Security and Authorization

School of Computer Science
University of Waterloo

CS 348
Introduction to Database Management
Fall 2007

Content of certain slides due to R. Ramakrishnan and J. Gehrke

Outline

Objectives in Securing an Information System

Secrecy Information should only be shown to people who are allowed to see it.

Integrity Information should only be modified by people who are allowed to modify it.

Availability If someone is allowed to see and/or modify data, they should be able to do so.

Access Control

A security policy defines who should be allowed to see and/or modify specific data in the system.

- A DBMS provides access control mechanisms to help implement a security policy.
- Two complementary types of mechanisms:
  1. *Discretionary access control*
  2. *Mandatory access control*

Discretionary Access Control

Idea

Achieve security by specifying which schema objects a user may access.

- Users are given privileges to access the appropriate schema objects (tables, views).
- Users can grant privileges to other users at their own discretion.
- Implementation: GRANT and REVOKE commands

In SQL-92, privileges are assigned to users.
In SQL:1999, privileges are assigned to *roles*, which are then granted to users.

Granting/Revoking Privileges

```
GRANT privileges ON object TO users [WITH GRANT OPTION]

REVOKE [GRANT OPTION FOR] privileges ON object
FROM users { RESTRICT | CASCADE }
```

- Possible privileges:
  - `SELECT`
  - `INSERT(column)`
  - `UPDATE(column)`
  - `DELETE`
  - `REFERENCES(column)`
- `WITH GRANT OPTION` allows user to pass on privilege (with or without passing on grant option)
- When a privilege is revoked from user X, it is also revoked from all users that were granted the privilege solely from X

Views

- Views can be used to allow access to only certain tuples from a table
- The view creator has same privileges on the view as on the underlying tables
- A view is dropped if the view creator loses SELECT privileges on underlying tables/views

Mandatory Access Control

Idea

Achieve security by specifying which data (i.e. instance) objects a user may access.

- Discretionary AC is susceptible to *Trojan Horse attacks*:
  - If user X tricks user Y into copying data from table A into table B, then the access control on table A doesn't apply to the copy of the data in table B

- In Mandatory AC, system-wide policies govern who can see which data objects, independent of the data lineage

The Bell-LaPadula Model

- Objects (tables, views, rows, columns) are assigned security classes
- Subjects (users, roles, programs) are assigned security clearances
- Sample classes/clearances: Top Secret, Secret, Confidential, Unclassified

$$TS > S > C > U$$

Goal

Information should never flow from a higher to a lower class.

Restrictions enforced by the DBMS:

1. Subject S can read object O only if clearance$(S) \geq$ class(O)
2. Subject S can write object O only if clearance$(S) \leq$ class(O)

Multilevel Relations

Individual tuples or columns can be assigned security classes

$\Rightarrow$ users with different clearances see different tables

Fighters

Name	Threat	Security Class
Sopwith Pup	Harmless	Unclassified
MiG-29 Fulcrum	Extremely Dangerous	Top Secret

Users with clearance *TS* see two rows; other users see only one.

To avoid revealing any information about the MiG-29 Fulcrum, the **Security Class** must be treated as part of the key.