Introduction to Security
COMS W4995.002

Prof. Steven M. Bellovin

smb@cs.columbia.edu

http://www.cs.columbia.edu/~smb/classes/f05

What is this Course?

- Security primitives
- Security architecture
- How to think about security
- ☞ How to think about insecurity. . .
- *Not* 4180 — complementary to it

Security Primitives

- What are the basic mechanisms you can use to secure a system?
- What are the properties of these mechanisms?
- What is the *assurance* associated with them?

Security Architecture

- How to put the pieces together
- How to spot the risky parts
- How to evaluate an architecture
- “Security Architecture” is actually a better title for this course

How to Think About Security

- Security is a property of the overall design
- You do *not* get security by sprinkling on crypto or by forcing people to change their passwords frequently
- Those can sometimes help — but bad guys go around strong security, not through it

How to Think About Insecurity. . .

- The bad guys don't follow the rules
- To understand how to secure a system, you have to understand what sort of attacks are possible
- Note that that is *not* the same as actually launching them. . .

Course Structure

- Lecture format
- Syllabus subject to change to discuss current events
- Approximate grading percentages:

Homework	45%
Midterm	25%
Final	30%

- Grading will be via Courseworks:
  `https://courseworks.columbia.edu`

Textbook

- Matt Bishop
  Introduction to Computer Security
  Addison-Wesley-Longman
  ISBN: 0-321-24744-2
- Occasional assigned readings of other material
- I assume you all know how to use the library and/or electronic resources. (Hint: Google does not (yet?) have access to all of the world's knowledge.)

Homework

- Approximately 6 homework assignments
- The usual time for an assignment is two weeks
- Late homeworks lose 10% of their value for each *day* late. (5% for the first 12 hours.) Homeworks must be submitted electronically before the start of class.
- You have 3 (integral) penalty-free late days, which you may apply at your choice to one or more homework assignments.
- Exceptions granted only for *unforeseeable* events. Workload, day job, etc., are quite foreseeable.

Programming Assignments

- All programming homework *must* be done in C or C++ unless otherwise instructed. Don't bother asking for exceptions.
- Turn in a single `tar` file, including a Makefile.
- If necessary, include test data and a README file with execution instructions
- All programs *must* compile and run on Linux, on the CLIC machines (see `http://www1.cs.columbia.edu/CLIC/index.html`).
- Zero credit for programs that don't compile.
- Because most security problems are due to buggy code, there will be copious deductions for bugs or for inadequate documentation

Co-operation versus Dishonesty

- Discussing homework with others is encouraged
- All programs and written material *must* be individual work unless otherwise instructed.
- Please use appropriate file permission mechanisms to protect your homework. (Looking at other people's work is not allowed.)
- Zero tolerance for cheating
- See the department's policy on academic honesty: `http://www.cs.columbia.edu/education/honesty`. I will assume that you have all read it; you are in any event responsible for its terms and provisions.

The Ethics of Security

- Taking a computer security class is *not* an excuse for hacking
- “Hacking” is any form of unauthorized access, including exceeding authorized permissions
- The fact that a file or computer is not properly protected is no excuse for unauthorized access
- *If* the owner of a resource invites you to attack it, such use is authorized
- For more details, see `http://www.columbia.edu/cu/policy/network_use.html`
- *Absolutely no Trojan horses, back doors, or other malicious code in homework assignments*
- No, I’m not joking

Contacting Me

- Feel free to drop in during office hours: 1:30-2:30 Mondays and 1:00-2:00 Tuesdays, in 454 CSB
- I'll announce changes on my home page
- I'm amenable to meeting other times, by appointment. You're welcome to drop in if my office door is open, but I reserve the right to ask you to come back later
- If you have any questions, please use email rather than telephone; I travel a lot and am not very reachable by phone

Class Schedule

- The class may occasionally be rescheduled
- All lectures are available via CVN — feel free to watch it that way
- The midterm and final will be during regular class sessions — October 19 and December 12.

What is Security?

What is Security?

Security is keeping unauthorized entities from doing things you don't want them to do.

This definition is too informal. . .

What is Security?

- Confidentiality
- Integrity
- Avaibility

Confidentiality

- "The property that information is not made available or disclosed to unauthorized individuals, entities, or processes [i.e., to any unauthorized system entity]." [definitions from RFC 2828]
- Not the same as *privacy*.
- **Privacy**: "The right of an entity (normally a person), acting in its own behalf, to determine the degree to which it will interact with its environment, including the degree to which the entity is willing to share information about itself with others."
- Privacy is a reason for confidentiality
- The traditional primary focus of computer security

Integrity

- **data integrity**: “The property that data has not been changed, destroyed, or lost in an unauthorized or accidental manner.”
- **system integrity**: “The quality that a system has when it can perform its intended function in a unimpaired manner, free from deliberate or inadvertent unauthorized manipulation.”
- Often of more commercial interest than confidentiality

Availability

- “The property of a system or a system resource being accessible and usable upon demand by an authorized system entity, according to performance specifications for the system; i.e., a system is available if it provides services according to the system design whenever users request them.”
- Turning off a computer provides confidentiality and integrity, but hurts availability. . .
- Denial of service attacks are direct assaults on availability

They Interact

- It's obvious that violations of integrity can be used to compromise confidentiality
- In some situations, violations of availability can be used that way as well

More Definitions

vulnerability An error or weakness in the design, implementation, or operation of a system

attack A means of exploint some vulnerability in a system

threat An adversary that is motivated and capable of exploiting a vulnerability

(Definitions from *Trust in Cyberspace*)

Vulnerabilities

- The technical failing in a system
- The primary focus of most computer security classes
- If you can close the vulnerabilities, the threats don’t matter
- Or do they?

Threats

- Different enemies have different abilities
- Teenage joy-hackers can’t crack a modern cryptosystem
- Serious enemies can exploit the “three Bs”: burglary, bribery, and blackmail
- You can’t design a security system unless you know who the enemy is

The Human Element

"Humans are incapable of securely storing high-quality cryptographic keys, and they have unacceptable speed and accuracy when performing cryptographic operations. They are also large, expensive to maintain, difficult to manage, and they pollute the environment. It is astonishing that these devices continue to be manufactured and deployed, but they are sufficiently pervasive that we must design our protocols around their limitations."

Network Security: Private Communication in a Public World

Designing Security

- The problem is overconstrained
- Among the contraints are cost, human behavior, and ease of operation
- In the real world, realistic security is often far more important than theoretical security
- *What are you trying to protect against whom?*

What this Course is About

- Mechanisms
- Threat analysis
- Security architecture
- Assurance
- In short, *engineering* secure systems