# Computer Security: Principles and Practice

Fourth Edition, Global Edition

By: William Stallings and Lawrie Brown

# Chapter 4

## Access Control

# Access Control Definitions 1/2

NISTIR 7298 defines access control as:

"the process of granting or denying specific requests to: (1) obtain and use information and related information processing services; and (2) enter specific physical facilities"

# Access Control Definitions 2/2

RFC 4949 defines access control as:

"a process by which use of system resources is regulated according to a security policy and is permitted only by authorized entities (users, programs, processes, or other systems) according to that policy"

# Access Control Principles

- Access control implements a security policy that specifies who or what (e.g., in the case of a process) may have access to each specific system resource, and the type of access that is permitted in each instance.

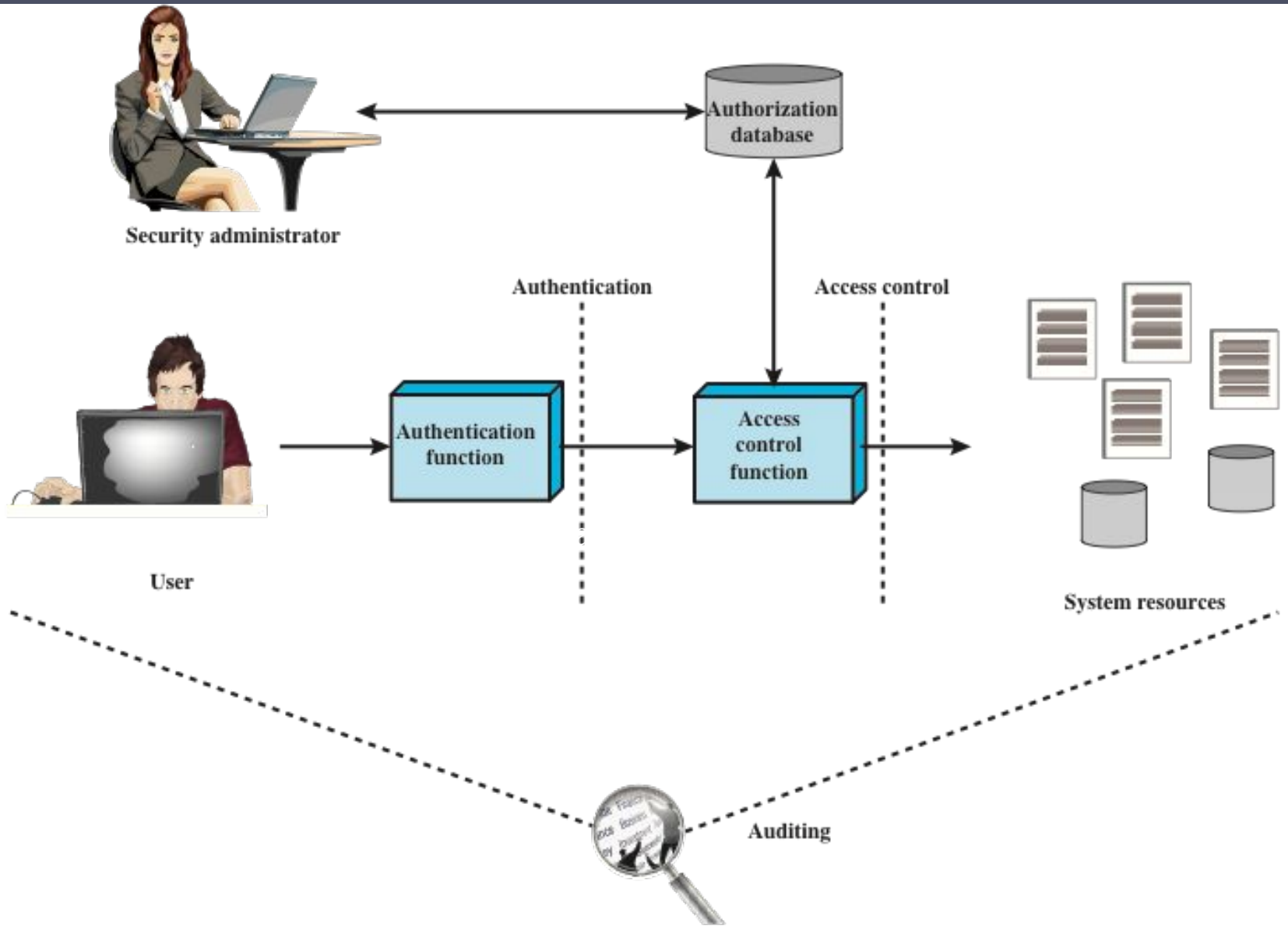



**Figure 4.1 Relationship Among Access Control and Other Security Functions**

*Source*: Based on [SAND94].

# Access Control Policies

- Discretionary access control (DAC)
  - Controls access based on the identity of the requestor and on access rules (authorizations) stating what requestors are (or are not) allowed to do
- Mandatory access control (MAC)
  - Controls access based on comparing security labels with security clearances
- Role-based access control (RBAC)
  - Controls access based on the roles that users have within the system and on rules stating what accesses are allowed to users in given roles
- Attribute-based access control (ABAC)
  - Controls access based on attributes of the user, the resource to be accessed, and current environmental conditions

# Subjects, Objects, and Access Rights

## Subject

An entity capable of accessing objects

Three classes
- Owner
- Group
- World

## Object

A resource to which access is controlled

Entity used to contain and/or receive information

## Access right

Describes the way in which a subject may access an object

Could include:
- Read
- Write
- Execute
- Delete
- Create
- Search

# Discretionary Access Control (DAC)

- Scheme in which an entity may be granted access rights that permit the entity, by its own violation, to enable another entity to access some resource
- Often provided using an access matrix
  - One dimension consists of identified subjects that may attempt data access to the resources
  - The other dimension lists the objects that may be accessed
- Each entry in the matrix indicates the access rights of a particular subject for a particular object

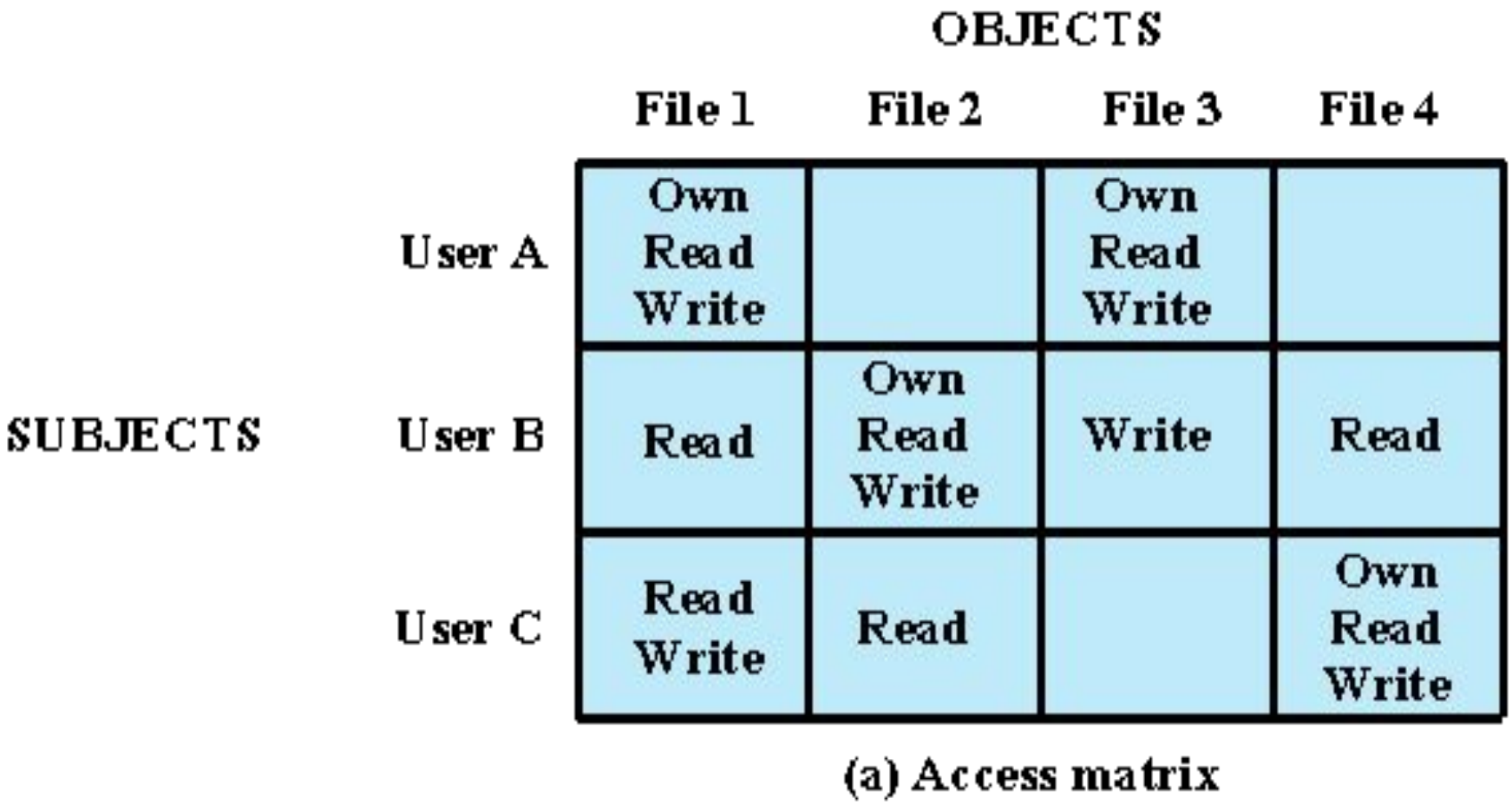
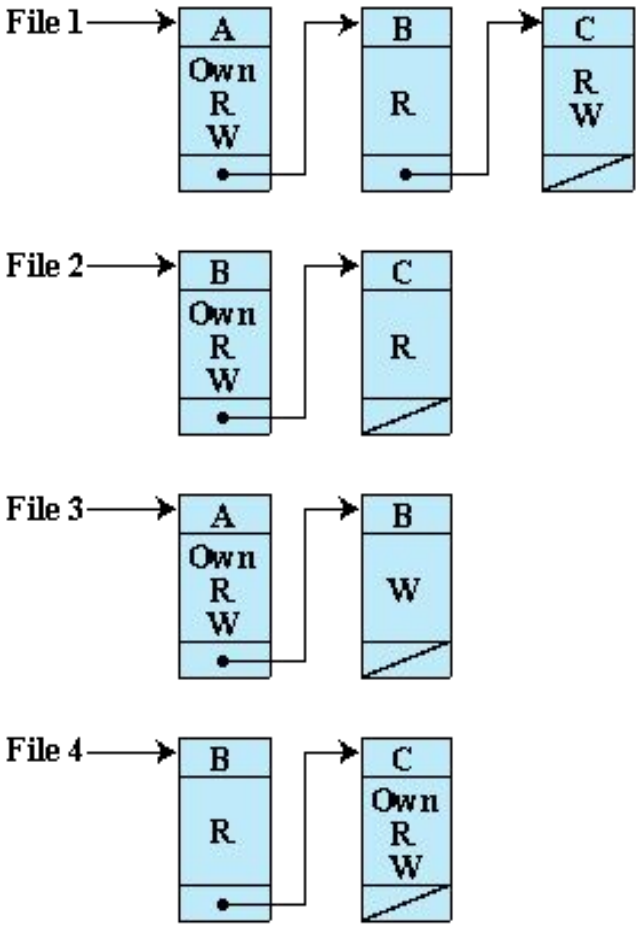
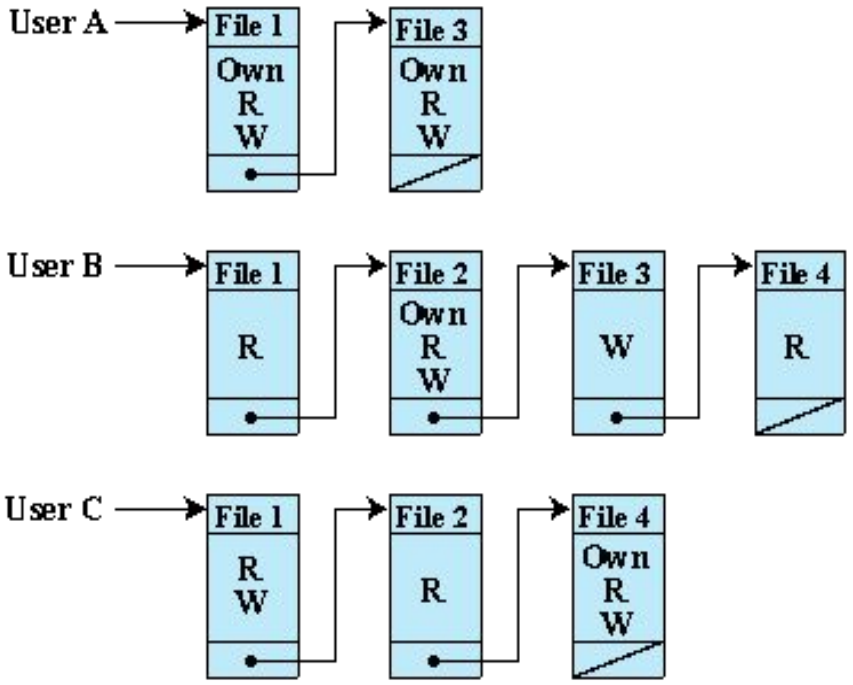



**Figure 4.2 Example of Access Control Structures**




(b) Access control lists for files of part (a)




(c) Capability lists for files of part (a)

**Figure 4.2 Example of Access Control Structures**

| Subject | Access Mode | Object |
|---|---|---|
| A | Own | File 1 |
| A | Read | File 1 |
| A | Write | File 1 |
| A | Own | File 3 |
| A | Read | File 3 |
| A | Write | File 3 |
| B | Read | File 1 |
| B | Own | File 2 |
| B | Read | File 2 |
| B | Write | File 2 |
| B | Write | File 3 |
| B | Read | File 4 |
| C | Read | File 1 |
| C | Write | File 1 |
| C | Read | File 2 |
| C | Own | File 4 |
| C | Read | File 4 |
| C | Write | File 4 |

# Table 4.2

## Authorization Table for Files in Figure 4.2

(Table is on page 113 in the textbook)

# UNIX File Access Control

## UNIX files are administered using inodes (index nodes)

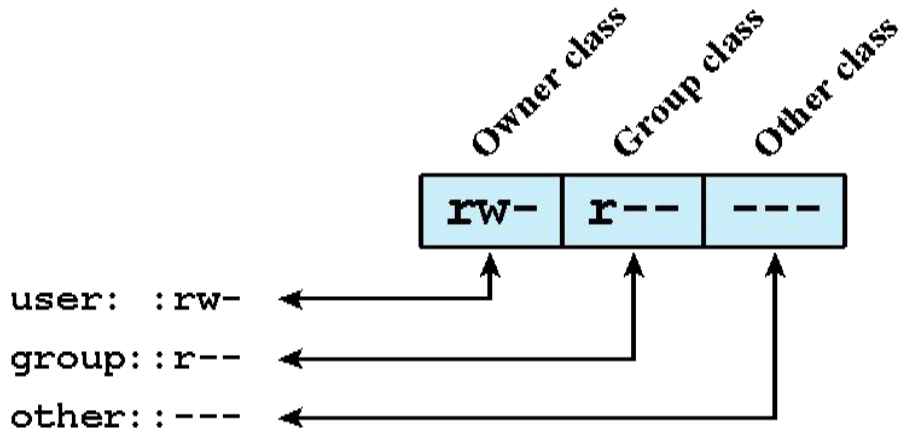
- Control structures with key information needed for a particular file
- An active inode is associated with exactly one file
- File attributes, permissions and control information are sorted in the inode
- On the disk there is an inode table, or inode list, that contains the inodes of all the files in the file system
- When a file is opened its inode is brought into main memory and stored in a memory resident inode table

## **Directories are structured in a hierarchical tree**

- May contain files and/or other directories
- Contains file names plus pointers to associated inodes

# UNIX
## File Access Control

- Unique user identification number (user ID)
- Member of a primary group identified by a group ID
- Belongs to a specific group
- 12 protection bits
  - Specify read, write, and execute permission for the owner of the file, members of the group and all other users
- The owner ID, group ID, and protection bits are part of the file's inode




(a) Traditional UNIX approach (minimal access control list)

**Figure 4.5 UNIX File Access Control**

# Access Control Lists (ACLs) in UNIX

**Modern UNIX systems support ACLs**

- FreeBSD, OpenBSD, Linux, Solaris

**FreeBSD**

- Setfacl command assigns a list of UNIX user IDs and groups
- Any number of users and groups can be associated with a file
- Read, write, execute protection bits
- A file does not need to have an ACL

**When a process requests access to a file system object two steps are performed:**

- Step 1 selects the most appropriate ACL
- Step 2 checks if the matching entry contains sufficient permissions

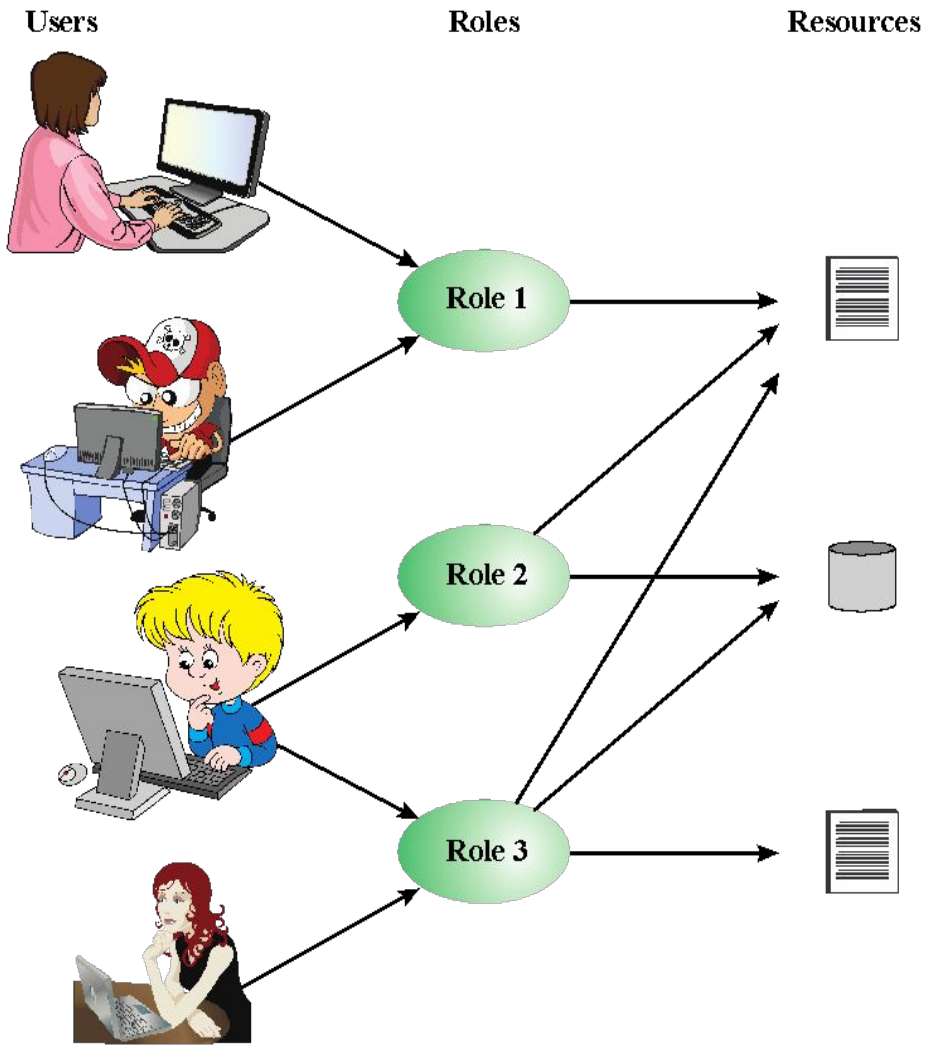
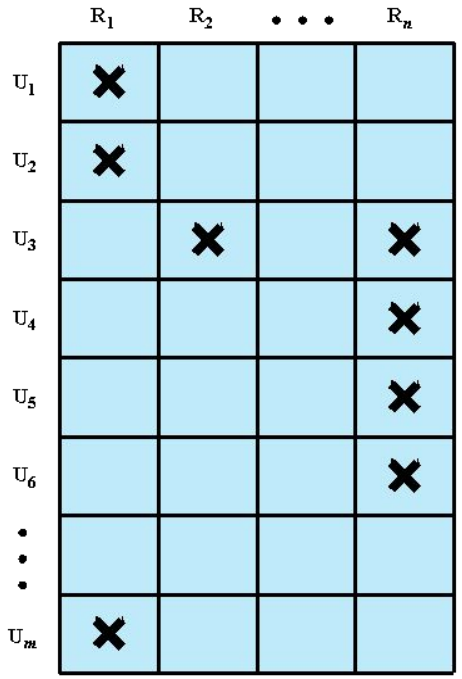


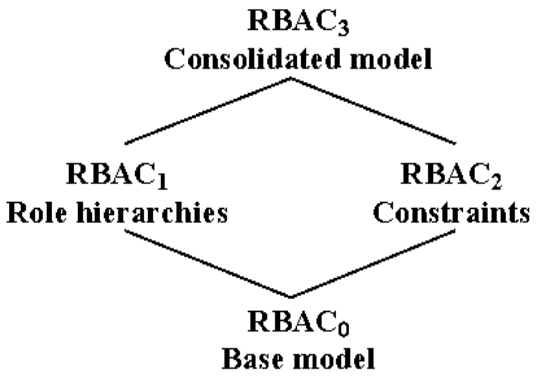

Figure 4.6 Users, Roles, and Resources



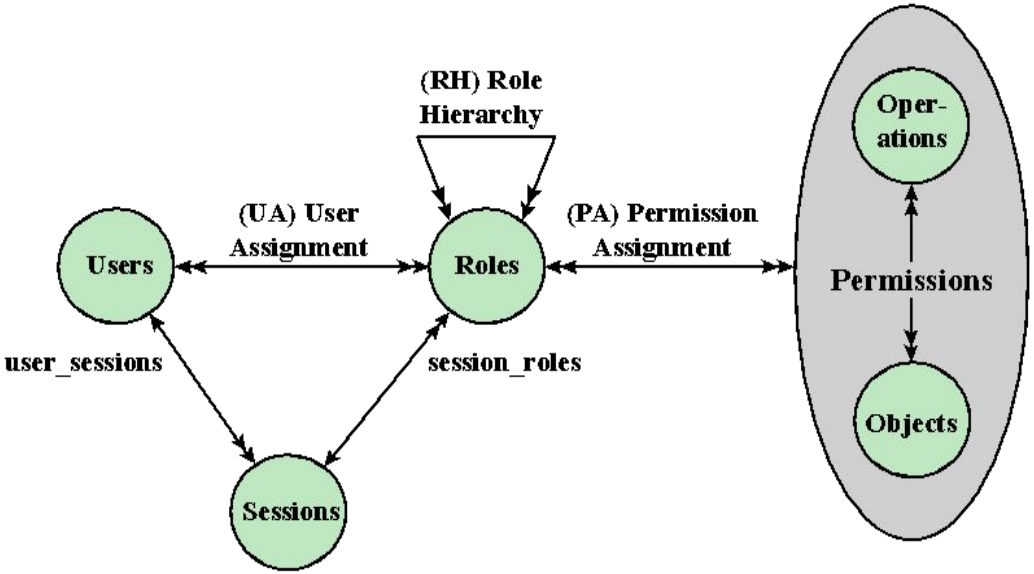

OBJECTS

| ROLES | $R_1$ | $R_2$ | $R_n$ | $F_1$ | $F_1$ | $P_1$ | $P_2$ | $D_1$ | $D_2$ |
|---|---|---|---|---|---|---|---|---|---|
| $R_1$ | control | owner | owner control | read * | read owner | wakeup | wakeup | seek | owner |
| $R_2$ | | control | | write * | execute | | | owner | seek * |
| • | | | | | | | | | |
| • | | | | | | | | | |
| • | | | | | | | | | |
| $R_n$ | | | control | | write | stop | | | |

**Figure 4.7 Access Control Matrix Representation of RBAC**




(a) Relationship among RBAC models




(b) RBAC models

**Figure 4.8 A Family of Role-Based Access Control Models.**

# Table 4.4
# Scope RBAC Models

| Models | Hierarchies | Constraints |
|---|---|---|
| $RBAC_0$ | No | No |
| $RBAC_1$ | Yes | No |
| $RBAC_2$ | No | Yes |
| $RBAC_3$ | Yes | Yes |

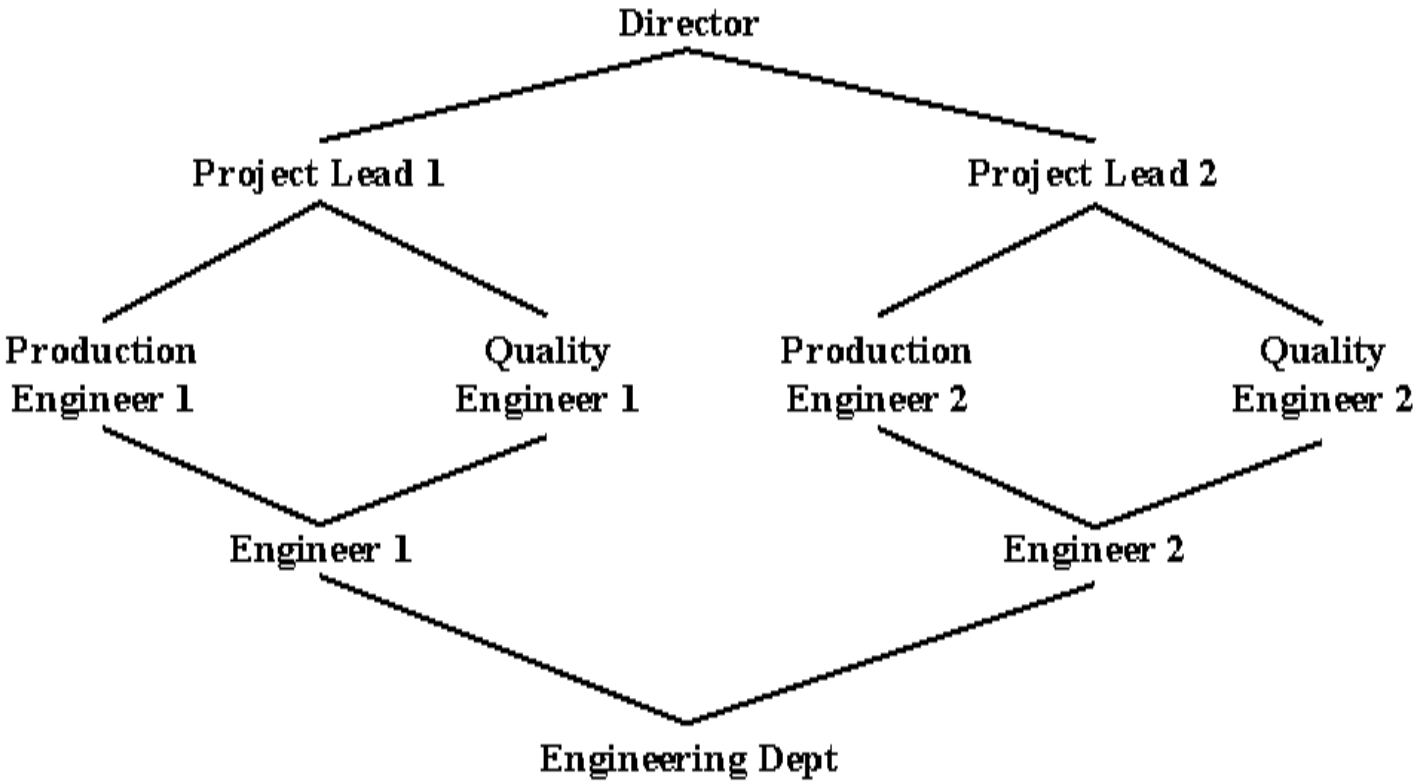



Figure 4.9 Example of Role Hierarchy

# Constraints - RBAC

- Provide a means of adapting RBAC to the specifics of administrative and security policies of an organization
- A defined relationship among roles or a condition related to roles
- Types:

**Mutually exclusive roles**

- A user can only be assigned to one role in the set (either during a session or statically)
- Any permission (access right) can be granted to only one role in the set

**Cardinality**

- Setting a maximum number with respect to roles

**Prerequisite roles**

- Dictates that a user can only be assigned to a particular role if it is already assigned to some other specified role

# Summary

- Access control principles
  - Access control context
  - Access control policies
- Subjects, objects, and access rights
- Discretionary access control
- UNIX file access control
  - Traditional UNIX file access control
  - Access control lists in UNIX
- Role-based access control
  - RBAC reference models