Chapter 4
Access Control

Access Control

• “The prevention of unauthorized use of a resource, including the prevention of use of a resource in an unauthorized manner”
• Central element of computer security
Authorization & Access Control

Who is trying to access a protected resource?

Authentication

Access Control Models

Authorization

Who should be allowed to access which protected resources?

Access Control Architectures

Enforcement

How does the system enforce the specified authorization?

who should be allowed to change the access?

Access Control Principles

Access Control Function
Access Control Policies

Access Control Requirements

• Reliable input
• Fine and coarse specifications of policies
• Least privilege
• Separation of duty
• Open and closed policies
• Policy combinations, conflict resolution
• Administrative policies

Access Control Elements

• Subject - entity that can access objects
  – A process representing user/application
  – Often have 3 classes: owner, group, world
• Object - access controlled resource
  – e.g. files, directories, records, programs etc
  – Number/type depend on environment
• Access right - way in which subject accesses an object
  – e.g. read, write, execute, delete, create, search
Discretionary Access Control

- Often provided using an access matrix
  - Lists subjects in one dimension (rows)
  - Lists objects in the other dimension (columns)
  - Each entry specifies access rights of the specified subject to that object

- Access matrix is often sparse
- Can be decomposed by either row or column

Access Control Model

Access Matrix Model

<table>
<thead>
<tr>
<th>SUBJECTS</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>r, w</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td></td>
<td>r, w</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Authorization System

- Consists of
  - A set of access control policies
    - Defined in terms of subjects, objects, and permissions
  - Statements about how to change the policies
    - Who can modify
    - What can be modified
    - What is the modified privilege
- Protection state
  - Set of information that specifies the access rights for each subject with respect to each object at a given point in time

Safety Problem

Initial Access Matrix

Rules for Modifying the Access Matrix

Is it possible to enter the right r in the cell A[V,C] of the access matrix?

Yes / No

Given an access matrix M and a access privilege, r, verifying the safety of M with respect to r is an undecidable problem

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 stored 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

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
  - Is exempt from usual access control restrictions
  - Has system-wide access
Access Control Lists (ACLs) in UNIX

- Modern UNIX systems support ACLs
  - FreeBSD, OpenBSD, Linux, Solaris

- 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
- Includes an additional protection bit that indicates whether the file has an extended ACL

When a process requests access to a file system object two steps are performed:
- Step 1: Select the most appropriate ACL
- Step 2: Check if the matching entry contains sufficient permissions

Need for MAC – Bypassing Discretionary Authorization

File A
X: r, w
Y: 

File B
X: w
Y: r, w

Program Goodies
Trojan Horse

Mandatory Access Control

- Access control at the system level that is more fundamental than anything determined by a subject
- When a system mechanism controls access to an object and an individual user cannot alter that access, the control in mandatory access control (MAC, occasionally called rule-based access control)
Multilevel Security (MLS)

- A class of system that has system resources (particularly stored information) at more than one security level (i.e., has different types of sensitive resources) and that permits concurrent access by users who differ in security clearance and need-to-know, but is able to prevent each user from accessing resources for which the user lacks authorization.

Multi-level Security with MAC

Two security levels
- High, Low
- Low < High

- L(X) = High
- L(Y) = Low
- L(A) = High
- L(B) = Low

Program Goodies
- Trojan Horse
- X: read, write

File A
- High
- X: r, w
- Y: 

File B
- Low
- X: w
- Y: r, w