Computer Security Introduction

Peter Chapin

Vermont State University

Last Revised: 2025-08-21

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 (mostly)
  - … about specific tools or techniques (mostly)
  - We will use some of the above things to illustrate the concepts

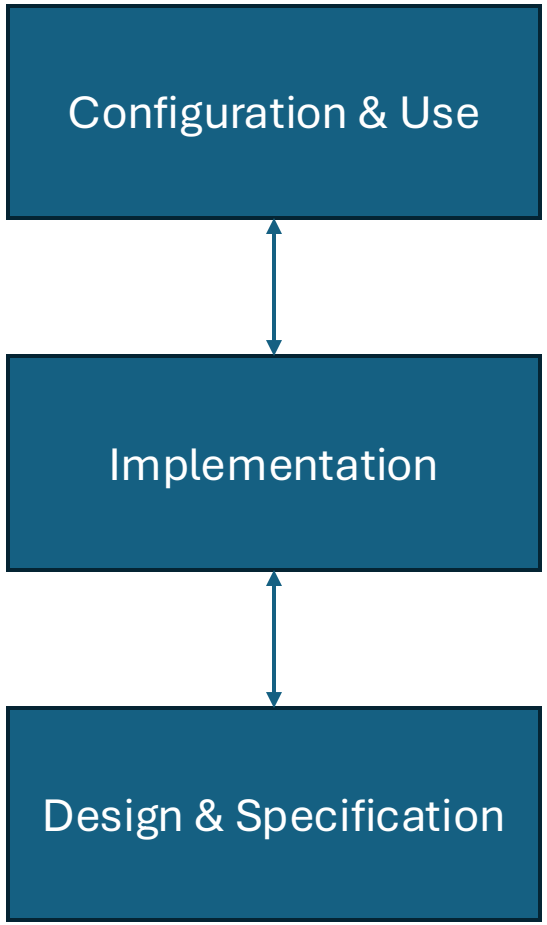
What is Computer Security?

- *A computer system is secure if unexpected behavior cannot occur or is not "problematic."*
  - "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 with software engineering and administration.

What is Computer Security? (Take 2)

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

Security Layers

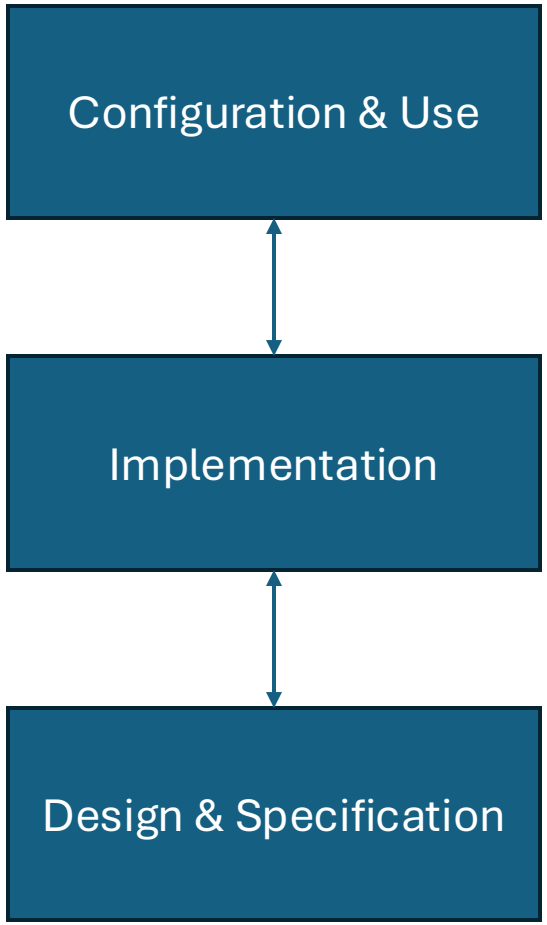


Is the system configured and used correctly?
(Administration)

Is the system implemented correctly?
(Engineering)

Is the system secure in principle?
(Security Theory)

Complexity is Bad for Security



Complex systems are difficult to use.

Complex systems are difficult to build.

Complex systems are difficult to understand.

Examples of Excessive Complexity*

- NTFS (Windows file system) permissions
  - Many complex interacting options
- IPsec (IP security protocol)
  - There are too many ways of doing essentially the same thing. There are too many interaction options.
- Linux `iptables` configuration
  - Like many firewalls, it offers numerous features. How can the administrator be sure everything is okay?

* What constitutes “excessive complexity” is in the eye of the beholder

Notes on Terminology

- Insecure vs Unsecured
  - *Insecure* is a general term: “Due to an insecure database, customer records were stolen.”
  - *Unsecured* is used more specifically: “Table permissions in the database were unsecured, resulting in unauthorized read access.”
- Hacker
  - *Hacker* has flipped between good and bad, and people distinguish between *white-hat* and *black-hat* hackers. It is best to avoid the term.
- Adversary
  - A neutral term without political or moral/ethical judgments. Best choice in a professional context.

Alice and Bob

- The security community has traditionally used Alice and Bob instead of A and B
  - “Alice sends Bob message M…”
  - “A sends B message M…”
- This makes describing security-related operations more relatable and easier to comprehend.
- If necessary, we can introduce Carol (C), Dave (D), Eve (E), etc.
- We will continue this tradition

Security Services

- *Service* in this context is a type of security, not a server
- Question: “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 areas, weak in others
  - *Match the security services you use to your needs!*

The 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 tools: digital signatures)
- These two services are *duals* of one another. They have an intimate theoretical relationship.

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 resubmitted for processing
- Sequence Control
  - The ability to detect that the order 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
  - The adversary is only able to look at the data but not touch it. Observe, but do not interfere.
- Active
  - The adversary can modify, insert, remove, and reorder data.
- It is essential to use the correct model when analyzing a security system
  - **Be realistic**
  - *No security system can protect against an adversary with god-like powers*!

Dolev-Yau (D-Y) Adversary Model

- Commonly used to analyze network protocols
- Adversary can…
  - … read every message everywhere on the network
  - … modify any message anywhere on the network
  - … block, reorder, reroute, or replay messages at will
- Adversary cannot…
  - … defeat any encryption technology used
  - … access any information on the hosts
- *The Attacker carries the message*

Security Through Obscurity

- *Always assume your adversary has complete knowledge of the methods and algorithms you use*
  - They will figure those things out eventually
  - Assuming your methods and algorithms remain secret is *security through obscurity*.
  - It can be helpful as a temporary barrier, but don't rely on it indefinitely!
- Red Flag
  - "We've developed our own network security protocol. Trust us, it's great!"
- Safer
  - "We use standard AES encryption with TLS as described by RFC-XXXX."