

Announcements

- Project #6
- Reading Chapter 15 (Networks)

Project #6 Notes

- Uid

- First process has uid of 0
- Spawned processes
 - Inherit uid of parent
 - Unless setuid bit is set on program to run, then the uid of the owner of that file is used

- ACLs

- First ACL entry is owner
- Others are for other users
 - Can delete these entries with `setACL(file, uid, 0)`
- Uid 0 can open any file regardless of ACLs

Access Matrix

- **Abstraction of protection for objects in a system.**
 - Rows are domains (users or groups of users)
 - Columns are objects (files, printers, etc.)
 - Items are methods permitted by a domain on an objects
 - read, write, execute, print, delete, ...
- **Representing the Table**
 - simple representation (dense matrix) is large
 - sparse representation possible: each non-zero in the matrix
 - observation: same column used frequently
 - represent groups of users with a name and just store that
 - create a default policy for some objects without a value
- **Revocation of access**
 - when are access rights checked?
 - selective revocation vs. global

Access Matrix

	F1	F2	F3	Laser Printer	
D1	read		execute		
D2			execute	print	
D3	read, write		execute		
D4			execute		
D5		delete			

- Rows represent users or groups of users
- Columns represent files, printers, etc.

Capabilities

- Un-forgeable Key to access something
- Implementation: a string
 - I.e. a long numeric sequence for a copier
- Implementation: A protected memory region
 - tag memory (or procedures) with access rights
 - example - x86 call gate abstraction
 - permit rights amplification

Monitoring

- Record (log) significant events
 - attempts to login to the system
 - changes to selected files or directories
- Possible to compromise the log
 - the user or software breaking in could delete all or part of the logs
 - could record logs to non-erasable storage
 - have a line printer attached to the machine
 - use DVD-ROM drives
 - send data to a secure remote host

Tripwire

- Compute a set of expectations about system
 - Hash of file contents
 - Dates on files
- Store database of values
 - On read-only media
 - Offline
- Periodically
 - Compare database to current system
 - Report any differences

Encryption: protecting info from being read

- Given a message m
 - use a key k , and function E_k to compute $E_k(m)$
 - store or send only $E_k(m)$
 - use a second key k' and function $D_{k'}$ such that
 - $D_{k'}(E_k(m)) = m$
 - E_k and $D_{k'}$ need not be kept a secret
- If $k=k'$ it's called private key encryption
 - need to keep k secret
 - example AES-256
- if $k \neq k'$, it's called public key encryption
 - need only keep one of them secret
 - if k' is secret, anyone can send a private message
 - if k is secret, it is possible to “sign” a message
 - still need a way to authenticate k or k' for a user
 - example RSA

Public Key Encryption

- Split into public and private keys
 - public key used to encrypt messages
 - publish this key widely
 - private key used to decrypt messages
 - keep this key a secret
- RSA
 - algorithm for computing public/private key pairs
 - based on problems involved in factoring large primes
 - for an n bit message P , $C = (P^e \bmod n)$, and $P = (C^d \bmod n)$
- Other Public Key Algorithms
 - knapsack
 - given a large collection of objects with different weights
 - public key is the total weight of a subset of the objects
 - private key is the list of objects

One Time Pad

- Key Idea: randomness in key
- Create a random string as long as the message
 - each party has the pad
 - xor each bit of the message with the a bit of the key
- Almost impossible to break
- Some practical problems
 - need to ensure key is not captured
 - a one bit drop will corrupt the rest of the message

Secure Socket Layer

- Goal:
 - Provide secure access to remote services
 - Authenticate remote servers to local users
 - Allow remote systems to authenticate users
 - Permit encrypted communication
- Approach
 - Public Key Cryptography
 - Certificates (signed by certificate authorities)
 - Server sends:
 - Certificate (signed with CA's private key)
 - Certificate contains server's public key
 - Client responds by encrypting reply using server's public key
 - Server checks response with private key