PROTECTION

- Goals of protection,
- Principles of protection,
- Protection Rings,
- Domain of protection,
- Access matrix.

Introduction

- Protection involves controlling the access of processes and users to the resources defined by a computer system.
- The processes in an operating system must be protected from one another's activities.
- To provide this protection, we can use various mechanisms to ensure that only processes that have gained proper authorization from the operating system can operate on the files, memory segments, CPU, networking, and other resources of a system.

1. Goals of protection

- **Prevent Unauthorized Access** – Ensure that users and processes can only access resources they are permitted to, avoiding security breaches.
- **Ensure Data Integrity** – Protect system resources from accidental or malicious modification, maintaining accuracy and consistency.
- **Enforce User Authentication** – Verify user identities before granting access to sensitive data or system functions.
- **Limit Resource Usage** – Restrict the usage of CPU, memory, files, and I/O devices to prevent monopolization by a single process or user.
- **Facilitate Controlled Sharing** – Allow authorized users and processes to share system resources while preventing unauthorized interference.
- **Maintain System Stability** – Prevent one process from negatively affecting others, ensuring smooth system operations.
- **Support Audit and Accountability** – Track access and modifications to resources to identify security violations and ensure compliance.

2.Principles of Protection

- **Principle of Least Privilege** – Each process or user should have the minimum privileges necessary to perform its task, reducing potential security risks.
- **Access Control** – The system must define and enforce policies on who can access what resources and in what manner (read, write, execute).
- **Domain of Protection** – Users and processes operate within a defined "domain" that specifies their access rights. Domains can be user-based or process-based.
- **Access Matrix** – A conceptual model that defines which subjects (users/processes) have what type of access to which objects (resources).
- **Capability Lists and Access Control Lists (ACLs)** – Two common mechanisms for implementing access control:
  - **Capability Lists** associate each process with the resources it can access.
  - **Access Control Lists (ACLs)** define the access rights for each resource.

- **Revocation of Access Rights** – The system should support revoking privileges dynamically when needed.
- **Role-Based Access Control (RBAC)** – Assigns permissions based on roles rather than individual users, simplifying management.
- **Security Policies** – The OS must enforce policies such as Mandatory Access Control (MAC) and Discretionary Access Control (DAC) to regulate access.
- **Implementation of Protection Mechanisms** – Use secure methods like encryption, authentication, and auditing to ensure compliance.
- **Separation of Privilege** – Access to critical operations should require multiple conditions to be met, reducing the risk of single points of failure.

3.Protection Rings

- Refers to a hierarchical security model used in computer systems to regulate access to resources. The model is structured in multiple levels, typically four, with **Ring 0** being the most privileged and **Ring 3** the least privileged.

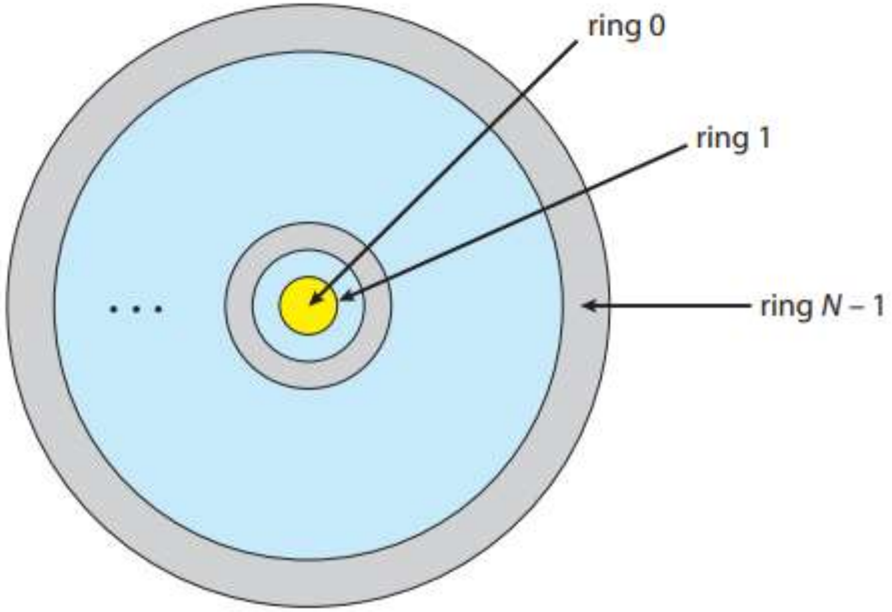


Figure 17.1 Protection-ring structure.

- **Ring 0 (Kernel Mode):**
  - Highest privilege level.
  - Executes critical operating system code (e.g., process scheduling, memory management).
  - Has direct access to hardware and all system resources.
- **Ring 1 (OS Services):**
  - Used for system services and device drivers that require elevated privileges but do not need full kernel access.
- **Ring 2 (Middleware and Drivers):**
  - Intended for less critical drivers and privileged applications that need some hardware access.
- **Ring 3 (User Mode):**
  - Lowest privilege level.Used for executing user applications.
  - Restricted access to system resources;
  - must use system calls to request OS services.

4. Domain of Protection

- Rings of protection separate functions into domains and order them hierarchically. A generalization of rings is using domains without a hierarchy.
- A computer system can be treated as a collection of processes and objects.
- The domain of protection provides a structured way to control which processes can access system resources and how they can interact with them.
- It enhances security, reliability, and fault isolation, preventing processes from interfering with each other or with critical system functions.
- The **protection domain** consists of a set of access rights, which specify the **objects** (resources like files, memory, or devices) and the **operations** (such as read, write, or execute) that a process can perform on those objects.

Key Aspects of the Domain of Protection:

1. **Definition of a Domain:**
    1. A **domain** is a set of **(object, access rights)** pairs.
    2. Each process executes within a particular domain, determining its permitted actions.
2. **Multiple Domains:**
    1. A system may have **multiple protection domains** (e.g., kernel mode, user mode, administrative mode).
    2. A process can belong to one domain at a time but may **switch domains** under specific conditions (e.g., executing system calls).
3. **Access Rights:**
    1. Each domain specifies **which resources a process can access** and **what operations it can perform** on them.
    2. Examples of access rights include **read, write, execute, delete, and modify permissions.**

4. **Dynamic Domain Switching:**

- Some systems allow a process to switch domains to gain temporary access to more privileges.
- This can be done using mechanisms like system calls or role-based access control.

5. **Implementation of Protection Domains:**

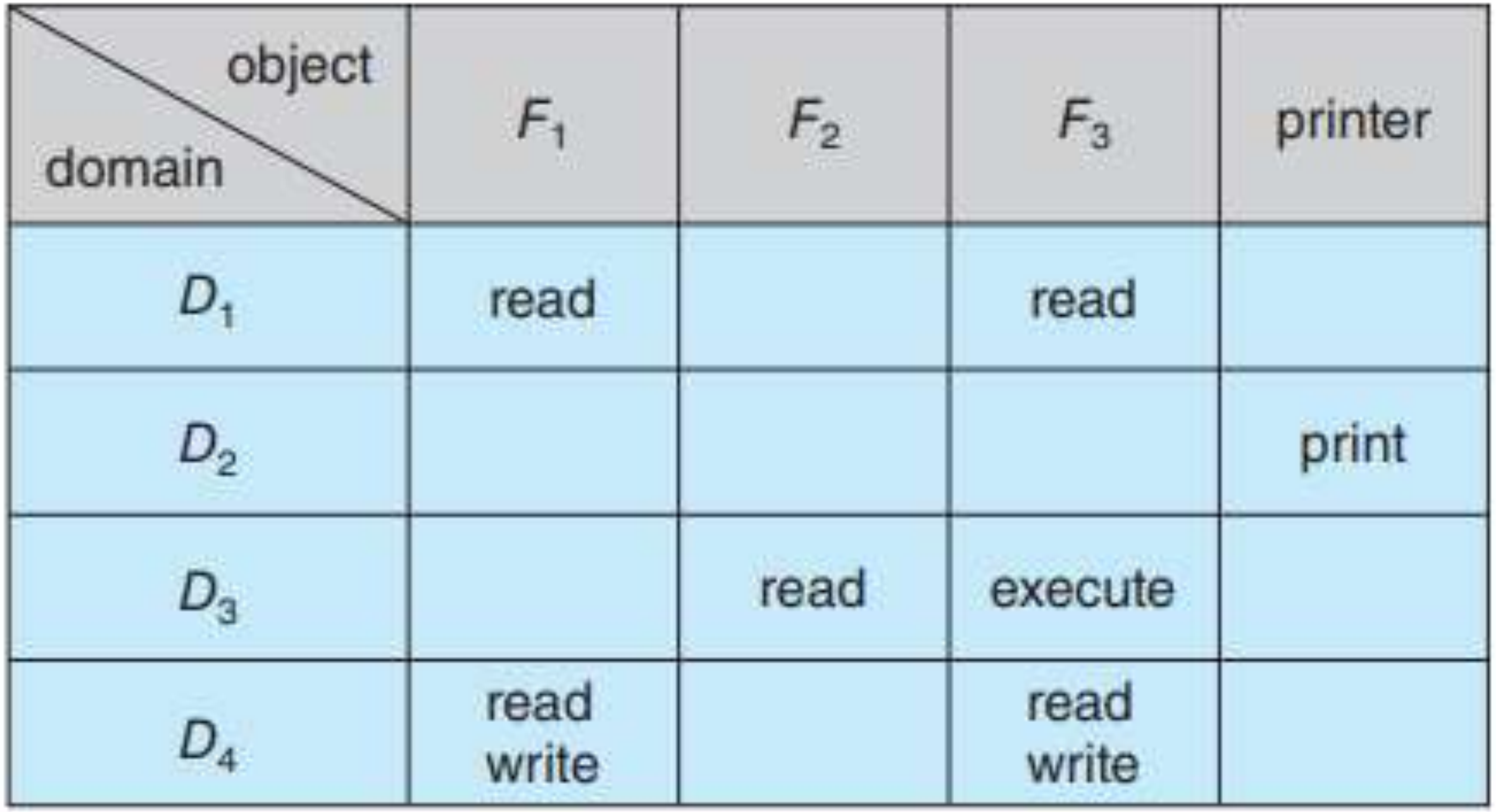
- Protection domains can be implemented using Access Control Lists (ACLs) or Capability Lists.
- Access Control List (ACL): Each object has a list specifying which domains can access it.
- Capability List: Each domain has a list specifying which objects it can access and how.

6. **Domain Enforcement:**

- The operating system ensures that a process cannot exceed its privileges.
- Unauthorized access attempts are blocked through hardware and software mechanisms.

5. Access matrix

- The general model of protection can be viewed abstractly as a matrix, called an access matrix.
- The rows of the access matrix represent domains, and the columns represent objects.
- Each entry in the matrix consists of a set of access rights. Because the column defines objects explicitly, we can omit the object name from the access right.
- The entry access(i,j) defines the set of operations that a process executing in domain Di can invoke on object Oj.
- We must ensure that a process executing in domain Di can access only those objects specified in row i, and then only as allowed by the access-matrix entries.



object domain	F_1	F_2	F_3	printer
D_1	read		read	
D_2				print
D_3		read	execute	
D_4	read write		read write	

Figure 17.5 Access matrix.

- There are four domains and four objects— three files (F1, F2, F3) and one laser printer.
- A process executing in domain D1 can read files F1 and F3.
- A process executing in domain D4 has the same privileges as one executing in domain D1; but in addition, it can also write onto files F1 and F3.
- The laser printer can be accessed only by a process executing in domain D2