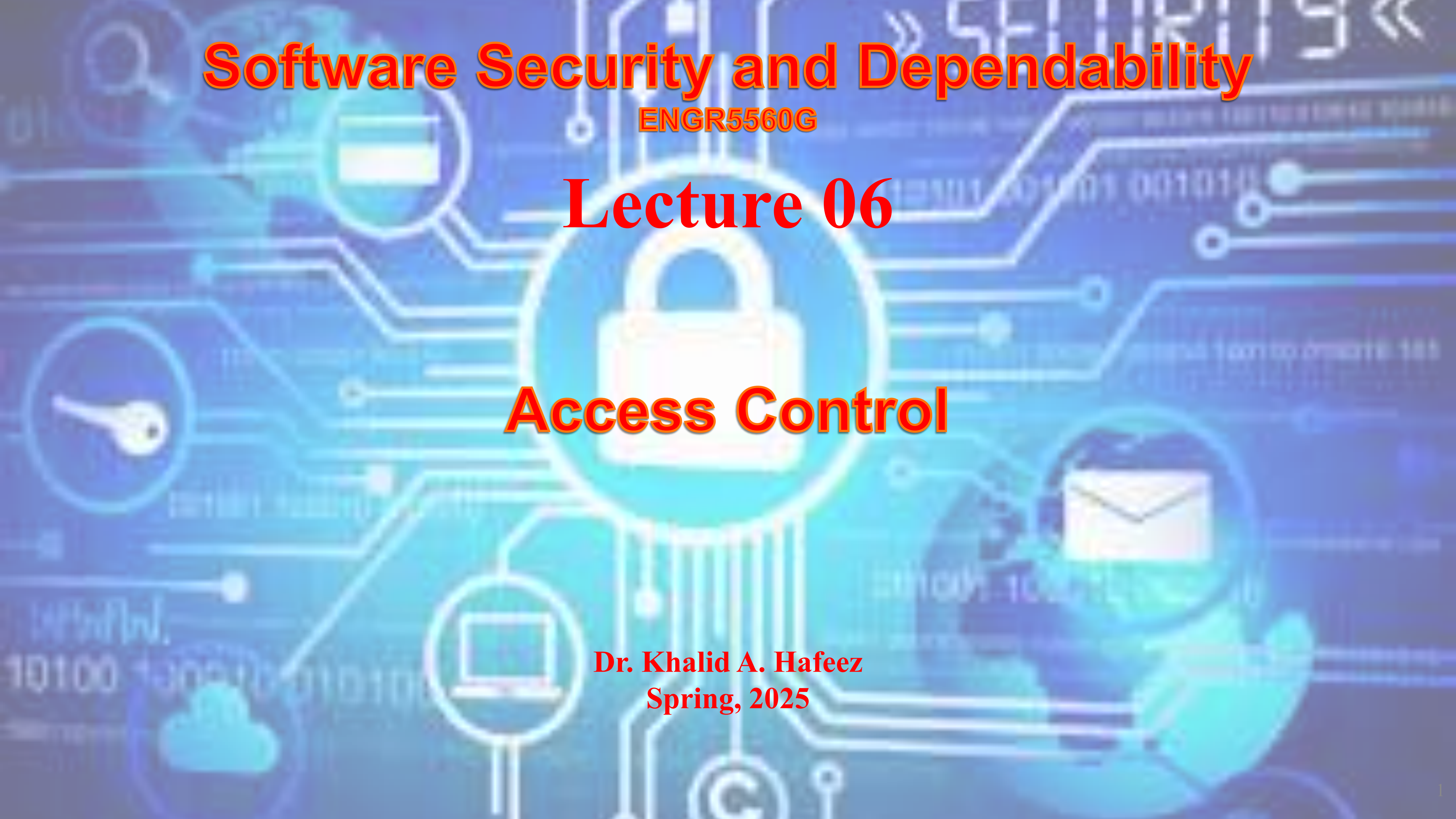


Software Security and Dependability

ENGR5560G

Lecture 06

Access Control

Dr. Khalid A. Hafeez
Spring, 2025

Objectives

- Explain how access control fits into the broader context that includes authentication, authorization, and audit.
- Define the four major categories of access control policies.
- Distinguish among subjects, objects, and access rights.
- Discuss the principal concepts of role-based access control.
- Summarize the NIST RBAC model.

Access Control Principles

- Access Control:
  - "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"
  - In a broad sense, all of computer security is concerned with access control
  - **The principal objectives of computer security are to:**

1. Prevent unauthorized access to resources,
2. Prevent legitimate users from accessing resources in an unauthorized manner,
3. Enable legitimate users to access resources in an authorized manner.

Access Control Principles

- Access Control Context: **involves the following entities and functions:**
  - Authentication: to verify the credentials of a user or system entity.
  - Authorization: to determine who is trusted for a given purpose.
  - Audit: to test for adequacy of system controls, to ensure compliance with established policy and operational procedures, to detect breaches in security, and to recommend any indicated changes in control, policy and procedures.

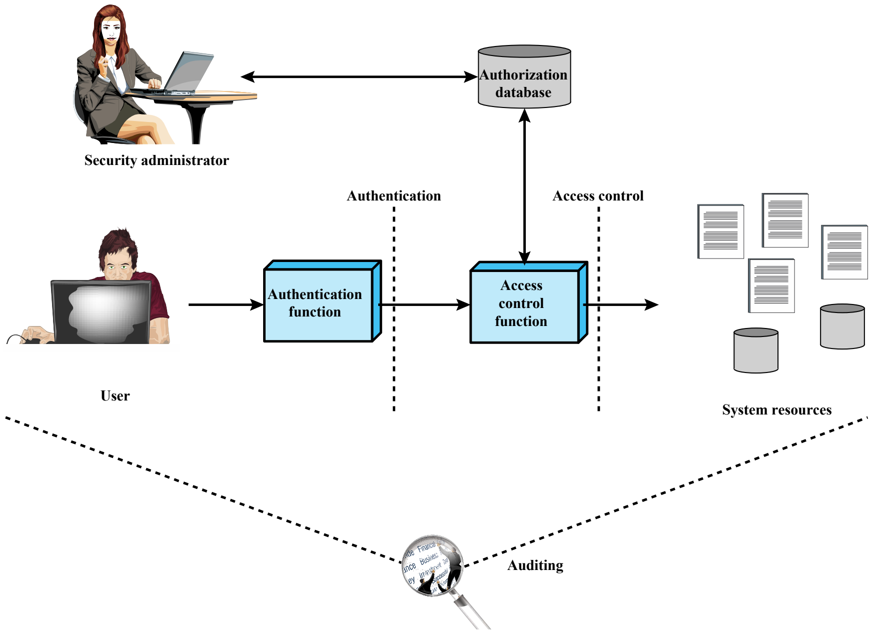


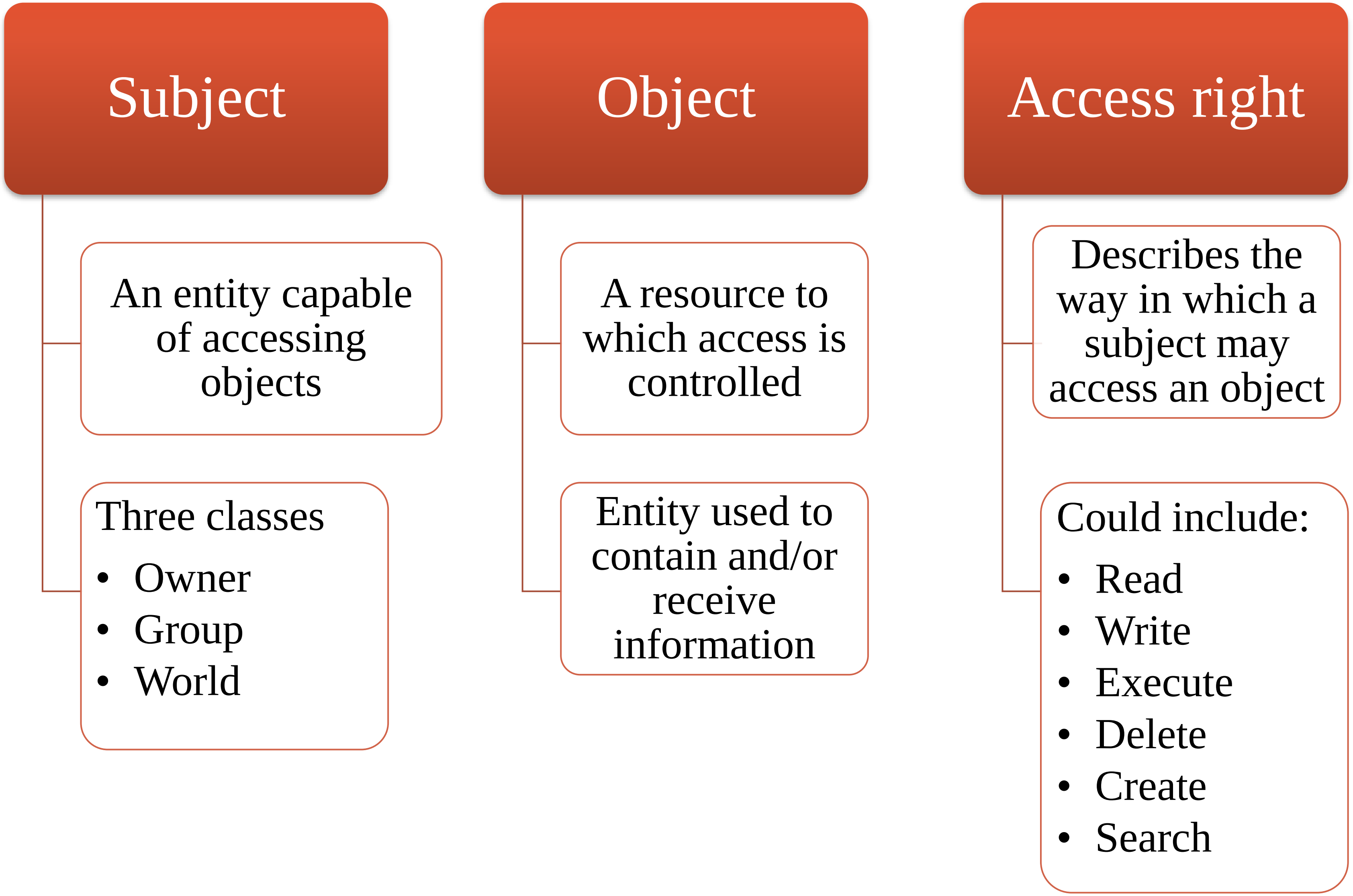
Figure 4.1 Relationship Among Access Control and Other Security Functions

Access Control Policies

- They dictates what types of access are permitted, under what circumstances, and by whom.
- **They are grouped into the following categories:**
  - **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.
    - An entity that has access rights may permit access to another entity.
  - **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

- The basic elements of access control are:



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

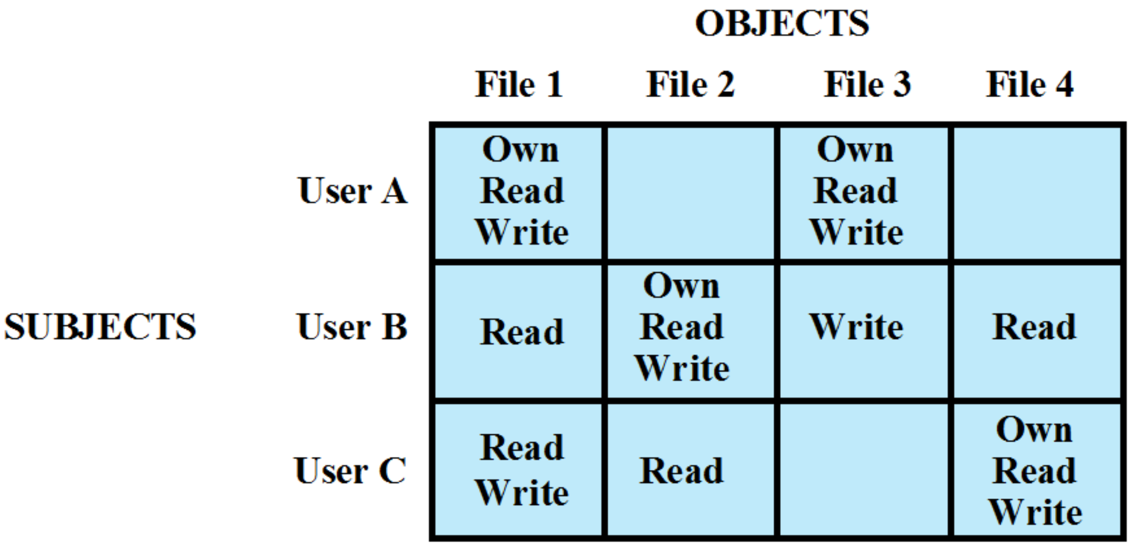
Access right

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

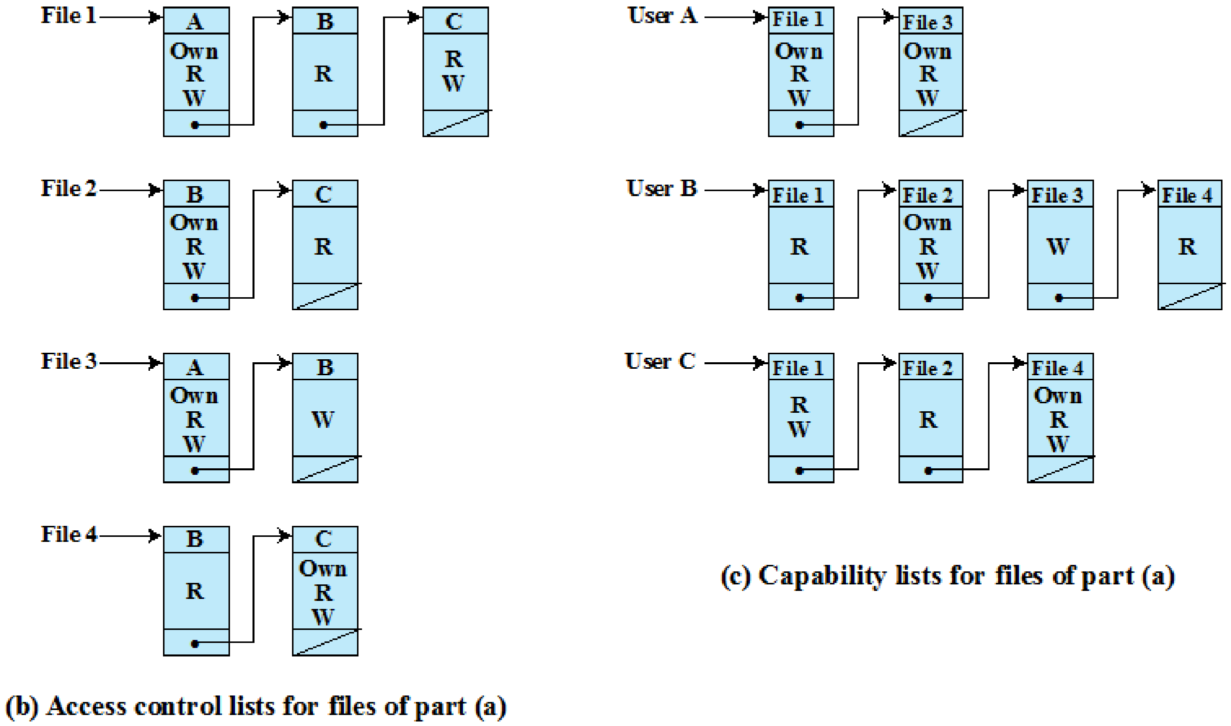


		OBJECTS			
		File 1	File 2	File 3	File 4
SUBJECTS	User A	Own Read Write		Own Read Write	
	User B	Read	Own Read Write	Write	Read
	User C	Read Write	Read		Own Read Write

(a) Access matrix

Discretionary Access Control (DAC)

- The matrix may be decomposed by columns, yielding access control lists (ACLs)
- Decomposition by rows yields capability tickets.
    - A capability ticket specifies authorized objects and operations for a particular user.



(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

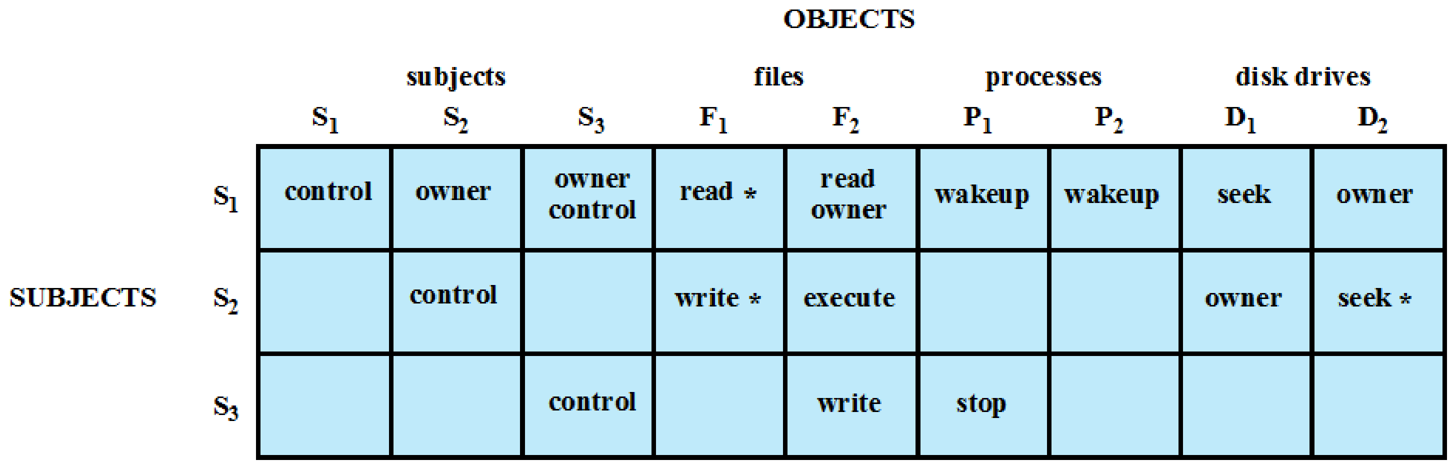
Discretionary Access Control (DAC)

- An **authorization table** contains one row for one access right of one subject to one resource.
  - Sorting or accessing the table by subject is equivalent to a capability list.
  - Sorting or accessing the table by object is equivalent to an ACL.
- Can be implemented by a relational database (SQL).

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

Discretionary Access Control (DAC)

- Extended Access Control Matrix
    - The universe of objects in the access control matrix is extended to include:
        - Processes: ability to delete a process, stop (block), and wake up a process.
        - Devices: ability to read/write the device, to control its operation (e.g., a disk seek), and to block/unblock the device for use.
        - Memory locations or regions: the ability to read/write certain regions of memory
        - Subjects: ability to grant or delete access rights of that subject to other objects.



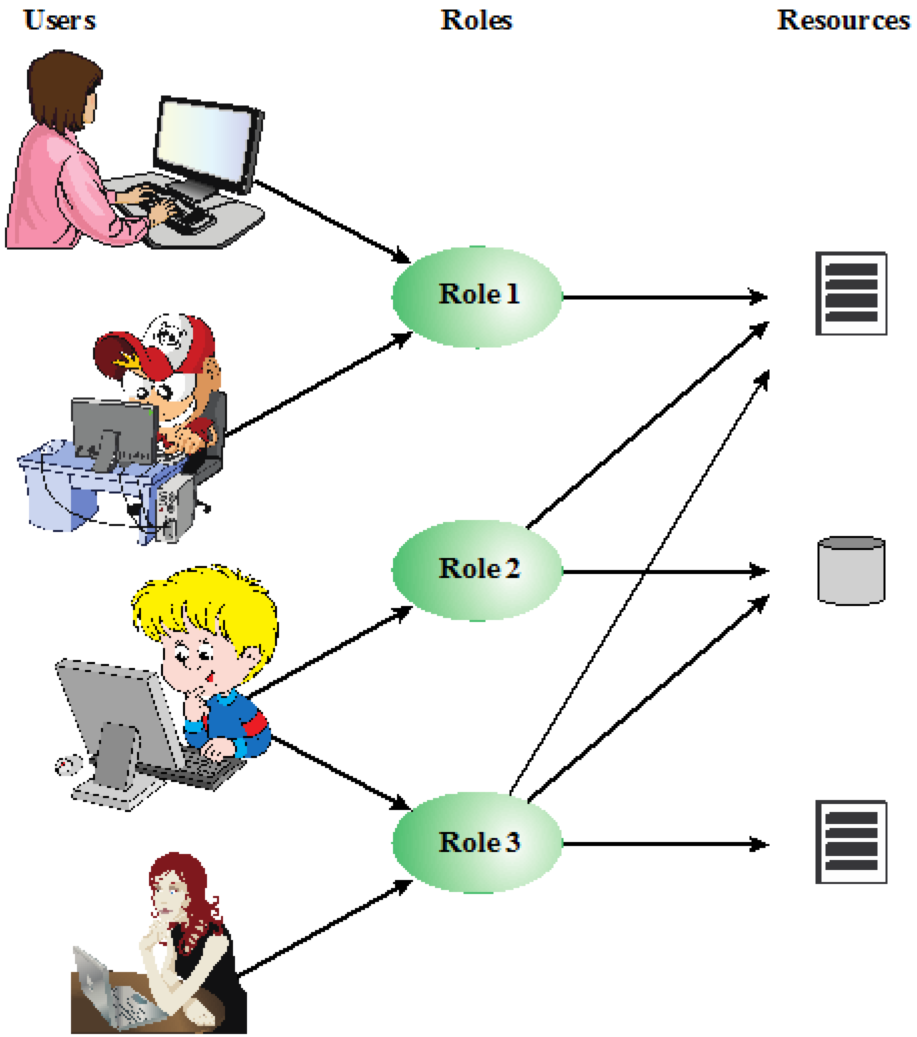
		OBJECTS								
		subjects			files		processes		disk drives	
		S_1	S_2	S_3	F_1	F_2	P_1	P_2	D_1	D_2
SUBJECTS	S_1	control	owner	owner control	read *	read owner	wakeup	wakeup	seek	owner
	S_2		control		write *	execute			owner	seek *
	S_3			control		write	stop			

* - copy flag set

*-copy flag: The subject can transfer the access right (with or without the copy flag)

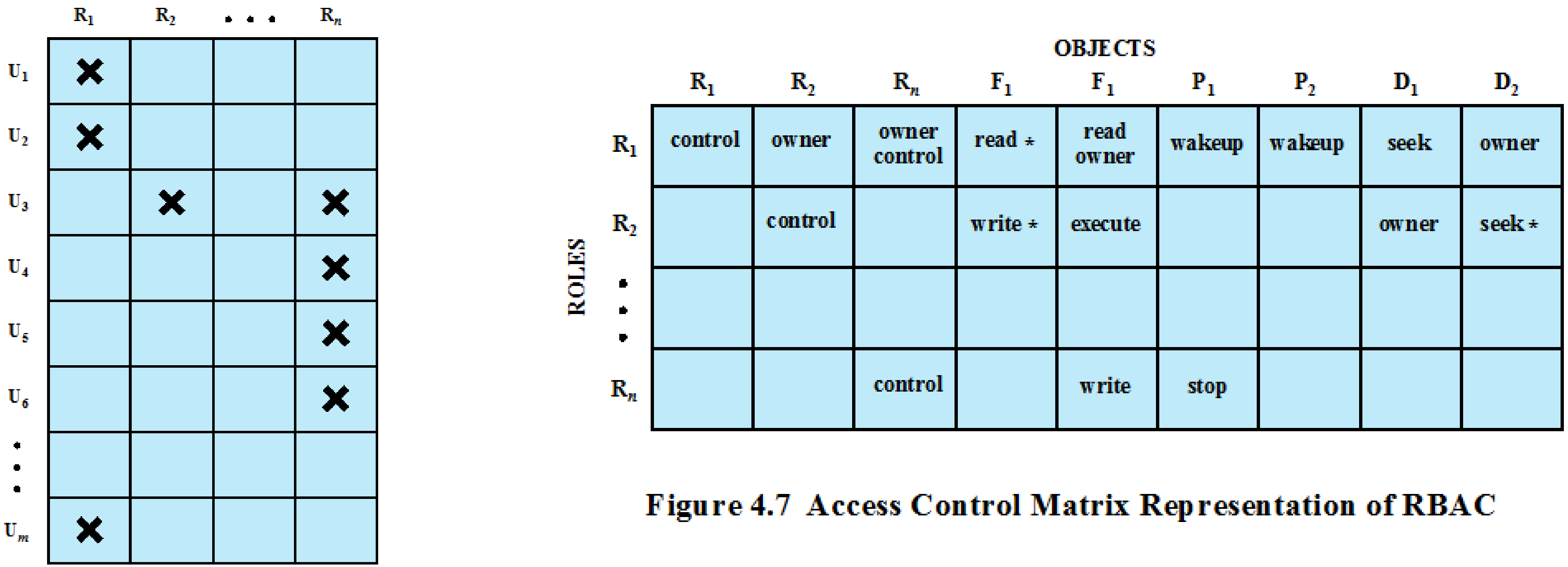
Role-Based Access Control

- Users, Roles, and Resources
  - RBAC is based on the roles that users assume in a system rather than the user's identity
    - RBAC defines a role as a job function within an organization.
    - RBAC systems assign access rights to roles instead of individual users.
    - In turn, users are assigned to different roles, either statically or dynamically, according to their responsibilities.
    - The set of roles in an organization is static (does not change much)



Role-Based Access Control (RBAC)

- Access Control Matrix Representation for RBAC
  - The access matrix can be used to depict the key elements of an RBAC system in simple terms
  - The left matrix relates individual users to roles.
  - The right matrix relates roles to objects with specific rights.
  - Each role should contain the minimum set of access rights needed for that role.



	R_1	R_2	. . .	R_n
U_1	✗			
U_2	✗			
U_3		✗		✗
U_4				✗
U_5				✗
U_6				✗
⋮				
U_m	✗			

	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

Attribute-Based Access Control (ABAC)

- **Example**: consider a configuration in which each resource has an attribute that identifies the subject that created the resource.
    - Then, a single access rule can specify the ownership privilege for all the creators of every resource.

Can define authorizations that express conditions on properties of both the resource and the subject

Strength is its flexibility and expressive power

Main obstacle to its adoption in real systems has been concern about the performance impact of evaluating predicates on both resource and user properties for each access

Web services have been pioneering technologies through the introduction of the eXtensible Access Control Markup Language (XACML)

There is considerable interest in applying the model to cloud services

Attribute-Based Access Control (ABAC)

- ABAC Model: Attributes

Subject attributes

- A subject is an active entity that causes information to flow among objects or changes the system state
- Attributes define the identity and characteristics of the subject such as (subject’s identifier, name, organization, job title, …

Object attributes

- An object (or resource) is a passive information system-related entity containing or receiving information
- Objects have attributes that can be leverages to make access control decisions
  - A MS Word document, may have attributes such as title, subject, date, and author
  - Web service attributes: ownership, service taxonomy, or even Quality of Service (QoS) attributes.

Environment attributes

- Describe the operational, technical, and even situational environment or context in which the information access occurs
- These attributes have so far been largely ignored in most access control policies
- Example: current date and time, the current virus/hacker activities, and the network’s security level

Attribute-Based Access Control (ABAC)

- ABAC is a logical access control model that:

Distinguishable because it controls access to objects by evaluating rules against the attributes of entities, operations, and the environment relevant to a request

Relies upon the evaluation of attributes of the subject, attributes of the object, and a formal relationship or access control rule defining the allowable operations for subject-object attribute combinations in a given environment

ABAC Systems are capable of enforcing DAC, RBAC, and MAC concepts

ABAC allows an unlimited number of attributes to be combined to satisfy any access control rule