CS 5430

Principles

Prof. Clarkson
Spring 2017

Review

Attacks

are perpetrated by

threats

that inflict

harm

by exploiting

vulnerabilities

which are controlled by

countermeasures.

Approaches to security

- Prevention: build systems that are completely free of vulnerabilities

- Risk management: invest wisely in countermeasures

- Deterrence through accountability: attribute attacks to humans and legally prosecute

Principles of Prevention

[Saltzer and Schroeder, *The Protection of Information in Computer Systems*, 1975]

- Accountability
- Complete Mediation
- Least Privilege
- Failsafe Defaults
- Separation of Privilege
- Defense in Depth
- Economy of Mechanism
- Open Design
- Psychological Acceptability

EXERCISE: BINGO

Accountability

Hold principals responsible for their actions

Accountability

Hold principals responsible for their actions

- **Authorization:** mechanisms that govern whether actions are permitted

- **Authentication:** mechanisms that bind principals to actions

- **Audit:** mechanisms that record and review actions

Accountability

Hold principals responsible for their actions

- **Au**thorization: mechanisms that govern whether actions are permitted

- **Au**thentication: mechanisms that bind principals to actions

- **Au**dit: mechanisms that record and review actions

... Gold Standard [Lampson 2000]

Butler Lampson

(b. 1943)

Turing Award Winner 1992

*For contributions to the development of distributed, personal computing environments and the technology for their implementation: workstations, networks, operating systems, programming systems, displays, **security** and document publishing.*

Complete Mediation

Every operation requested by a principal must be intercepted and determined to be acceptable according to the security policy

Complete Mediation

Every operation requested by a principal must be intercepted and determined to be acceptable according to the security policy

- Component that does the interception and determination is the reference monitor

- Related to Accountability

- Restricts caching of information, including previous decisions

Least Privilege

Principals should be given the minimum privileges necessary to accomplish their task

- Limits the damage that can result from accident or malice
- Cf. "need to know"

Failsafe Defaults

Base decisions on the presence of privilege, not the absence of prohibition

- The default answer is "no"
- Say "yes" only when there is an explicit reason to do so
- Principals who discover they don't have access will complain
- Attackers who discover they do have access won't complain!

Failsafe Defaults

Java stack inspection circa 1998:

```
checkPermission(T) {
  // loop newest to oldest stack frame
  foreach stackFrame {
    if (local policy forbids access to T by class executing in
        stack frame) throw ForbiddenException;

    if (stackFrame has enabled privilege for T)
      return;  // allow access

    if (stackFrame has disabled privilege for T)
      throw ForbiddenException;
  }

  // end of stack
  if (Netscape || …) throw ForbiddenException;
  if (MS IE4.0 || JDK 1.2 || …) return;
}
```

Separation of Privilege

- Different operations should require different privileges

- Supports Least Privilege
- In tension with usability: too many operations and objects and principals

Separation of Privilege

- Different operations should require different privileges

- Disseminate privileges for an operation amongst multiple principals (Separation of Duty)

[Wargames 1983]
[Inside Out 2015]

Defense in Depth

Prefer a set of complementary mechanisms over a single mechanism

Complementary:

- **Independent:** attack that compromises one mechanism is unlikely to compromise others

- **Overlapping:** attacks must compromise multiple mechanisms to succeed

Economy of Mechanism

Prefer mechanisms that are simpler and smaller

- Easier to understand, construct, analyze
- Hence less likely to have unknown vulnerabilities
- Applies to any aspect of system, not just security

Trusted computing base (TCB): mechanisms that implement the core security functionality

...keep the TCB small

Open Design

Security shouldn't depend upon the secrecy of design or implementation


```
/*      efdtt.c    Author:  Charles M. Hannum <root@ihack.net>            */
#define m(i)(x[i]^s[i+84])<<
unsigned char x[5],y,s[2048];main(n){for(read(0,x,5);read(0,s,n=2048);write(1,s
,n))if(s[y=s[13]%8+20]/16%4==1){int i=m(1)17^256+m(0)8,k=m(2)0,j=m(4)17^m(3)9^k
*2-k%8^8,a=0,c=26;for(s[y]-=16;--c;j*=2)a=a*2^i&1,i=i/2^j&1<<24;for(j=127;++j<n
;c=c>y)c+=y=i^i/8^i>>4^i>>12,i=i>>8^y<<17,a^=a>>14,y=a^a*8^a<<6,a=a>>8^y<<9,k=s
[j],k="7Wo~'G_\216"[k&7]+2^"cr3sfw6v;*k+>/n."[k>>4]*2^k*257/8,s[j]=k^(k&k*2&34)
*6^c+~y;}}
```

Open Design

Security shouldn't depend upon the secrecy of design or implementation

Arguments for open design:

- Secrets eventually come out: reverse engineering is possible, employees move around

- Making details public increases chance of identifying and repairing vulnerabilities

Open Design

Security shouldn't depend upon the secrecy of design or implementation

Arguments against open design:

- Secrecy supports Defense in Depth by making it harder to find vulnerabilities

- Lack of hard evidence that Linus' Law really holds ("given enough all eyeballs, all bugs are shallow")

- After identification, some vulnerabilities cannot quickly or easily be repaired

Psychological Acceptability

Minimize the burden of security mechanisms on humans

- Don't make operations (much) more difficult to complete than if security mechanisms were absent
- Don't make configuration difficult
- Produce comprehensible error messages

...always a tradeoff between security and usability

Principles of Security

- Accountability
- Complete Mediation
- Least Privilege
- Failsafe Defaults
- Separation of Privilege
- Defense in Depth
- Economy of Mechanism
- Open Design
- Psychological Acceptability

EXERCISE: BINGO

ASSIGNMENTS

A1

- Out today
- Due in 2 weeks
  - The *deadline* is the time by which you must upload to CMS and confirm you are happy with the file it records
  - But can be submitted after that for a penalty
  - See [late policy in syllabus](#)
- Each assignment weighted equally in final grade, lowest assignment dropped
- A1 is individual work, not partners nor teams

Academic Integrity

- You are bound by Academic Integrity policies linked from course syllabus
  - AI violation + grade penalty: dishonest about source of ideas
  - Only grade penalty: honest about source, which isn't you
- If you have a question about what is or is not allowed, please ask
- If you fear you have committed a violation, tell me before grading commences
- Given the subject matter of this course, I take ethics extremely seriously

Upcoming events

- [today] A1 out by end of day, consulting hours start

"Important principles may, and must, be inflexible."
– Abraham Lincoln