Computer Security Overview

Reading: Ch. 1, Van Oorschot

Furkan Alaca

University of Toronto Mississauga

CSC347H5, Fall 2018

Today's Objectives

- CSC347 overview and objectives
- Logistics
- Computer security goals
- Threat modelling
- Security design principles

What is Computer Security?

- **Computer security** is the practice of protecting computer-related assets from unauthorized access and its consequences
  - **Prevent** unauthorized actions
  - If that fails, **detect** them and **recover** from them

- The security of a system can only be evaluated within a given **context**.

- Consider the following fundamental questions:
  1. What assets do we need to protect?
  2. How are those assets threatened?
  3. What can we do to counter those threats?

Definition of Computer Security

"Measures and controls that ensure **confidentiality**, **integrity**, and **availability** of information system assets including hardware, software, firmware, and information being processed, stored, and communicated."

(NISTIR 7298 Rev. 2)

Computer Security Objectives

Confidentiality

Data confidentiality: Assures that confidential information is not disclosed to unauthorized entities.

Privacy: Gives individuals control over what information related to them may be collected and stored, and by whom and to whom that information may be disclosed.

Integrity

Data integrity: Assures that information and programs are changed only in a specified and authorized manner.

System integrity: Assures that a system functions as intended, free from any unauthorized manipulation.

Availability

Ensures timely and reliable access to services by authorized users.

Computer Security Objectives (2)

Authenticity

The property of being verifiably genuine. Allows one to be confident in the validity of a transmission, message (could be data or a program), or message originator. Implies that each entity's claimed identity is verified and that each input arriving at the system came from a trusted source.

Accountability

The quality where actions of an entity are uniquely traceable to that entity. Requires the system to keep tamper-proof records of all transactions to permit forensic analysis in the event of a security breach or transaction dispute. This supports **nonrepudiation**, which is the property whereby an entity cannot credibly deny an action which they performed.

Attack Surfaces

Terminology

Attack: A deliberate action that (if successful) causes a security violation.

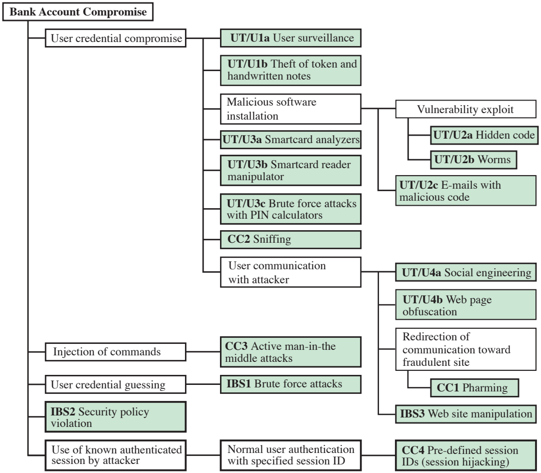
Vulnerability: A flaw or weakness in a system's design, implementation, or operation and management that could be **exploited** (i.e., in an **attack**) to violate the system's security policy.

Attack surface: Set of reachable and exploitable vulnerabilities in a system.

Three broad categories of attack surfaces:

- **Network attack surface:** Vulnerabilities over an enterprise network, wide-area network, or the Internet
- **Software attack surface:** Vulnerabilities in application, utility, or operating system code
- **Human attack surface:** Vulnerabilities created by personnel or outsiders, such as social engineering, human error, and trusted insiders

Attack Trees



- **Bank Account Compromise**
  - User credential compromise
    - **UT/U1a** User surveillance
    - **UT/U1b** Theft of token and handwritten notes
    - Malicious software installation
      - Vulnerability exploit
        - **UT/U2a** Hidden code
        - **UT/U2b** Worms
      - **UT/U2c** E-mails with malicious code
    - **UT/U3a** Smartcard analyzers
    - **UT/U3b** Smartcard reader manipulator
    - **UT/U3c** Brute force attacks with PIN calculators
    - **CC2** Sniffing
    - User communication with attacker
      - **UT/U4a** Social engineering
      - **UT/U4b** Web page obfuscation
      - Redirection of communication toward fraudulent site
        - **CC1** Pharming
      - **IBS3** Web site manipulation
  - Injection of commands
    - **CC3** Active man-in-the middle attacks
  - User credential guessing
    - **IBS1** Brute force attacks
  - **IBS2** Security policy violation
  - Use of known authenticated session by attacker
    - Normal user authentication with specified session ID
      - **CC4** Pre-defined session IDs (session hijacking)

Figure 1.4 An Attack Tree for Internet Banking Authentication

Figure source: Computer Security Principles and Practice 3e, Stallings & Brown

Analysis considers three components involved in the authentication process:

- **UT/U**: User terminal and user
- **CC**: Communication channel
- **IBS**: Internet banking server

Five overall attack strategies:

- User credential compromise
- Injection of commands
- User credential guessing
- Security policy violation
- Use of known authenticated session

Attack Trees (cont'd)

- When designing and implementing security countermeasures, you should understand your **threat model**
  - What type of attackers are you trying to defend against? What are their motivations and goals, and what type of attack strategies are they likely to employ?
  - Failure to do so can result in disaster, e.g., file access permissions will not offer protection against an attacker with physical access to the machine who can boot into a live OS from a USB key
- Branches in the attack tree can be assigned values, e.g. complexity of attack, cost to defend against
- Attack trees are a useful method to represent knowledge of known attacks and strategically allocate resources to implementing countermeasures
  - You might not have the resources to defend against some relatively advanced threats, e.g. electromagnetic or acoustic emissions from computer equipment, but the government does consider these threats: see TEMPEST

Security Design Principles (1)

Economy of mechanism

Keep the design and implementation **simple**. This facilitates analysis and verification, resulting in fewer vulnerabilities.

Fail-safe defaults

The default configuration of a system should be **safe**.

Complete mediation

Each access to every resource should be checked to ensure that it is allowed (or **authorized**).

Open design

While cryptographic keys must be kept secret, the design of a security mechanism or algorithm should be open, to allow public scrutiny. **Security by obscurity** (reliance on a secret design) can be dangerous.

Security Design Principles (2)

Separation of privilege

Require multiple conditions or privilege attributes for access to a restricted resource.

Least privilege

Every entity (e.g., user, process) should operate using the least set of privileges necessary to perform its tasks.

Least common mechanism

Minimize reliance on sharing resources (e.g., shared state, libraries) between users.

Psychological acceptability

Security mechanisms should not interfere unduly with the work of users – otherwise, users may circumvent those mechanisms.

Security Design Principles (3)

Work factor*

For security mechanisms susceptible to direct work-factor calculation, design so that the cost to defeat the mechanism safely exceeds resources of expected attackers.

Compromise recording*

Reliably record that a compromise of information has occurred.

*More relevant to physical security, but may still apply (albeit imperfectly) to computer security.

Security Design Principles (4)

Isolation

Preventing disclosure or tampering by isolating:
1) Public-access systems from critical resources.
2) Users' processes and files from one another.
3) Security mechanisms from other parts of the system.

Defense in depth

Use multiple, overlapping protection approaches so that the failure of any individual protection approach will not leave the system unprotected.

Least surprise

The program of user interface should always respond in the way that is least likely to surprise the user.

Computer Security Challenges

1. Computer security is not as simple as it might first appear to the novice.
2. Potential attacks on the security features themselves must be considered.
3. Procedures used to provide particular services are often counterintuitive.
4. Physical and logical placement of security mechanisms needs to be determined.
5. The task of developing a security mechanism can be complicated by reliance on other protocols.
6. Attackers only need to find a single weakness, the developer needs to find all weaknesses.
7. Users and system managers tend to not see the benefits of security until a failure occurs.
8. Security requires regular and constant monitoring.
9. Security is often an afterthought to be incorporated into a system after the design is complete.
10. Security is thought of as an impediment to efficiency and user-friendliness.

Source: Lawrie Brown

Summary

- Computer security goals: Confidentiality, Integrity, Availability
- Fundamental security design principles
  - We will refer back to these principles
- Attack surfaces, attack trees, and threat modelling
  - Always know who/what you are trying to defend against
- Computer security challenges