Introduction
CS 136
Computer Security
Peter Reiher
October 2, 2014

Purpose of Class

- To introduce students to computer security issues
- To familiarize students with secure software development
- To learn to handle security in today's installations and systems

Description of Class

- Topics to be covered
- Prerequisites
- Grading
- Reading materials
- Homework
- Office hours
- Web page

Topics to Be Covered

- Cryptography and authentication
  - Use, not design and analysis
- Access control and security models
- Secure software design and programming
- Secure protocols
- Network security – threats and countermeasures
- Operating systems security
- Security analysis and forensics
- Malware, common attacks, and important defenses
- Privacy
- Practical computer security defenses

Prerequisites

- CS111 (Operating Systems)
- CS118 (Computer Networks)
- Or equivalent classes elsewhere
- If you aren't familiar with this material, you'll be at a disadvantage
  - People have had serious problems with this unfamiliarity recently

Teaching Assistant

- Turker Garip
  - mtgarip@CS.UCLA.EDU
- Weekly recitation sections Fridays
  - Section 1A: 8-10, MS 5128
  - Section 1B: 10-12, MS 5128
  - Won't cover new material
  - May help with problems with lectures
- Will also handle all homework issues
- Office hours: TBA

Grading

- Midterm – 25%
- Exercises – 35%
- Final – 40%

Class Format

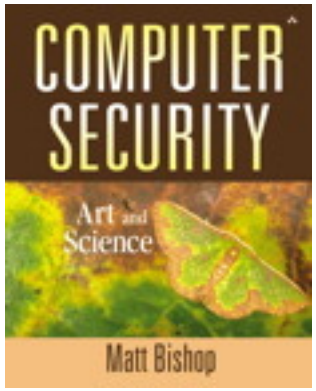
- A lecture class
- Questions and discussions always welcomed

Reading Materials

- Textbook
- Non-required supplemental text
- Optional papers and web pages

Textbook

- *Computer Security: Art and Science*
  - By Matt Bishop
- Available in UCLA bookstore
- Bishop has a shorter version
  - That's not the one we're using
- First reading assignment: Chapter 1



Supplemental Text

- *Secrets and Lies*
  - By Bruce Schneier
- Not a textbook at all
- A philosophy of computer security
- Great for appreciating the field and problems
- Not great for depth of technical details
- Not required
  - No readings will be assigned from this book
  - But if you plan to work in this field, read it

Papers and Web Pages

- Non-required reading material
- Might or might not be assigned each week
- Usually made available electronically
  - Through class web page
- Generally relevant news stories or discussion of security topics

Exercises

- Five assignments
- Requiring practical work
- Performed on the Deter testbed
  - Accessible via the web from any connected location
- Individual, not group, assignments

Exercise Topics

1. Access control and permissions
   - Week 3
2. Exploits
   - Week 4
3. Man in the middle attacks
   - Week 6
4. TCP SYN flood DDoS Attacks
   - Week 7
5. Botnet detection and defense
   - Week 8

More on Exercises

- Each exercise has an associated web page
  - With full instructions and pointers to necessary tools
- Due by midnight on Thursday of indicated week
- Class TA will provide advise and assistance on exercises

The Deter Testbed

- A set of machines devoted to security research and education
- Located at ISI and SRI
- Accessible remotely
- Special accounts set up for this class
- First discussion section will provide instructions on using Deter
  - With further assistance from TA
  - Enrollment key: CS136KEY

Tests

- Midterm – Thursday, November 6 in classroom
- Final – Friday, December 19, 8:00 – 11:00 AM in classroom
- Closed book/notes tests

Office Hours

- TTh 1-2
- Held in 3532F Boelter Hall
- Other times possible by appointment

Class Web Page

http://www.lasr.cs.ucla.edu/classes/136_fall14

- Slides for classes will be posted there
  - By 5 PM the previous afternoon
  - In Powerpoint and PDF
- Readings will be posted there
  - With links to web pages

Introduction to Computer Security

- Why do we need computer security?
- What are our goals and what threatens them?

Why Is Security Necessary?

- Because people aren't always nice
- Because a lot of money is handled by computers
- Because a lot of important information is handled by computers
- Because our society is increasingly dependent on correct operation of computers

History of the Security Problem

- In the beginning, there was no computer security problem
- Later, there was a problem, but nobody cared
- Now, there's a big problem and people care
  - Only a matter of time before a real disaster
  - At least one company went out of business due to a DDoS attack
  - Identity theft and phishing claim vast number of victims
  - Stuxnet seriously damaged Iran's nuclear capability
  - Video showed cyberattack causing an electric transformer to fail
  - There's an underground business in cyber thievery
  - Increased industry spending on cybersecurity

Some Examples of Large Scale Security Problems

- Malicious code attacks
- Distributed denial of service attacks
- Vulnerabilities in commonly used systems

Malicious Code Attacks

- Multiple new viruses, worms, botnets, and Trojan horses appear every week
- Recent estimate of $10 billion annual damages from botnets
- Stuxnet worm targeted at nuclear facilities
  - Unspecified amounts of damage done to Iran's nuclear program
- IM and smartphone attacks are popular

Distributed Denial of Service Attacks

- Use large number of compromised machines to attack one target
  - By exploiting vulnerabilities
  - Or just generating lots of traffic
- Very common today
- A favored tool for hacktivists
  - Recent large DDoS attacks on China and others
- In general form, an extremely hard problem

Vulnerabilities in Commonly Used Systems

- 802.11 WEP is fatally flawed
- Recently, critical vulnerabilities in Intel processor microcode, Linksys routers
- Many popular applications have vulnerabilities
  - Recent vulnerabilities in Android WebView, Android OS, Internet Explorer, HP backup software, Microsoft Office, Adobe Flash, Apache Tomcat, etc.
- Many security systems have vulnerabilities
  - Gnu TLS, Apple iOS SSL, and Symantec Endpoint Protection recently

Electronic Commerce Attacks

- As Willie Sutton said when asked why he robbed banks,
  - "Because that's where the money is"
- Increasingly, the money is on the Internet
- Criminals have followed
- Common problems:
  - Credit card number theft (often via phishing)
  - Identity theft (phishing, again, is a common method)
  - Loss of valuable data from laptop theft
  - Manipulation of e-commerce sites
  - Extortion via DDoS attacks or threatened release of confidential data
- 2010's Sony data breach estimated to cost the company $170 million

Some Recent Statistics

- Bit9 survey in 2013 reports 47% of surveyed organizations knew they'd suffered a cyberattack
  - But 52% doubted their ability to detect attacks
  - 13% didn't even know if they'd been attacked
- 2013 Verizon report said that 66% of breaches took months to years to discover
  - And 69% of breaches were not discovered by the compromised organization itself
- Ponemon Institute 2014 survey showed 94% of healthcare organizations lost data in past two years

Cyberwarfare

- Nation states have developed capabilities to use computer networks for such purposes
- DDoS attacks on Estonia and Georgia
  - Probably just hackers
- Some regard Stuxnet as real cyberwarfare
  - Pretty clear it was done by US
- Continuous cyberspying by many nations
- Vulnerabilities of critical infrastructure
  - The smart grid will only increase the danger

Something Else to Worry About

- Are some of the attempts to deal with cybersecurity damaging liberty?
- Does data mining for terrorists and criminals pose a threat to ordinary people?
  - The NSA is looking at a lot of stuff . . .
  - And they aren't the only ones
- Can I trust Facebook/Google/MySpace/Twitter/ whoever with my private information?
- Are we in danger of losing all privacy?

Why Aren't All Computer Systems Secure?

- Partly due to hard technical problems
- But also due to cost/benefit issues
- Security costs
- Security usually only pays off when there's trouble
- Many users perceive no personal threat to themselves
  - "I don't have anything valuable on my computer"
  - "I don't have any secrets and I don't care what the government/Google/my neighbor knows about me"
- Ignorance also plays a role
  - Increasing numbers of users are unsophisticated
  - Important that computer security professionals don't regard this ignorance as a character flaw
  - It's a fact of life we must deal with

Legacy and Retrofitting

- We are constrained by legacy issues
  - Core Internet design
  - Popular programming languages
  - Commercial operating systems
- All developed before security was a concern
  - With little or no attention to security
- Retrofitting security works poorly
  - Consider the history of patching

Problems With Patching

- Usually done under pressure
  - So generally quick and dirty
- Tends to deal with obvious and immediate problem
  - Not with underlying cause
- Hard (sometimes impossible) to get patch to everyone
- Since it's not organic security, patches sometimes introduce new security problems

Speed Is Increasingly Killing Us

- Attacks are developed more quickly
  - Often easier to adapt attack than defense
- Malware spreads faster
  - Slammer got 75,000 nodes in 30 minutes
- More attackers generating more attacks
  - US DoD computers targeted at least 43,000 times in first half of 2009
  - US military doctrine says cyber attack could be an act of war

Some Important Definitions

- Security
- Protection
- Vulnerabilities
- Exploits
- Trust

Security and Protection

- *Security* is a policy
  - E.g., "no unauthorized user may access this file"
- *Protection* is a mechanism
  - E.g., "the system checks user identity against access permissions"
- Protection mechanisms implement security policies

Vulnerabilities and Exploits

- A *vulnerability* is a weakness that can allow an attacker to cause problems
  - Not all vulnerabilities can cause all problems
  - Most vulnerabilities are never exploited
- An *exploit* is an actual incident of taking advantage of a vulnerability
  - Allowing attacker to do something bad on some particular machine
  - Term also refers to the code or methodology used to take advantage of a vulnerability

Trust

- An extremely important security concept
- You do certain things for those you trust
- You don’t do them for those you don’t
- Seems simple, but . . .

Problems With Trust

- How do you express trust?
- Why do you trust something?
- How can you be sure who you're dealing with?
- What if trust is situational?
- What if trust changes?

Trust Is Not a Theoretical Issue

- Most vulnerabilities that are actually exploited are based on trust problems
- Attackers exploit overly trusting elements of the computer
  - From the access control model to the actual human user
- Taking advantage of misplaced trust
- Such a ubiquitous problem that some aren't aware of its existence

Transitive Trust

I trust Alice

Alice trusts Bob

So do I trust Carol?

Should I?

Bob trusts David

David trusts Carol

Examples of Transitive Trust

- Trust systems in peer applications
- Chains of certificates
- But also less obvious things
  - Like a web server that calls a database
  - The database perhaps trusts the web server
  - But does the database necessarily trust the user who invoked the server?
  - Even if the web server trusts the user
- Programs that call programs that call programs are important cases of transitive trust

What Are Our Security Goals?

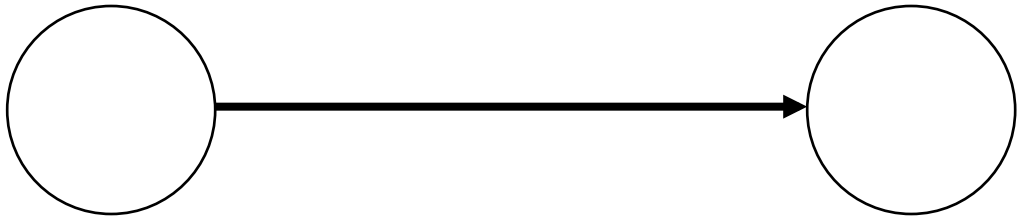
- CIA
- **C**onfidentiality
  - If it's supposed to be a secret, be careful who hears it
- **I**ntegrity
  - Don't let someone change something they shouldn't
- **A**vailability
  - Don't let someone stop others from using services

What Are the Threats?

- Theft
- Privacy
- Destruction
- Interruption or interference with computer-controlled services

Thinking About Threats

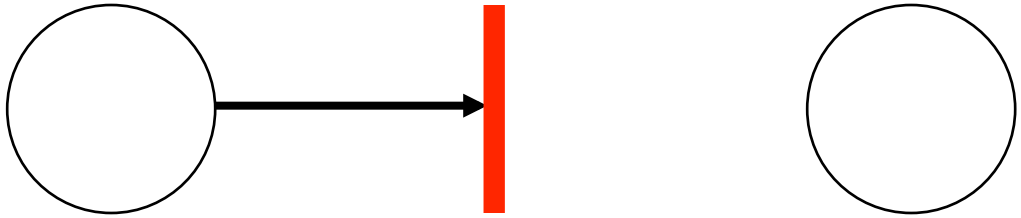
- Threats are viewed as types of attacks on normal services
- So, what is normal service?



Information Source

Information Destination

Interruption



Information Source

Information Destination

The information never reaches the destination

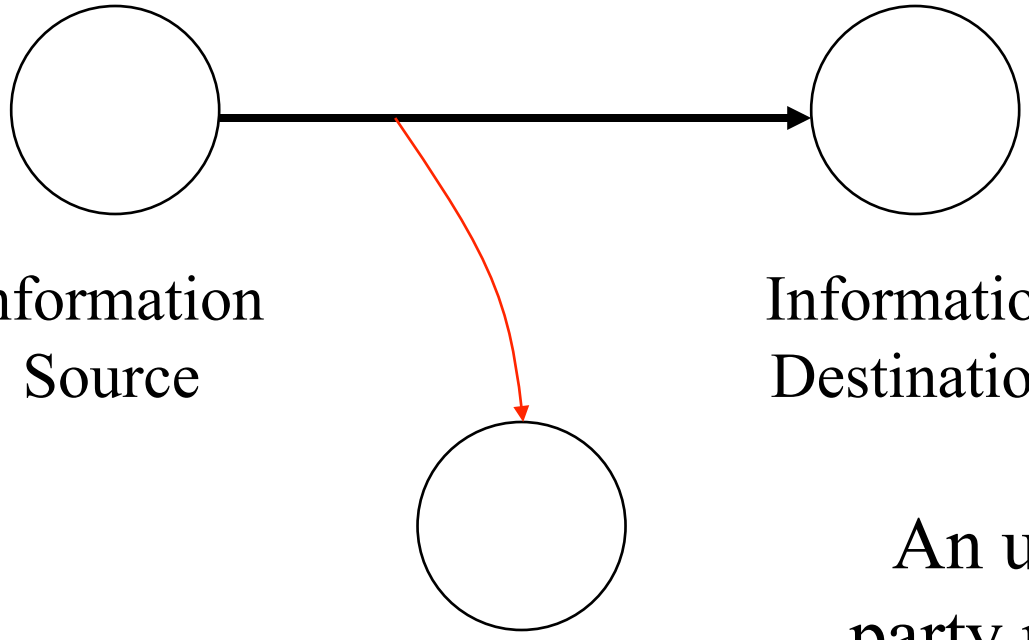
Interruption Threats

- Denial of service
- Prevents source from sending information to receiver
- Or receiver from sending requests to source
- A threat to availability

How Do Interruption Threats Occur?

- Destruction of hardware, software, or data
- Interference with a communications channel
- Overloading a shared resource

Interception



An unintended party receives the information

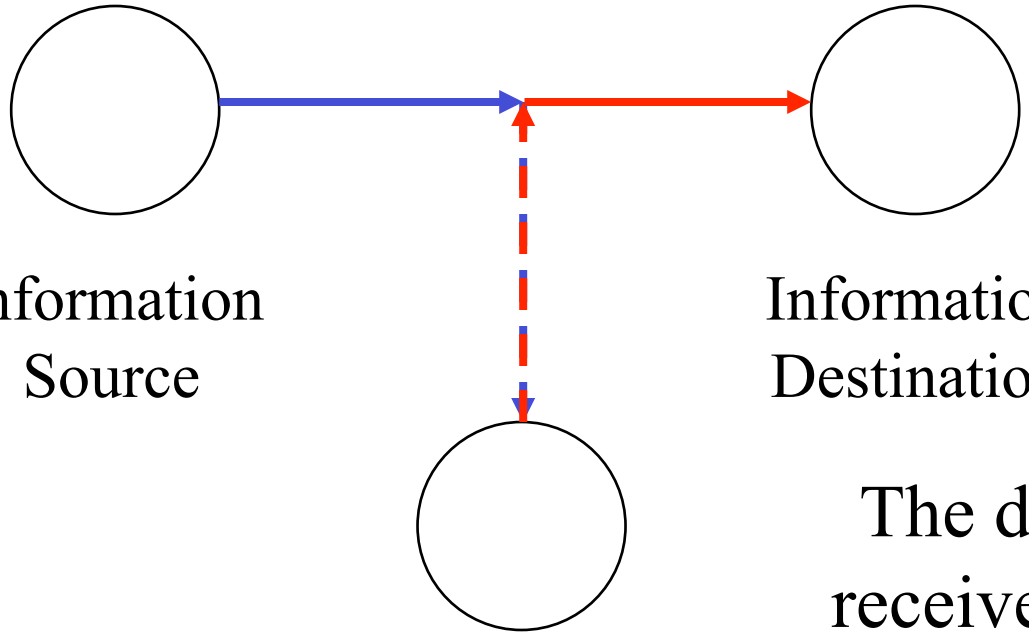
Interception Threats

- Data or services are provided to an unauthorized party
- Either in conjunction with or independent of a legitimate request
- A threat to confidentiality

How Do Interception Threats Occur?

- Eavesdropping
- Masquerading
- Break-ins
- Illicit data copying

Modification



The destination receives different information than what was originally sent

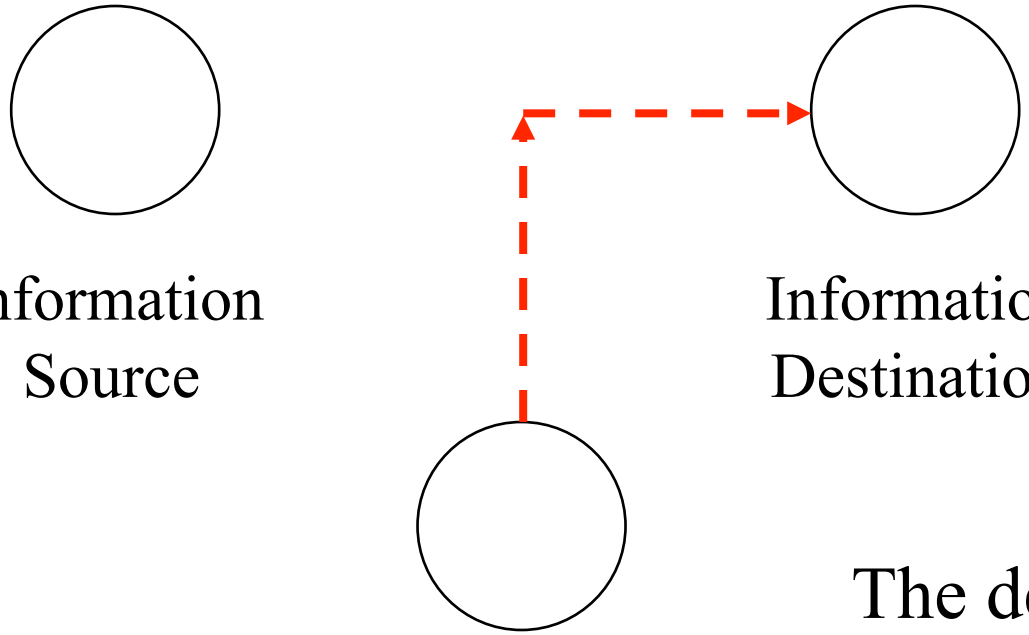
Modification Threats

- Unauthorized parties modify the data
- Either on the way to the users
- Or permanently at the servers
- A threat to integrity

How Do Modification Threats Occur?

- Interception of data requests/replies
- Masquerading
- Break-ins
- Flaws in applications allowing unintended modifications
- Other forms of illicit access to servers and their services

Fabrication



The destination receives information the source never sent

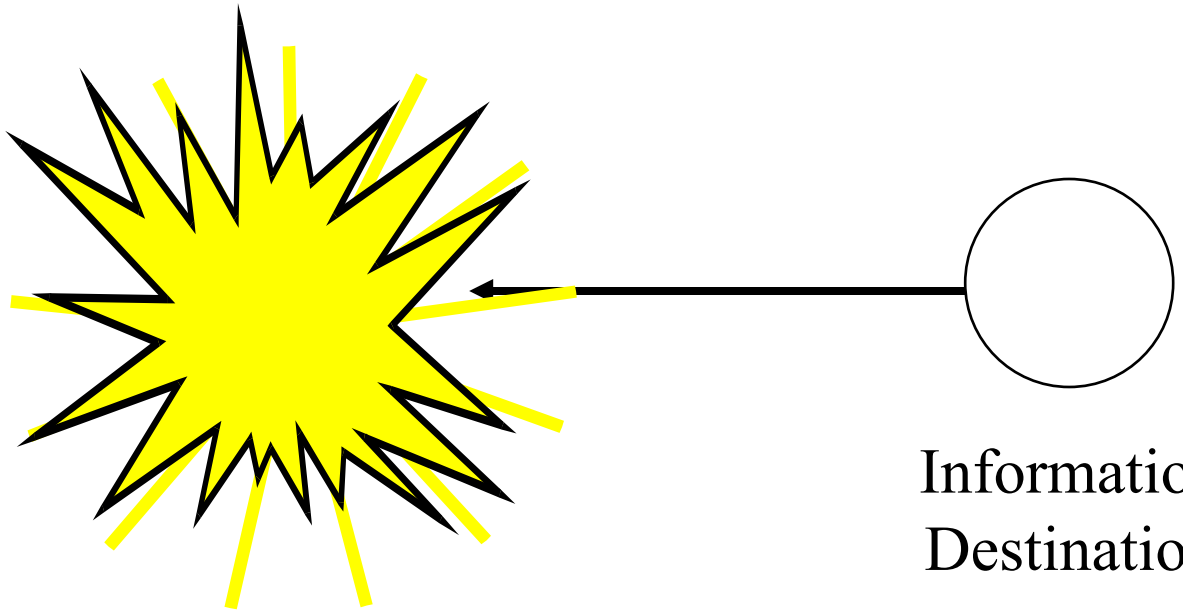
Fabrication Threats

- Unauthorized parties insert counterfeit objects into the system
- Causing improper changes in data
- Or improper use of system resources
- Or other bad behavior
- A threat to integrity

How Do Fabrication Threats Occur?

- Masquerading
- Bypassing protection mechanisms
- Duplication of legitimate requests/ responses

Destruction Threats



Information
Destination

The information is no longer accessible to a legitimate user

Destruction Threats

- Destroy data, hardware, software, etc.
- Often easier to destroy something than usefully modify it
- Often (not always) requires physical access
  - As counterexample, consider demo of destroying power generator[1]
  - Stuxnet destroyed centrifuges
- Destruction threats primarily threaten availability

[1] http://www.cnn.com/2007/US/09/26/power.at.risk/index.html?iref=newssearch#cnnSTCVideo

Active Threats Vs. Passive Threats

- *Passive threats* are forms of eavesdropping
  - No modification, injections of requests, etc.
- *Active threats* are more aggressive
- Passive threats are mostly to secrecy
- Active threats are to all properties

Social Engineering and Security

- The best computer security practices are easily subverted by bad human practices
  - E.g., giving passwords out over the phone to anyone who asks
  - Or responding to bogus email with your credit card number
- Social engineering attacks tend to be cheap, easy, effective
- So all our work may be for naught

Social Engineering Example

- Phishing
- Attackers send plausible email requesting you to visit a web site
- To "update" your information
- Typically a bank, popular web site, etc.
- The attacker controls the site and uses it to obtain your credit card, SSN, etc.
- Likelihood of success based on attacker's ability to convince the victim that he's real
  - And that the victim had better go to the site or suffer dire consequences

How Popular is Phishing?

- Anti-Phishing Work Group reported 10% increase in phishing for 1st quarter 2014[1]
  - 125,000 phishing attacks reported
  - Targeting 557 different brands
- Based on gullibility of humans more than computer vulnerability
- But can computer scientists do something to help?

[1] http://www.antiphishing.org/

Why Isn't Security Easy?

- Security is different than most other problems in CS
- The "universe" we're working in is much more hostile
- Human opponents seek to outwit us
- Fundamentally, we want to share secrets in a controlled way
  - A classically hard problem in human relations

What Makes Security Hard?

- You have to get everything right
  - Any mistake is an opportunity for your opponent
- When was the last time you saw a computer system that did everything right?
- So, must we wait for bug-free software to achieve security?

How Common Are Software Security Flaws?

- SANS used to publish weekly compendium of newly discovered security flaws
- About 1500 security flaws found per year
  - Only counting popular software
  - Only flaws with real security implications
  - And only those that were publicized
- SANS stopped doing this because it's not reasonable to expect anyone to keep up

Security Is Actually Even Harder

- The computer itself isn't the only point of vulnerability
- If the computer security is good enough, the foe will attack:
  - The users
  - The programmers
  - The system administrators
  - Or something you never thought of

A Further Problem With Security

- Security costs
  - Computing resources
  - People's time and attention
- If people use them badly, most security measures won't do the job
- Security must work 100% effectively
- With 0% overhead or inconvenience or learning

Another Problem

- Most computer practitioners know little or nothing about security
- Few programmers understand secure programming practices
- Few sysadmins know much about secure system configuration
- Typical users know even less

The Principle of Easiest Penetration

- *An intruder must be expected to use any available means of penetration. This is not necessarily the most obvious means, nor is it necessarily the one against which the most solid defense has been installed.*
- Put another way,
  - The smart opponent attacks you where you're weak, not where you're strong
  - And most opponents aren't stupid

But Sometimes Security Isn't That Hard

- The Principle of Adequate Protection:
  - *Computer items must be protected only until they lose their value. They must be protected to a degree consistent with their value.*
- So worthless things need little protection
- And things with timely value need only be protected for a while

Conclusion

- Security is important
- Security is hard
- A security expert's work is never done
  - At least, not for very long
- Security is full-contact computer science
  - Probably the most adversarial area in CS
- Intensely interesting, intensely difficult, and "the problem" will never be solved