CSE 227
Computer Security
Winter 2008
Stefan Savage
Course info

- Stefan Savage
  - Web: http://www.cs.ucsd.edu/~savage
  - E-mail: savage@cs.ucsd.edu
  - Office hours: M 3-4pm (or by appt)
    CSE 3106

- Course Web pages (empty now)
  - http://www.cse.ucsd.edu/classes/wi08/cse227/
This is a relatively “new” class

You

Me
Goals and non-goals

- **Goals**
  - Explore range of current problems and tensions in modern computer security
  - Understand how to identify security issues in your own research and how to address them
  - Figure out if security is an area of interest for you
  - Get feet wet in security research (mini research project)

- **Non-goals**
  - Review of all std security mechanisms
    - Read a textbook or take CSE127
  - Significant examination of applied cryptography
    - Take one of our great crypto courses
Readings

- There is no textbook for this class
  - We’ll read a bunch of papers and occasionally from some books

- However, in general I recommend:
  - Security Engineering by Ross Anderson
    http://www.cl.cam.ac.uk/~rja14/book.html

- For those who want some general “backup”, check out
  - Security in Computing by Charles Pfleeger
  - Introduction to Computer Security by Matt Bishop
Misc crud

- **Grading**
  - 20% participation
  - 50% project
  - 30% final

- **Research project**
  - Of your choosing (more on this Thurs)
  - Two people (if you want more, justify it to me)
  - Short paper (e.g. 6 pages) & presentation (10 mins)
  - High standards
    - Last year: one published, two being submitted for publication later this year (expanded versions) and at least two more could have been
My background…

- Originally OS kernels…
  and networking…
- Came to Security by accident
  - Misbehaving TCP receivers – think like a bad guy
  - DDoS traceback – in response to 2000 attacks
- Startup
  and…
  synchronicity (David Moore @ UCSD found indirect evidence of spoofed DoS attacks, hmmm… general analysis possible)
  Startup was failure, analysis technique was golden
- Code Red
  - Same technique allowed measuring worm outbreaks
  - Interest * opportunity snowballed…
Recently

- Research
  - I direct the Collaborative Center for Internet Epidemiology and Defenses (CCIED) (www.ccied.org)
    - Joint UCSD/Berkeley ICSI effort
    - Focus on large-scale Internet attacks (bots, spyware, worms, underground economics, etc)

- Policy
  - National Research Council’s Cybersecurity Research group
  - Institute for Defense Analysis’ ISAT advisory group for DARPA

- Industry
  - Netsift (UCSD-originated worm/virus defense company) -> Cisco
  - Expert witness consulting
Topics we’ll be covering

- Human factors/usability
- Measurement/analysis studies
- Crypto engineering
- System design/implementation
  - Protection, small TCB, etc
- Information exposure
  - Privacy, anonymity, side & covert channels
- Software vulnerabilities & malware
  - Vulnerability research, viruses, botnets, defenses, etc
- I’m open to more topics… got any?
What is security?
What is security?

- Merriam-Webster online dictionary:
  Function: noun
  1: the quality or state of being secure:
     a: freedom from danger;
     b: freedom from fear or anxiety;
     c: freedom from the prospect of being laid off (job security)
  2 a: something given, deposited, or pledged to make certain the fulfillment of an obligation
     b: SURETY
  3: an instrument of investment in the form of a document (as a stock certificate or bond) providing evidence of its ownership
  4 a: something that secures: PROTECTION
     b: (1)
  Measures taken to guard against espionage or sabotage, crime, attack, or escape
Computer security?

- Most of computer science is about providing functionality:
  - User Interface
  - Software Design
  - Algorithms
  - Operating Systems/Networking
  - Compilers/PL
  - Vision/graphics
  - Microarchitecture
  - VLSI/CAD
- Computer security is **not** about functionality
- It is about how the embodiment of functionality behaves in the presence of an adversary
Two competing philosophies...

- **Binary** model
  - Traditional crypto and trustworthy systems
  - Assume adversary limitations \( X \) and define security policy \( Y \)
  - If \( Y \) cannot be violated without needing \( X \) then system is secure, else insecure

- **Risk management** model
  - Most commercial software development (and much real-world security… e.g., terrorism)
  - Try to minimize biggest risks and threats
  - Improve security where most cost effective (expected value)
Classic example (binary): perfect substitution cipher

Invited by combination of Vernam(1919) & Mauborgne

Choose a string of random bits the same length as the plaintext, XOR them to obtain the ciphertext.

Perfect Secrecy (proved by Claude Shannon)
  - Probability that a given message is encoded in the ciphertext is unaltered by knowledge of the ciphertext
  - Proof: Give me any plaintext message and any ciphertext and I can construct a key that will produce the ciphertext from the plaintext.
Classic example (risk mgmt): Concrete barricades

- Prevent incursion by car bombers
The problems with the binary model

- Hard to assume $X$ in real systems
  - Real artifacts fragile, imperfect
    » E.g. buffer overflow vulnerabilities
  - Implicit dependencies with exposed layer
    » Example: reading RSA bits off current draw

Courtesy Paul Kocher via Dan Boneh’s 155
The problems with the binary model (cont)

- Hard to know what policy Y is in advance?
  - What are the dangers?
- Examples:
  - SPAM
  - Exchange rate fraud in South Africa vs SWIFT bank balance controls
  - Mobile code

- Finally: *hugely expensive*... how many certified systems out there?
The problems with the risk management model

I only need to win once, you simpering dolt!
The problem with the risk management model

- Creates arms race – forced co-evolution

Diagram:
- Adversary invents new attack
- Defender creates new defense
The problem with the risk management model

- It's fine to say security is a spectrum, but how to evaluate risk or reward?

- And the best you can hope for is stalemate
  - And we’re losing stalemate in a number of situations (e.g., SPAM, Malware)
Key issues

- Risks
- Threats
- Value
- Protection (locks)
  - Confidentiality, integrity, authenticity, availability & access control
- Deterrence (police)
  - Non-repudiation, Accountability/Auditability
- Identity, reputation, and trust
Policy

- What is a bad thing?
- Remarkably tricky to define…
  - The software on your computer likely has 100s of security options
  - How should you set them?
- Can be non-intuitive
  - Should a highly privileged user have more rights on a system or less?
Risks & threats

- Risk
  - What bad things are possible?
  - How bad are they and how likely are they?

- Threats
  - Who is targeting the risk?
  - What are their capabilities?
  - What are their motivations?

- These tend to be well understood/formalized in some communities (e.g. finance sector) and less in others (e.g. computer science)
The Threat Landscape
(courtesy David Aucsmith, Microsoft)

- National Interest
  - Script-Kiddy
  - Hobbyist
  - Expert
  - Specialist

- Personal Gain
  - Thief
  - Trespasser
  - Author

- Personal Fame
  - Vandal

- Curiosity
  - Spy

January 8, 2008
CSE 227 -- Lecture 1 -- Introduction
The Threat Landscape

National Interest
- Spy

Personal Gain
- Fastest growing segment
- Thief
- Trespasser

Personal Fame
- Author

Curiosity
- Vandal
- Script-Kiddy
- Hobbyist Hacker
- Expert
- Specialist

Hacking and the bad guys
The Threat Landscape

National Interest

Personal Gain

Personal Fame

Curiosity

Script-Kiddy

Hobbyist Hacker

Expert

Specialist

Spy

Fastest growing segment

Vandal

Thief

Trespasser

Author

Hacking and the bad guys
The Threat Landscape

- **National Interest**
- **Personal Gain**
- **Personal Fame**
- **Curiosity**

**Tools created by experts now used by less skilled attackers and criminals**

- **Spy**
- **Hacker**
- **Trespasser**
- **Author**
- **Expert**
- **Specialist**
- **Script-Kiddy**
- **Hobbyist Hacker**
- **Vandal**
Value

- What is the cost if the bad thing happens?
- What is the cost of preventing the bad thing?

- Example: Visa/Mastercard fraud
- Example: Permission Action Links for nuclear weapons
Protection (locks)

- The mechanisms used to protect resources against threats
  - This is most of academic and industrial computer security

- Many classes of protections
  - Cryptographic protection of data
  - Software guards
  - Communication guards

- Can be either proactive or reactive
Deterrence

- There is some non-zero expectation that there is a future cost to doing a bad thing
  - i.e. going to jail, having a missile hit your house, having your assets seized, etc
  - Criminal cost-benefit: \( M_b + P_b > O_{cp} + O_{cm} P_a P_c \) [Clark&Davis 95]
    - \( M_b \): Monetary benefit
    - \( P_b \): Psychological benefit
    - \( O_{cp} \): Cost of committing crime
    - \( O_{cm} \): Monetary cost of conviction
    - \( P_a \): Probability of getting caught
    - \( P_c \): Probability of conviction

- Need meaningful forensic capabilities
  - Audit actions, assign identity to evidence, etc
  - Must be cost effective relative to positive incentives
Identity & reputation

- What is identity?
  - Why is it valuable?
  - What’s the difference between an identity and an identifier?

- Reputation?
  - Why is it valuable?

- Key issues:
  - Difference between identity and identifier
  - Binding of identity to reputation
  - Always be clear on how these qualities relate to action
Difference between Due diligence and trust

- **Due diligence**
  - Work to acquire multiple independent pieces of evidence establishing identity/reputation linkage; particularly via direct experience
  - Expensive

- **Trust**
  - Allows cheap form of due-diligence: third-party attestation
  - Economics of third-party attestation? Cost vs limited liability
  - What is a third-party qualified to attest to?
That’s it for today

- Any questions?

- For next time, read:

(both will have pointers on the Web page shortly)