Computer Security

Vermont Technical College

CIS-4040

Peter C. Chapin

What We Cover

- This is a fundamentals course
  - Basic concepts of security
  - Important security tools (e. g., cryptography)
  - Important security protocols (e. g., key exchange)
  - Network and host security
- This course is not…
  - … about specific vulnerabilities or attack methods
  - … about specific tools or techniques

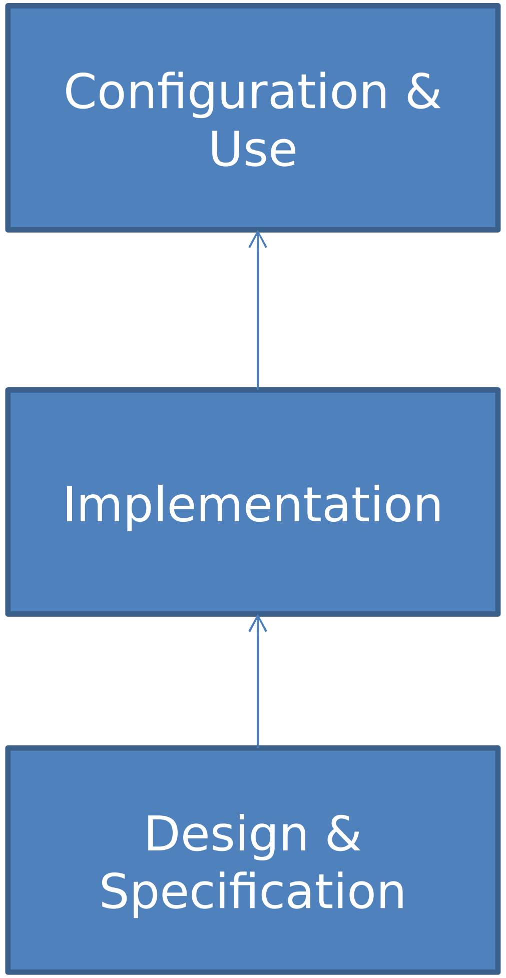
What is Computer Security?

- *A computer system is secure if <u>unexpected behavior</u> cannot occur or is <u>not problematic</u>*.
  - “Unexpected behavior” includes, but is not limited to:
    - Revealing data to unauthorized entities.
    - Letting unauthorized entities modify data.
    - Ordinary software faults.
    - Hardware failure.
  - **Very broad definition**! It overlaps software engineering and system administration.

What is Computer Security?

- *A computer system is secure if it behaves as expected when attacked by an unauthorized intelligence*.
  - Focuses on the issue of **malicious attack** (bad guy)
  - Random errors are not a security problem unless they introduce an exploitable vulnerability
  - **More intuitive definition**, but it overlooks a practical reality: *Data loss is data loss no matter what causes it*.

Security Layers



Configuration & Use

Is the system configured and used properly?
(Administration)

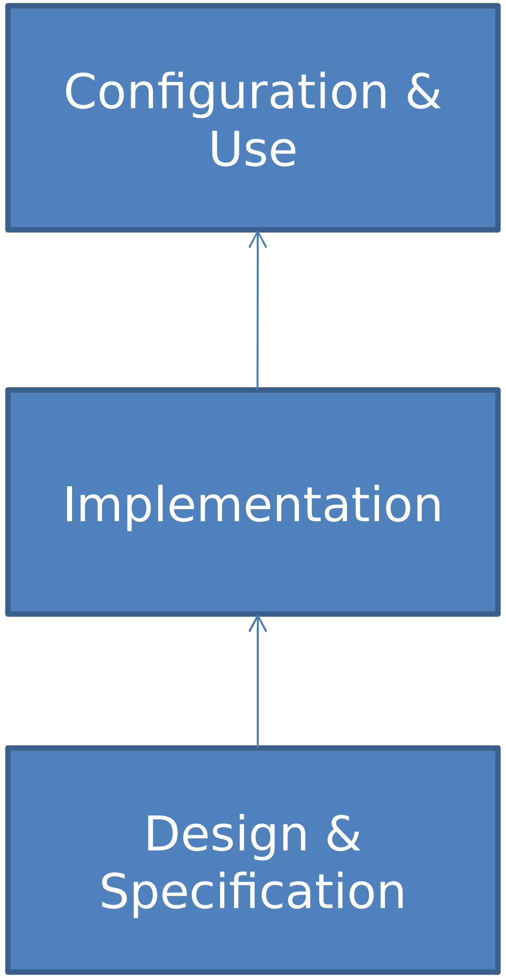
Implementation

Is the system built correctly?
(Engineering)

Design & Specification

Is the system secure in principle?
(Security theory)

Complexity is Bad for Security



Configuration & Use

Complex systems are hard to use.

Implementation

Complex systems are hard to build.

Design & Specification

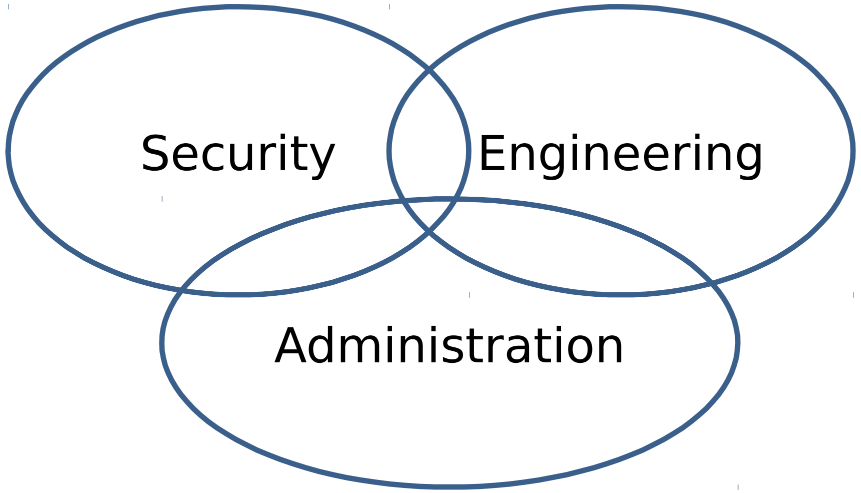
Complex systems are hard to understand.

Examples of Excessive* Complexity

- NTFS (Windows file system) permissions
  - Many complex interacting options
- IPsec (IP Security Protocol)
  - Too many ways of doing essentially the same thing. Too many interacting options.
- Linux `iptables` configuration
  - Like many firewalls, provides a large number of features. How can administrator be sure it’s okay?

* Complexity is “excessive” if the corresponding capability is either unnecessary OR can be obtained in another, simpler way.

Security, Engineering, Administration



- Security + Engineering
  - Design
  - Reliability
- Security + Admin
  - Software configuration
  - Contingency planning
- Engineering + Admin
  - Administrative tools
  - Hardware support

Notes on Terminology

- “Insecure” vs “Unsecure” vs “Unsecured”
  - Insecure is an emotional state. Unsecure is not a word. Unsecured implies no security activated.
- “Hacker”
  - A hacker used to be a good guy. Then hackers became bad. Now they are good again. *Don’t use the term, the meaning is ambiguous.*
- “Adversary” vs “Attacker”
  - Adversary is more neutral.

Alice and Bob

- Security community has traditionally used Alice and Bob instead of A and B.
  - “Alice sends Bob message M…”
  - “A sends B message M…”
- I will continue this tradition.

Security Services

- “Service” in this context is a type of security, not a server
- Q: “Is your system secure?”
  - Wrong answer: “Yes” (or “No”)
  - Right answer: “Secure in what sense?”
- Security is **not** a Boolean attribute.
  - Many possible security services exist.
  - A system might be strong in some ways, weak in others.
  - *Match the security services you use to your needs!*

“Big Two” Security Services

- **Confidentiality**
  - The property of blocking unauthorized users from reading data. (Common tool: Encryption)
- **Data Integrity**
  - The property of blocking unauthorized users from writing data. (Common tool: Digital signatures)

These two services are *duals* of one another. They have an intimate theoretical relationship that we will explore as the course progresses.

Other Security Services

- Authentication
  - The ability to determine the *identity of a principal*
- Authorization
  - Determining *what a principal can do* once authenticated
- Anti-Replay
  - The ability to detect when an old transaction is inappropriately resubmitted for processing
- Sequence Control
  - The ability to detect when the ordering of events has been rearranged
- Availability
  - The ability to continue working despite attempts to shut you down

Example

- *Alice sends Bob packets over the network. Alice encrypts and signs the packets so*…
  - Confidentiality, data integrity, and authentication are provided.
- BUT…
  - Without sequence control an adversary could rearrange the packets
  - Without anti-replay an adversary could send the packets again

Adversary Models

- Passive
  - Adversary only able to look at data, but not touch it. “Observe, but do not interfere.”
- Active
  - Adversary able to modify, insert, remove, reorder data. “Go ahead and interfere.”

It’s important to use the right model when analyzing a security system. **Be realistic**. *No security is sufficient against an attacker with god-like power!*

Dolev-Yau Adversary Model

- Common when analyzing network protocols
  - Adversary can…
    - … read every message everywhere on the network
    - … modify any message anywhere on the network
    - … block, reorder, or replay messages at will
  - Adversary cannot…
    - … break any cryptographic methods in use
    - … access any information on any of the hosts
  - “Attacker carries the message”

“Security Though Obscurity”

- *Always assume adversary has full knowledge of the methods and algorithms used.*
  - They will figure them out eventually
  - Assuming your methods and algorithms remain secret is “security through obscurity.”
    - Can be useful as a barrier…
    - … <u>don’t rely on it!</u>

Summary

- *A computer system is secure if* it behaves in the expected way.
- *Security concerns overlap with* those from engineering and system administration.
- *Security must be applied at all levels*: Design, implementation and deployment.
- *Complexity is bad for security*. Unfortunately feature-hungry users gravitate toward complex systems.
- *Security services define the kinds of security* one might be interested in having.
- Before analyzing security, *understand the assumed capabilities of your adversary*.
- *Always assume your adversary knows your methods*.