CS 3IS3 Fall 2007

06 Design for Security

William M. Farmer

Department of Computing and Software
McMaster University

11 November 2007

Design for Security

- For an information system to be secure, it must be designed for security.
    - Security is very difficult to add on as an afterthought.
- Due to security's pervasive nature, it is difficult to design for security.
- The principles of good security design are largely the same as the principles of good software design.
- Saltzer and Schroeder (1975) give eight principles for the design and implementation of security mechanisms.

1. Principle of Least Privilege

- The principle of least privilege states that a subject should be given only those privileges that it needs in order to complete its task.
- The need-to-know principle is a special case of this principle.
- As a consequence of this principle, access rights should be revoked when they are no longer needed.
- Most systems do not have the granularity in privileges to apply this principle precisely.

2. Principle of Fail-Safe Faults

- The principle of fail-safe faults states that, unless a subject is given explicit access to an object, it should be denied access to that object.
- According to this principle, the default access to an object should be none.
- As a consequence of this principle, an information system should not be able to fail when it is in its initial state.

3. Principle of Economy of Mechanism

- The principle of economy of mechanism states that security mechanisms should be as simple as possible.
- “Everything should be made as simple as possible, but not one bit simpler.” — Albert Einstein.
- A simple design usually makes everything else simpler: implementation, testing, maintenance, documentation, and application.
- Complexity often leads to errors because crucial assumptions are missed or misunderstood.

4. Principle of Complete Mediation

- The principle of complete mediation requires that all accesses to objects be checked to ensure they are allowed.
- Some trusted system must be the mediator.
- Examples of mediators:
  - Operating system.
  - Type system.
  - Security manager.
- Many systems cache the results of the initial access check so that subsequent checks can be abbreviated.
  - Is this a violation of the principle?

5. Principle of Open Design

- The principle of open design states that the security of a mechanism should not depend on the secrecy of its design or implementation.
- The opposite of this principle is often called security through obscurity.
- Keeping a design or implementation secret does not improve security in practice.
  - Eventually the secret will be revealed or discovered, by accident or intent.
  - Weaknesses in design or implementation may take longer to be discovered by the developers.
  - The approach can lead to a false sense of security.

6. Principle of Separation of Privilege

- The principle of separation of privilege states that a system should not grant permission based on a single condition.
- The principle of separation of duty is a special case of this principle.
- Many Unix systems violate this principle with the `root` account.
  - Some Unix systems do not allow an `su` to the `root` account unless the user is currently in an account in the `wheel` group (with GID 0).

7. Principle of Least Common Mechanism

- The principle of least common mechanism states that mechanisms used to access resources should not be shared.
- Sharing resources provides a communication channel that may not be intended.

8. Principle of Psychological Acceptability

- The principle of psychological acceptability states that security mechanisms should not make the resource more difficult to access than if the security mechanisms were not present.
- If a security mechanism adds an excessive or unreasonable burden then:
  - Administrators will be more likely to make mistakes.
  - Users will be more likely to try to go around the mechanism.
- However, security mechanisms should not unnecessarily reveal information for the sake of user-friendliness.

Security Design Concepts

- A reference monitor is a abstract access control machine that mediates all accesses to objects by subjects.
- A reference validation mechanisms (RVM) is an implementation of a reference monitor.
- Requirements of an RVM:
  1. Tamper proof.
  2. Complete: Invoked in all accesses to objects.
  3. Simple: Small enough to be adequately analyzed.
- A security kernel is a small, self-contained part of an information system that implements a security monitor.
  - It can include both hardware and software.
  - It is often a module in the operating system.

Trusted Computing Base

- A trusted computing base (TCB) is the collection of all the security mechanisms in an information system that are responsible for enforcing a security policy.
  - Can include hardware, firmware, and software.
  - Is a generalization of a security kernel.
- Requirements of a TCB:
  1. Satisfies its target security policy.
  2. Protects itself, especially its software components.
  3. Small as possible.