Computer Security: Principles and Practice

Chapter 4 – Access Control

First Edition
by William Stallings and Lawrie Brown
Lecture slides by Lawrie Brown
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

• assume have users and groups
  - authenticate to system
  - assigned access rights to certain resources on system
Access Control Principles
Access Control Policies

- Discretionary access control policy
- Mandatory access control policy
- Role-based access control policy
Access Control Requirements

- reliable input
- fine and coarse specifications
- 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 decompose by either row or column
Access Control Structures
# Access Control Model

<table>
<thead>
<tr>
<th>SUBJECTS</th>
<th>S₁</th>
<th>S₂</th>
<th>S₃</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₁</td>
<td>control</td>
<td>owner</td>
<td>owner control</td>
</tr>
<tr>
<td>S₂</td>
<td>control</td>
<td>write *</td>
<td>execute</td>
</tr>
<tr>
<td>S₃</td>
<td>control</td>
<td>write</td>
<td>stop</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OBJECTS</th>
<th>F₁</th>
<th>F₁</th>
<th>P₁</th>
<th>P₂</th>
<th>D₁</th>
<th>D₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>subjects</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S₁</td>
<td>read *</td>
<td>read owner</td>
<td>wakeup</td>
<td>wakeup</td>
<td>seek</td>
<td>owner</td>
</tr>
<tr>
<td>S₂</td>
<td></td>
<td>execute</td>
<td></td>
<td></td>
<td>owner</td>
<td>seek *</td>
</tr>
<tr>
<td>S₃</td>
<td></td>
<td>stop</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* - copy flag set
Access Control Function
Protection Domains

- set of objects with associated access rights
- in access matrix view, each row defines a protection domain
  - but not necessarily just a user
  - may be a limited subset of user’s rights
  - applied to a more restricted process
- may be static or dynamic
UNIX File Concepts

- UNIX files administered using inodes
  - control structure with key info on file
    - attributes, permissions of a single file
  - may have several names for same inode
  - have inode table / list for all files on a disk
    - copied to memory when disk mounted
- directories form a hierarchical tree
  - may contain files or other directories
  - are a file of names and inode numbers
UNIX File Access Control

- Owner can read, write, and execute the file
- Any user in the owner's group can read and write the file
- Users outside the group cannot read, write, or execute the file.
UNIX File Access Control

- “set user ID” (SetUID) or “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
  - on directory limits rename/move/delete to owner

- superuser
  - is exempt from usual access control restrictions
UNIX Access Control Lists

- modern UNIX systems support ACLs
- can specify any number of additional users / groups and associated rwx permissions
- ACLs are optional extensions to std perms
- group perms also set max ACL perms
- when access is required
  - select most appropriate ACL
    - owner, named users, owning / named groups, others
  - check if have sufficient permissions for access
Role-Based Access Control
Role-Based Access Control
Role-Based Access Control

(a) Relationship among RBAC models

(b) RBAC models
NIST RBAC Model

SSD = static separation of duty
DSD = dynamic separation of duty
RBAC For a Bank

Human Resources Department
- User IDs
- Assigns
- Roles
  - Functions
  - Positions

Application Administration
- Application
  - Access Right

Authorization Administration
- Role
  - Application

1 \rightarrow 1-4

N \rightarrow M

M \rightarrow N
Summary

• introduced access control principles
  - subjects, objects, access rights
• discretionary access controls
  - access matrix, access control lists (ACLs), capability tickets
  - UNIX traditional and ACL mechanisms
• role-based access control
• case study