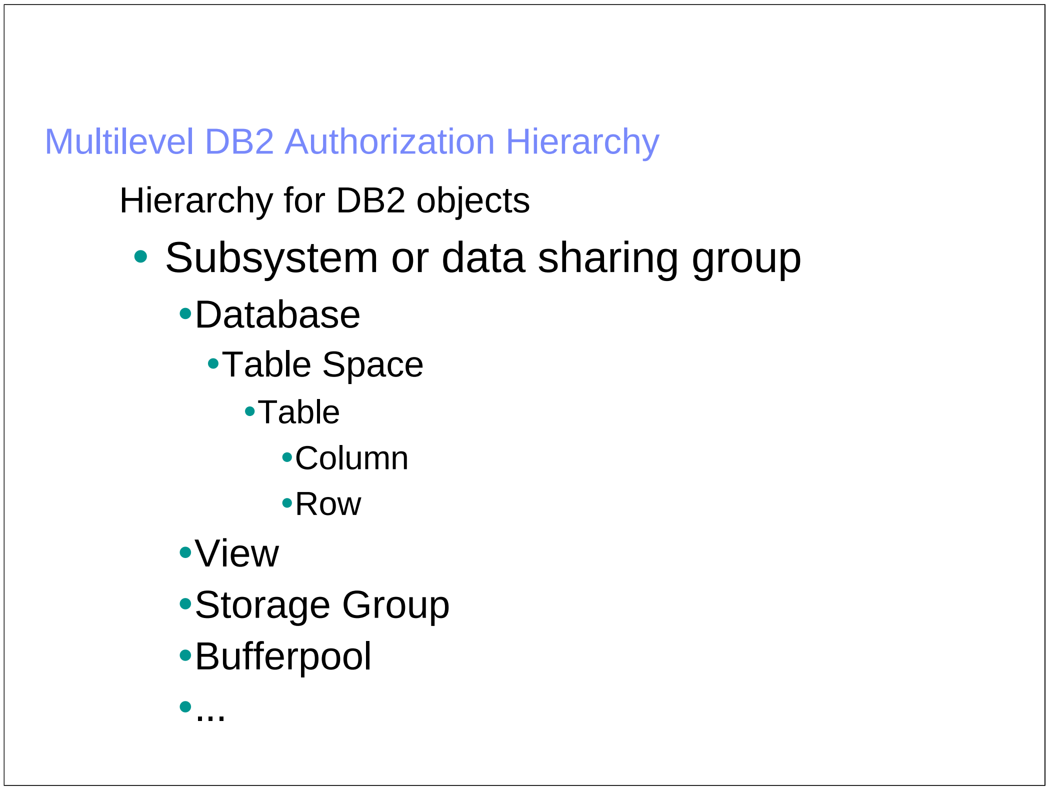


Multilevel security for RACF Access Control

- Ability to use multilevel security with RACF access control for objects: views, tables, databases, …
- Use security profile definitions, not PERMITs
- Ship access control authorization exit with DB2
  - prefix.SDSNSAMP instead of SYS1.SAMPLIB
- Requires z/OS V1R5 & Security Server V1R5

If you use RACF access controls, then you can define multilevel security for other objects. Then, access will require both the discretionary access control (PERMIT) and the mandatory access control (seclabel comparison).
While earlier exits came with RACF in SYS1.SAMPLIB, the new exit comes with DB2 in prefix.SDSNSAMP.

The RACF Access Control Module Guide book is available with other DB2 books on the DB2 UDB for z/OS Library web pages.

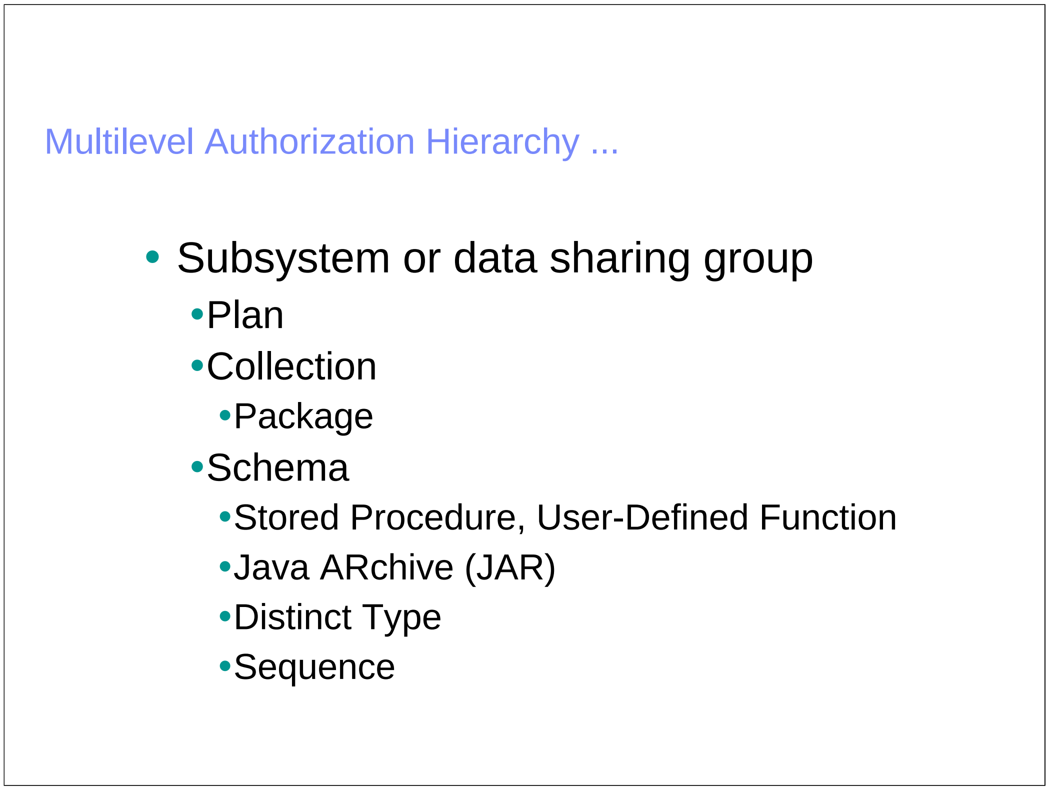


Multilevel DB2 Authorization Hierarchy

Hierarchy for DB2 objects

- Subsystem or data sharing group
  - Database
    - Table Space
      - Table
        - Column
        - Row
  - View
  - Storage Group
  - Bufferpool
  - ...

MLS security can be defined for the DB2 objects, but in general, you will want the seclabel of an object higher in the object hierarchy to dominate all objects within it. There will be some exceptions, similar to a write-down capability. Managing the relationships among DB2 objects is still a manual process. Using RACF should allow the number of seclabels and permits to be small enough to manage this way, because of groups and generic authority.

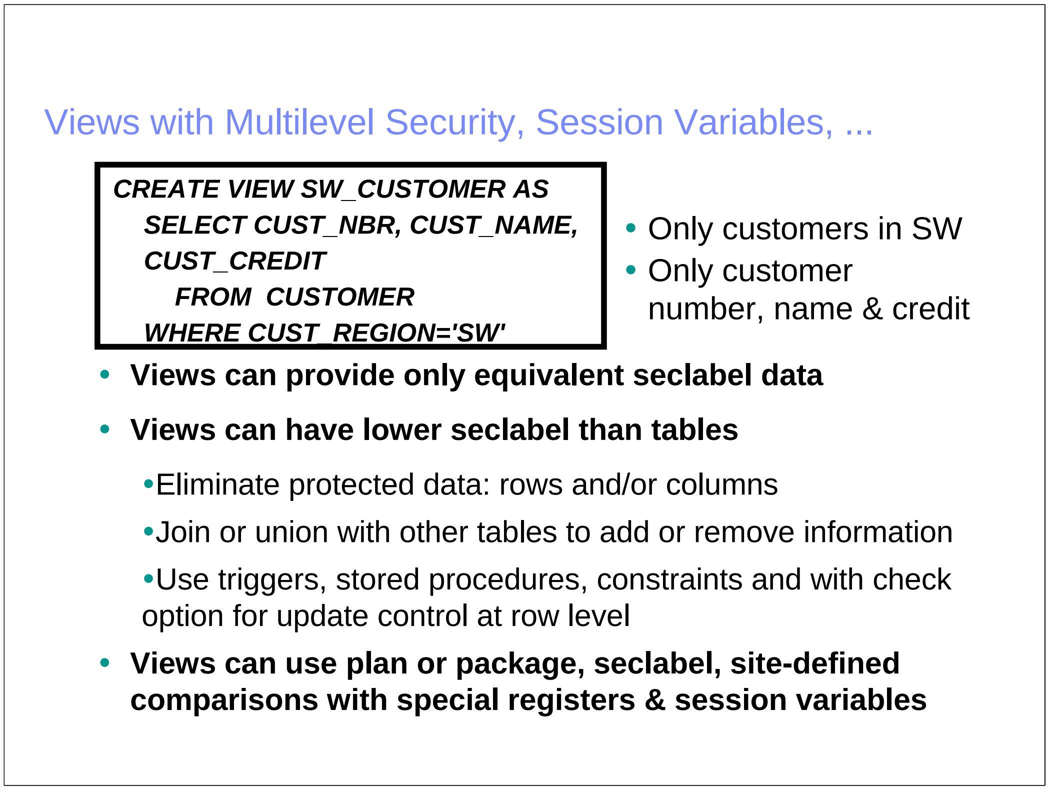


Multilevel Authorization Hierarchy ...

- Subsystem or data sharing group
  - Plan
  - Collection
    - Package
  - Schema
    - Stored Procedure, User-Defined Function
    - Java ARchive (JAR)
    - Distinct Type
    - Sequence

This is the second half of the hierarchy. While this hierarchy is not enforced, meaningful authorization rules will generally require that the higher level in this hierarchy have a seclabel which dominates the objects lower in the hierarchy.

In some cases, you may need to use some of the system built-in seclabels, such as SYSMULTI.



Views with Multilevel Security, Session Variables, ...

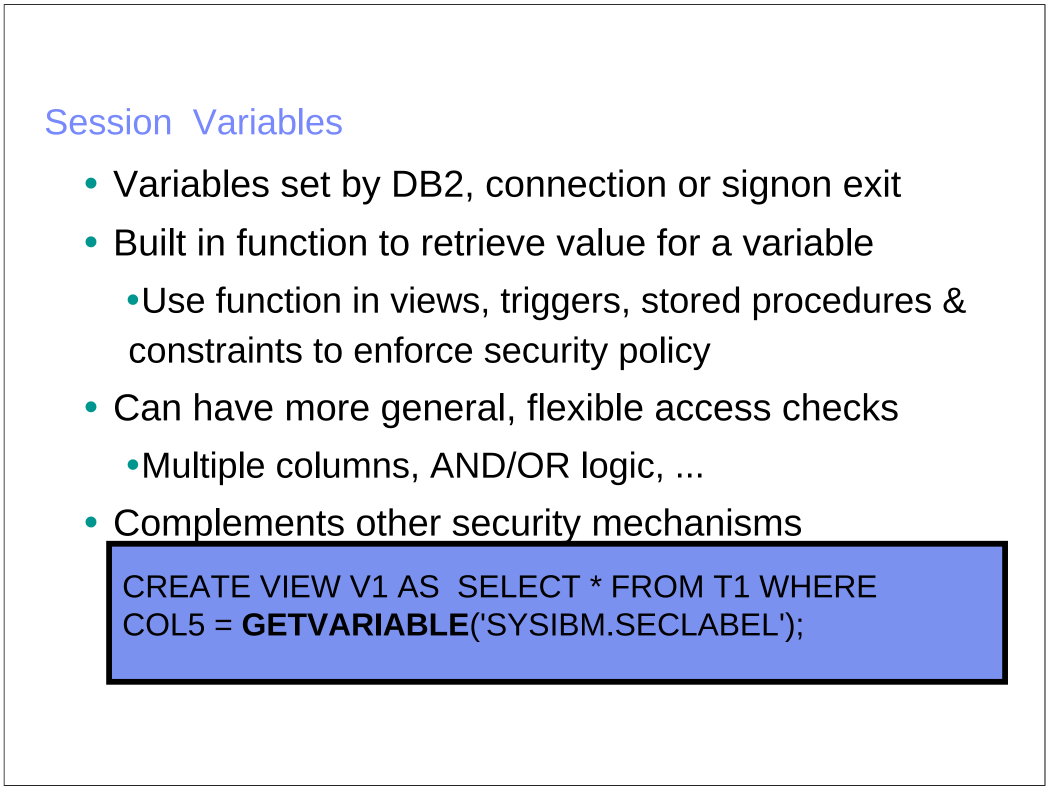
```
CREATE VIEW SW_CUSTOMER AS
   SELECT CUST_NBR, CUST_NAME,
   CUST_CREDIT
      FROM  CUSTOMER
   WHERE CUST_REGION='SW'
```

- Only customers in SW
- Only customer number, name & credit

- **Views can provide only equivalent seclabel data**
- **Views can have lower seclabel than tables**
  - Eliminate protected data: rows and/or columns
  - Join or union with other tables to add or remove information
  - Use triggers, stored procedures, constraints and with check option for update control at row level
- **Views can use plan or package, seclabel, site-defined comparisons with special registers & session variables**

Views can be used to hide data. They can subset to provide only certain columns or fields. Views are often used to simplify access to data by being able to define the view once and use it many times.

Table privileges DELETE, INSERT, SELECT, UPDATE, or ALL can be granted individually on a view. By creating a view and granting privileges on it, you can give someone access only to a specific combination of data. This capability is sometimes called field-level access control or field-level sensitivity.



Session Variables

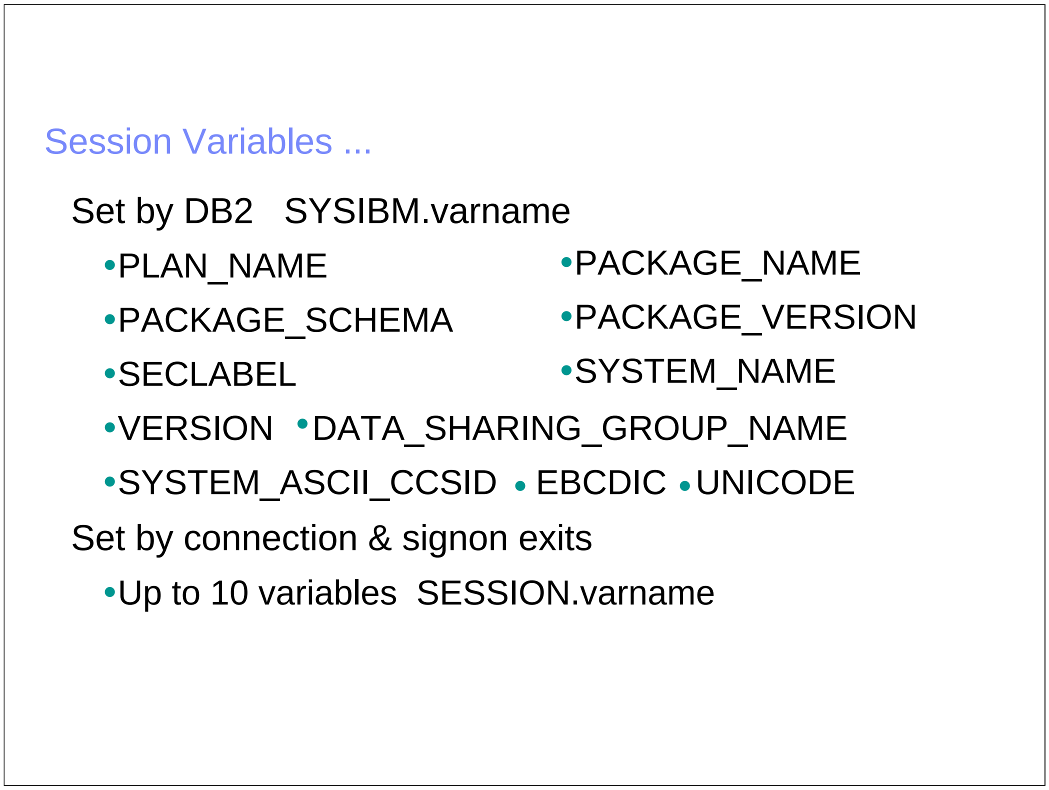
- Variables set by DB2, connection or signon exit
- Built in function to retrieve value for a variable
  - Use function in views, triggers, stored procedures & constraints to enforce security policy
- Can have more general, flexible access checks
  - Multiple columns, AND/OR logic, ...
- Complements other security mechanisms

```
CREATE VIEW V1 AS  SELECT * FROM T1 WHERE
COL5 = GETVARIABLE('SYSIBM.SECLABEL');
```

Session Variables provide another way to provide information to applications. Some variables will be set by DB2. Others can be set in the connection and signon exits to set these session variables

A new built-in function GETVARIABLE is added to retrieve the values of a session variable. This function can be used in views, triggers, stored procedures and constraints to help enforce a security policy. If your primary security need is more general, flexible controls, this information complements other security mechanisms.

For example, you can have a view which provides data that is at the user's current security label.



Session Variables ...

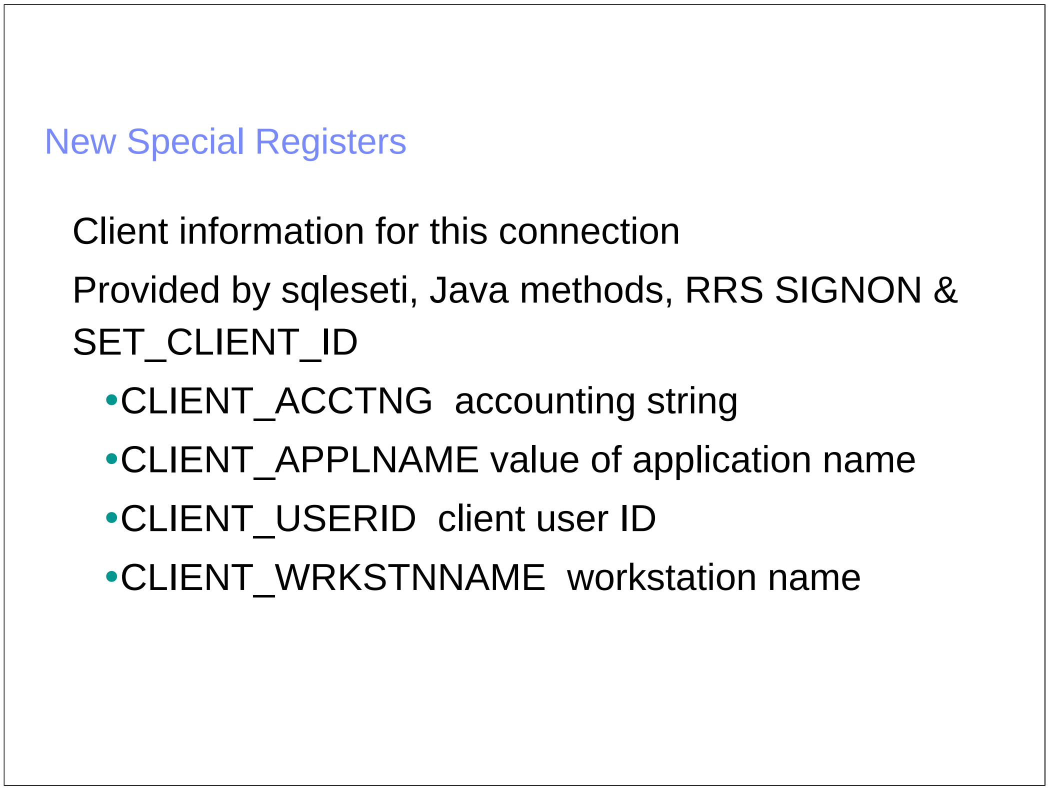
Set by DB2   SYSIBM.varname

- PLAN_NAME
- PACKAGE_NAME
- PACKAGE_SCHEMA
- PACKAGE_VERSION
- SECLABEL
- SYSTEM_NAME
- VERSION
- DATA_SHARING_GROUP_NAME
- SYSTEM_ASCII_CCSID
- EBCDIC
- UNICODE

Set by connection & signon exits

- Up to 10 variables SESSION.varname

The session variables set by DB2 are qualified by SYSIBM. You can get the plan name, package name, the user's seclabel, DB2 subsystem version and CCSID information. This information is useful for security controls, but programmers have other needs for this information as well. Customers can add up to ten variables, with the qualifier SESSION, by setting the name and value in the connection and signon exits. Both the name and the value allow up to 128 characters. Session variables can be accessed, but not changed, in applications.



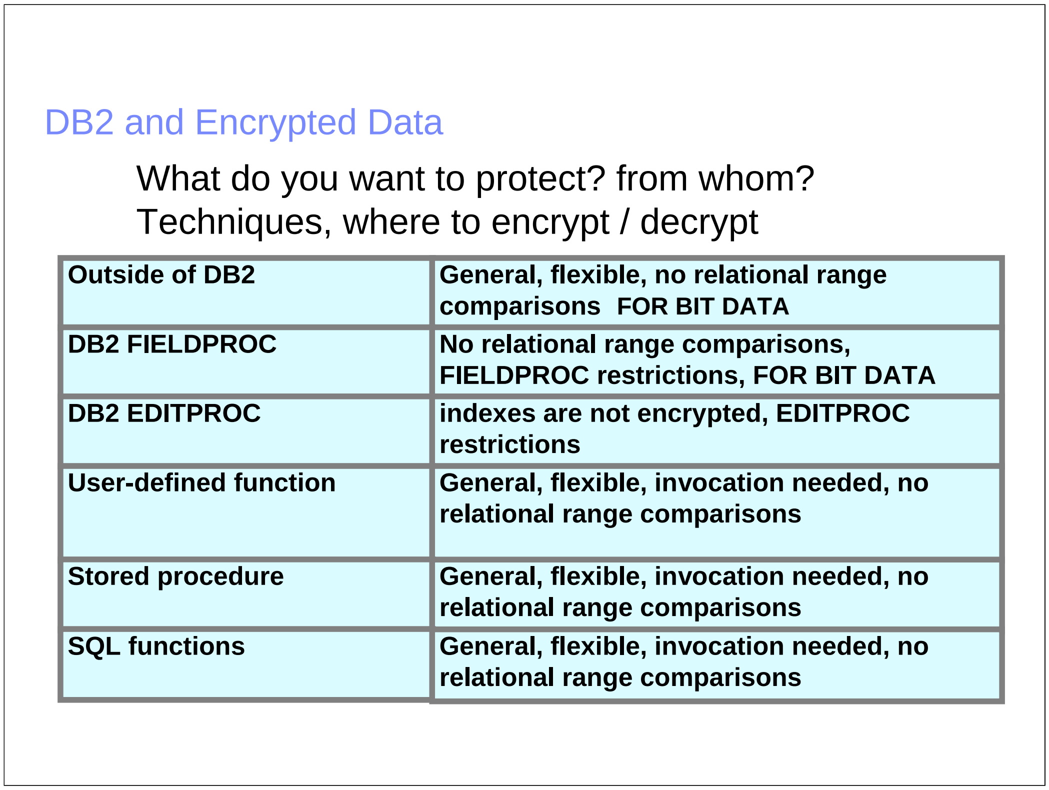
New Special Registers

Client information for this connection

Provided by sqleseti, Java methods, RRS SIGNON & SET_CLIENT_ID

- CLIENT_ACCTNG accounting string
- CLIENT_APPLNAME value of application name
- CLIENT_USERID client user ID
- CLIENT_WRKSTNNAME workstation name

Four new SPECIAL REGISTERS are added to the product. These special registers are CLIENT_ACCTNG, CLIENT_APPLNAME, CLIENT_USERID, and CLIENT_WRKSTNNAME. The information is provided through a number of application programming interfaces.
Similar special registers (without the underscore) were added to DB2 UDB for Linux, UNIX & Windows, V8. Since applications can change the information, it is not as secure.

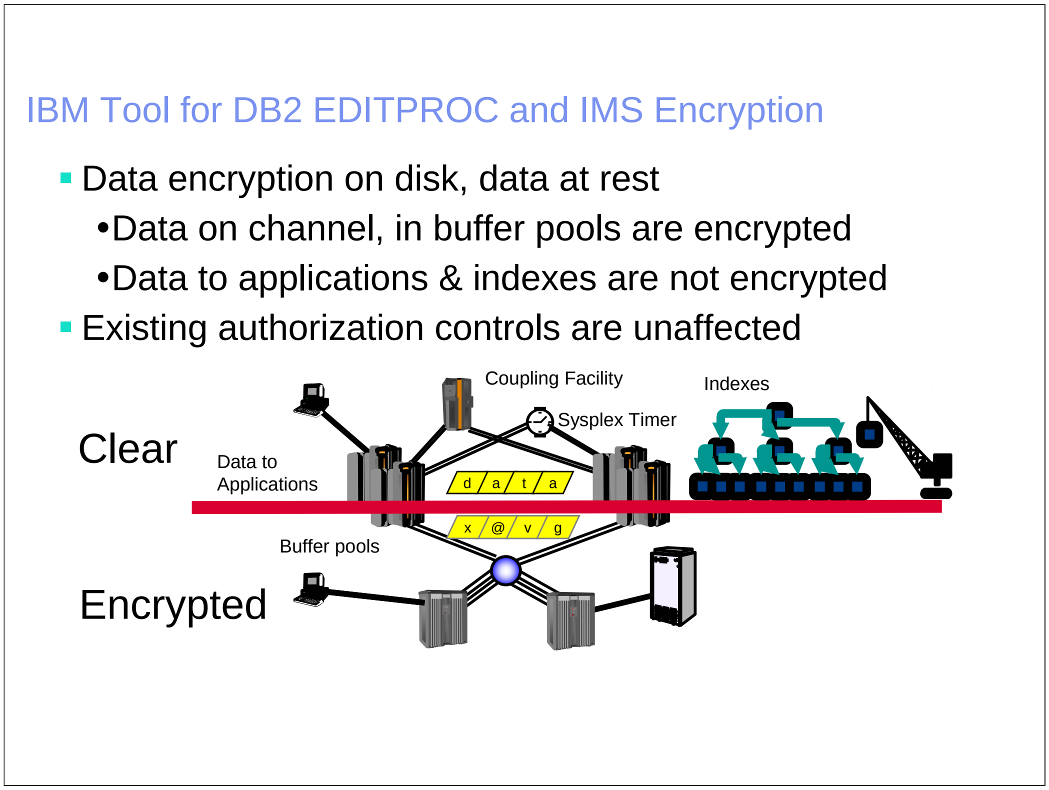


DB2 and Encrypted Data

What do you want to protect? from whom?
Techniques, where to encrypt / decrypt

Outside of DB2	**General, flexible, no relational range comparisons FOR BIT DATA**
DB2 FIELDPROC	**No relational range comparisons, FIELDPROC restrictions, FOR BIT DATA**
DB2 EDITPROC	**indexes are not encrypted, EDITPROC restrictions**
User-defined function	**General, flexible, invocation needed, no relational range comparisons**
Stored procedure	**General, flexible, invocation needed, no relational range comparisons**
SQL functions	**General, flexible, invocation needed, no relational range comparisons**

There are a number of ways to encrypt data in DB2. The answers to the questions, "What do you want to protect and from whom?" are generally needed to determine which technique to use and where to encrypt and decrypt. Encryption does mean some tradeoffs in function, usability and performance. Either the indexes are not encrypted or encrypted data will not give correct results for comparisons other than equals or not equals. All greater than, less than and range predicates are not usable. An EDITPROC is the most used technique, and one is provided to use cryptography hardware with an IBM Tool. The CMOS Cryptographic Coprocessors feature and ICRF hardware were designed to address high volume cryptographic transaction rates and bulk security requirements on z/OS and OS/390. The Integrated Cryptographic Service Facility provides the interface to service routines supported by the hardware, such as key management.



IBM Tool for DB2 EDITPROC and IMS Encryption

- Data encryption on disk, data at rest
  - •Data on channel, in buffer pools are encrypted
  - •Data to applications & indexes are not encrypted
- Existing authorization controls are unaffected

Coupling Facility

Indexes

Sysplex Timer

Clear

Data to Applications

d / a / t / a

x / @ / v / g

Buffer pools

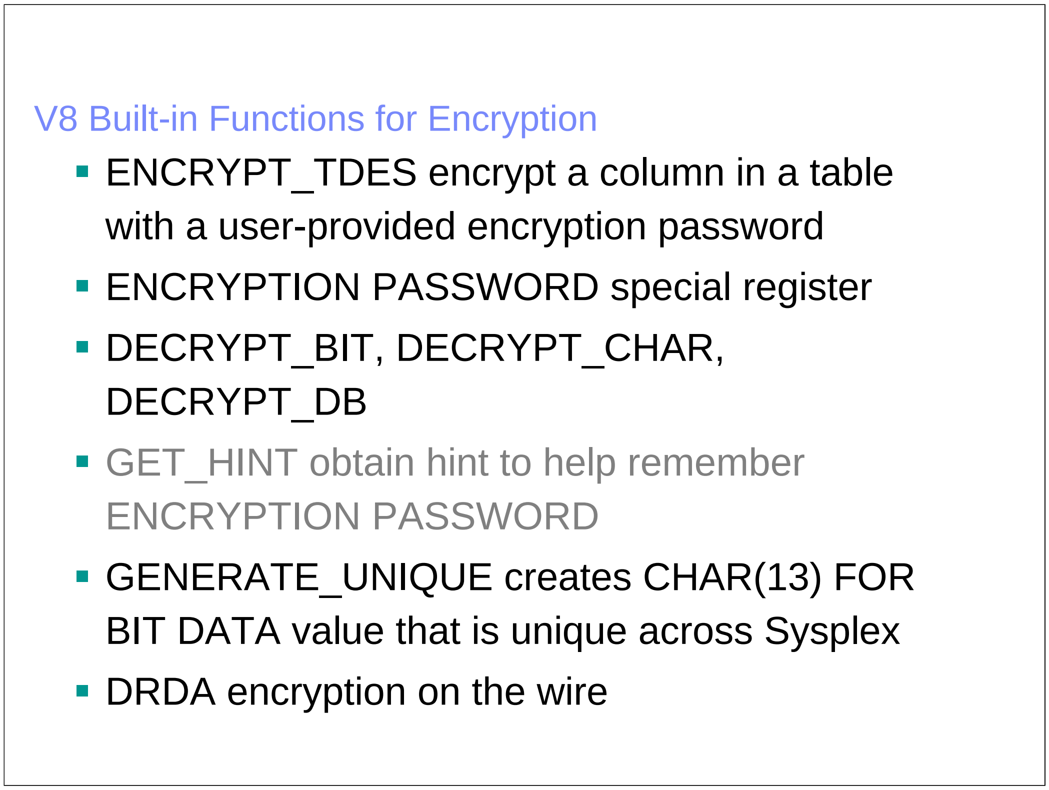
Encrypted

On this slide, data above the middle line is not encrypted and data below the line is encrypted.

In this example, the encryption is not providing an additional level of security for your DB2 or IMS applications. It is providing encryption of the data on the disk.

Once the data is brought, for example, into DBMS working storage, you are using the existing DB2 and IMS authorizations to secure data. As the data is written out to disk, that is once it leaves the channel and enters the network to move to the disk, the data is encrypted.

The example on this slide shows two subsystems with disks. The circle on the bottom half of the picture might be what we have known as an ESCON director in the past. The processor on the right hand side, below the line, might also be attached to that same I/O device; however, if the processor is a zSeries system that does not have the encryption key it will not be able to read the data.

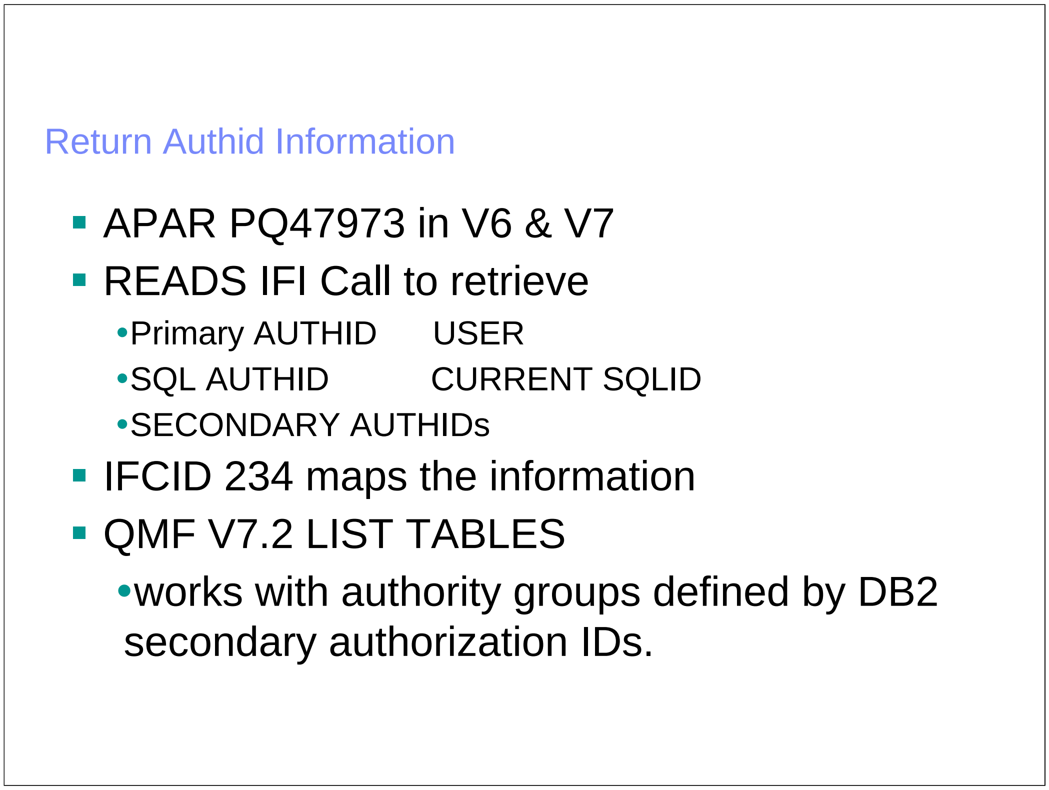


V8 Built-in Functions for Encryption

- ENCRYPT_TDES encrypt a column in a table with a user-provided encryption password
- ENCRYPTION PASSWORD special register
- DECRYPT_BIT, DECRYPT_CHAR, DECRYPT_DB
- GET_HINT obtain hint to help remember ENCRYPTION PASSWORD
- GENERATE_UNIQUE creates CHAR(13) FOR BIT DATA value that is unique across Sysplex
- DRDA encryption on the wire

Functions ENCRYPT_TDES (triple DES), DECRYPT_BIN, DECRYPT_CHAR, and GETHINT are added. The SET ENCRYPTION PASSWORD statement allows the application to specify a password
The ability to generate a unique value is also included. These changes came in DB2 for Linux, UNIX and Windows V8, so this change improves DB2 family consistency.

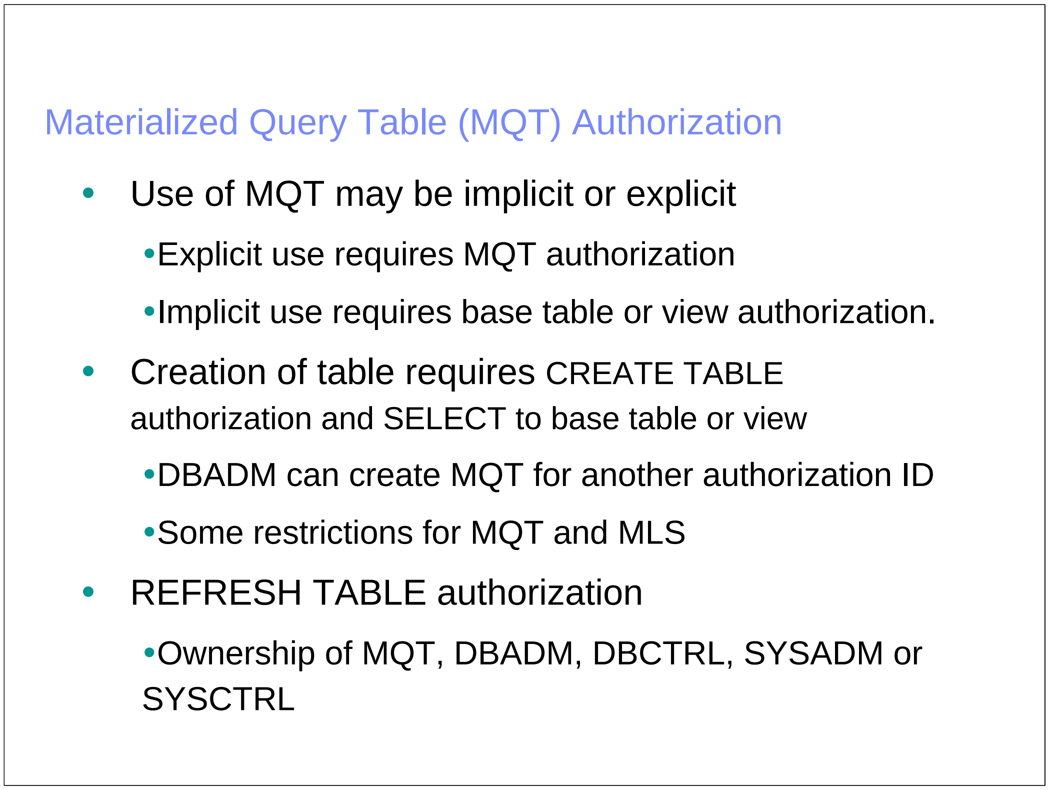
DRDA is extended to allow encryption of the data being sent. The DB2 Connect change is provided in V8.2.



Return Authid Information

- APAR PQ47973 in V6 & V7
- READS IFI Call to retrieve
  - Primary AUTHID   USER
  - SQL AUTHID   CURRENT SQLID
  - SECONDARY AUTHIDs
- IFCID 234 maps the information
- QMF V7.2 LIST TABLES
  - works with authority groups defined by DB2 secondary authorization IDs.

A function was added to DB2 V6 & V7 via APAR PQ47973 in late 2001. Customers have been asking for a technique that will return the list of secondary authids to a program.
Customers can use the Instrumentation Facility interface or IFI to retrieve this information with a synchronous READS call. QMF V7.2 uses this function in the LIST TABLES command and provides a table UDF which makes the secondary ids available in SQL.



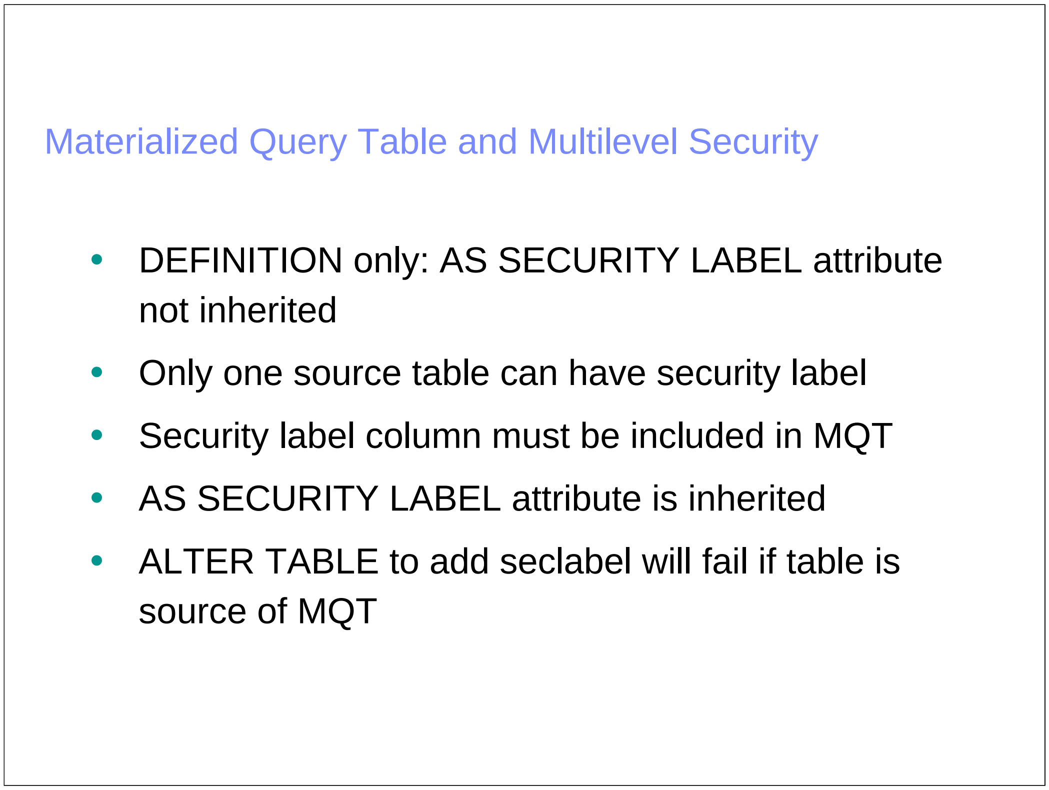
Materialized Query Table (MQT) Authorization

- Use of MQT may be implicit or explicit
  - •Explicit use requires MQT authorization
  - •Implicit use requires base table or view authorization.
- Creation of table requires CREATE TABLE authorization and SELECT to base table or view
  - •DBADM can create MQT for another authorization ID
  - •Some restrictions for MQT and MLS
- REFRESH TABLE authorization
  - •Ownership of MQT, DBADM, DBCTRL, SYSADM or SYSCTRL

The materialized query table is often a summary table. DB2 optimization can rewrite a query on the base tables to use the MQT. No authorization on a MQT is required for it to be used in automatic query rewrite or implicitly. Authorization: To ALTER, the privilege set that is defined below must include at least one of the following:

The ALTER privilege on the table, Ownership of the table, DBADM authority for the database, or SYSADM or SYSCTRL authority

Additional privileges might be required when: FOREIGN KEY, DROP PRIMARY KEY, DROP FOREIGN KEY, or DROP CONSTANT is specified; The data type of a column that is added to the table is a distinct type; or a fullselect is specified.



Materialized Query Table and Multilevel Security

- DEFINITION only: AS SECURITY LABEL attribute not inherited
- Only one source table can have security label
- Security label column must be included in MQT
- AS SECURITY LABEL attribute is inherited
- ALTER TABLE to add seclabel will fail if table is source of MQT

If any table in the fullselect of the materialized query definition contains a security label column,

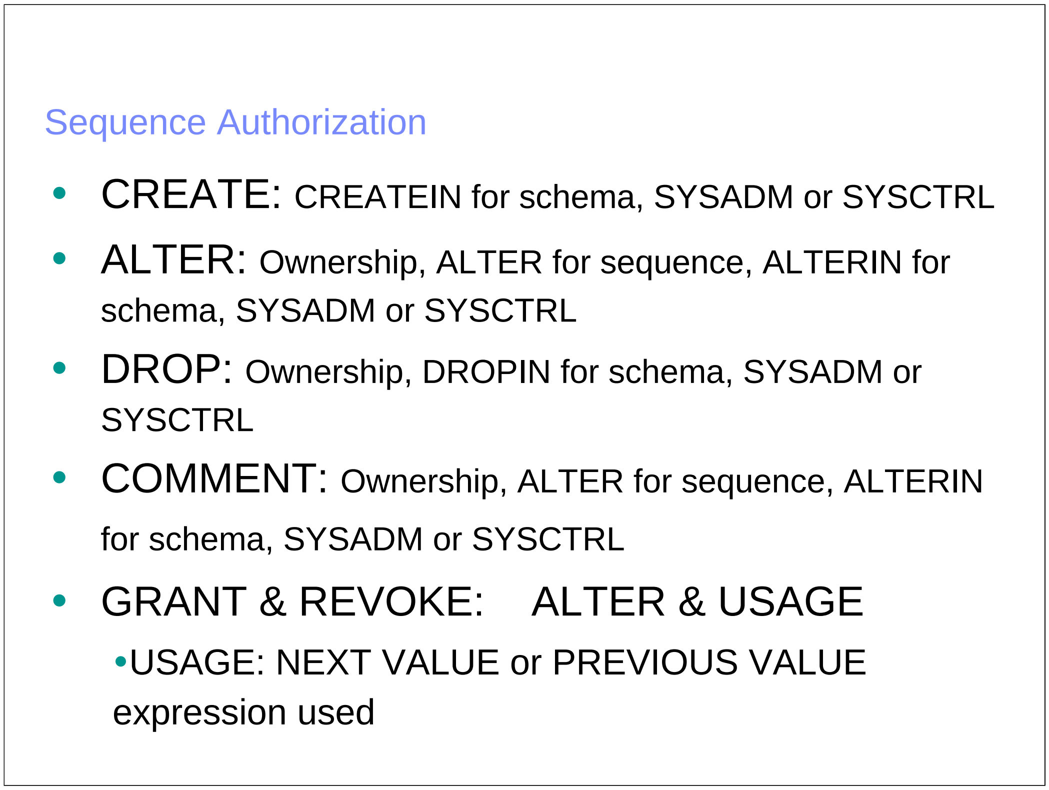
1. If for DEFINITION ONLY: The column attribute AS SECURITY LABEL is not inherited from table.

2. If only one table contains the security label column, the security label column must be included in the MQT. The MQT will inherit the column attribute AS SECURITY LABEL. The MAINTAINED BY USER option is allowed.

3. If more than one source table contains a security label column, an error is returned.

ALTER MQT source TABLE by adding security label column: An ALTER TABLE to add a security label column will fail if the table is a source table of an MQT.

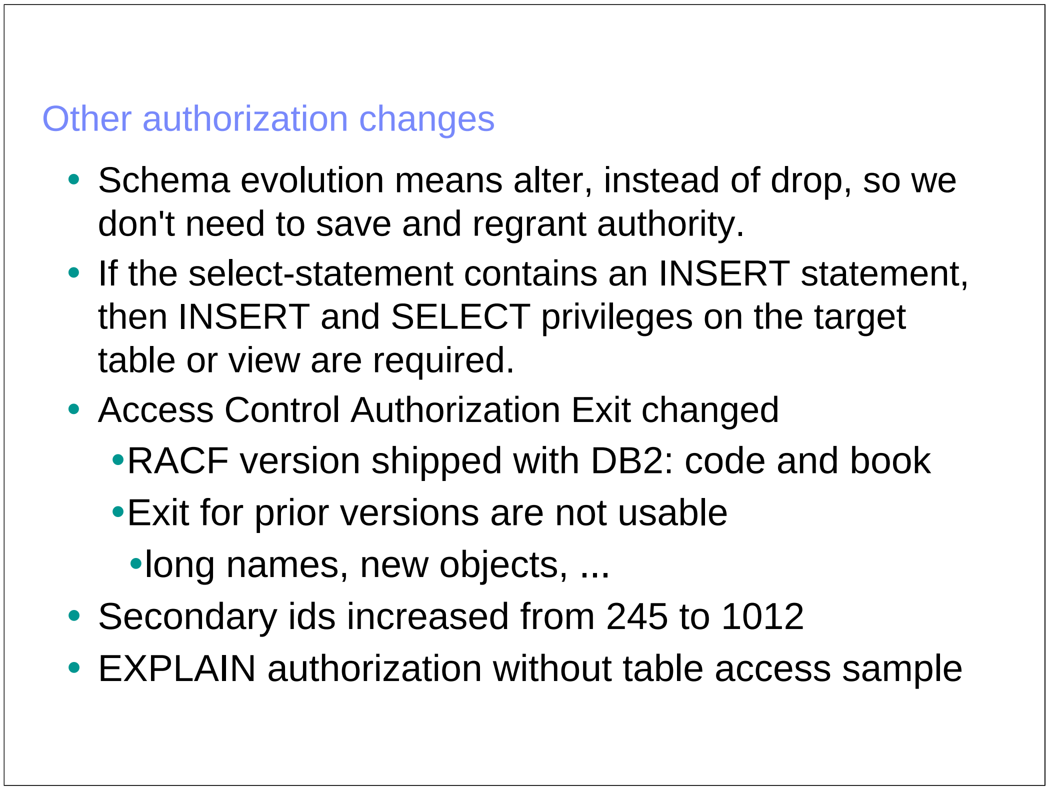
REFRESH TABLE for MQT: The REFRESH TABLE SQL statement, used to delete the data currently in the materialized query table and then to repopulate the materialized query table by executing the fullselect, does not check for MLS with row level granularity. The MLS row level granularity check is enforced when using the MQT, either by exploiting the MQT or by using the MQT directly.



Sequence Authorization

- CREATE: CREATEIN for schema, SYSADM or SYSCTRL
- ALTER: Ownership, ALTER for sequence, ALTERIN for schema, SYSADM or SYSCTRL
- DROP: Ownership, DROPIN for schema, SYSADM or SYSCTRL
- COMMENT: Ownership, ALTER for sequence, ALTERIN for schema, SYSADM or SYSCTRL
- GRANT & REVOKE: ALTER & USAGE
  - USAGE: NEXT VALUE or PREVIOUS VALUE expression used

The sequence is a new DB2 object, and these new SQL statements require authorization. The USAGE privilege for the sequence will be the most common privilege, I expect.
There are new authorization rules for data definition and for use of this new function.



Other authorization changes

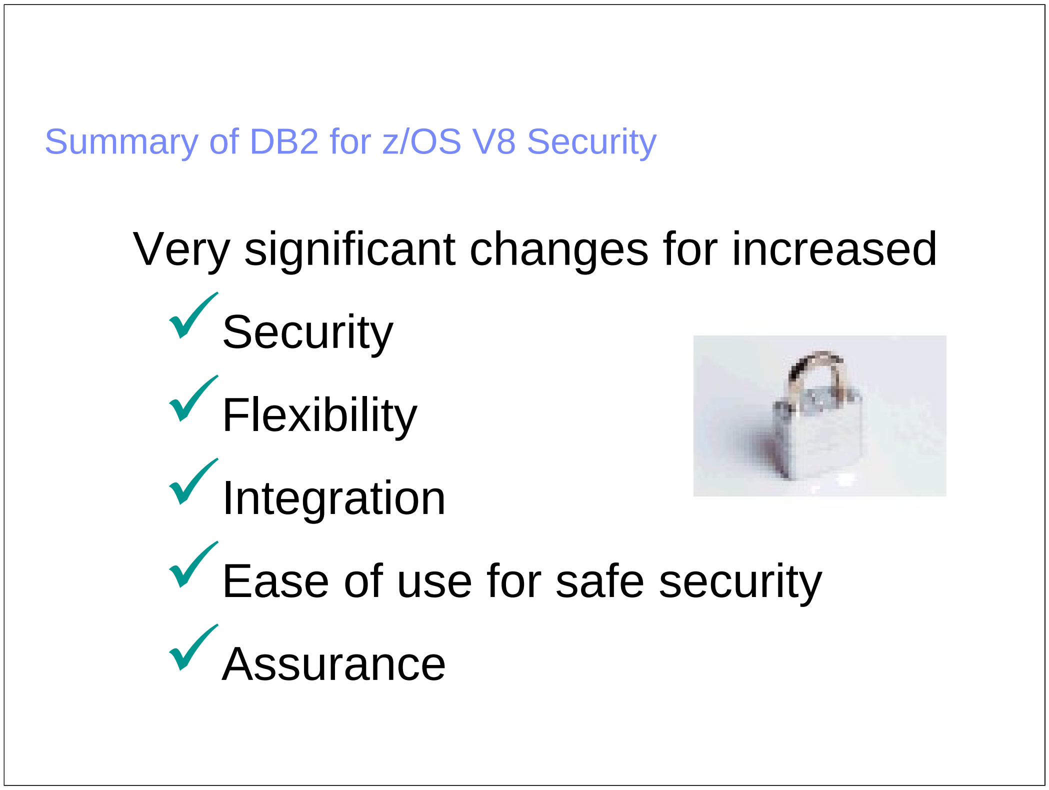
- Schema evolution means alter, instead of drop, so we don't need to save and regrant authority.
- If the select-statement contains an INSERT statement, then INSERT and SELECT privileges on the target table or view are required.
- Access Control Authorization Exit changed
  - RACF version shipped with DB2: code and book
  - Exit for prior versions are not usable
    - long names, new objects, ...
- Secondary ids increased from 245 to 1012
- EXPLAIN authorization without table access sample

For many database administrators, the biggest change in authorization will be the ability to avoid the cascade revoke caused by deleting a table or table space to change attributes. Being able to ALTER is faster, more available and safer.

The RACF access controls are changed very substantially. The exit has additional capabilities for sequences, and changes in the interface to handle long names. The exit is provided names converted from Unicode to EBCDIC. While the RACF exit has been shipped by RACF in SYS1.SAMPLIB, now it will come with DB2 in prefix.SDSNSAMP.

The maximum number of secondary authorization ids has increased from 245 to 1012 with APAR PQ90147.

An example showing how to have EXPLAIN authorization without direct access to the tables is provided with the V8 Visual Explain. http://www.ibm.com/software/data/db2/zos/osc/ve/index.html

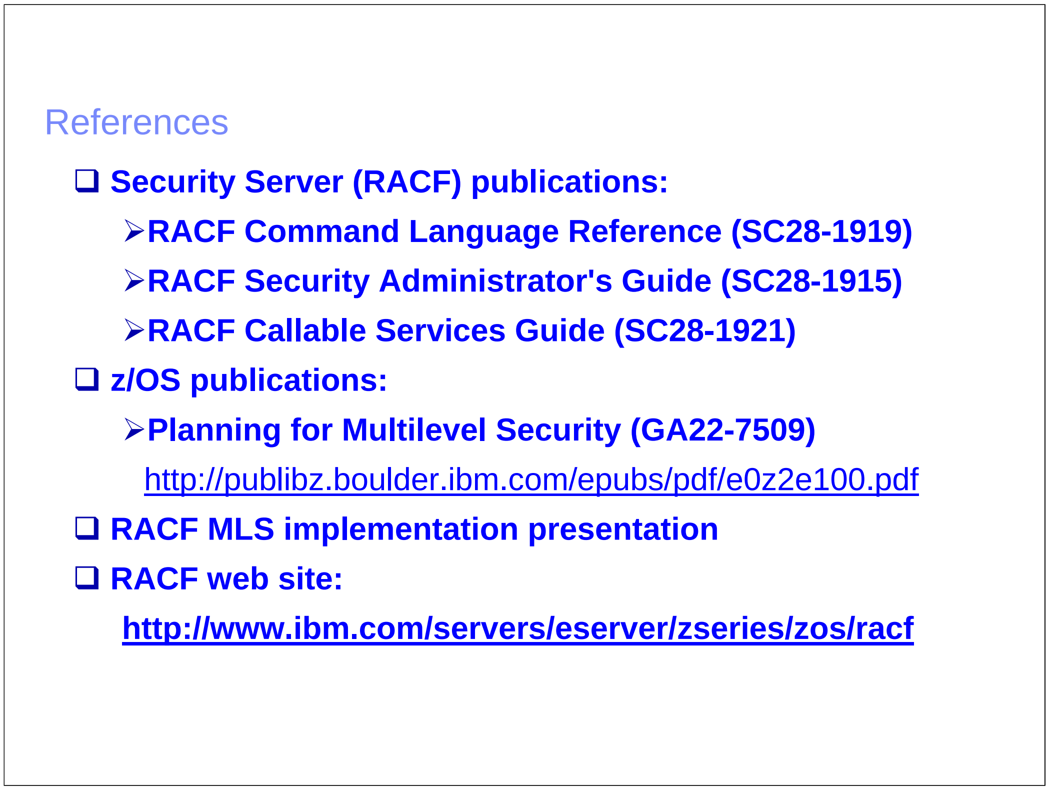


Summary of DB2 for z/OS V8 Security

Very significant changes for increased

- ✓ Security
- ✓ Flexibility
- ✓ Integration
- ✓ Ease of use for safe security
- ✓ Assurance

DB2 for z/OS provides many enhancements for security. There are new options for tighter security, more granularity, and more information for additional flexibility in applications and SQL. Integration has been improved with other platforms and with z/OS and the Security Server (RACF). The changes are intended to improve your ability to implement and use security safely. Let’s be safe out there.



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)

    http://publibz.boulder.ibm.com/epubs/pdf/e0z2e100.pdf
- RACF MLS implementation presentation
- RACF web site:

  http://www.ibm.com/servers/eserver/zseries/zos/racf

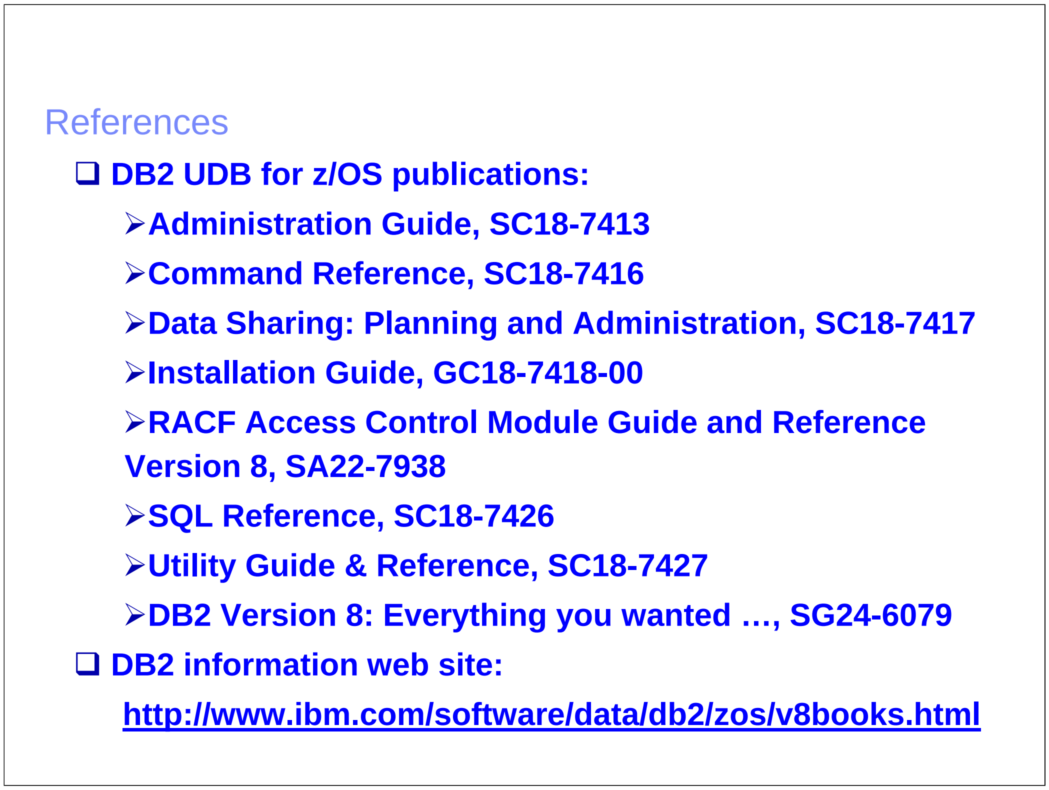
Here are some additional pointers for information about RACF. z/OS V1R5 is available. Check for the additional information and see one DB2 & RACF book shipped as a pdf file with DB2, RACF Access Control Module Guide and Reference Version 8 on the next page.

Planning for Multilevel Security is on the web under z/OS library, System level books, or

ibm.com/servers/eserver/zseries/zos/bkserv/r5pdf/zsys.html

DB2 V8 books including the RACF Access Control Module Guide are located at

http://www.ibm.com/software/data/db2/os390/v8books.html



References

- ❑ **DB2 UDB for z/OS publications:**
  - ➢**Administration Guide, SC18-7413**
  - ➢**Command Reference, SC18-7416**
  - ➢**Data Sharing: Planning and Administration, SC18-7417**
  - ➢**Installation Guide, GC18-7418-00**
  - ➢**RACF Access Control Module Guide and Reference Version 8, SA22-7938**
  - ➢**SQL Reference, SC18-7426**
  - ➢**Utility Guide & Reference, SC18-7427**
  - ➢**DB2 Version 8: Everything you wanted …, SG24-6079**
- ❑ **DB2 information web site:**
  http://www.ibm.com/software/data/db2/zos/v8books.html

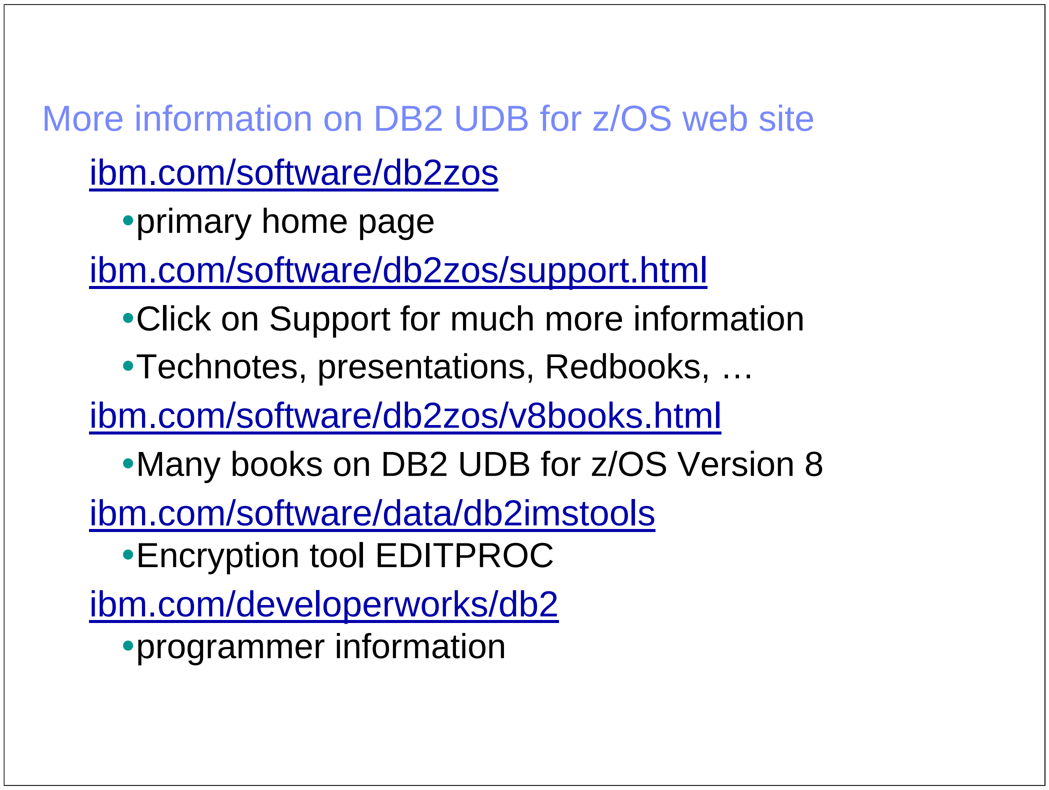
Here are some additional pointers for information about DB2 & RACF. One new RACF book is shipped as a pdf file with DB2 and on the DB2 books web page, RACF Access Control Module Guide and Reference Version 8.

Planning for Multilevel Security is on the web under z/OS library, System level books, or

ibm.com/servers/eserver/zseries/zos/bkserv/r5pdf/zsys.html

DB2 V8 books including the RACF Access Control Module Guide are located at

http://www.ibm.com/software/data/db2/os390/v8books.html



More information on DB2 UDB for z/OS web site

ibm.com/software/db2zos
- primary home page

ibm.com/software/db2zos/support.html
- Click on Support for much more information
- Technotes, presentations, Redbooks, …

ibm.com/software/db2zos/v8books.html
- Many books on DB2 UDB for z/OS Version 8

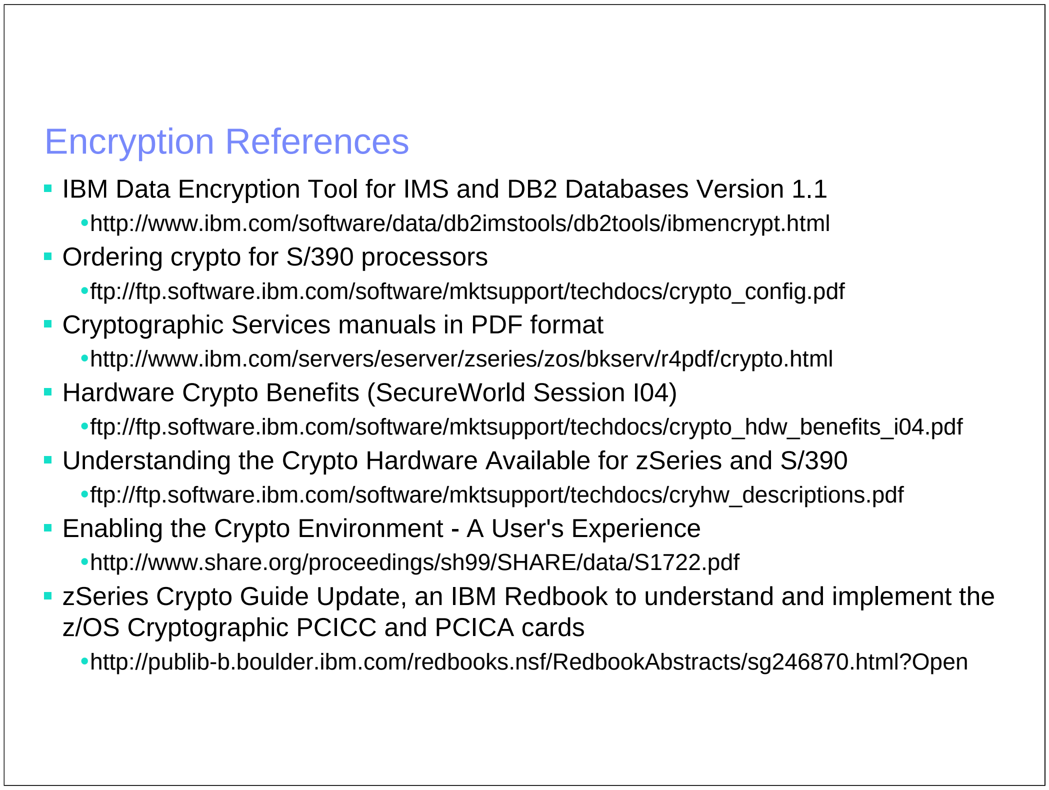
ibm.com/software/data/db2imstools
- Encryption tool EDITPROC

ibm.com/developerworks/db2
- programmer information

Here are the primary places to look for additional information. Check the primary home page to see what's new in the product.
The Support page has hundreds of items ranging from answers to frequently asked questions to redbooks and technical presentations. There is a new redbook, DB2 UDB for z/OS Version 8 Technical Preview, SG24-6871 on the web.

The presentations page has many presentations from conferences, so that customers can get the latest information even if they can't come to every conference.

For the latest on DB2 UDB for z/OS V8, check the V8 page.



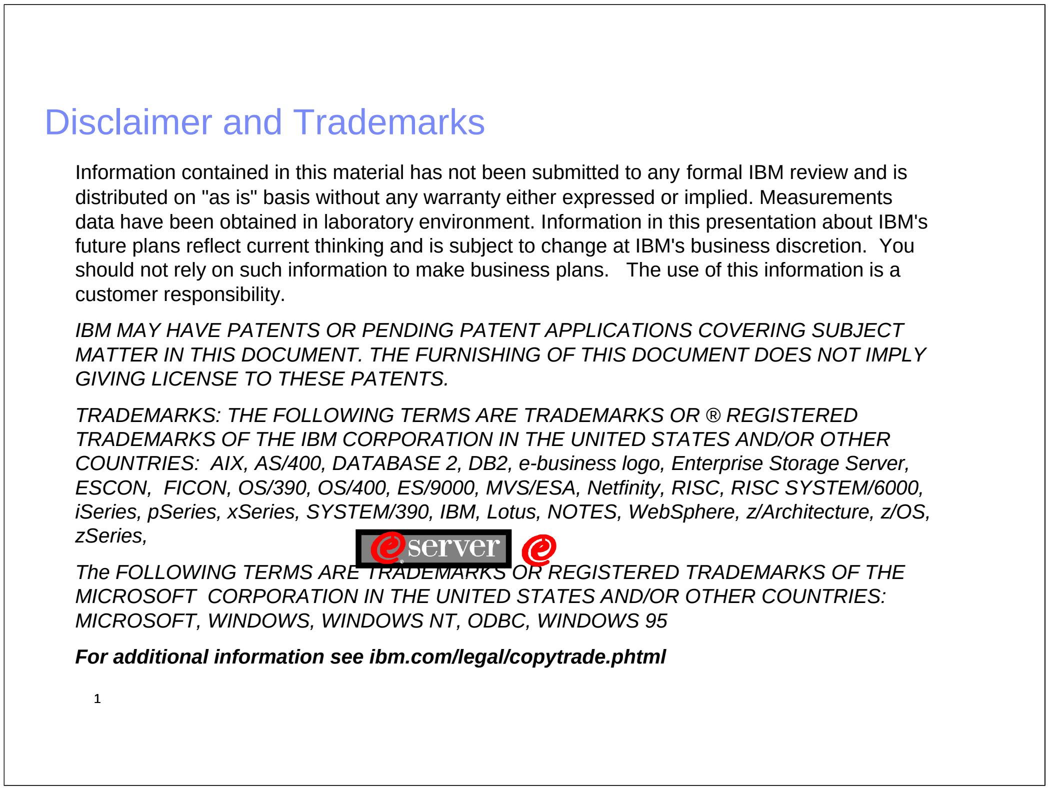
Encryption References

- IBM Data Encryption Tool for IMS and DB2 Databases Version 1.1
  - http://www.ibm.com/software/data/db2imstools/db2tools/ibmencrypt.html
- Ordering crypto for S/390 processors
  - ftp://ftp.software.ibm.com/software/mktsupport/techdocs/crypto_config.pdf
- Cryptographic Services manuals in PDF format
  - http://www.ibm.com/servers/eserver/zseries/zos/bkserv/r4pdf/crypto.html
- Hardware Crypto Benefits (SecureWorld Session I04)
  - ftp://ftp.software.ibm.com/software/mktsupport/techdocs/crypto_hdw_benefits_i04.pdf
- Understanding the Crypto Hardware Available for zSeries and S/390
  - ftp://ftp.software.ibm.com/software/mktsupport/techdocs/cryhw_descriptions.pdf
- Enabling the Crypto Environment - A User's Experience
  - http://www.share.org/proceedings/sh99/SHARE/data/S1722.pdf
- zSeries Crypto Guide Update, an IBM Redbook to understand and implement the z/OS Cryptographic PCICC and PCICA cards
  - http://publib-b.boulder.ibm.com/redbooks.nsf/RedbookAbstracts/sg246870.html?Open

These references provide much more information about encryption.

Disclaimers: 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. Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

The licensed program described in this information 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.



Disclaimer and Trademarks

Information contained in this material has not been submitted to any formal IBM review and is distributed on "as is" basis without any warranty either expressed or implied. Measurements data have been obtained in laboratory environment. Information in this presentation about IBM's future plans reflect current thinking and is subject to change at IBM's business discretion. You should not rely on such information to make business plans. The use of this information is a customer responsibility.

IBM MAY HAVE PATENTS OR PENDING PATENT APPLICATIONS COVERING SUBJECT MATTER IN THIS DOCUMENT. THE FURNISHING OF THIS DOCUMENT DOES NOT IMPLY GIVING LICENSE TO THESE PATENTS.

TRADEMARKS: THE FOLLOWING TERMS ARE TRADEMARKS OR ® REGISTERED TRADEMARKS OF THE IBM CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: AIX, AS/400, DATABASE 2, DB2, e-business logo, Enterprise Storage Server, ESCON, FICON, OS/390, OS/400, ES/9000, MVS/ESA, Netfinity, RISC, RISC SYSTEM/6000, iSeries, pSeries, xSeries, SYSTEM/390, IBM, Lotus, NOTES, WebSphere, z/Architecture, z/OS, zSeries,

The FOLLOWING TERMS ARE TRADEMARKS OR REGISTERED TRADEMARKS OF THE MICROSOFT CORPORATION IN THE UNITED STATES AND/OR OTHER COUNTRIES: MICROSOFT, WINDOWS, WINDOWS NT, ODBC, WINDOWS 95

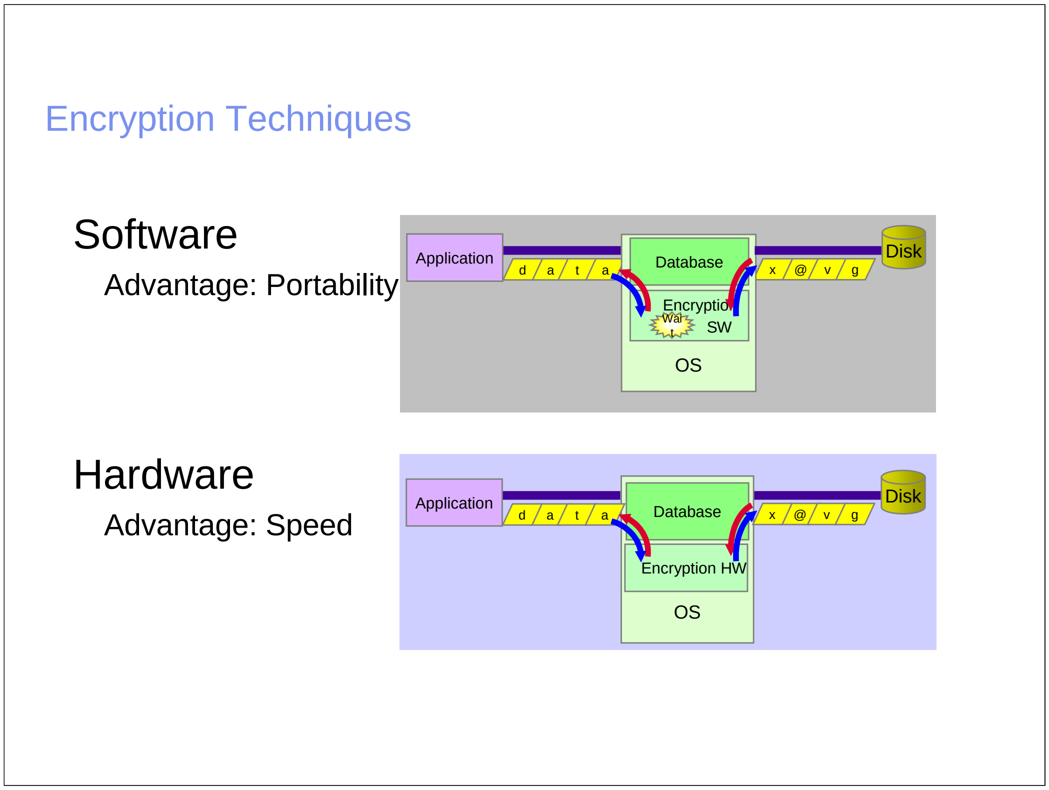
For additional information see ibm.com/legal/copytrade.phtml

1

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in the 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 may contain 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.



Encryption Techniques

Software

Advantage: Portability

Hardware

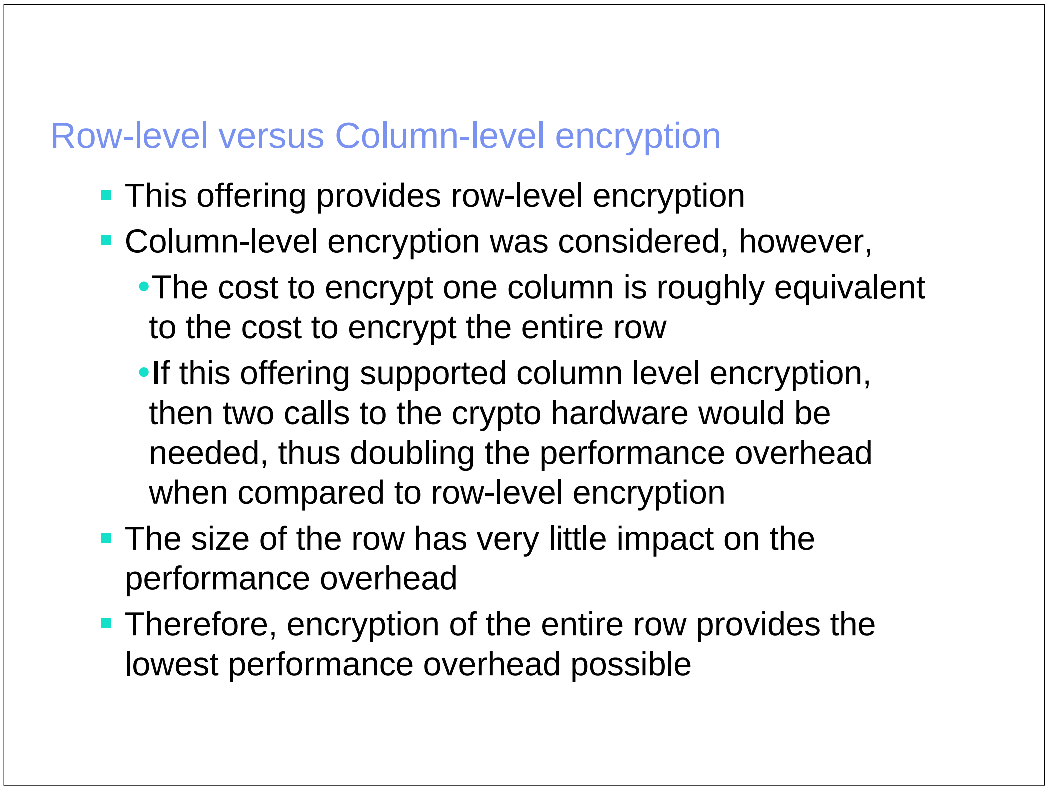
Advantage: Speed

Now we will look at encryption itself and how encryption techniques and issues apply to the tool. In general, there are two encryption techniques, encryption using software and encryption using hardware.

If you do your encryption in software, the data on the disk is encrypted as shown in the top picture. As data comes in, it goes through some encryption software to get the data decrypted out to an application. Encryption using software is inherently slower than encryption using hardware.

The bottom picture shows the same flow, but here some type of hardware assist is used to do encryption and decryption. This is very similar to the concept of compression and decompression. We use a hardware assist to do encryption in the same way that we used a hardware assist to do compression.

The advantage of software encryption is that you have portability. You can take the encryption software and put it on any platform as long as you have it coded in some language that lets you compile it on the different operating systems. The advantage of hardware encryption is speed.

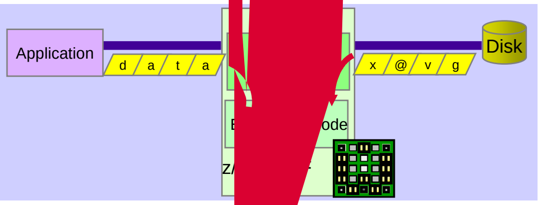


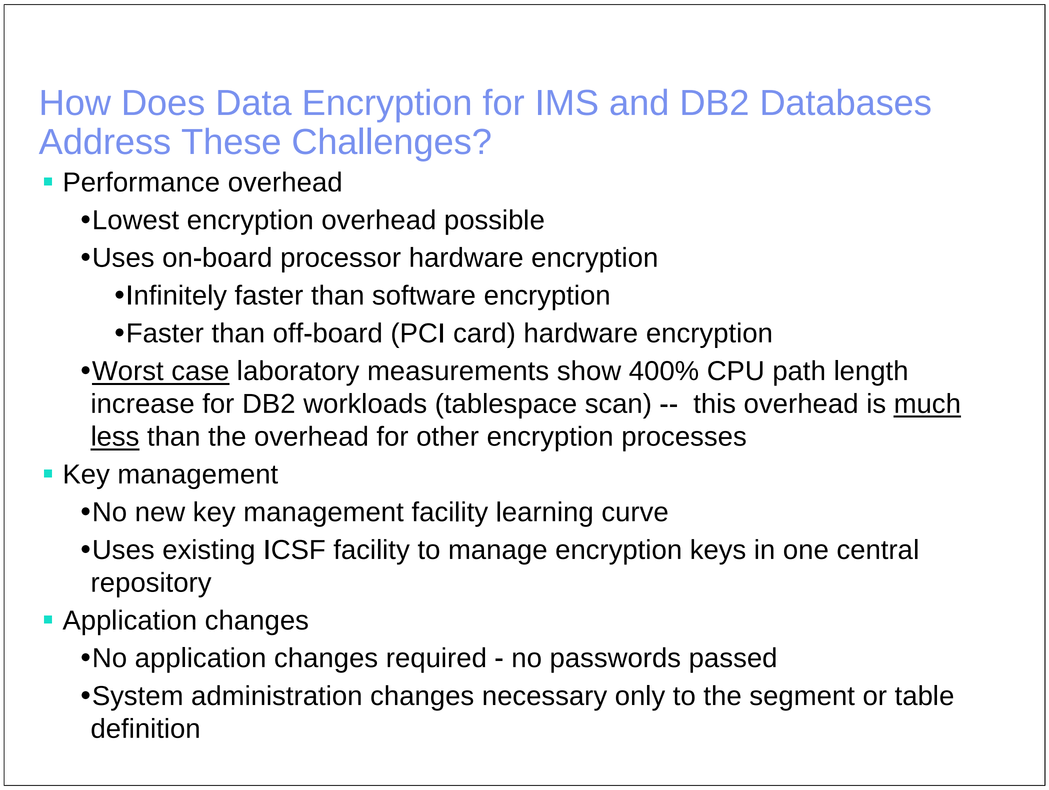
Row-level versus Column-level encryption

- This offering provides row-level encryption
- Column-level encryption was considered, however,
  - The cost to encrypt one column is roughly equivalent to the cost to encrypt the entire row
  - If this offering supported column level encryption, then two calls to the crypto hardware would be needed, thus doubling the performance overhead when compared to row-level encryption
- The size of the row has very little impact on the performance overhead
- Therefore, encryption of the entire row provides the lowest performance overhead possible

If you want column-level encryption, rather than row-level encryption, there are some tradeoffs. The real difference is what you are trying to protect and from whom. If your objective is to protect the data at rest, this offering does the job, ensuring that access is through the primary interfaces and uses the security checking. If you want column-level encryption or field-by-field encryption, then V8 provides functions for the encryption and decryption. This is application level or user level encryption, and your applications or users must do the key management.

Hardware Encryption Techniques





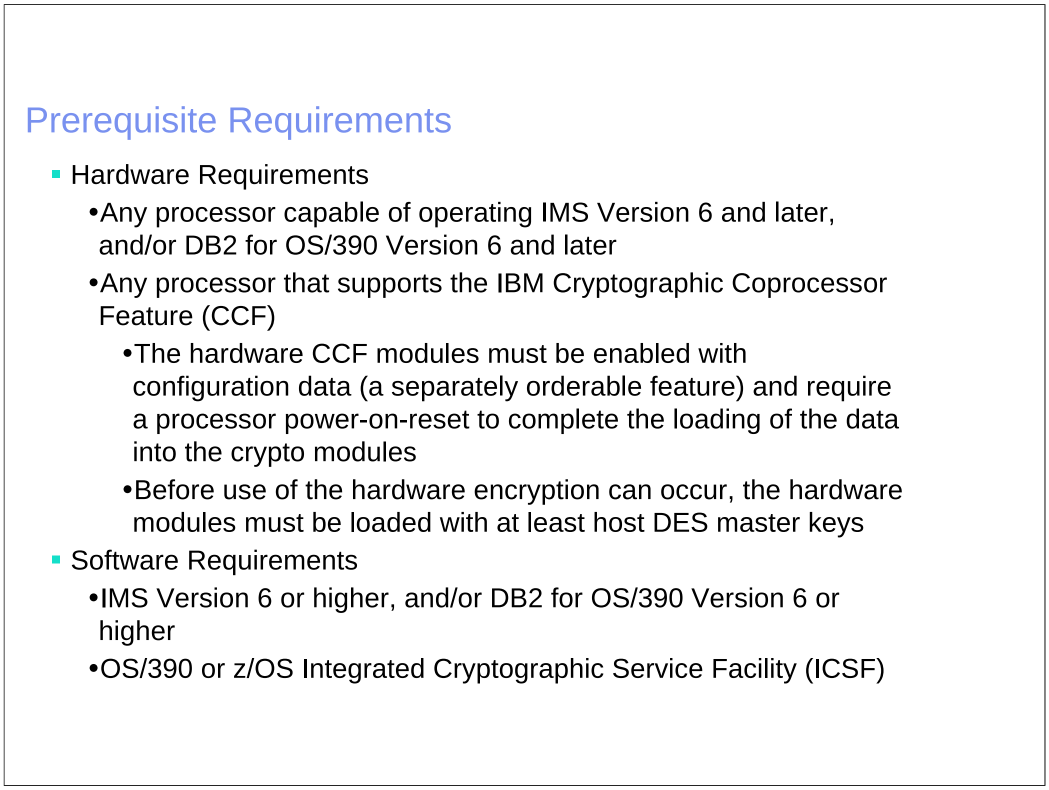
How Does Data Encryption for IMS and DB2 Databases Address These Challenges?

- Performance overhead
  - Lowest encryption overhead possible
  - Uses on-board processor hardware encryption
    - Infinitely faster than software encryption
    - Faster than off-board (PCI card) hardware encryption
  - Worst case laboratory measurements show 400% CPU path length increase for DB2 workloads (tablespace scan) -- this overhead is much less than the overhead for other encryption processes
- Key management
  - No new key management facility learning curve
  - Uses existing ICSF facility to manage encryption keys in one central repository
- Application changes
  - No application changes required - no passwords passed
  - System administration changes necessary only to the segment or table definition

The challenge for data encryption is in the areas of performance overhead, key management, and application changes. How do we address these challenges with the IBM Data Encryption for IMS and DB2 Databases tool?

Performance overhead. This tool has very low encryption overhead. Performance is better than any software encryption that can be performed. And it is faster than outboard or PCI-based compression; this is because of the location of the processors and because all processors share the same I/O bus. There are no calls to other pieces of hardware. Worst case laboratory measurements show about a 400 percent CP path link increase for a DB2 workload. In this case, the workload was a table space scan. A table space scan should be the worst case performer, because every row has to be decrypted while it is being accessed. This is not the case when you are going through an index. Indexes are not encrypted in DB2. The overhead to access an index is going to be much less. In some cases, the overhead may even be unnoticeable.

Key management. There is no new key management facility to learn. Existing ICSF services are used. **Application changes.** There are no application changes required. There are no passwords that need to be stored in applications. Passwords are passed at an exit level, and passwords are all managed. Note, however, that in order to implement data encryption in DB2 the table must be redefined. For example, if you want to add an EDITPROC into a DB2 table you cannot alter the table and add the EDITPROC. Instead, you have to UNLOAD the data, DROP and RECREATE the table and all of its dependent objects, RELOAD the data, and REDEFINE your applications. If you have the IBM DB2 Administration Tool installed, that tool will do these steps for you (the UNLOAD, DROP and RECREATE, and RELOAD of the encrypted data by way of the exit).

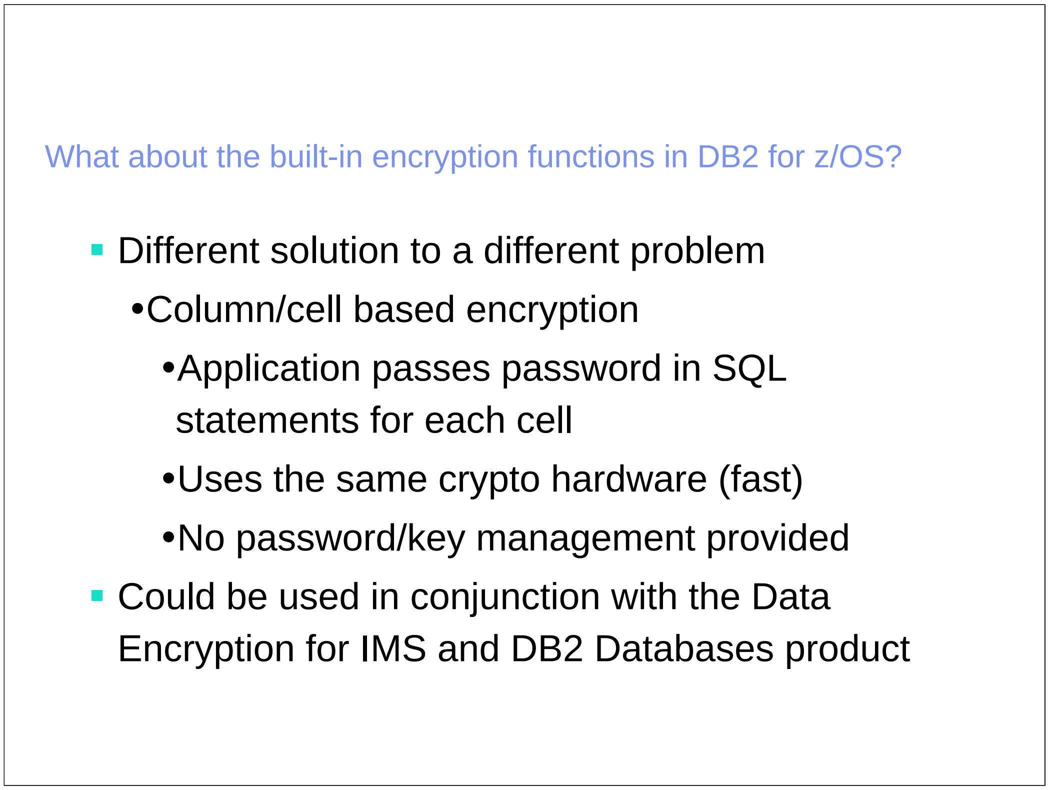


Prerequisite Requirements

- Hardware Requirements
  - Any processor capable of operating IMS Version 6 and later, and/or DB2 for OS/390 Version 6 and later
  - Any processor that supports the IBM Cryptographic Coprocessor Feature (CCF)
    - The hardware CCF modules must be enabled with configuration data (a separately orderable feature) and require a processor power-on-reset to complete the loading of the data into the crypto modules
    - Before use of the hardware encryption can occur, the hardware modules must be loaded with at least host DES master keys
- Software Requirements
  - IMS Version 6 or higher, and/or DB2 for OS/390 Version 6 or higher
  - OS/390 or z/OS Integrated Cryptographic Service Facility (ICSF)

Hardware requirements - any hardware that supports IMS version 6 or later or DB2 Version 6 or later is supported. This means that all the processors that DB2 V6 and IMS V6 run on have the crypto hardware. The crypto hardware must be enabled, and you must do a power-on-reset to enable the CCF modules.

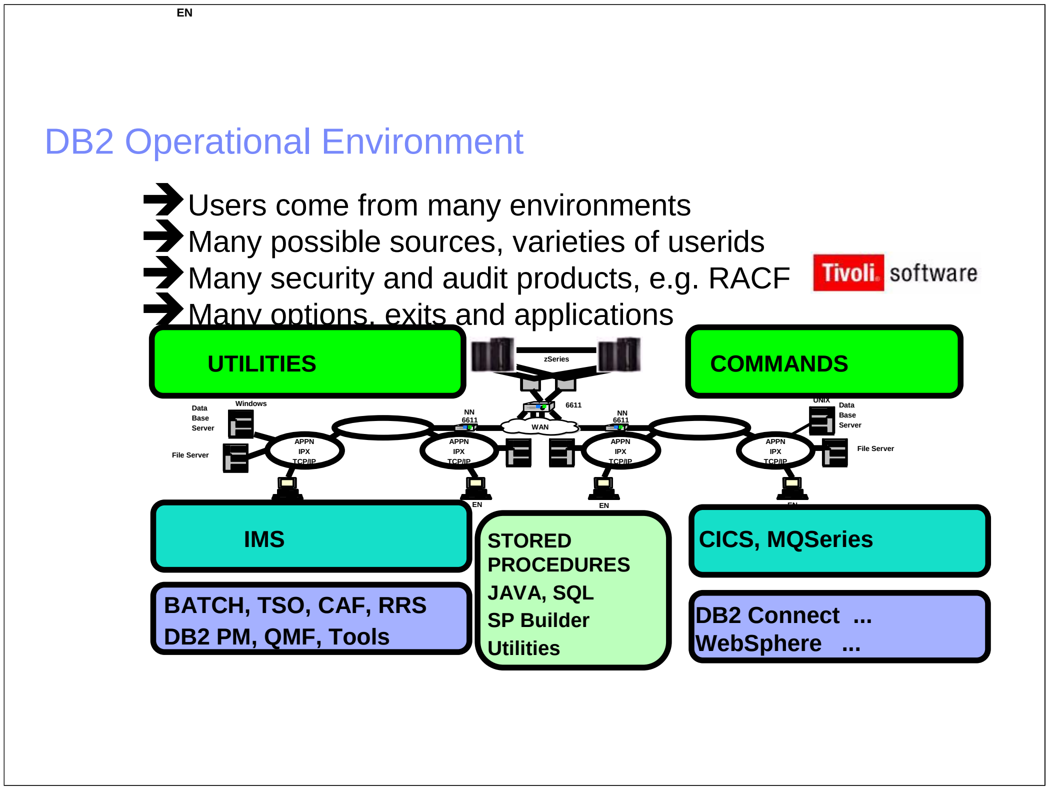
Software requirements - IMS Version 6 or higher and/or DB2 Version 6 or higher. ICSF, which is a part of the z/OS operating system, is also required. ICSF is not an add-on price feature. It is a base element of z/OS and OS/390.



What about the built-in encryption functions in DB2 for z/OS?

- Different solution to a different problem
  - Column/cell based encryption
    - Application passes password in SQL statements for each cell
    - Uses the same crypto hardware (fast)
    - No password/key management provided
- Could be used in conjunction with the Data Encryption for IMS and DB2 Databases product

Comparisons of encryption functions in DB2 V8 versus encryption of the data on disk versus encryption of data on the wire: All use encryption, but the techniques and objectives are very different.


EN

DB2 Operational Environment

- ➔ Users come from many environments
- ➔ Many possible sources, varieties of userids
- ➔ Many security and audit products, e.g. RACF
- ➔ Many options, exits and applications

Tivoli software

UTILITIES

COMMANDS

IMS

BATCH, TSO, CAF, RRS
DB2 PM, QMF, Tools

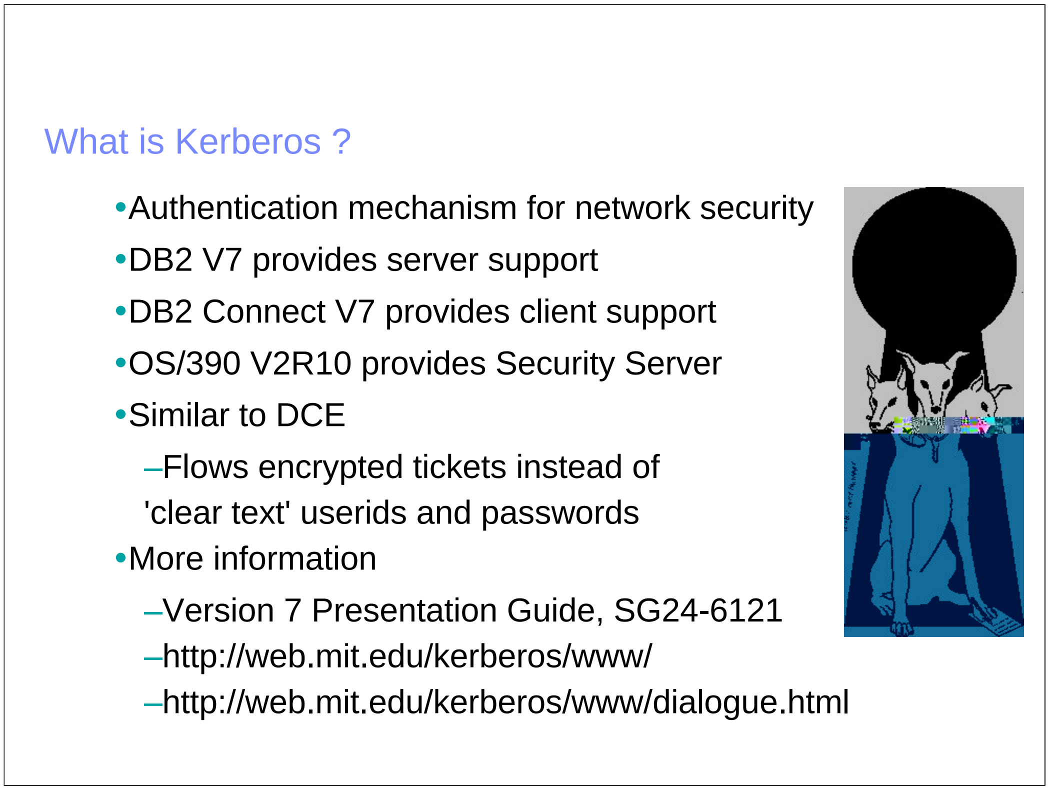
STORED PROCEDURES
JAVA, SQL
SP Builder
Utilities

CICS, MQSeries

DB2 Connect ...
WebSphere ...

There are many different environments for DB2, with different connections and security. DB2 uses the security context when possible, so batch jobs, TSO users, IMS and CICS transactions have security that uses a consistent identification and authentication. This is true for stored procedures from these environments as well. The large number of options, exits, environments and asynchronous or parallel work provide challenges for security. Some key applications manage security differently.

For some work, such as distributed database serving, DB2 is the initial point on this platform. For this work (DRDA distributed access, remote stored procedures), DB2 establishes the security context and manages the work.



What is Kerberos ?

- Authentication mechanism for network security
- DB2 V7 provides server support
- DB2 Connect V7 provides client support
- OS/390 V2R10 provides Security Server
- Similar to DCE
  - Flows encrypted tickets instead of 'clear text' userids and passwords
- More information
  - Version 7 Presentation Guide, SG24-6121
  - http://web.mit.edu/kerberos/www/
  - http://web.mit.edu/kerberos/www/dialogue.html

Kerberos security is an option for network security with DB2 Version 7, DB2 Connect Version 7 and OS/390 Version 2 Release 10 Security Server.

Kerberos is an industry accepted standard that provides better integration with other platforms and a single signon capability.

This function is for DB2 as a server, not as a requester.

This third party authentication flows encrypted tickets, rather than userids and passwords.