Filesystem Security

Daniel Andresen

CIS520 – Operating Systems

File System Security

• protecting information from unauthorized disclosure or modification.

• Issues:

- secrecy prevent disclosure.
- integrity prevent modification.
- **Protection Mechanisms:** a method used to implement a policy to safeguard data.
- **Policy vs. Mechanism**: A policy is a statement used to specify whose data are to be protected from whom.
- A mechanism is how the policy is actually enforced by the system, and this will be our emphasis.

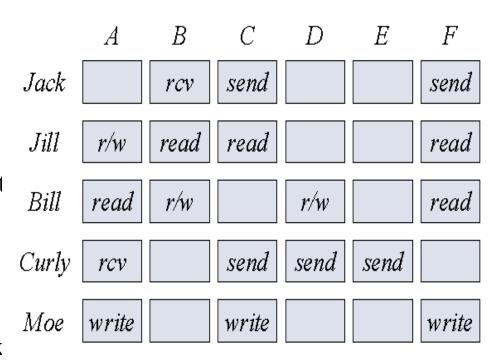
Protection Domains & Objects

- **object** = computer resource, either hardware (CPU, printer, etc.) or software (files, processes, etc.).
- **right** = an appropriate operation on an object. (read, write)
- **protection domain** = set of (object, rights) pairs.
- At every instance in time, each process runs in some protection domain.
 - (e.g. in UNIX the domain of a process is defined by a user's id (uid) and group id (gid))
- A system call causes a domain switch
 - e.g. when a process EXECs a file with the **SETUID** or **SETGID** bits on, the process acquires a new effective UID or GID with a different (UID, GID) combination. For example, passwd.

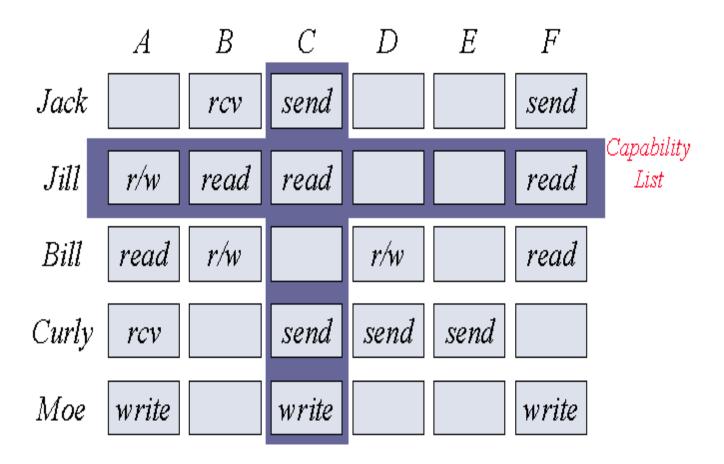
How do we keep track of which object belongs to which domain?

Access Control Matrix

- Authorization problems can be represented abstractly by an *access matrix*.
- each **row** represents a subject/principal/domain
- each column represents an object
- each **cell**: accesses permitted for the {subject, object} pair (read, write, delete, execute, search, control, etc.)
- In real systems, the access matrix is sparse and dynamic.
- We need a *flexible* and *efficient* representation, and a model for governing changes to the access matrix.



Slicing the Matrix



Access Control List

Access Control Lists

- *Approach*: represent the access matrix by storing its columns with the objects.
- Tag each object with an *access control list* (ACL) of authorized subjects/principals.
- Example: AFS access control
 - To authorize an access requested by S for O:
 - search O's ACL for an entry matching S
 - compare requested access with permitted access
- access checks are often made only at bind time
- The ACL may also control which subjects may modify the access matrix by updating the ACL itself.

Capabilities List

- *Approach*: represent the access matrix by storing its rows with the subjects.
- Tag each subject with a list of *capabilities* for the objects it is permitted to access.
- *capabilities* = unforgeable object reference, like a pointer.
 - e.g., Mach port rights are equivalent to capabilities
- Endows holder with permission to operate on the object.
 - e.g., permission to invoke specific methods
- Typically, capabilities may be passed between subjects.
- *confinement problem*: "The friend of my friend is my friend."

Security vs. Extensibility

- *Problem*: how to endow software modules with appropriate privilege?
- What mechanism exists to bind principals to subjects?
 - e.g., setuid syscall, setuid bit
- How do subjects change identity to execute a more privileged module?
 - protection domain, protection domain switch
- What principals should a software module bind to?
 - privilege of creator: not sufficient to do the service
 - privilege of user or system: dangerous

Unix security model

- Have three operations read, write and execute.
- Each file has an owner and a group.
- Protections are given for each operation on basis of everybody, group and owner.
- Like everything else in Unix, is a fairly simple and primitive protection strategy.
- Unix file listing:

```
      drwxr-xr--
      2 dan
      faculty
      2048 May 15 21:03 ./

      drwxr-xr-x
      7 dan
      faculty
      512 May 3 17:46 ../

      -rw-r---
      1 dan
      faculty
      213 Apr 19 22:27 a0.aux

      -rw-r----
      1 dan
      faculty
      3488 Apr 19 22:27 a0.dvi

      -rw-r----
      1 dan
      faculty
      1218 Apr 19 22:27 a0.log

      -rw-r--r--
      1 dan
      faculty
      36617 Apr 19 22:27 a0.ps

      -rwxr-xr-x
      1 dan
      faculty
      2599 Apr 5 18:07 a0.tex*
```

Most modern Unix versions also implement ACLs.