Chapter 2: Security Policies

DATABASE SECURITY

Overview

- what security policy is to be enforced by the system?
    - First question before designing a secure system
    - a set of rules that enforce security
    - mandatory security policies
        - policies that are "mandatory" in nature and are application independent
        - Bell and LaPadula
    - discretionary security policies
        - policies that are specified by who is responsible for the environment in which the system will operate
    - This chapter focuses on discretionary security policies

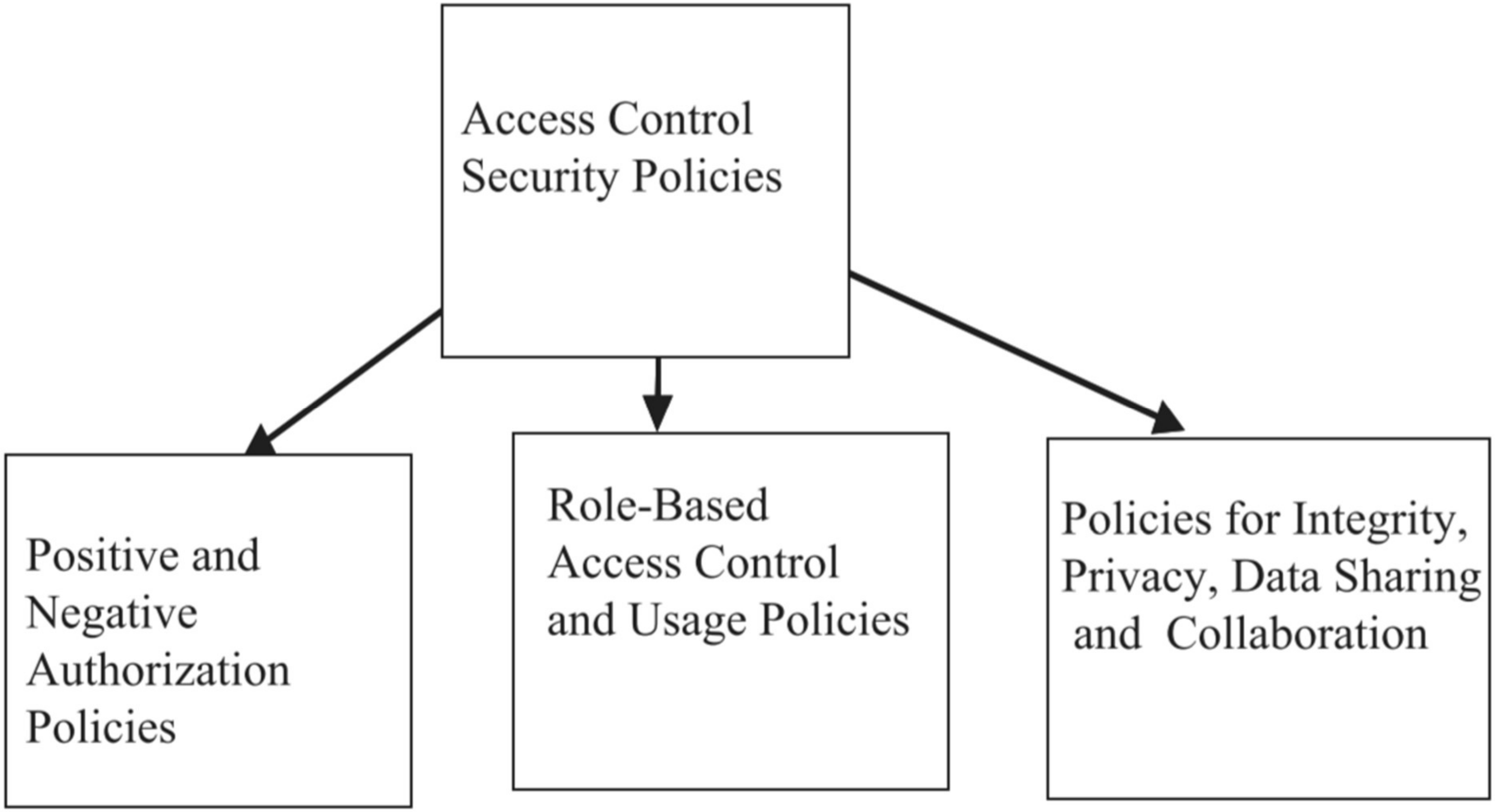
Overview

- Access control
  - most popular discretionary security policy
  - First studied for operating systems
  - Two first database systems that investigate it
    - System R and INGRES
- Other discretionary policies
  - administration policies
  - identification and authentication policies

Access-Control Policies

- first examined for operating systems
  - essential point: whether a process can be granted access to a file.
  - Access could be read access or write access
    - Write access could include access to modify, append, or delete.
- were transferred to database systems
  - various forms have been studied
    - Notable: role-based access-control policies
      - now implemented in several commercial systems

Access-Control Policies



Access Control Security Policies
Positive and Negative Authorization Policies
Role-Based Access Control and Usage Policies
Policies for Integrity, Privacy, Data Sharing and Collaboration

Authorization Policies

- Users are granted access to data based on authorization rules
    - Positive Authorizations
        - John has read access to attribute Salary and write access to attribute Name in relation EMP.
        - Write access could include
            - Append
            - Modify
            - delete

Authorization Policies

- Users are granted access to data based on authorization rules
  - Negative Authorization
    - What if access to an object is not specified?
      - Implicit: authorization rule that is not specified is taken to be a negative authorization
      - Explicit: negative authorizations are explicitly specified.
        - Example: John does not have access to relation EMP or Jane does not have access to relation DEPT

Authorization Policies

- Users are granted access to data based on authorization rules
  - Conflict Resolutions
    - how do we resolve the conflicting rules?
      - a rule grants John read access to relation EMP
      - another rule does not grant John read access to the salary attribute in EMP.
      - Usually a system enforces the least privilege rule
        - John has access to EMP except for the salary values

Authorization Policies

- Users are granted access to data based on authorization rules
  - Strong and Weak Authorization
    - strong authorization: the rule holds regardless of conflicts.
    - weak authorizations: the rule does not hold in case of conflict
    - Example
      - John is granted access to EMP with a strong authorization rule
      - the rule where John is not granted access to salary attribute is a weak authorization
      - The strong authorization will hold

Authorization Policies

- Users are granted access to data based on authorization rules
    - Propagation of Authorization Rules
        - how do the rules get propagated?
        - John has read access to relation EMP
            - does it automatically mean that John has read access to every element in EMP?
            - Usually this is the case
                - unless we have a rule that prohibits automatic propagation of an authorization rule.

Authorization Policies

- Users are granted access to data based on authorization rules
    - Special Rules
        - Content-based rules
            - access is granted depending on the content of the data
                - John has read access to tuples only in DEPT D100.
        - Context-based rules
            - access is granted depending on the context in which the data is displayed
                - John does not have read access to names and salaries taken together, however, he can have access to individual names and salaries.
        - Event-based rules
            - after the election, John has access to all elements in relation EMP

Authorization Policies

- Users are granted access to data based on authorization rules
  - Consistency of Rules
    - do we have conflict resolution rules that will resolve the conflicts?
  - Completeness of Rules
    - Are all of the entities specified in access-control rules for a user?
    - what assumptions do we make about entities that do not have either positive or negative authorizations for a particular user or a class of users?

Authorization Policies

- Role-Based Access Control
  - Idea:
    - grant access to users depending on their roles and functions
  - Issues:
    - does access propagate upwards in the hierarchy ?
    - What about the downward propagation?
    - What about the multiple parents?

Administration Policies

- Specify who is to administer the data
  - keeping the data current
  - making sure the metadata is updated whenever the data is updated
  - ensuring recovery from failures
  - …

Administration Policies

- Typically
  - DBA is responsible for updating
    - the metadata
    - the index
    - access methods
    - also ensuring that the access-control rules are properly enforced.
  - SSO may also have a role.
    - DBA and SSO may share the duties
      - security-related issues might be the responsibility of the SSO
      - data-related issues might be the responsibility of the DBA.

Administration Policies

- Other administration policies
  - assigning caretakers
    - Usually owners have control of the data that they create
    - owners may not be available to manage the data
      - Assign caretakers

Identification and Authentication

- By identification we mean
  - users must identify themselves with their user ID and password.
- Authentication means
  - the system must then match the user ID with the password to ensure that this is indeed the person
- We discuss identity management later

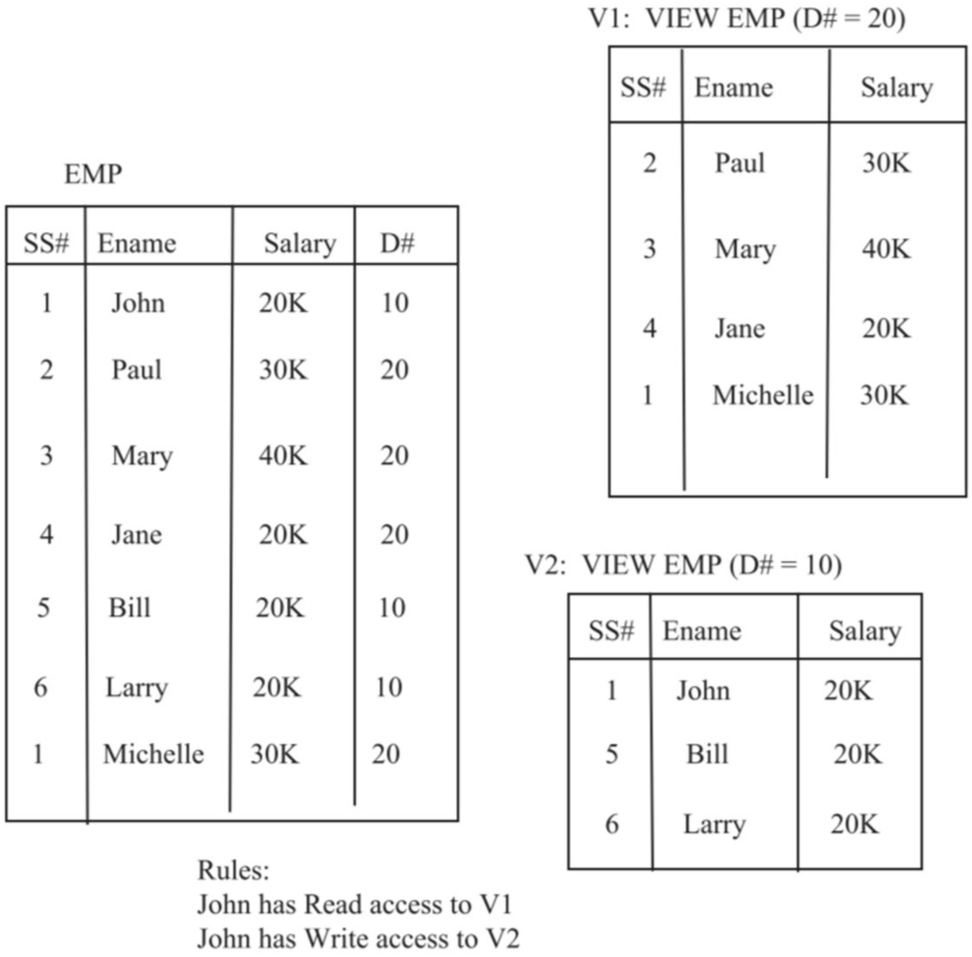
Auditing a Database System

- Databases are audited for multiple purposes
  - to keep track of
    - the number of queries posed
    - the number of updates made
    - the number of transactions executed
    - the number of times the secondary storage is accessed
  - Also for security purposes
    - have any of the access-control rules been bypassed by releasing information to the users?
    - Has the inference problem occurred?
    - Has privacy been violated?
    - Have there been unauthorized intrusions?

Views for Security

- DBA could form views and grant access to the views
  - views could be assigned security levels
  - have problems associated with them
    - view update problem

Views for Security



EMP

SS#	Ename	Salary	D#
1	John	20K	10
2	Paul	30K	20
3	Mary	40K	20
4	Jane	20K	20
5	Bill	20K	10
6	Larry	20K	10
1	Michelle	30K	20

V1: VIEW EMP (D# = 20)

SS#	Ename	Salary
2	Paul	30K
3	Mary	40K
4	Jane	20K
1	Michelle	30K

V2: VIEW EMP (D# = 10)

SS#	Ename	Salary
1	John	20K
5	Bill	20K
6	Larry	20K

Rules:
John has Read access to V1
John has Write access to V2

SQL Extensions for Security

- SQL has GRANT and REVOKE
  - GRANT JOHN EMP READ
  - REVOKE JOHN EMP READ
- also extended with more complex constraints

```
GRANT JOHN READ
EMP.SALARY
GRANT JOHN READ
EMP.NAME
NOT GRANT JOHN READ
Together (EMP.NAME, EMP.SALARY)
```

```
GRANT JOHN READ
EMP
Where EMP.SALARY < 30K
```

These are not standards

Query Modification

- was first proposed in the INGRES
- The idea is to modify the query based on the constraints
  - John only has read access to tuples with
    - salary < 30K
    - employee is not in the Security department

```
Select * from EMP ➔ Select * from EMP
                    Where EMP.Salary < 30K
                    And EMP.D# = DEPT.D#
                    And DEPT.Name is not Security
```