

Introduction
CS 239
Computer Security
Peter Reiher
January 9, 2006

CS 239, Winter 2006

Lecture 1
Page 1

Description of Class

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

CS 239, Winter 2006

Lecture 1
Page 2

Topics to Be Covered

- Cryptography and authentication
- Design of secure protocols
- Network security – threats and countermeasures
- Secure operating systems design
- Practical application of security principles
- If time permits, other neat stuff

CS 239, Winter 2006

Lecture 1
Page 3

Prerequisites

- Must have taken CS111 and CS118, or equivalents
- Desirable to have taken an advanced OS course and advanced networking course

CS 239, Winter 2006

Lecture 1
Page 4

Grading

- Midterm – 25%
- Project – 50%
- Final – 25%

CS 239, Winter 2006

Lecture 1
Page 5

Class Format

- Typically we'll start each session with a discussion of material from last session
- Followed by lecture on new material
- Always feel free to stop me for questions or interesting discussions

CS 239, Winter 2006

Lecture 1
Page 6

Reading Materials

- Textbook
- Non-required supplemental texts
- Papers and web pages

CS 239, Winter 2006

Lecture 1
Page 7

Textbook

- *Computer Security: Art and Science*
 - By Matt Bishop
 - First edition
- Should be available in UCLA bookstore
- First reading assignment: Chapter 1

CS 239, Winter 2006

Lecture 1
Page 8

Supplemental Text 1

- *Applied Cryptography*
 - By Bruce Schneier
- Only covers what its title implies
 - And, as Schneier himself argues, there's a lot more to security
- But an excellent book on its subject
- Not required
 - No reading assignments from this book

CS 239, Winter 2006

Lecture 1
Page 9

Supplemental Text 2

- *Secrets and Lies*
 - Also 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

CS 239, Winter 2006

Lecture 1
Page 10

Papers and Web Pages

- Usually one paper per week and a couple of web pages
- Usually made available electronically
 - Through class web page
- Material in papers might or might not be lectured on
 - But it can appear on tests, regardless

CS 239, Winter 2006

Lecture 1
Page 11

Projects

- Either individual or small group
 - Depending on size of class
- Usually requiring program development
- Related to some topic covered in class
- Must be approved by instructor

CS 239, Winter 2006

Lecture 1
Page 12

Choosing a Project Topic

- Submit a 1 page proposal
 - By end of 3^d week of classes (January 27)
 - Email submissions OK
- I will approve them and offer suggestions
- Must be submitted, but not part of grade

CS 239, Winter 2006

Lecture 1
Page 13

What Makes a Good Project?

- Something new
- Something you're interested in
- Maybe it can turn into a paper for you
- Feasible to demonstrate something interesting within the quarter
 - Running code or other practical demonstration, not just a paper

CS 239, Winter 2006

Lecture 1
Page 14

Possible Project Topics

- Security for Internet infrastructure
- Defenses against spam, phishing and click fraud
- Handling botnets
- Security for ad hoc wireless networks and peer systems
- Methods for measuring and evaluating security
- Intrusion and insider threat detection
- DDoS and worm defense mechanisms
- Security for sensor networks
- Security evaluations of local labs
- Language-based approaches to secure coding
- OS enhancements for security

CS 239, Winter 2006

Lecture 1
Page 15

Project Updates

- Due at the end of the 7th week of class
 - February 24th
- 1 page report on your group's progress on its project
 - Email submission OK
- Not graded, but required
 - And should describe actual progress

CS 239, Winter 2006

Lecture 1
Page 16

Project Reports

- Written report on the project
- Should:
 - Describe project
 - Discuss how project was performed
 - Cover difficulties and interesting points
 - Describe the implementation
- Expected to be around 15 pages

CS 239, Winter 2006

Lecture 1
Page 17

Project Demos

- Must show working version of project to instructor
- Schedule time individually for this
- Must be done by middle of finals week

CS 239, Winter 2006

Lecture 1
Page 18

Project Deadlines

- Submit project proposal – January 27th
- Submit project update – February 24th
- Demonstration of project to instructor and project reports – March 24th

CS 239, Winter 2006

Lecture 1
Page 19

Tests

- Midterm – February 8 in class
- Final – March 22 (3-6 PM)
- Both tests will be open book
 - Essay questions concentrating on applying knowledge

CS 239, Winter 2006

Lecture 1
Page 20

Office Hours

- MW 2-3
- Held in 3532F Boelter Hall
- Other times available by prior arrangement

CS 239, Winter 2006

Lecture 1
Page 21

Class Web Page

www.lasr.cs.ucla.edu/classes/239_1.winter06

- Slides for classes will be posted there
 - By 5 PM the previous afternoon
 - In 6-up PDF form
- Readings will be posted there
 - With links to papers
- Also links to other interesting info

CS 239, Winter 2006

Lecture 1
Page 22

Introduction to Computer Security

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

CS 239, Winter 2006

Lecture 1
Page 23

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

CS 239, Winter 2006

Lecture 1
Page 24

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
 - Many individuals have been harmed by phishing and identity theft
 - A cyberattack released a large quantity of sewage in Australia
 - Companies continue to increase spending on cybersecurity

CS 239, Winter 2006

Lecture 1
Page 25

Some Examples of Large Scale Security Problems

- The Internet Worm
- New malicious code attacks
- Distributed denial of service attacks
- Vulnerabilities in commonly used systems

CS 239, Winter 2006

Lecture 1
Page 26

The Internet Worm

- Launched in 1988
- A program that spread over the Internet to many sites
- Around 6,000 sites were shut down to get rid of it
- And (apparently) its damage was largely unintentional
- The holes it used have been closed
 - But the basic idea still works

CS 239, Winter 2006

Lecture 1
Page 27

Malicious Code Attacks

- Multiple new viruses, worms, and Trojan horses appear every week
- The Virkel.f Trojan horse attacks instant messaging
 - Clicking on a link in the instant message infects your machine
- IM attacks becoming increasingly popular
 - And cell phone attacks appearing

CS 239, Winter 2006

Lecture 1
Page 28

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
- Attacks are increasing in sophistication
- In general form, an extremely hard problem

CS 239, Winter 2006

Lecture 1
Page 29

The DNS DDoS Attack

- Attack on the 13 root servers of the DNS system
- Ping flood on all servers
- Interrupted service from 9 of the 13
- But did not interrupt DNS service in any noticeable way

CS 239, Winter 2006

Lecture 1
Page 30

Vulnerabilities in Commonly Used Systems

- 802.11 WEP is fatally flawed
- Vulnerabilities pop up regularly in Windows and Linux
 - E.g., current WMF format flaw
- Many popular applications have vulnerabilities
- Many security systems have vulnerabilities
 - Symantec's antivirus products recently found to have buffer overflow

CS 239, Winter 2006

Lecture 1
Page 31

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 will follow
- Common problems:
 - Credit card number theft (often via phishing)
 - Extortion for stolen on-line information
 - Identity theft (phishing, again, is a common method)
 - Manipulation of e-commerce sites
 - Extortion via DDoS attacks

CS 239, Winter 2006

Lecture 1
Page 32

Some Recent Statistics

- From Computer Security Institute/FBI Computer Crime and Security Survey, 2005¹
- 53% of respondents reported unauthorized use of their systems
- Total estimated losses by respondents: \$130 million
 - Primarily costs of handling viruses, unauthorized access, and data theft

<http://www.usdoj.gov/criminal/cybercrime/FBI2005.pdf>

CS 239, Winter 2006

Lecture 1
Page 33

How Much Attack Activity Is There?

- Blackhole monitoring on a small (8 node) network¹
- Detected 640 **billion** attack attempts over four month period
- At peak of Nimda worm's attack, 2000 worm probes per **second**

¹ Unpublished research numbers from Farnham Jahanian, U. of Michigan, DARPA PTN PI meeting, January 2002.

CS 239, Winter 2006

Lecture 1
Page 34

But Do We Really Need Computer Security?

- The preceding examples suggest we must have it
- Yet many computers are highly insecure
- Why?
- Ultimately, because many people don't think they need security
 - Or don't understand what they need to do to get it

CS 239, Winter 2006

Lecture 1
Page 35

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
- And, relatively speaking, the computer/network environment is still fairly benevolent
- Ignorance also plays a role
 - Increasing numbers of users are unsophisticated

CS 239, Winter 2006

Lecture 1
Page 36

Well, What About Tomorrow?

- Will security become more important?
- Yes!
- Why?
 - More money on the network
 - More sophisticated criminals
 - More leverage from computer attacks
 - More complex systems

CS 239, Winter 2006

Lecture 1
Page 37

What Are Our Security Goals?

- Confidentiality
 - If it's supposed to be a secret, be careful who hears it
- Integrity
 - Don't let someone change something they shouldn't
- Availability
 - Don't let someone stop others from using services
- Exclusivity
 - Don't let someone use something he shouldn't

CS 239, Winter 2006

Lecture 1
Page 38

What Are the Threats?

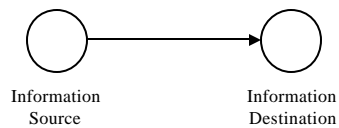
- Theft
- Privacy
- Destruction
- Interruption or interference with computer-controlled services

CS 239, Winter 2006

Lecture 1
Page 39

Thinking About Threats

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



CS 239, Winter 2006

Lecture 1
Page 40

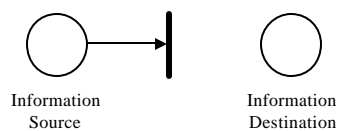
Classification of Threats

- Secrecy
- Integrity
- Availability
- Exclusivity

CS 239, Winter 2006

Lecture 1
Page 41

Interruption



The information never reaches the destination

CS 239, Winter 2006

Lecture 1
Page 42

Interruption Threats

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

CS 239, Winter 2006

Lecture 1
Page 43

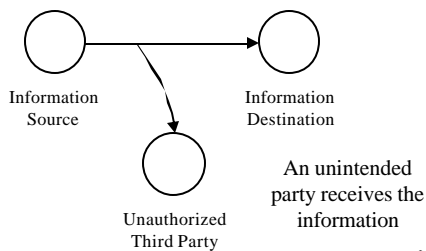
How Do Interruption Threats Occur?

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

CS 239, Winter 2006

Lecture 1
Page 44

Interception



CS 239, Winter 2006

Lecture 1
Page 45

Interception Threats

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

CS 239, Winter 2006

Lecture 1
Page 46

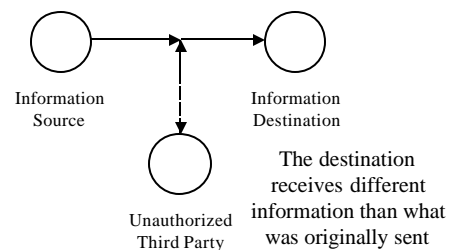
How Do Interception Threats Occur?

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

CS 239, Winter 2006

Lecture 1
Page 47

Modification



CS 239, Winter 2006

Lecture 1
Page 48

Modification Threats

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

CS 239, Winter 2006

Lecture 1
Page 48

- # Modification Threats
- Unauthorized parties modify the data
 - Either on the way to the users
 - Or permanently at the servers
 - A threat to integrity
- CS 239, Winter 2006
- Lecture 1
Page 48

Modification Threats

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

CS 239, Winter 2006

Lecture 1
Page 48

Modification Threats

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

CS 239, Winter 2006

Lecture 1
Page 48

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

CS 239, Winter 2006

Lecture 1
Page 50

- # 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
- CS 239, Winter 2006
- Lecture 1
Page 50

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

CS 239, Winter 2006

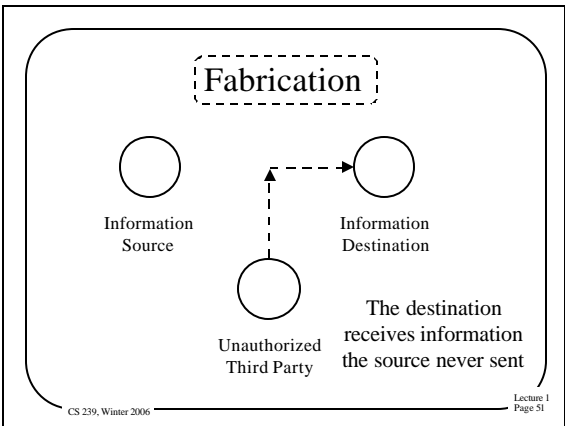
Lecture 1
Page 50

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

CS 239, Winter 2006

Lecture 1
Page 50

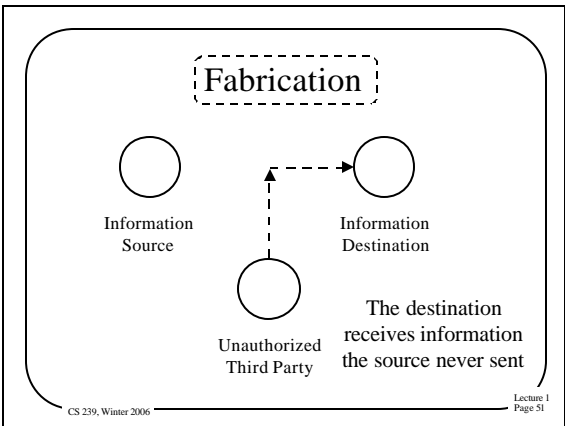


```
graph LR;
    A((Information Source)) -.-> B((Information Destination));
    C((Unauthorized Third Party)) -.-> B;
```

The diagram shows three nodes: Information Source, Information Destination, and Unauthorized Third Party. A dashed arrow points from Information Source to Information Destination. Another dashed arrow points from Unauthorized Third Party to Information Destination. A text box labeled 'Fabrication' is at the top. A text box at the bottom right states: 'The destination receives information the source never sent'.

CS 239, Winter 2006

Lecture 1
Page 51



```
graph LR;
    A((Information Source)) -.-> B((Information Destination));
    C((Unauthorized Third Party)) -.-> B;
```

The diagram shows three nodes: Information Source, Information Destination, and Unauthorized Third Party. A dashed arrow points from Information Source to Information Destination. Another dashed arrow points from Unauthorized Third Party to Information Destination. A text box labeled 'Fabrication' is at the top. A text box at the bottom right states: 'The destination receives information the source never sent'.

CS 239, Winter 2006

Lecture 1
Page 51

```
graph LR;
    A((Information Source)) -.-> B((Information Destination));
    C((Unauthorized Third Party)) -.-> B;
```

The diagram shows three nodes: Information Source, Information Destination, and Unauthorized Third Party. A dashed arrow points from Information Source to Information Destination. Another dashed arrow points from Unauthorized Third Party to Information Destination. A text box labeled 'Fabrication' is at the top. A text box at the bottom right states: 'The destination receives information the source never sent'.

CS 239, Winter 2006

Lecture 1
Page 51

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
 - And possibly exclusivity

CS 239, Winter 2006

Lecture 1
Page 52

- ## 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
 - And possibly exclusivity
- CS 239, Winter 2006
- Lecture 1
Page 52

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
 - And possibly exclusivity

CS 239, Winter 2006

Lecture 1
Page 52

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
 - And possibly exclusivity

CS 239, Winter 2006

Lecture 1
Page 52

How Do Fabrication Threats Occur?

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

CS 239, Winter 2006

Lecture 1
Page 53

- # How Do Fabrication Threats Occur?
- Masquerading
 - Bypassing protection mechanisms
 - Duplication of legitimate requests/responses
- CS 239, Winter 2006
- Lecture 1
Page 53

How Do Fabrication Threats Occur?

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

CS 239, Winter 2006

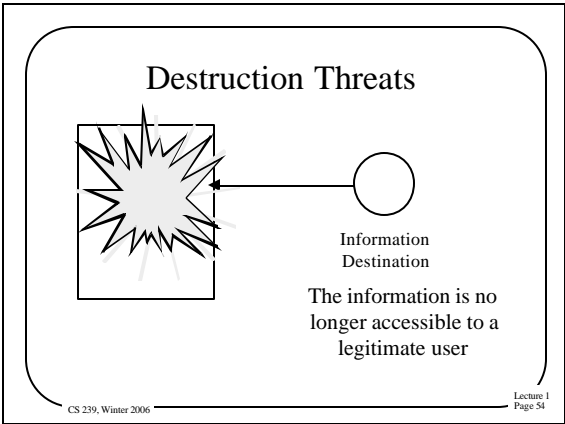
Lecture 1
Page 53

How Do Fabrication Threats Occur?

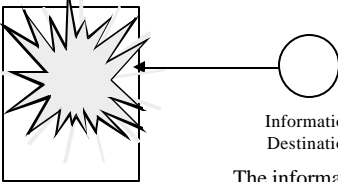
- Masquerading
- Bypassing protection mechanisms
- Duplication of legitimate requests/responses

CS 239, Winter 2006

Lecture 1
Page 53



Destruction Threats

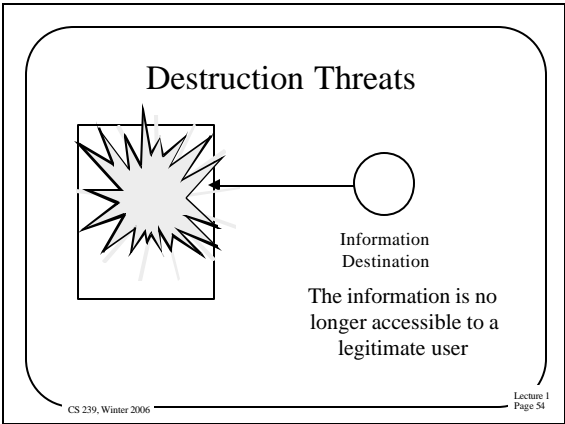


Information
Destination

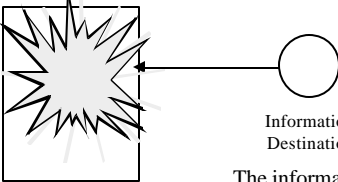
The information is no
longer accessible to a
legitimate user

CS 239, Winter 2006

Lecture 1
Page 54



Destruction Threats



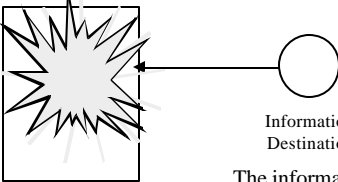
Information
Destination

The information is no
longer accessible to a
legitimate user

CS 239, Winter 2006

Lecture 1
Page 54

Destruction Threats



Information
Destination

The information is no
longer accessible to a
legitimate user

CS 239, Winter 2006

Lecture 1
Page 54

Destruction Threats

- Destroy data, hardware, messages, or software
- Often easier to destroy something than usefully modify it
- Often (but not always) requires physical access

CS 239, Winter 2006

Lecture 1
Page 55

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

CS 239, Winter 2006

Lecture 1
Page 56

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

CS 239, Winter 2006

Lecture 1
Page 57

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

CS 239, Winter 2006

Lecture 1
Page 58

How Popular is Phishing?

- Anti-Phishing Work Group reported 15,820 new phishing schemes in October 2005 alone¹
- Up from 6957 in October 2004
- Based on gullibility of humans more than computer vulnerability
- But can computer scientists do something to help?

¹<http://www.antiphishing.org/>

CS 239, Winter 2006

Lecture 1
Page 59

Another New Form of Cyberattack

- Click fraud
- Based on popular pay-per-click model of Internet advertising
- Two common forms:
 - Rivals make you pay for “false clicks”
 - Profit sharers “steal” or generate bogus clicks to drive up profits

CS 239, Winter 2006

Lecture 1
Page 60

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

CS 239, Winter 2006

Lecture 1
Page 61

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?

CS 239, Winter 2006

Lecture 1
Page 62

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

CS 239, Winter 2006

Lecture 1
Page 63

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

CS 239, Winter 2006

Lecture 1
Page 64

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

CS 239, Winter 2006

Lecture 1
Page 65

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

CS 239, Winter 2006

Lecture 1
Page 66

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