CS 356 – Lecture 7
Access Control

Fall 2013
Review

• Chapter 1: Basic Concepts and Terminology
  – Integrity, Confidentiality, Availability, Authentication, and Accountability
  – Types of threats: active vs. passive, insider/outsider
  – Lots of terminology and general concepts

• Chapter 2: Basic Cryptographic Tools
  – Symmetric key encryption and secure hashing
  – Public key cryptography
  – Random Numbers

• Chapter 3 – User Authentication
  – Passwords
  – Checking passwords and other user auth techniques
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

• assume have users and groups
  – authenticate to system
  – assigned access rights to certain resources on system
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 Model

* - copy flag set
Access Control Rules for S0

• Transfer a (or a*) to S,X
  – if a* in A[S0,X] then store a (or a*) in A[S,X]

• Grant a (or a*) to S,X
  – if “owner” in A[S0,X] then store a (or a*) in A[S,X]

• Delete a (or a*) from S,X
  – Control in A[S0,S] or Owner in A[S0,X], then delete a (or a*) from A[S,X]

• Read S,X
  • Control in A[S0,S] or Owner in A[S0,X], then display A[S,X]

• Create and Destroy Subjects and Objects
Access Control Function

Subject: $S_i$  
- read $F$  
- wake up $P$

Access control mechanisms:
- File system
- Memory addressing hardware
- Process manager
- Terminal & device manager
- Instruction decoding hardware
- Access matrix monitor

Objects:
- Files
- Segments & pages
- Processes
- Terminal & devices
- Instructions

System intervention:
- $S_i$  
- $S_j$  
- $S_k$  
- $S_m$  
- $S_{m'}$

Access matrix:
- read
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

- rwxrwx------
  - 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
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
Don’t Forget

- New homework set assigned, see the web site; due next Tuesday at start of class

- a LOT of you are procrastinating with the 1st project. This is a really bad idea. Time flies. no, really!!