# Unit 6: Protection and Security

## 6.1. The Security Problem

# The Security Problem

- System is secure if its resources are utilized and access is as intended under all circumstances
- Security violations:
  - Unauthorized reading of data (theft of information)
  - Unauthorized modification of data
  - Unauthorized destruction of data
- Security measures:
  - Physical
  - User authorization
- Weakness at high-level security may circumvent low-level (operating system) measures

# Authentication

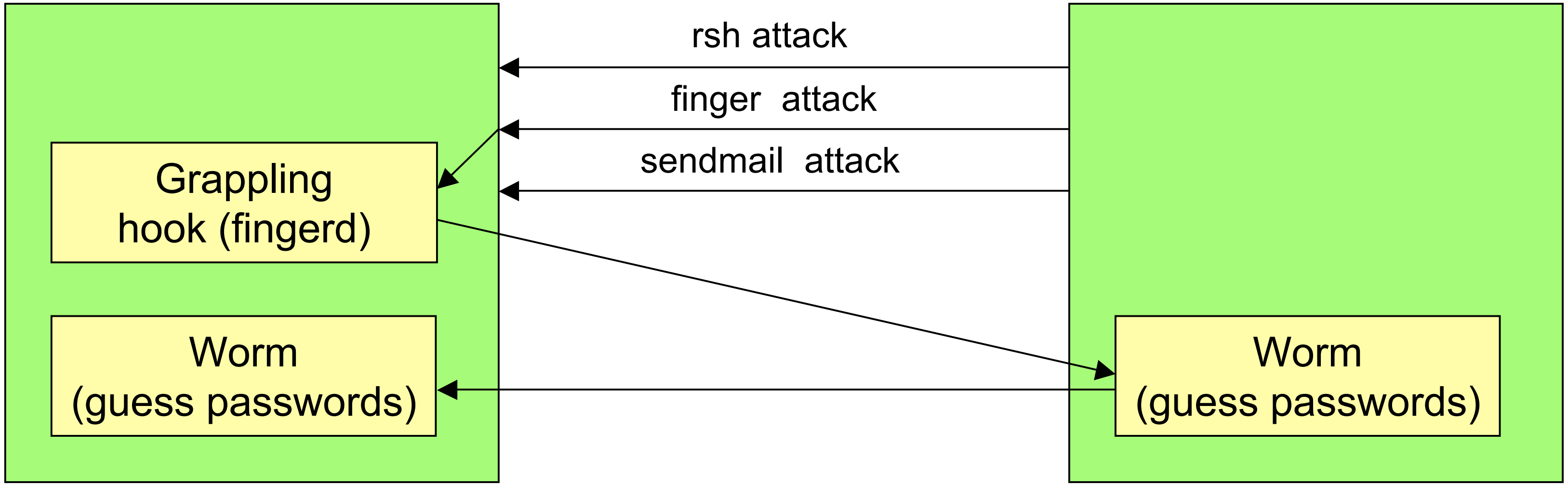
- Username/password
  - Special case of keys/capabilities
  - System generated vs. User generated passwords (hard to remember/easy to guess)
  - Paired passwords: system selects one/user responds appropriately
- How to store passwords securely:
  - one-way functions
  - Shadow passwords

# Program Threats

- Trojan horses
  - User programs executed by other users
  - Where to place the . in search path (current dir) ?
- Trap door
  - Self-reproducing code / self-modifying programs (ACM article by Ken Thompson)
  - Code may check for specific user identifier
  - Compiler may check for specific executable (login)
  - Compiler may check name of source file and reproduce the trap door in new compiler executables

  -> even compiler sources would show no evidence

# System Threats



- Worms & Viruses
  - Nov. 2, 1988, Robert Tappan Morris, Jr., Cornell grad. stud.
  - Overwrite buffer in finger daemon and execute buffer data
    - Create /bin/sh as root
  - Use debug switch in sendmail to execute programs remotely

# Threat Monitoring & Encryption

- System checks for suspicious patterns of activity
- Audit log: time/user/access type for all system objects
- Scan system periodically
  - Short / easy to guess passwords
  - Unauthorized set-UID programs
  - Unauthorized programs in system directories
  - Unexpected long running processes
  - Improper directory protections, dangerous entries in search path
  - Changes to system programs – keep checksums of system programs
- $D_k(E_k(m)) = m$ -- Data Encryption Standard (DES)
  - $D_k$ and $E_k$ can be computed efficiently
  - Security depends only on secrecy of the key, not on secrecy of algorithms E and D – *key distribution problem*