Cyber Security: The Disconnect (or how we dug ourselves into a deep hole)

Ming Chow
mchow@cs.tufts.edu
Twitter: @0xmchow

Principles of Security

- The CIA triad:
  - Confidentiality
  - Integrity
  - Availability
- Also often included are authenticity and non-repudiation

Important Definitions

- **Event** - Could be anything
- **Incident** - A malicious event
- **Bug** - An error that exists in the implementation-level (i.e. only exist in source code); very correctable
- **Flaw** - An error at a much deeper level, particularly in the design, and likely in the code level; can be very difficult and costly to correct
- **Hacker** - A creative programmer; a positive connotation
- **Cracker** - The bad guy, the attacker, what media coins "hacker" (the negative connotation). We'll use attacker in this class.
- **Black hat** - An attacker with malicious intents
- **White hat** - An attacker with good intents (i.e., the white knight)
- **Gray hat** - An attacker with good and bad intents
- **Script kiddie or skiddie** - Nuisance; not going away any time soon; 1337 wannabes; use scripts and exploits written by others (and do not understand how they really work); always a lamer
- **Vulnerability** - A security bug; a weakness in a system that can potentially be exploited by an attacker
- **Exploiting or exploitation** - The act of taking advantage of a vulnerability
- **Exploit** - Software program that performs the exploiting
- **Risk** - The likelihood that an attacker will take advantage of that vulnerability
- **Threat** - The likelihood that an incident will happen

What Does a Bad Guy Wants

- Information, including intellectual property
- Financials, money
- Administrative access to a remote computer, unauthorized access
- Cause a denial of service (DoS)
- Cause trauma

How to Break into Computer Systems

- Social engineering (low-tech) including plain-old just asking for it
- Taking advantage of exploits in software, starting with reconnaissance (high-tech)

Why Are Attacks Possible?

- The “trinity of trouble” according to my dear colleague Gary McGraw
  - Connectivity
  - Extensibility
  - Complexity

Who to Blame?

- In no particular order:
  - Companies
  - Users
  - Developers
  - Technology itself
  - Media

What Everyone Now Needs: Foundational Knowledge

- *The nature of computers and code, what they can and cannot do*
- How computer hardware works: chips, cpu, memory, disk
- Terminology: Bits, bytes, megabytes, gigabytes, hexadecimals, octals
- How software works: what is a program, what is "running"
- How software is built
- How digital images work
- Computer code: loops and logic
- Big ideas: abstraction, logic, bugs, cost and complexity
- How structured data works
- How the Internet works and the seven network layers
- How the World Wide Web works: content, layout and styling, interactive content, persistent data
- Security, privacy, politics, and ethics
- Digital media, images, sounds, video, compression
- How and where: a good Introduction to Computer Science course, https://www.coursera.org/course/cs101, https://www.edx.org/course/mit/6-00x/introduction-computer-science/586

Why is the Foundational Knowledge Necessary?

- To understand the limitations of computers and code (what they can and cannot do)
- A significant number of security issues are software-related
- There are many information security people who do not know how to code or have the foundational knowledge (that's scary)
- There is far too much misinformation, misconceptions, and FUD (fear-uncertainty-doubt) out there

Still Much More Needs to be Done in These Areas

- Responsible vulnerability disclosure process.
- Information security and secure software development in a Computer Science curriculum.
- Secure software development training and outreach.
- Informing and communicating with others on security-related matters including the general public and government.