

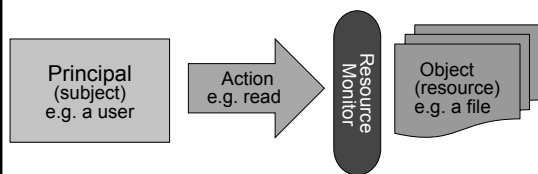
Access Control

Tom Chothia
Computer Security, Lecture 5

Today's Lecture

- Access control models
 - Access Control Matrix
 - Access Control Lists
 - Capability Lists
 - Role Based Access Control
- Linux/Unix access control
- Confused Deputy Problem

Model of Access Control



Access Control Matrix

	Operating System	Accounts Program	Accounting Data	Audit Trial
Alice (manager)				
Bob (auditor)				
Accounts Program				
Sam (sys admin)				

Permissions: x: execute, r: read, w: write

Access Control Matrix

- ACM is a matrix of all principals and objects.
- The matrix entries describe the permissions.
- Problem: maintaining such a matrix can be difficult.
- If the matrix is corrupted then all controls is lost.

Access Control Lists (ACLs)

- We don't want to store one massive matrix.
- Instead we can store each column of the matrix with the object it refers to. e.g.

(Accounts data,
[(Sam,r), (Bob,r),(Accounts program, rw)])

Capability Lists

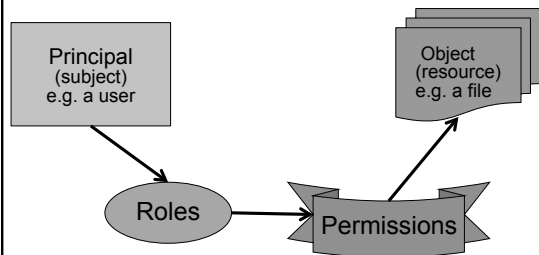
- ACLs work well for operating systems.
- But they are not so good for systems with huge numbers of users, e.g. Amazon.
- **Capability Lists** store the rights with the principal, e.g.

(Alice, [(Operating System, x),
(Accounts program, x)])

Capability Lists

- Capability Lists work well in web systems.
- The Capability List can be implemented as a:
 - Cookie
 - e.g. a server encrypts principals rights and stores this in the browser.
 - A Certificate
 - This lets a principal prove their rights to a third party.

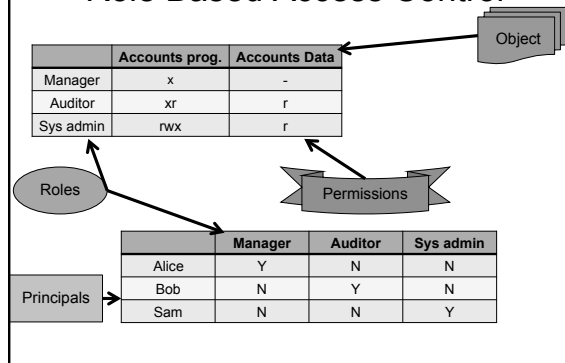
Role Based Access Control



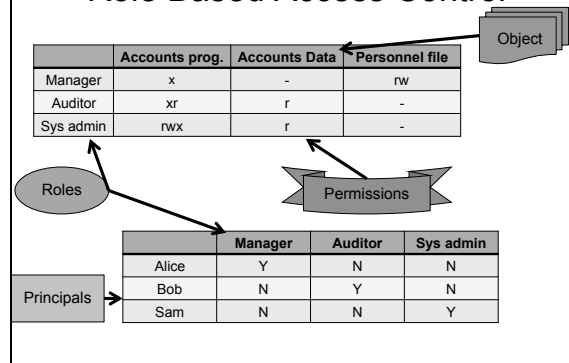
Access Control Matrix

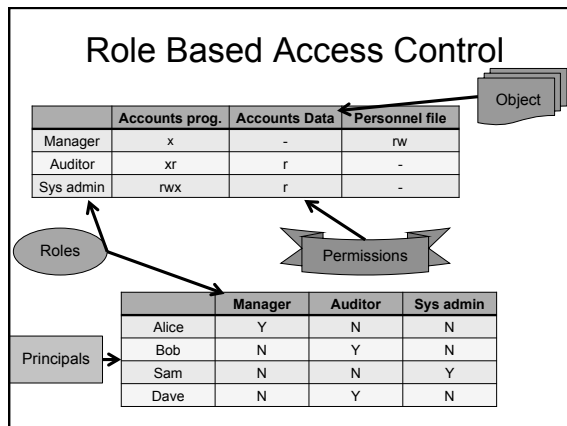
	Accounts Program	Accounting Data
Alice (manager)	x	-
Bob (auditor)	r	r
Sam (sys admin)	rw	r

Role Based Access Control



Role Based Access Control





Role Based Access Control

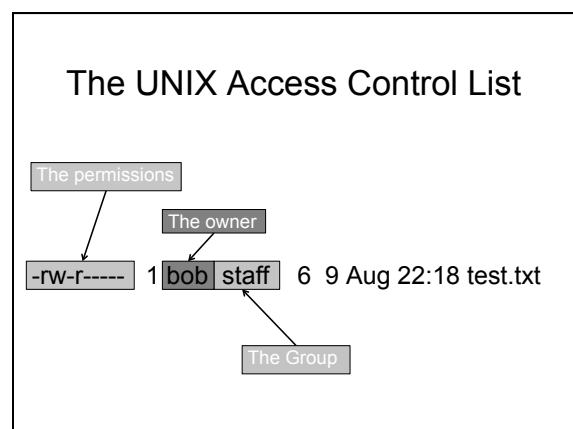
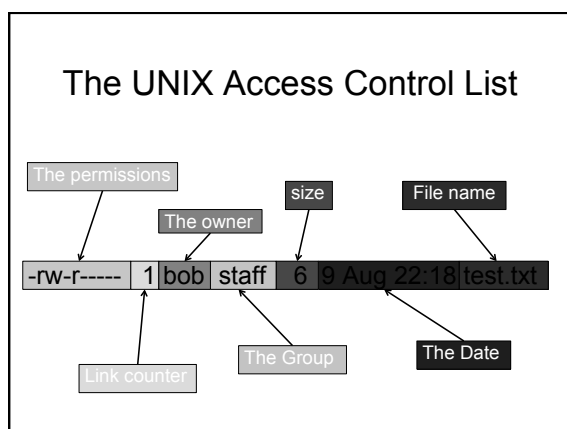
- Role Based Access Control makes it very easy to maintain large access control policies.
 - Good at expressing complex policies
 - Bad at expressing single user policies
- Used in Microsoft Active Directory, Microsoft SQL Server, PostgreSQL, SELinux, FreeBSD, Oracle DBMS, ...

Summary of Access Control Models

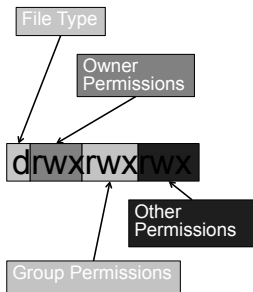
- Access Control Matrix:
 - Simple, but hard to use in practice
- Access Control Lists
 - Store access rights with the object
 - Good for OS access control
- Capability Lists
 - Store access rights with the principal
 - Good for web servers, cookies, certs.
- Role Based Access Control
 - Easy to administer

Access Control in Unix/Linux

- Unix/Linux/Mac use ACL, with groups.
- “uid” set when you log on.
- Linux Kernel then dynamically enforces the ACLS.
- `ls -l` displays files with their ACL
- `root` owns everything (“get root” = control the system)



UNIX File Permissions



Permissions:
r: read permission
w: write permission
x: execution permission
-: no permissions

File Type:
- : file
d : directory
b/c: device file

Access Control for Directories

For directories

- "r" is read only for directory contents
- "x" is permission to traverse, e.g. switch to, run.

No "x": I can't run any commands inside the directory

No "r": I can't list the files in the directory

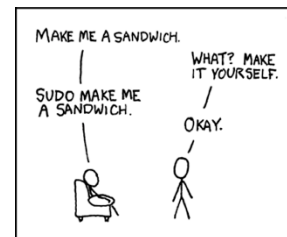
Access Control for Process

```
-r-sr-xr-x 1 root wheel 70352 19 Jun 2009 passwd
```

The "x" permission controls who can run a process

- in the case of `passwd`: anyone.

The "s" permission indicates that the process runs with the permission of its owner.



The Confused Deputy Problem

Users can run programs with more privileges

If there was a mistake in the `passwd` program we could use it do root only actions.

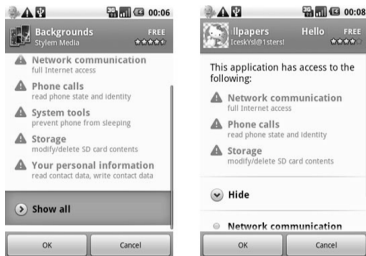
The Confused Deputy Problem, when a low level attacker gets a high level process to misusing its authority.

Make sure process have as low a level as possible.

Common Problems With Access Control

- Little protection if the attacker has physical access
- Poorly configured policies can be a problem
- Confused deputy problem:
 - low level users can get programs with high level access to do their dirty work.
- No defence against stack based attacks

Access Control in Android



Access Control in Android

- Android is a version of Linux
- Every app has it's own uid
- There is a group for access to each research.
- If you click "yes" when downloading, the uid of the app is added to the groups

Further Study

- Security Engineering, Ross Anderson
– Access Control Chapter
<http://www.cl.cam.ac.uk/~rja14/Papers/SE-04.pdf>
- Computer Security, Dieter Gollmann
– Chapters 4, 6 & 7
- Experiment with your own computer.

Next Time

- Hash functions.
- Getting around access control.
- Password protection
– what to do with that password file.