KARNAVATI UNIVERSITY

Operating Systems

Course Code: **71203002004**

File Access Control

by -
Minal Rajwar

File Access Control

Files often store sensitive data (e.g., passwords, credit card numbers).
Hence, file systems must control user access to prevent unauthorized use.

Why is it necessary?

- Any user can reference any pathname.
- Protects personal and sensitive info from unauthorized access.

Techniques for File Access Control

1. Access Control Matrix

- A 2D matrix with users (rows) and files (columns).
- Entry = `1` → user has access, `0` → no access.
- Example: User 5 can access all files, User 4 only file 1.

Disadvantages:

1. Very large and sparse for big systems.
2. Wastes storage.
3. Slower access time.
4. Becomes larger if multiple access types (read, write, execute, etc.) are added.

User \ File	1	2	3	4	5	6	7	8	9	10
1	1	1	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	0	0	0
3	0	1	0	1	0	1	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0
5	1	1	1	1	1	1	1	1	1	1
6	0	0	0	0	0	1	1	0	0	0
7	1	0	0	0	0	0	0	0	0	1
8	1	0	0	0	0	0	0	0	0	0
9	1	1	1	1	0	0	0	0	1	1
10	1	1	0	0	1	1	0	0	0	1

Techniques for File Access Control

2. File Permissions

Common permissions:

- **Read (R):** open and read.
- **Write (W):** modify and save.
- **Delete (D):** remove file/directory.
- **Execute (X):** run executable files.

Permissions can be:

- Granted/denied to **one person** or a **group**.
- Combined (e.g., give Read + Write but deny Delete).

Techniques for File Access Control

3. Access Control by User Classes

Uses **less space** than access control matrix.

Classification:

1. **Owner** → Creator, full access.
2. **Specified User** → Owner allows another user.
3. **Group** → Project team members share files.
4. **Public** → Accessible by all users (usually Read/Execute only).

Advantages:

1. Less storage overhead.
2. Easier to give permissions to groups with a single entry.

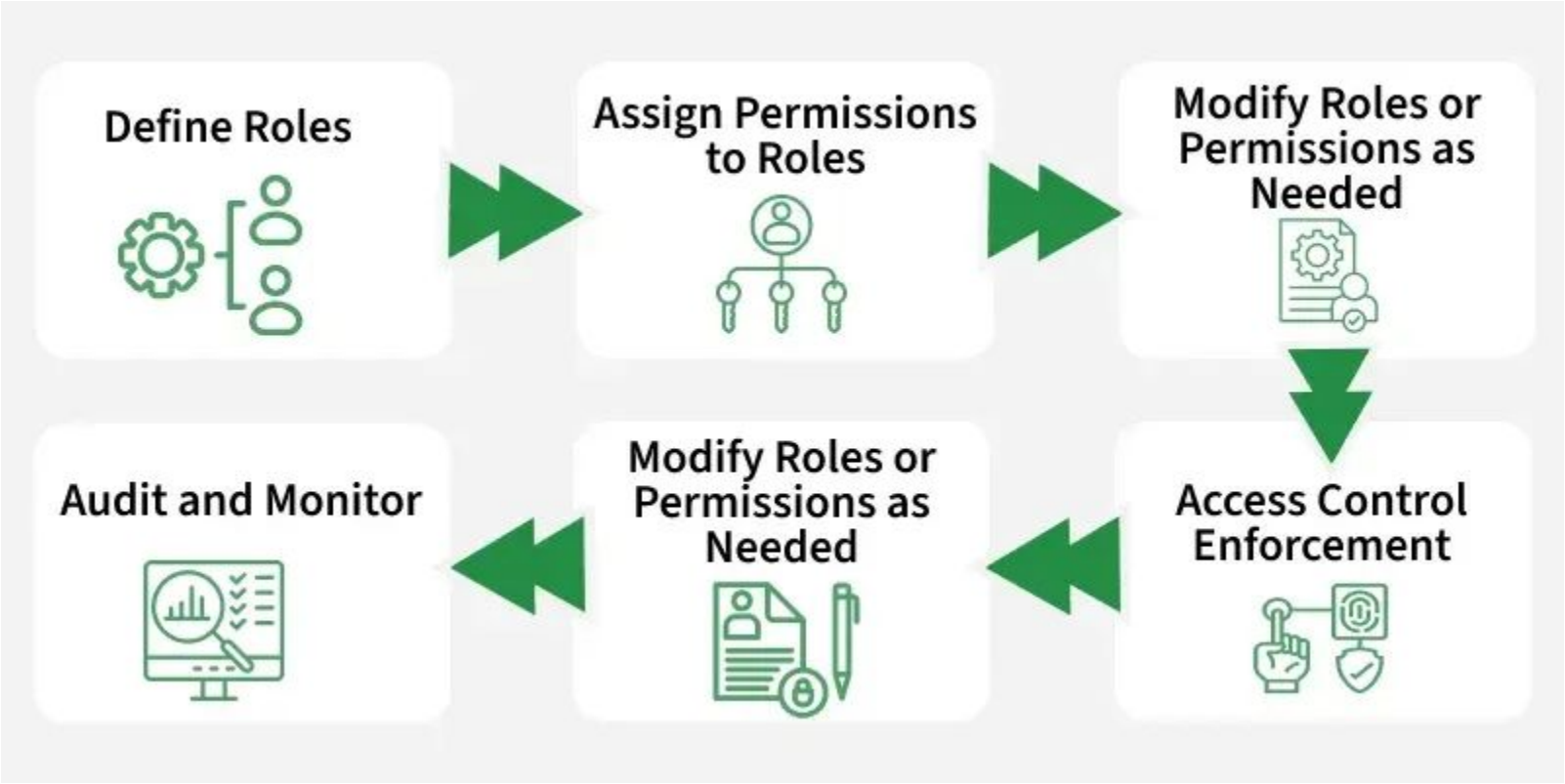
Protection vs Security

S. No.	Protection	Security
1	Works under trusted processes.	Works even if processes are malicious.
2	Guards against errors by non-malicious users.	Guards against malicious users/attacks.

Role-Based Access Control (RBAC)

RBAC is a security method where **access is based on a user's job role**, not individual permissions.

- Roles are created with specific privileges (read, write, delete, etc.).
- Users are assigned to roles and automatically inherit those permissions.
- Simplifies permission management and reduces unauthorized access.


Define Roles
Assign Permissions to Roles
Modify Roles or Permissions as Needed
Access Control Enforcement
Modify Roles or Permissions as Needed
Audit and Monitor

How RBAC Works in an OS

Define Roles:

- Example roles: Administrator, User, Guest, Editor.

Assign Permissions to Roles:

- Administrator → full access (read/write/delete)
- User → read-only access
- Editor → read/write access

Assign Users to Roles:

- Users inherit all permissions of their assigned roles.
- Changing a user's role automatically updates their permissions.

How RBAC Works in an OS

Example

- Role “Editor” → can edit and delete files.
- Role “Reader” → can only view files.
- New employee joins as Editor → automatically gets Editor permissions.
- Employee moves to a different role → permissions update automatically.

DISCUSSION & REVISION

1. What type of access control uses a matrix of users and files?
2. In file permissions, which right allows a user to run a program?
3. What is the access control method that assigns permissions based on roles?
4. Who has full access to a file in user classes?
5. Protection aims to guard against errors, while security guards against malicious entities.

REFERENCES

1. https://www.studocu.com/in/document/guru-gobind-singh-indraprastha-university/operating-systems/file-access-control-file-access-control-notes/6508882 9
2. https://www.geeksforgeeks.org/computer-networks/role-based-access-control/