Intro to Computer Security

Lujo Bauer

lbauer@cmu.edu

http://www.ece.cmu.edu/~lbauer

Fall 2011

Carnegie Mellon

Plan for Today

- What is computer security …
- … and why is it important?

- Types of computer misuse
- Basic security analysis
- A taxonomy of computer security

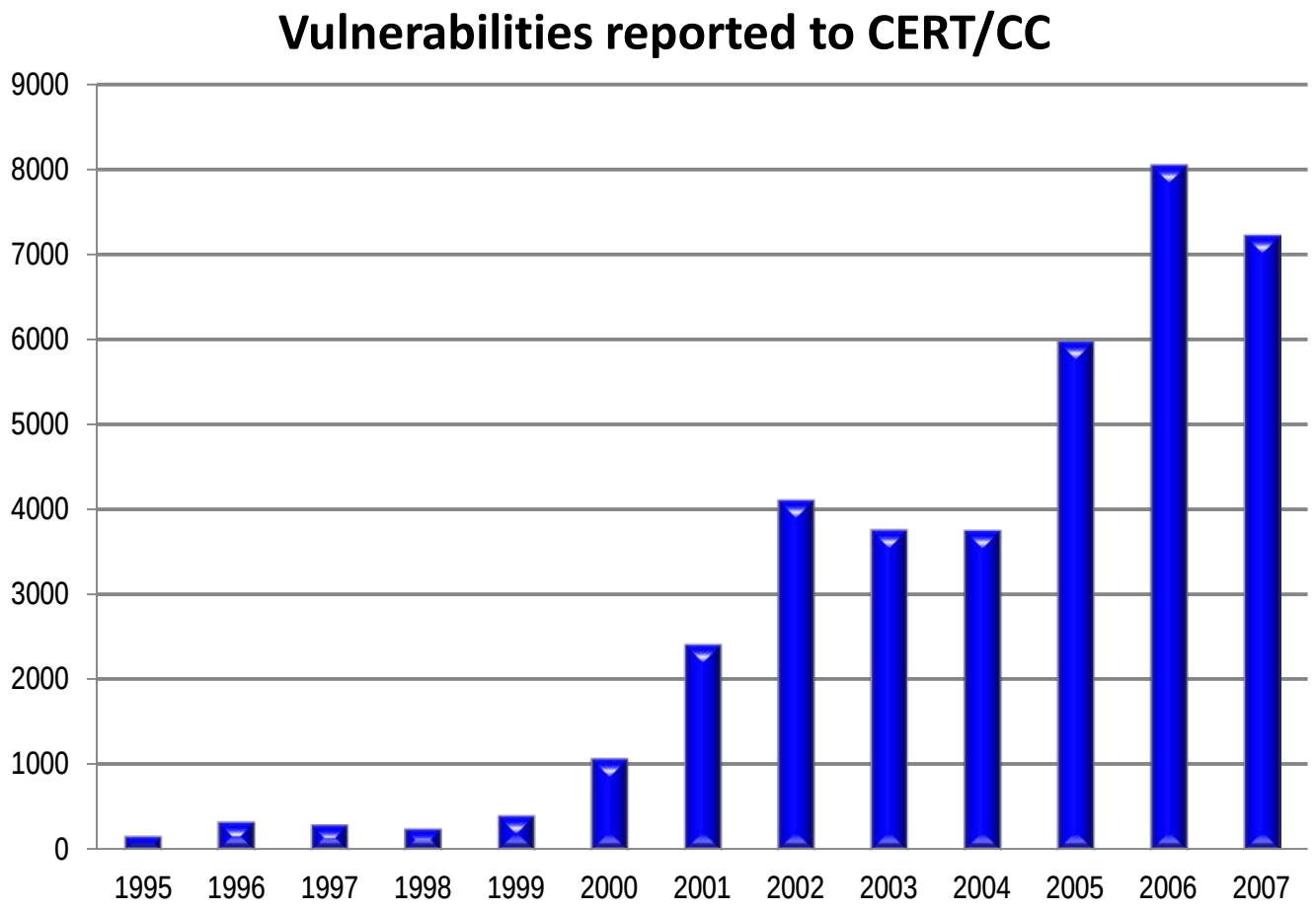
What Is Computer Security?

- **Protecting computers against misuse and interference**
- **Broadly comprised of three types of properties**
  - *Confidentiality*: information is protected from unintended disclosure
    - Secrecy, privacy
  - *Integrity*: system and data are maintained in a correct and consistent condition
  - *Availability*: systems and data are usable when needed
    - Also includes timeliness
- **These concepts overlap (and clash)**
- **These concepts are (perhaps) not all-inclusive**
  - Spam?
  - "Non-business related" surfing?

Why Is Computer Security Important?

There Are Lots of Bugs!

[http://www.cert.org/stats]



There Are Lots of Bugs!

- But is it a *computer security* problem?
- Computer security = protecting computers against misuse and interference
- Bugs can be (and are) purposefully exploited

Exploiting Bugs as a Nuisence

- **To be annoying**
  - Newsday technology writer & hacker critic found …
    - Email box jammed with thousands of messages
    - Phone reprogrammed to an out of state number where caller's heard an obscenity-loaded recorded message

  [Time Magazine, December 12, 1994]

Exploiting Bugs for Profit

- **Hacker convicted of breaking into a business' computer system, stealing confidential information and threatening disclosure if $200,000 not paid**

  [U.S. Dept. of Justice Press Release, Jul 2003]

- **11 people indicted for stealing more than 40 million credit card and debit card numbers**

  [CNN, Aug 2008]

Costs Can Be Staggering

- MyDoom (2004) - $38.5 billon
- SoBig (2003) - $37.1 billion
- Love Bug (2000) - $15 billion
- Code Red (2001) - $2 billion

Is It Just About Cost?

Software Bugs in the News

Unmanned European **rocket explodes on first flight**

Europe's newest unmanned satellite-launching rocket, the Ariane 5, intentionally was blown up Tuesday just seconds after taking off on its maiden flight. …

[http://edition.cnn.com/WORLD/9606/04/rocket.explode/]

… The internal SRI software exception was caused during execution of a data conversion from 64-bit floating point to 16-bit signed integer value. The **floating point number which was converted had a value greater than what could be represented by a 16-bit signed integer**. This resulted in an Operand Error. The data conversion instructions (in Ada code) were not protected from causing an Operand Error, although other conversions of comparable variables in the same place in the code were protected. …

[*ARIANE 5 Flight 501 Failure, Report by the Inquiry Board*, Paris, Jul 19 1996]

Software Bugs in the News

… A previously-unknown software flaw in a widely-deployed General Electric energy management system contributed to the devastating scope of the August 14th northeastern U.S. blackout …

[*Security Focus*, Feb 11 2004]

The Northeast Blackout of August 2003, the largest in North American history, shut down 62,000 MW of generation capacity, and cost businesses an estimated $13 billion in productivity. …

[*IEEE-USA Today's Engineer*, Feb 2005]

… "There was a couple of processes that were in contention for a common data structure, and through a software coding error in one of the application processes, they were both able to get write access to a data structure at the same time … And that corruption led to the alarm event application getting into an infinite loop and spinning." …

[*Security Focus*, Apr 7 2004]

Software Bugs in the News

E-voting vendor: Programming errors caused dropped votes

… E-voting machines from Premier Election Solutions, formerly called Diebold Election Systems, dropped hundreds of votes in 11 Ohio counties during the primary election, as the machine's memory cards uploaded to vote-counting servers. …

[*Network World*, Aug 22 2008]

Software Bugs in the News

… **Software bugs in a Soviet early-warning monitoring system nearly brought on nuclear war in 1983**, according to news reports in early 1999. The software was supposed to filter out false missile detections caused by Soviet satellites picking up sunlight reflections off cloud-tops, but failed to do so. Disaster was averted when a Soviet commander, based on a what he said was a '…funny feeling in my gut', decided the apparent missile attack was a false alarm. The filtering software code was rewritten. . …

[http://rajasriengg.wordpress.com/2008/07/16/recent-major-computer-system-failures-caused-by-software-bugs/]

Software Bugs in the News

- Accidents
- Monetary loss
- Effect on political process?
- Military conflict?

Types of Computer Misuse (1)

[Neumann and Parker 1989]

- **External**
  - Visual spying — Observing keystrokes or screens
  - Misrepresentation — Deceiving operators and users
  - Physical scavenging — "Dumpster diving" for printouts
- **Hardware misuse**
  - Logical scavenging — Examining discarded/stolen media
  - Eavesdropping — Intercepting electronic or other data
  - Interference — Jamming, electronic or otherwise
  - Physical attack — Damaging or modifying equipment
  - Physical removal — Removing equipment & storage media

Types of Computer Misuse (2)

[Neumann and Parker 1989]

- **Masquerading**
  - Impersonation — Using false identity external to computer
  - Piggybacking — Usurping workstations, communication
  - Spoofing — Using playback, creating bogus systems
  - Network weaving — Masking physical location or routing
- **Pest programs**
  - Trojan horses — Implanting malicious code
  - Logic bombs — Setting time or event bombs
  - Malevolent worms — Acquiring distributed resources
  - Viruses — Attaching to programs and replicating
- **Bypasses**
  - Trapdoor attacks — Utilizing existing flaws
  - Authorization attacks — Password cracking

Types of Computer Misuse (3)

[Neumann and Parker 1989]

- **Active misuse**
  - Basic — Creating false data, modifying data
  - Denials of service — Saturation attacks
- **Passive misuse**
  - Browsing — Making random or selective searches
  - Inference, aggregation — Exploiting traffic analysis
  - Covert channels — Covert data leakage
- **Inactive misuse** — Failing to perform expected duties
- **Indirect misuse** — Breaking crypto keys

The Internet Worm (Nov 2, 1988)

- **Probably the most famous exploit ever unleashed**
- **Program was released that iteratively spread itself across Berkeley Unix systems, and crippled those it infected**
- **Exploited three different vulnerabilities**
  - **`debug`** option of **`sendmail`**
  - **`gets`**, used in the implementation of **`finger`**
  - Remote logins exploiting **`.rhost`** files
- **Perpetrator was convicted under the Computer Fraud and Abuse Act of 1986**
- **Largely the cause for the creation of the Computer Emergency Response Team (CERT)**

A Cautionary Tale

- **Perpetrator was Robert Morris, a Cornell CS graduate student at the time**
- **Morris intended the worm as a "benign" experiment**
  - The worm's propagating behavior was intended
  - The worm's destructive behavior was not
- **Lesson: DO NOT try hacking experiments—even "benign" ones—on public networks**

Basic Security Analysis

- How do you secure X? Is X secure?

1. What are we protecting?
2. Who is the adversary?
3. What are the security requirements?
4. What security approaches are effective?

1. What Are We Protecting?

- **Enumerate assets and their value**
- **Understand architecture of system**
- **Useful questions to ask**
  - What is the operating value, i.e., how much would we lose per day/hour/minute if the resource stopped?
  - What is the replacement cost? How long would it take to replace it?

2. Who Is the Adversary?

- **Identify potential attackers**
  - How motivated are they?
- **Estimate attacker resources**
  - Time and money
- **Estimate number of attackers, probability of attack**

Common (Abstract) Adversaries

- **Attacker action**
  - Passive attacker: eavesdropping
  - Active attacker: eavesdropping + data injection
- **Attacker sophistication**
  - Ranges from script kiddies to government-funded group of professionals
- **Attacker access**
  - External attacker: no knowledge of cryptographic information, no access to resources
  - Internal attacker: complete knowledge of all cryptographic information, complete access
    - Result of system compromise

3. What Are the Security Requirements?

- **Enumerate security requirements**
  - Confidentiality
  - Integrity
  - Authenticity
  - Availability
  - Auditability
  - Access control
  - Privacy
  - …

Secrecy, Confidentiality, Privacy, Anonymity

- **Often considered synonymous, but are slightly different**
- **Secrecy**
  - Keep data hidden
  - E.g., Alice kept the incriminating information secret
- **Confidentiality**
  - Keep (someone else's) data hidden from unauthorized entities
  - E.g., banks keep much account information confidential
- **Privacy**
  - Keep data about a person secret
  - E.g., to protect Alice's privacy, company XYZ did not disclose any personal information
- **Anonymity**
  - Keep identity of a protocol participant secret
  - E.g., to hide her identity from the web server, Alice uses The Onion Router (TOR) to communicate

Integrity, Authentication

- **Sometimes used interchangeably, but different**
- **Data integrity**
  - Ensure data is “correct” (i.e., correct syntax & unchanged)
  - Prevents unauthorized or improper changes
  - E.g., Trent always verifies the integrity of his database after restoring a backup, to ensure that no incorrect records exist
- **Entity authentication or identification**
  - Verify the identity of another protocol participant
  - E.g., Alice authenticates Bob each time they establish a secure connection
- **Data authentication**
  - Ensure that data originates from claimed sender
  - E.g., For every message Bob sends, Alice authenticates it to ensure that it originates from Bob

Temporal Properties

- **Age**
  - Prove that data exists before a certain time
  - Lower bound on the duration of existence
- **Freshness**
  - Prove that data was created after an event
  - Upper bound on the duration of existence
- **Temporal order**
  - Verify ordering of a sequence of events

Other Properties

- **Auditability**
  - Enable forensic activities after intrusions
  - Prevent attacker from erasing or altering logging information
- **Availability**
  - Provide access to resource despite attacks
  - Denial-of-Service (DoS) attacks attempt to prevent availability

4. Approaches to Achieve Security

- **No security**
  - Legal protection (deterrence)
  - Innovative: patent attack, get protection through patent law
- **Build strong security defense**
  - Use cryptographic mechanisms
  - Perimeter defense (firewall), VPN
- **Resilience to attack**
  - Multiple redundant systems (“hot spares”)
- **Detection and recovery (& offense ?)**
  - Intrusion detection system
  - Redundancy, backups, etc.
  - Counterstrike? (Legal issues?)

Threat Models

- **Can't protect against everything**
  - Too expensive
  - Too inconvenient
  - Not worth the effort
- **Identify the most likely ways your system will be attacked**
  - Identify likely attackers and their resources
    - Dumpster diving or rogue nation?
  - Identify consequences of possible attacks
    - Mild embarrassment or bankruptcy?
  - Design security measures accordingly
    - Accept that they will not defend against all attacks

Think Like an Attacker

- **Adversary is targeting *assets,* not defenses**
- **Will try to exploit the *weakest* part of the defenses**
  - E.g., bribe human operator, social engineering, steal (physically) server with data

Case Study

- **Class discussion on security of a house**
  - What are we protecting?
  - Who is the adversary?
  - What are the security requirements?
  - What security approaches are effective?

Computer Security Overview

- **Foundations**
  - Security properties
  - Basic cryptography
  - Security protocols and analysis
- **Security in the real world**
  - Usable security
  - Economics of security
- **Host/software security**
  - Access control
  - Process isolation
  - Trusted Computing Group's Trusted Platform Modules
- **Network security**
  - Key establishment
  - Firewalls
  - Network intrusion detection

Computer Security Overview

Basic building blocks

- **ons**
  - Security properties
  - Basic cryptography
  - Security protocols and analysis
- **Security in the real world**
  - Usable security
  - Economics of security
- **Host/software security**
  - Access control
  - Process isolation
  - Trusted Computing Group's Trusted Platform Modules
- **Network security**
  - Key establishment
  - Firewalls
  - Network intrusion detection

Systems

Why Is Security Hard?

- **We have all these tools…**
- **Practical problems can't be solved by direct application of building blocks**
  - E.g., messages often need padding before they can be encrypted
- **Composing building blocks yields new vulnerabilities**
  - E.g., adversary can interact with valid users in protocol, obtain information that can allow him to impersonate valid user
    - Replay (freshness attacks)
    - Insert (e.g., type flaw attacks, man-in-the-middle attacks)
    - Initiate different protocol sessions (parallel session attacks)

Takeaways

- **Importance of computer security**
- **“Security” is not absolute**
  - Attacker
  - Properties
  - Cost
- **Security is about managing risk**