Cyber Warriors: A Comprehensive Introduction to Cybersecurity Tools and Techniques

June 24-28, 2024

Murtuza Jadliwala

murtuza.jadliwala@utsa.edu

SPrITE LAB

UTSA

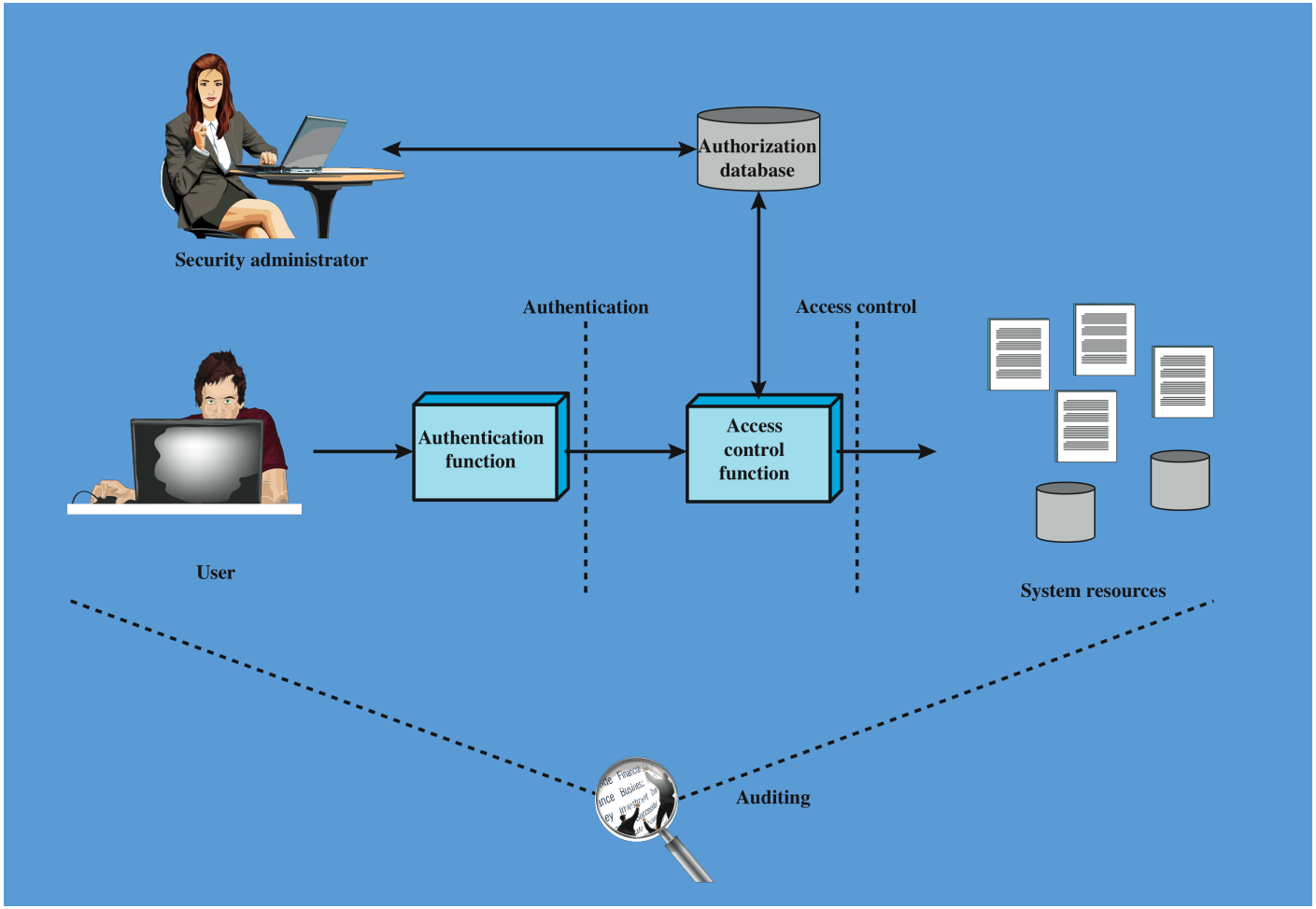
NSF

Introduction to Access Control

Access Control

"The prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner."

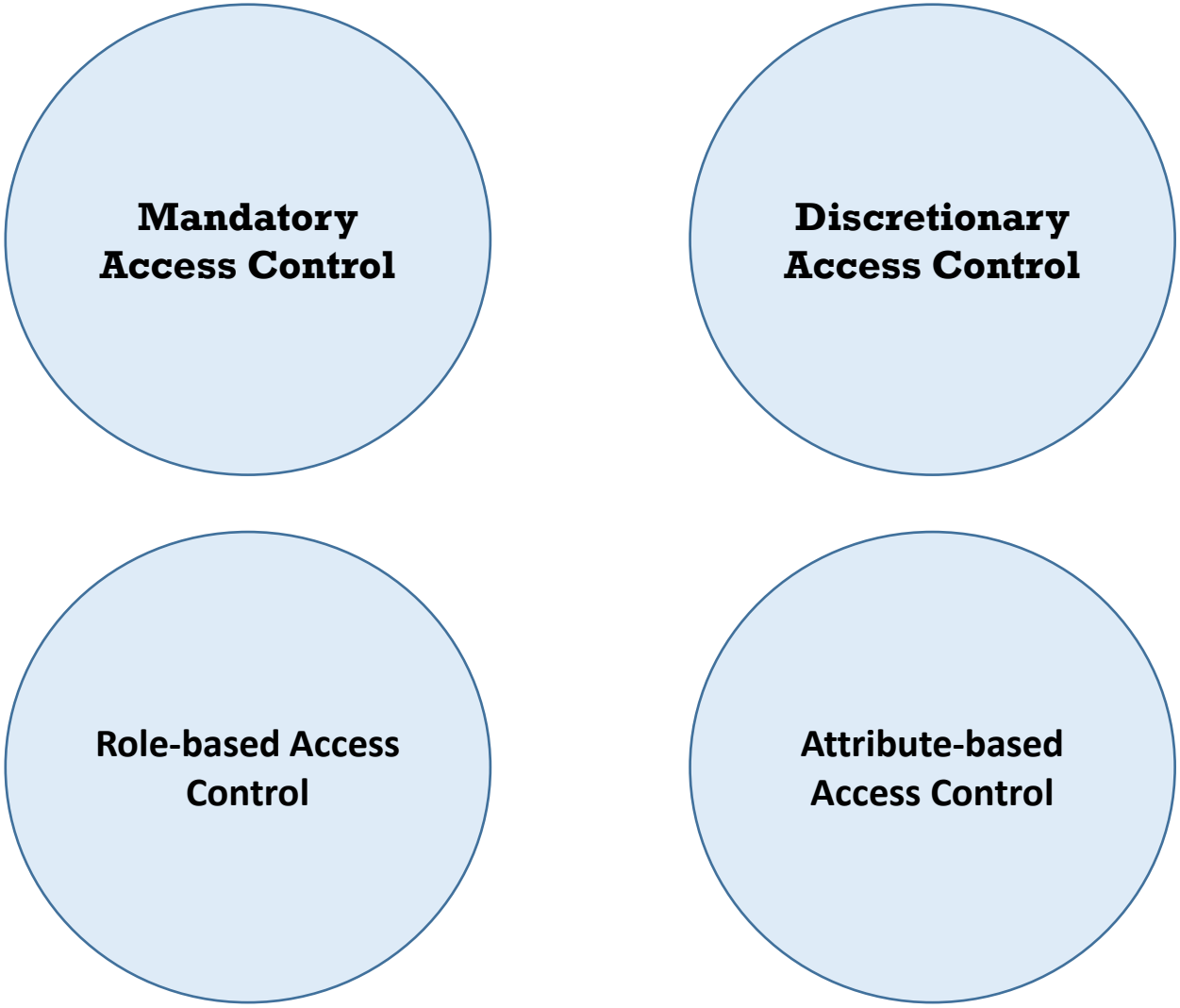
How Does Access Control Work?



Access Control Elements

- **Subject**
  - Entity capable of accessing objects - equates with process
  - Accountable for the actions they initiate
  - Three classes: owner, group, world
- **Object**
  - Resource to which access is controlled - entity that contains and/or receive information
  - Protection depends on the environment in which access control operates
- **Access right**
  - Describes the way in which a subject may access an object
  - e.g., read, write, execute, delete, create, search

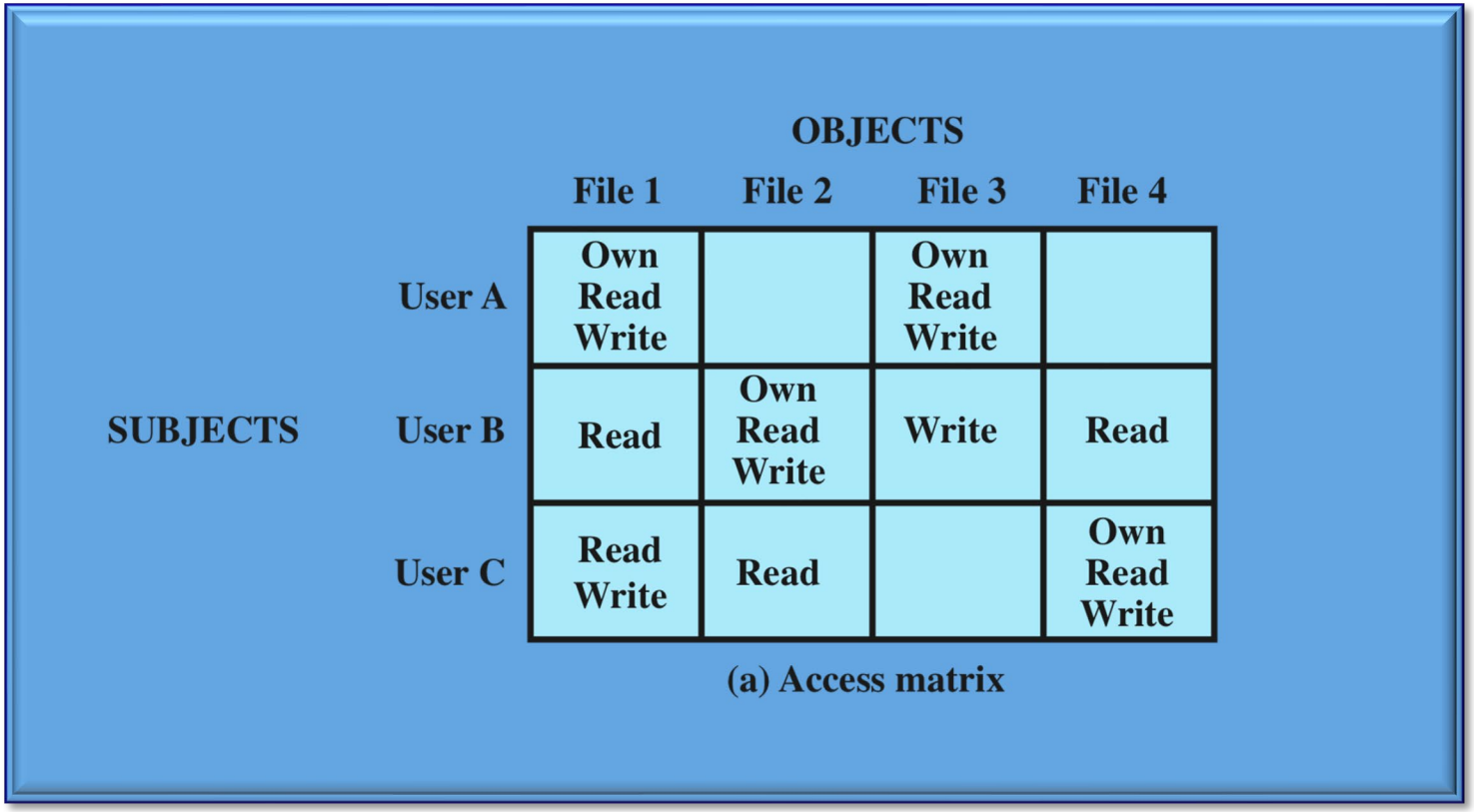
Access Control Policies



Discretionary Access Control (DAC)

- Scheme in which an entity may 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
  - 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

Access Matrix



		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

UNIX File Access Control

UNIX files are administered using inodes (index nodes)

- **Control structures with key information needed for a particular file**
- **Several file names may be associated with a single inode**
- **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

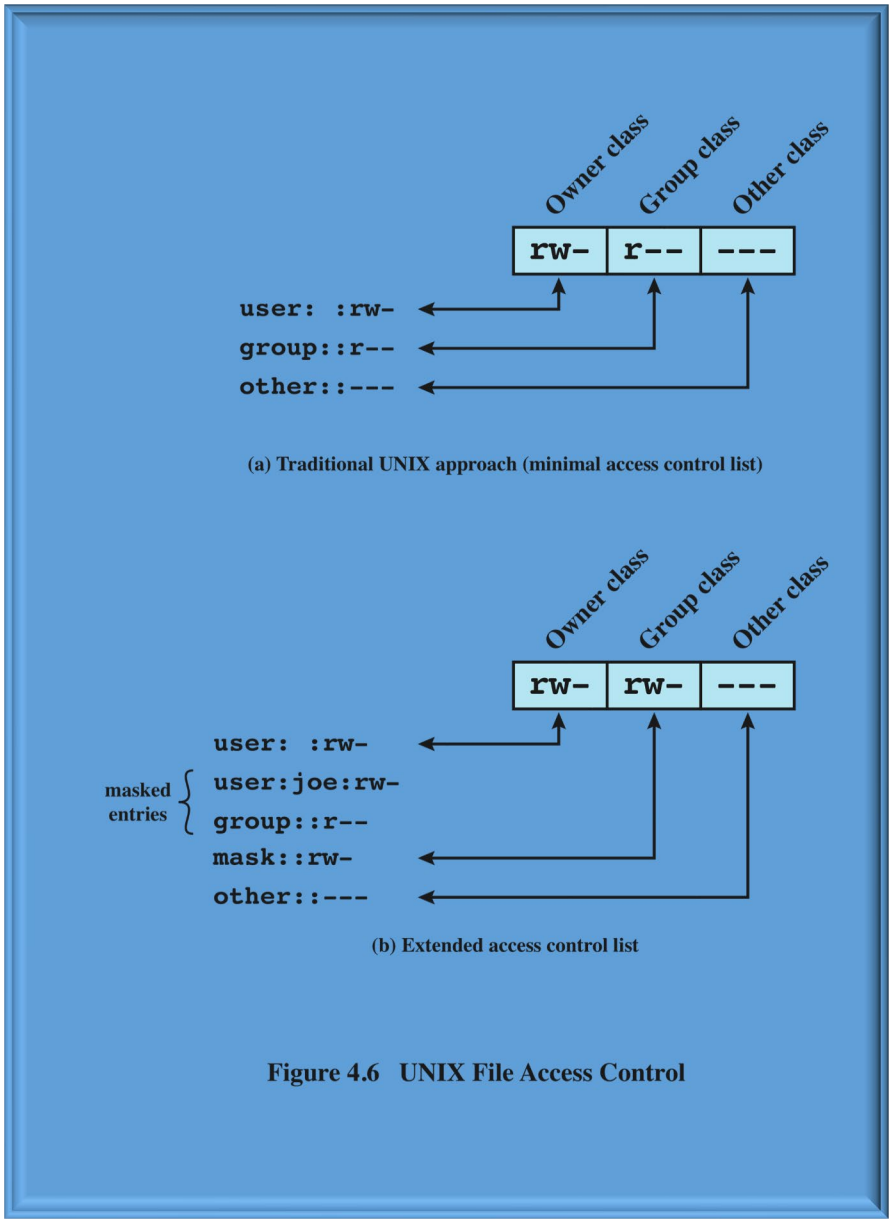


Figure 4.6 UNIX File Access Control

UNIX File Permissions – Decimal to Binary

Decimal	Read	Write	Execute
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Traditional UNIX File Access Control

- "set user ID" (SetUID)
- "set group ID" (SetGID)
  - System temporarily uses rights of the file owner / group in addition to the real user's rights when making access control decisions
  - Enables privileged programs to access files / resources not generally accessible
- Sticky bit
  - When applied to a directory it specifies that only the owner of any file in the directory can rename, move, or delete that file
- Superuser (root)
  - Exempt from usual access control restrictions
  - Has system-wide access