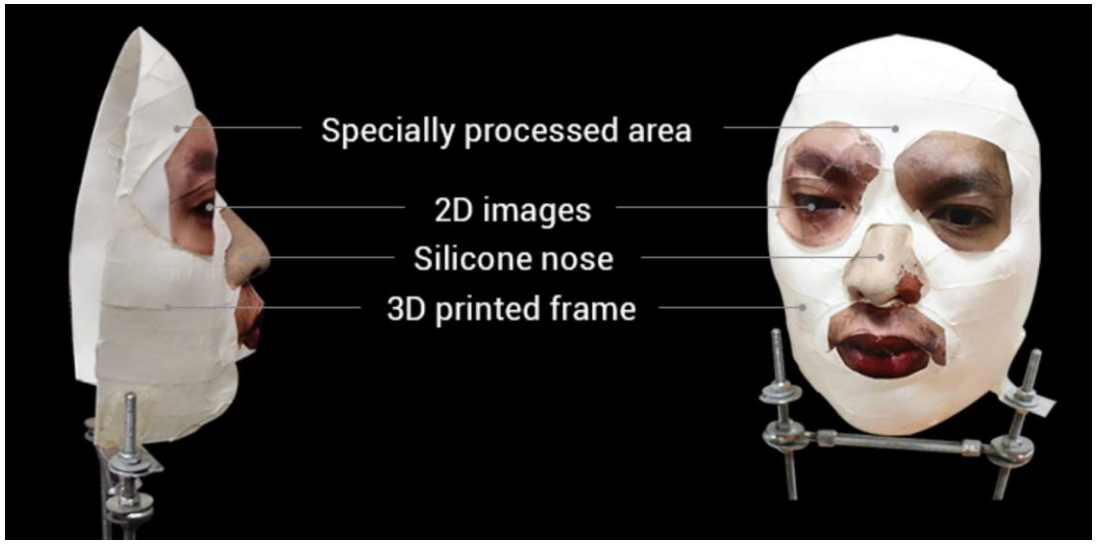
Attack something you are

2D photo



3D printed mask

Special paper

Something you have - token

- Active token
  - User action required
- Passive token
  - No user action required
- Static token
  - Fixed value generated
- Dynamic token
  - Has computing power to change internal state and generate different values

RSA SecurID

Time-Based Token Authentication

Login: mcollings
Passcode: 2468159759

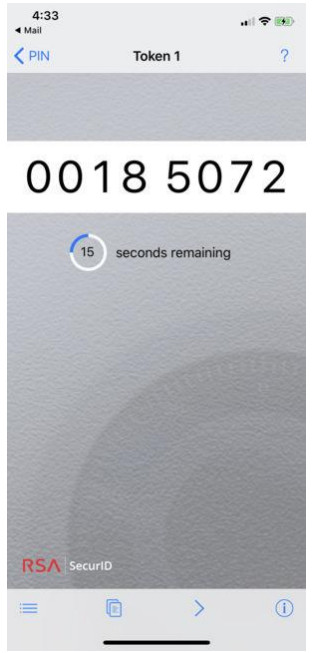
PASSCODE = PIN + TOKENCODE

Token code:
Changes every
60 seconds

Clock
synchronized to
UCT

Unique seed

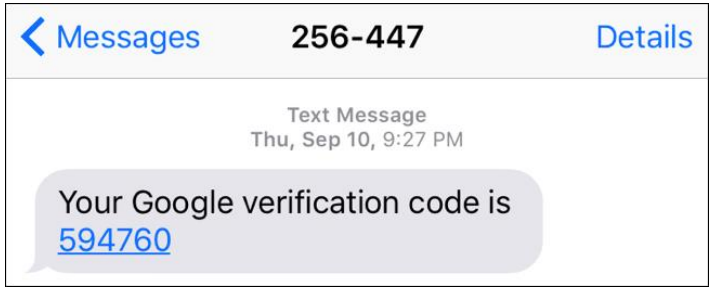
Hard token



Soft token

How to make authentication more secure?

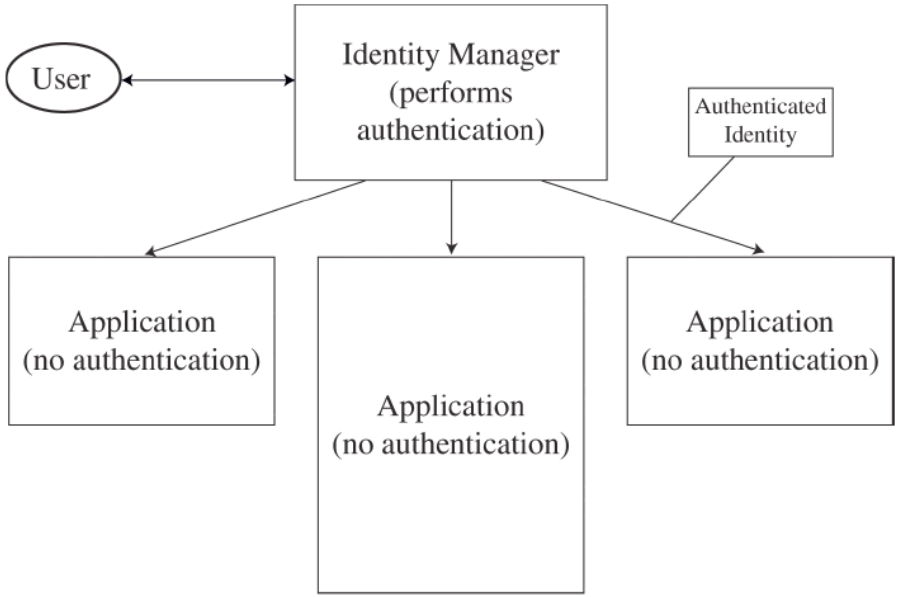
- Multi-factor authentication (MFA)
  - E.g., Password + SMS authentication code
- But how many factors are optimal?
  - Large number of factors lead to user frustration
- Remember overall security depends on weakest link…



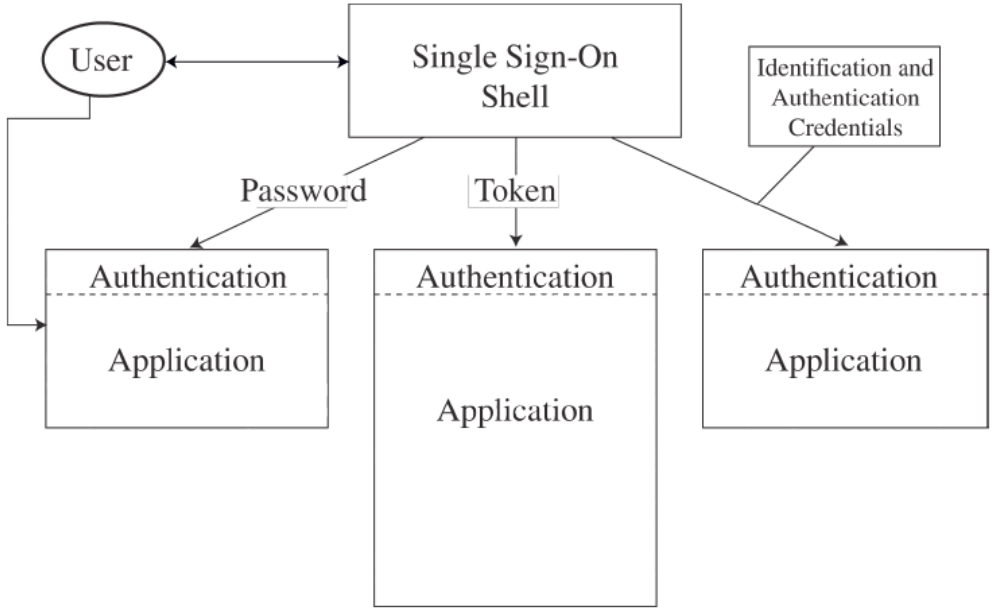
Reddit discloses hack, says SMS intercept allowed attackers to skirt 2FA protections

Federated Identity Management (FIM)

- Login & logout multiple systems are time-consuming
- FIM unifies the identification & authentication process for a group of systems
- Single sign-on (SSO) takes over sign-on & authentication for a user
- FIM replaces authentication module of individual systems, SSO doesn't



FIM



SSO

Example

Pros/cons?

Summary

- Authentication
  - The act of proving that a user is who she says she is
- Methods:
  - Something the user *knows (password, security questions)*
  - Something the user *is (biometrics)*
  - Something user *has (token)*
- Multi-factor authentication
- Federated Identity Management

Access Control

EECS 195

Spring 2019

Zhou Li

Why need access control?

Leo, doctor

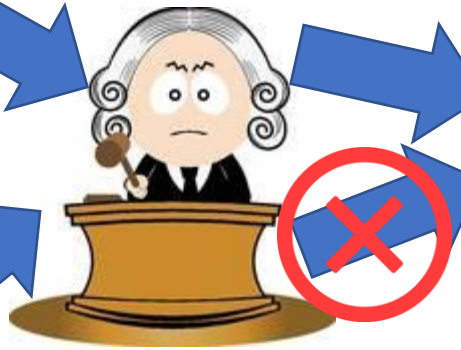
I have some special records. Protect them!

Sure!

Jim, programmer

Get me special records

Hospital Information System

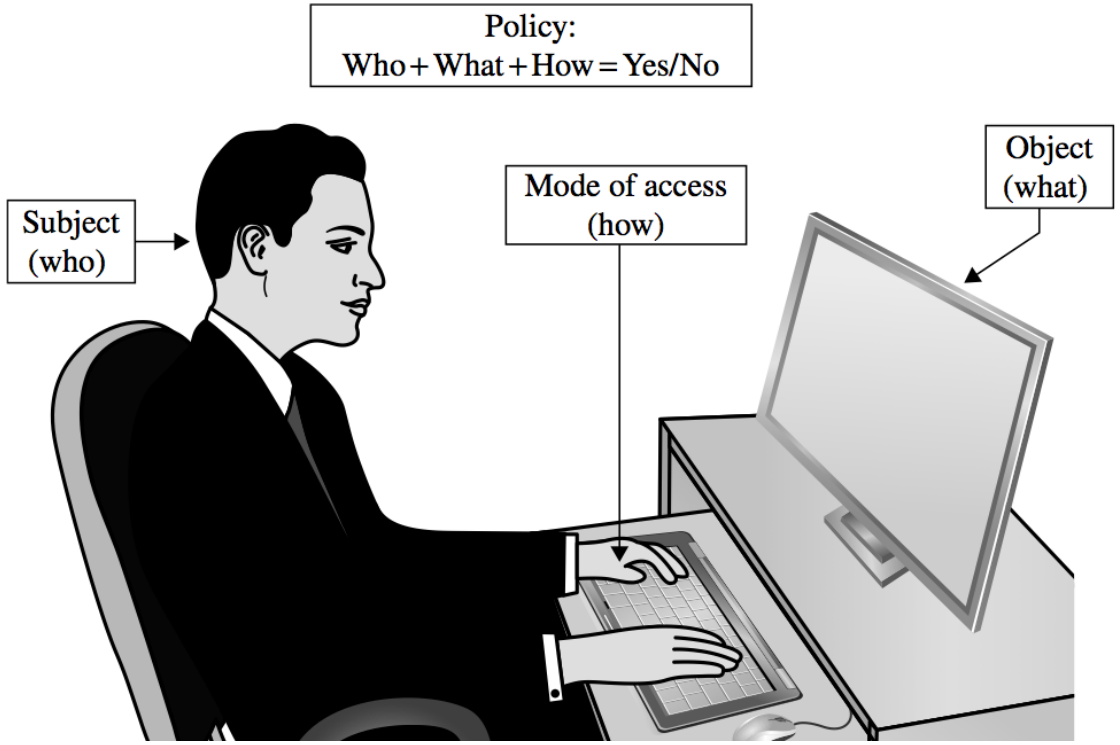


Get me same thing

Shared workstation

Access control

- Articulated by Scott Graham and Peter Denning [GRA72]
- Limiting who can access what in what ways



Effective policy implementation

- Check every access
  - No indefinite access
- Enforce least privilege
  - Access to the fewest resources necessary to complete a task
- Verify acceptable usage
  - Ensure the activity to be performed on an object is appropriate
  - E.g., stack only allows push, pop, clear

Tracking of access control

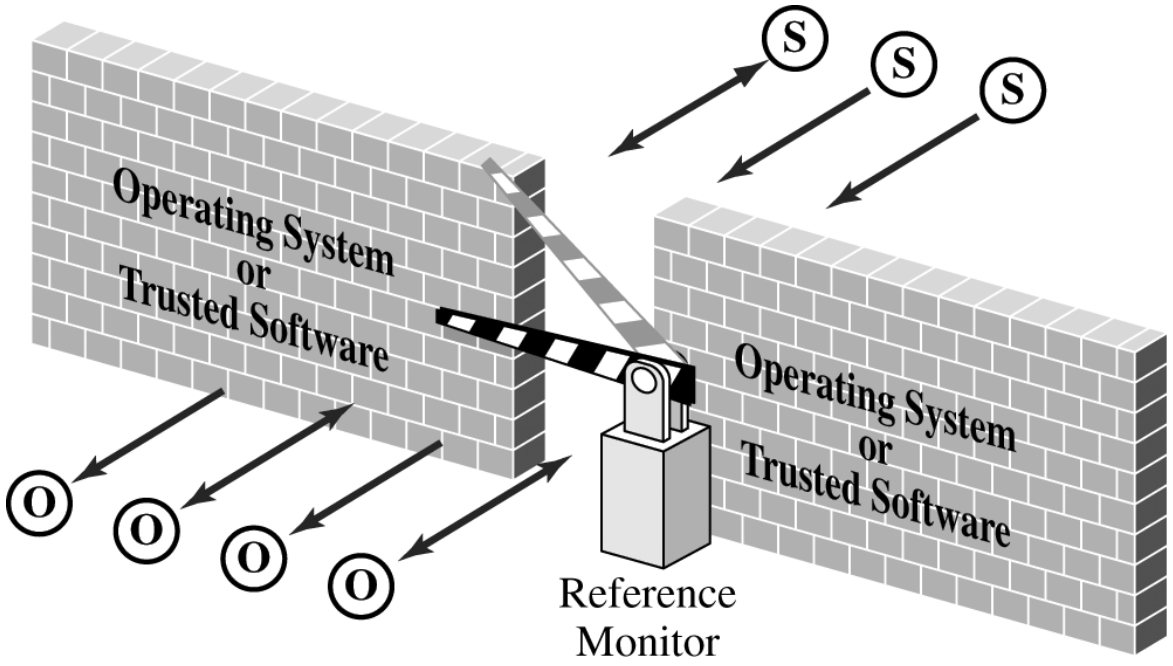
- Policies need to be revisited by admin frequently
  - Revoke authorization when account compromised/impersonated
  - Someone acquired a large number of no-longer-needed rights?
  - User has access to objects no longer needed to be controlled?
- Choice of granularity
  - Object granularity: bit, byte, word, file, computer…
  - File is the most common granularity
- Audit log
  - Recording what accesses have been permitted
  - Used for resource planning, causal analysis, …

Access control components

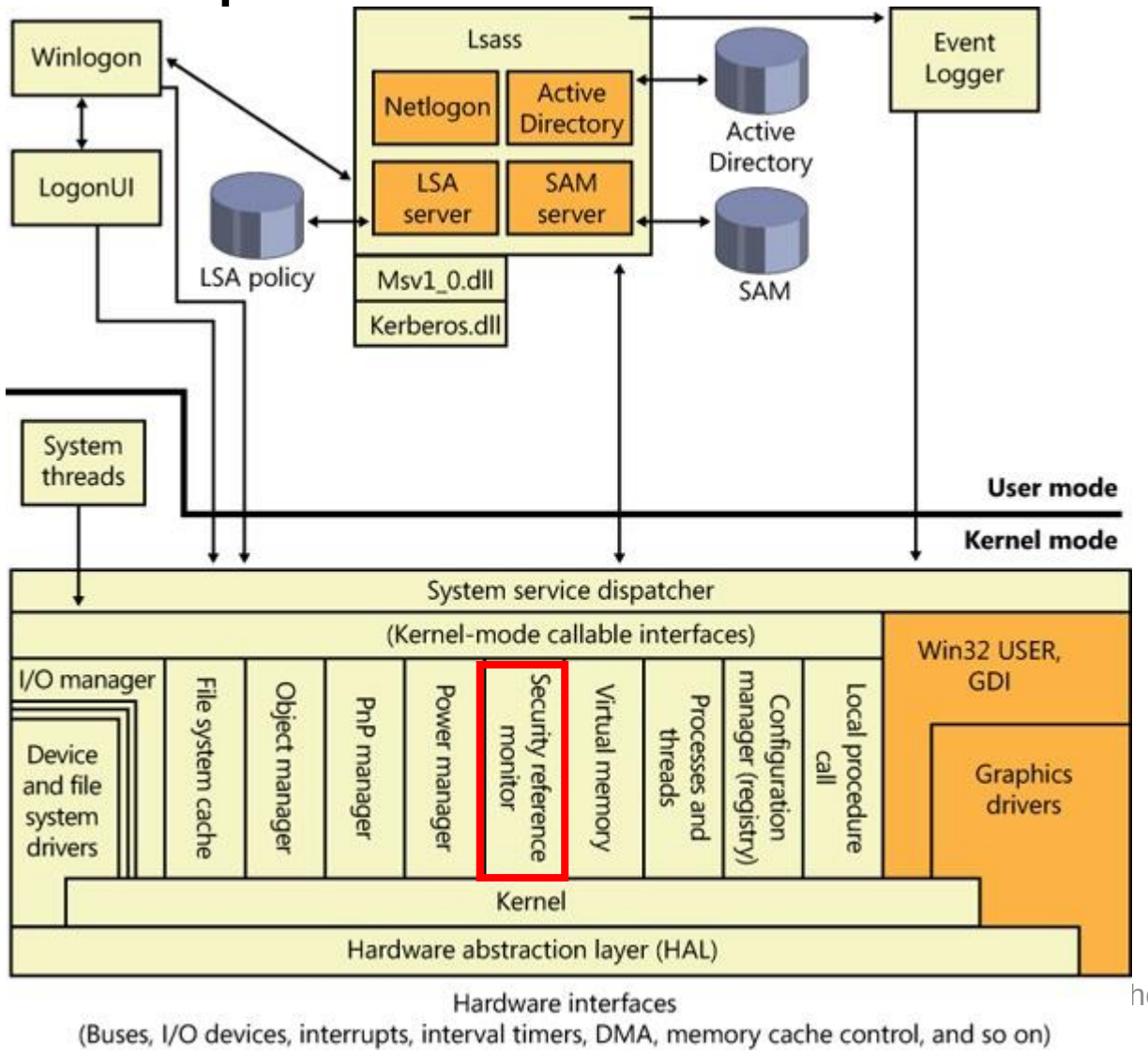
- Executor
  - Reference monitor
- Policy Storage
  - Access control directory
  - Access control matrix
  - Access control list
- Optimizations
  - Capability
  - ~~Procedure-oriented access control~~
  - Role-based access control

Reference Monitor

- James Anderson and his study committee [AND72] gave name and structure to the digital version of a concept.
- Key techniques: *isolation* & *managed access*
- Reference monitor: access control that is *always invoked, tamperproof, and verifiable*.



Example: Windows Kernel-Mode Security Reference Monitor

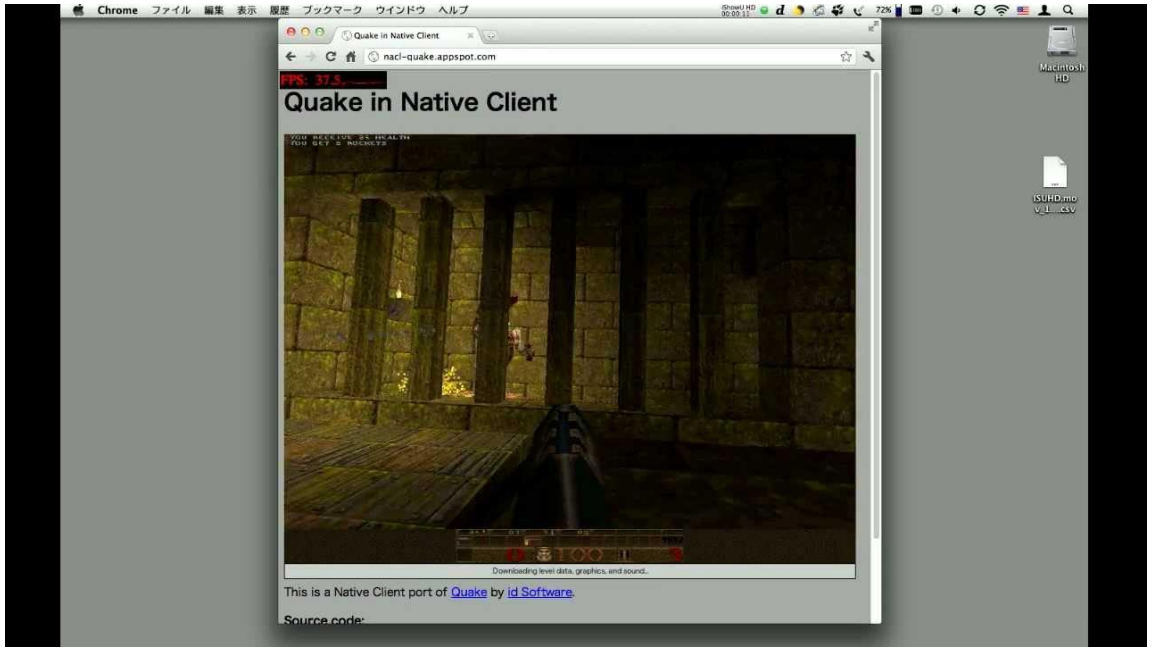


- **Security reference monitor (SRM)** (%SystemRoot%\System32\Ntoskrnl.exe):
  - Define the access token data structure to represent a security context
  - Perform security access checks on objects
  - Manipulate privileges (user rights)
  - Generate security audit messages

https://www.microsoftpressstore.com/articles/article.aspx?p=2228450&seqNum=2

Inline Reference Monitor (IRM)

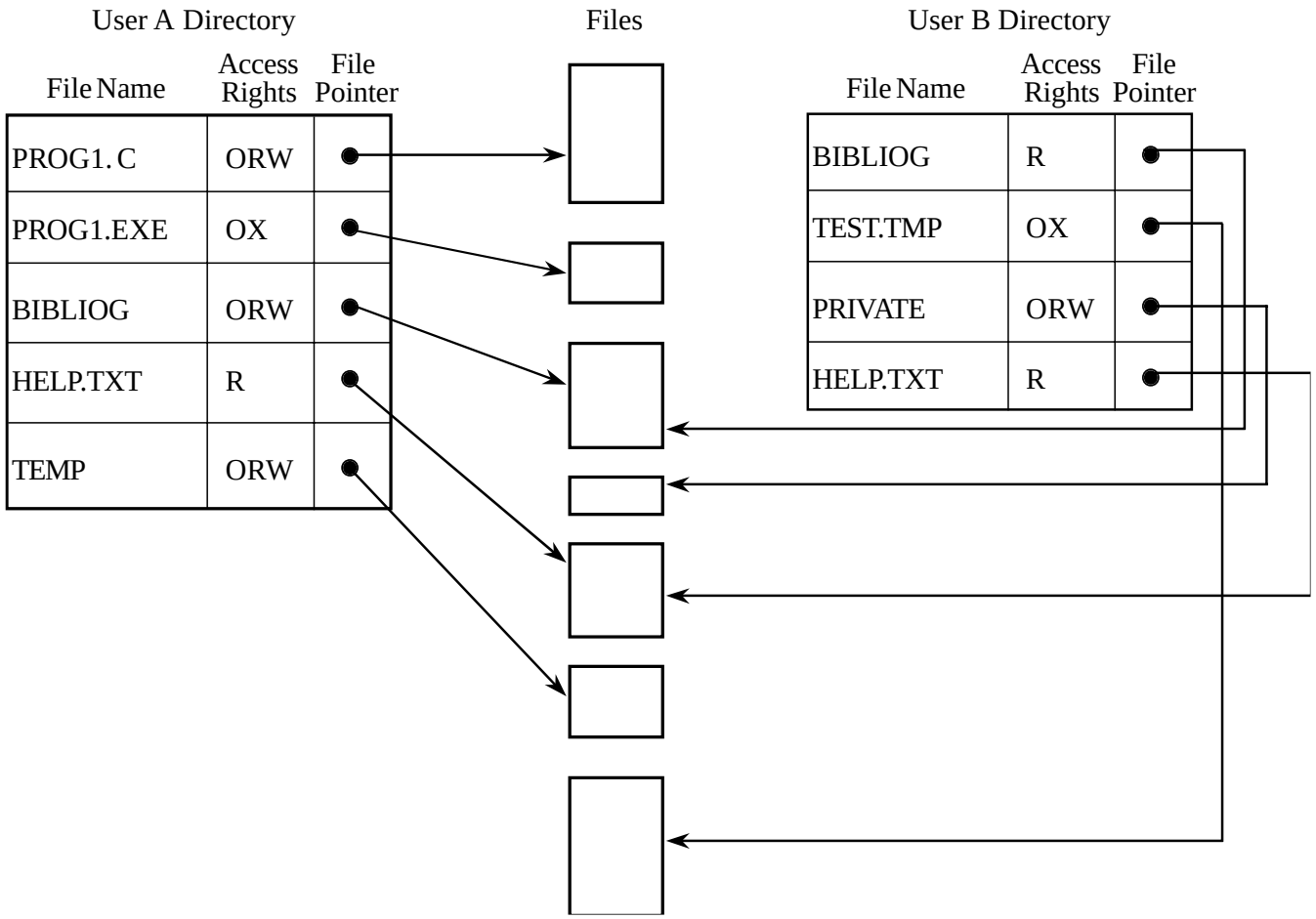
- RM above OS/Hardware
- E.g., Native Client (NaCl) Sandbox
- Goal: download an x86 binary and run it “safely”
  - Much better performance than JavaScript, Java, etc.
- Code is restricted to a subset of x86 assembly
  - Enables reliable disassembly and efficient validation



Play Quake in Google Chrome 14beta NaCl

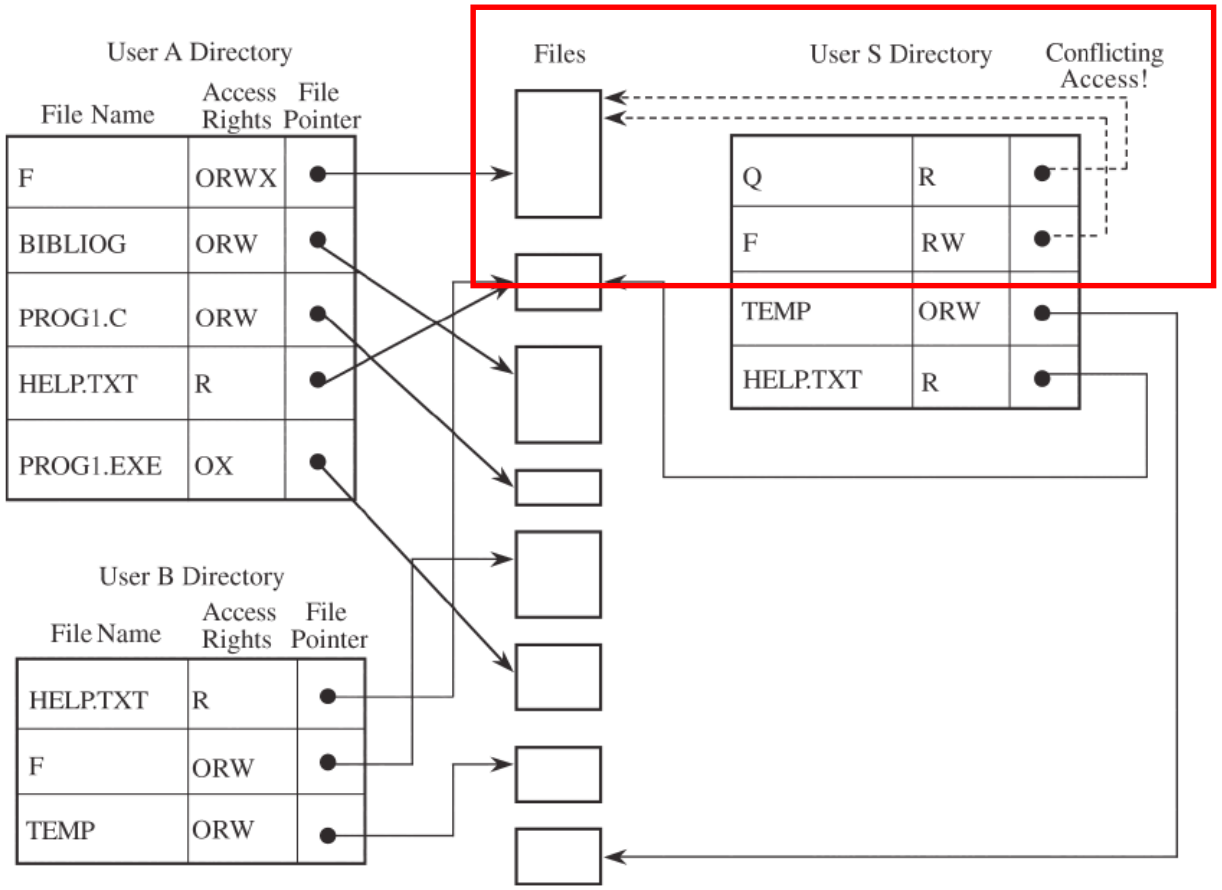
Storage: Access Control Directory

R: read
W: write
X: execute
O: own



- Like a file directory, writable by OS
- Every file has a unique owner
  - Can grant/revoke access of other users
- Each user has file directory
  - Each file has ORWX rights
- A file can be linked by multiple directories with different rights

Access Control Directory (Cond.)



Ambiguous access rights

- Pros:
  - Easy to implement
- Cons:
  - Directory per user becomes very large if there are many shared objects
  - Revocation of access is time-consuming (when A grants read access of file F to many users)
  - Inconsistent rights per object
    - Pseudonyms are allowed for one object

Storage: Access Control Matrix

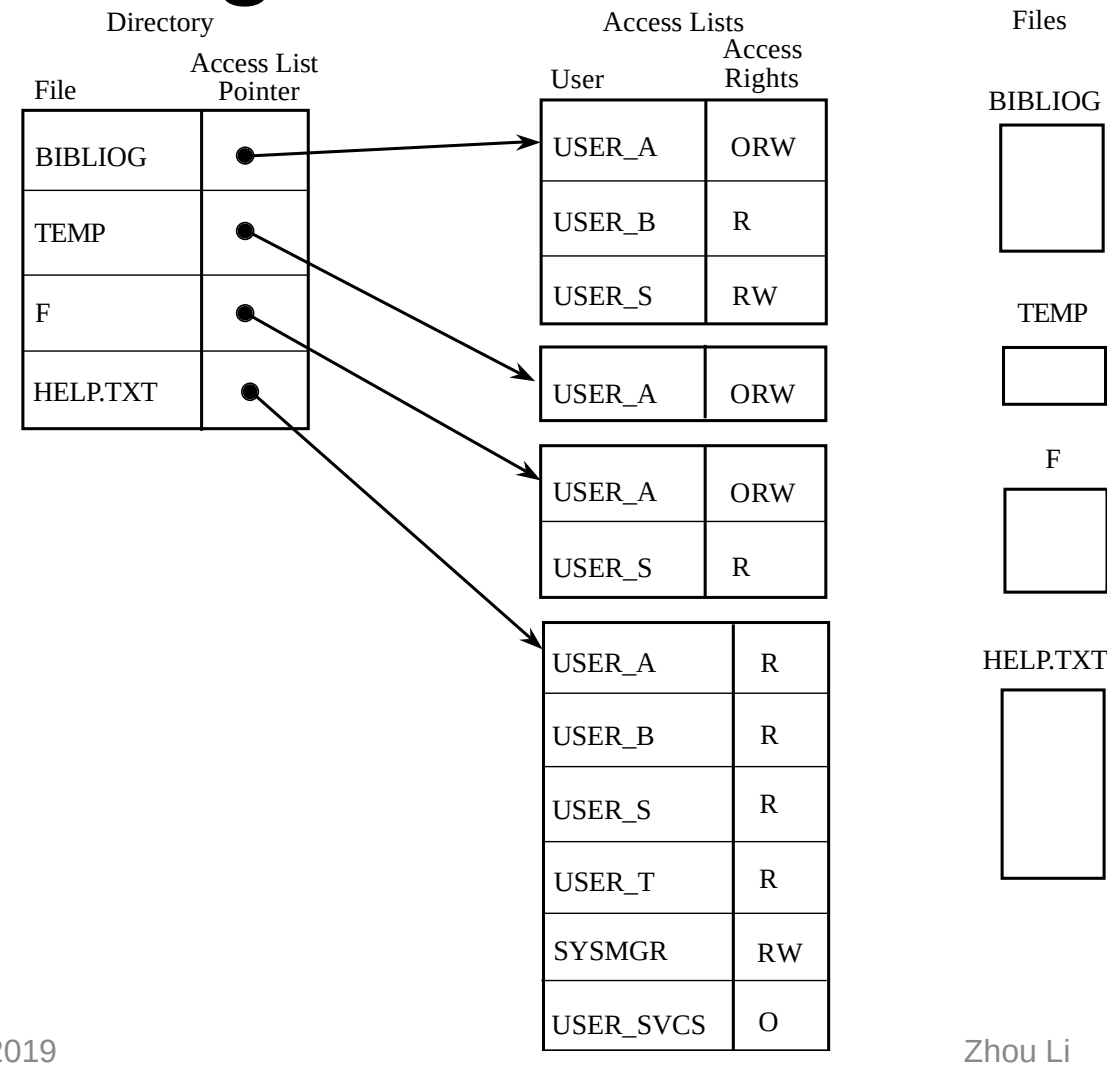
Object

Subject

	BIBLIOG	TEMP	F	HELP.TXT	C_COMP	LINKER	SYS_CLOCK	PRINTER
USER A	ORW	ORW	ORW	R	X	X	R	W
USER B	R	-	-	R	X	X	R	W
USER S	RW	-	R	R	X	X	R	W
USER T	-	-	-	R	X	X	R	W
SYS_MGR	-	-	-	RW	OX	OX	ORW	O
USER_SVCS	-	-	-	O	X	X	R	W

Privilege List

Storage: Access Control List

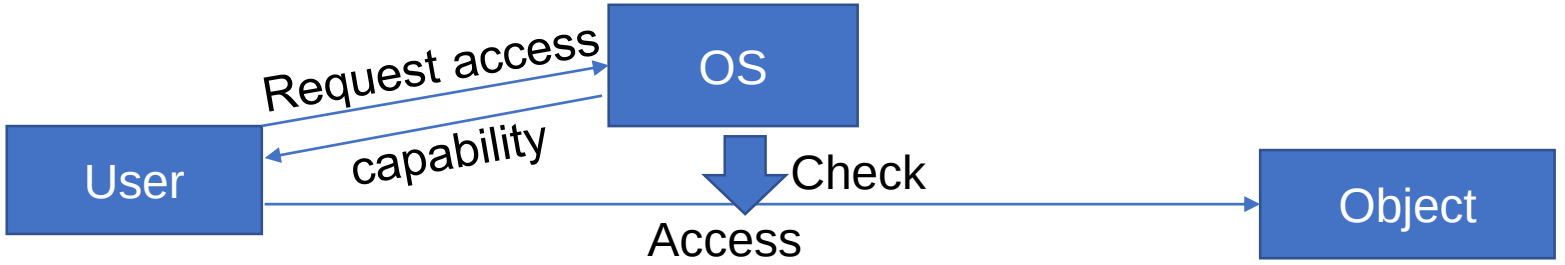


- Each object has a list
- An object can have a list of default rights (e.g., read) permissible to any subject

- Used by modern OS, like Windows (Security Reference Monitor) & Linux (permission bits)

Capability

- Can user define access control policy dynamically?
  - Yes, capability!
- Definition: unforgeable token (or ticket) that gives the user (or owned process) certain rights to an object
  - User must present token before accessing objects
  - Single- or multi-use
  - Unforgeability enforced by OS or encryption



Role-based access control (RBAC)

- Assigns permissions to specific *operations* with meaning in the organization, rather than to low level data objects
- Role: a collection of permissions
  - Group: a collection of users