Chapter 13: Design Principles

• Overview
• Principles
  – Least Privilege
  – Fail-Safe Defaults
  – Economy of Mechanism
  – Complete Mediation
  – Open Design
  – Separation of Privilege
  – Least Common Mechanism
  – Psychological Acceptability
Overview

• Simplicity
  – Less to go wrong
  – Fewer possible inconsistencies
  – Easy to understand

• Restriction
  – Minimize access
  – Inhibit communication
Least Privilege

• A subject should be given only those privileges necessary to complete its task
  – Function, not identity, controls
  – Rights added as needed, discarded after use
  – Minimal protection domain
Fail-Safe Defaults

- Default action is to deny access
- If action fails, system as secure as when action began
Economy of Mechanism

- Keep it as simple as possible
  - KISS Principle
- Simpler means less can go wrong
  - And when errors occur, they are easier to understand and fix
- Interfaces and interactions
Complete Mediation

- Check every access
- Usually done once, on first action
  - UNIX: access checked on open, not checked thereafter
- If permissions change after, may get unauthorized access
Open Design

• Security should not depend on secrecy of design or implementation
  – Popularly misunderstood to mean that source code should be public
  – “Security through obscurity”
  – Does not apply to information such as passwords or cryptographic keys
Separation of Privilege

- Require multiple conditions to grant privilege
  - Separation of duty
  - Defense in depth
Least Common Mechanism

• Mechanisms should not be shared
  – Information can flow along shared channels
  – Covert channels

• Isolation
  – Virtual machines
  – Sandboxes
Psychological Acceptability

- Security mechanisms should not add to difficulty of accessing resource
  - Hide complexity introduced by security mechanisms
  - Ease of installation, configuration, use
  - Human factors critical here
Key Points

- Principles of secure design underlie all security-related mechanisms

- Require:
  - Good understanding of goal of mechanism and environment in which it is to be used
  - Careful analysis and design
  - Careful implementation