CS431

Computer and Network Security

Introduction

Abhishek Bichhawat

09/01/2024

Course Overview

- Introduction to Security and Foundations
- Cryptography
- Software Security
- Web Security
- Network Security
- Crypto Applications
- Human Factors in Security

Course Details

- Reference Books
  - Cryptography and Network Security by W. Stallings
  - Security Engineering: A Guide to Building Dependable Systems, by Ross J. Anderson
  - Introduction to Computer Security, by Matt Bishop
  - https://textbook.cs161.org
- Lecture-slides will be online
- Exams on material covered in lectures
  - During the two examination slots
  - + some quizzes during lecture hours informed a week in advance

Course Details

- Assignments will involve hands-on exercises
  - 6, maybe 7, depending on available time
  - Due dates will not change
  - Expect you to learn & search for resources on your own to resolve issues
  - Defenses and/or attacks on fairly real systems.
  - Not a lot of coding, but lots of non-standard thinking.
    - Poke into obscure corners of x86 asm, C, Python, Javascript, ..
    - Also true for lectures! We are looking for loopholes everywhere…

Tentative Course Grading

- Assignments - 50%
- Exams/Quizzes - 40%
- Find a REAL exploit - 10%

Course Assignment

- Will be released on Canvas
  - Submission will include a PDF document
- Should be done individually unless mentioned
  - May discuss assignments but solutions must be your own
- Detailed instructions on how to do the practical exercises will be provided as the assignments are released
  - Will include CTF for majority of the exercises
- No grace hours
  - After deadline, 1% point deducted every hour up to 75%
- May have “random” viva to understand your assignments
- *Ethics* - do not break into any system w/o permission of the system’s owner

What is Security?

- “Building systems to remain dependable in the face of malice, error or mischance” (Ross Anderson)
- “Ensuring systems operate properly and remain secure from outside intrusion” (US Air Force)
- “The state or process of protecting and recovering networks, devices and programs from any type of cyberattack.” (Norton)
- “A set of techniques used to protect the integrity of an organization’s security architecture and safeguard its data against attack, damage or unauthorized access.” (Palo Alto Networks)

Security

- Security may be defined as the enforcement of a desired property (normally, assuming an attacker is present)
  - Confidentiality
  - Privacy
  - Integrity
  - Availability
- Why secure?
- Everything can be compromised
  - A Casino Gets Hacked Through a Fish-Tank Thermometer
  - No system is built secure

Building Secure Systems Is Hard

- **Example**: Your grade files stored on IMS
  - **Policy**: only TAs should be able to read and write the grades file.
  - Easy to implement the *positive* aspect of the policy:
    - There just has to be one code path that allows a TA to get at the file.
  - But security is a *negative* goal:
    - We want no tricky way for a non-TA to get at the file.
  - There are a huge number of potential attacks to consider!
    - Exploit a bug in the server's code.
    - Guess a TA's password.
    - Steal a TA's laptop, maybe it has a local copy of the grades file.
    - Intercept grades when they are sent over the network to the registrar.
    - Break the cryptographic scheme used to encrypt grades over the network.
    - Trick the TA's computer into encrypting grades with the attacker's key.
    - Get a job in the registrar's office, or as a TA.

Approaching Security

- Establish social norms
- Establish legal rules
- Make it “uneconomical”
- Establish defense


Master

“Security is a process, not a product”!

- Need to engineer security into a system
- During system design yields the best results
  - As more functionalities are added to the system, it becomes difficult to search and fix security bugs
  - Unfixed bugs lead to huge losses
- Security is not absolute
  - Cannot have full security
  - Tradeoff between attacker strength, cost and required security!

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary

"An impressive résumé, General, but remember—department-store security is different from national security."

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
    - Make it hard enough for an adversary so that the adversary spends their energy elsewhere
    - Build a model of the adversary, their motives and their capabilities
    - Who can attack you and why?

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary (assumptions on the adversary)
    - Knows general information about systems and can interact
    - May try a brute-force approach
    - May collude to perform complex attacks
    - Has the resources required for the attack
    - Can obtain privileges

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
    - Limit the **trusted computing base (TCB)**
    - TCB are the components of a system that you trust
    - Security of a system is built on top of the TCB
    - Determine what components should be in the TCB
      - Should be secure and “unhackable”
      - “Old”-code is the most vulnerable
    - KISS principle

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
    - Simply put, the cost of defense should be lesser than the cost of resource being secured
    - Security is directly proportional to costs
      - Example : Safes and lockers

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent

Building Secure Systems

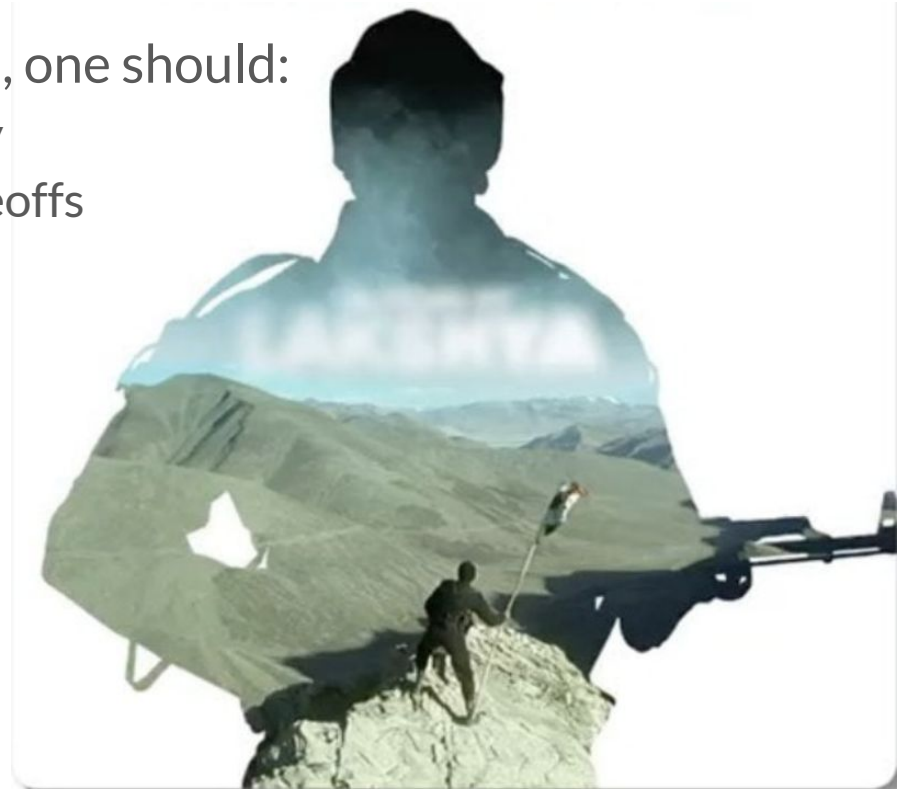
- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
    - Build systems to deter or prevent all attacks
    - If we can't stop an attack, we should be able to detect (and recover)
    - Prepare for the worst.
    - Do not rely on prevention; recovery is as important

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense



Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense
    - Multiple types of defenses should be layered together
      - Attacker should have to breach all defenses
    - Remember the costs

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense
  - Provide least privilege

Building Secure Systems

Building Secure Systems

- For building secure systems, one should:
  - Understand the adversary
  - Understand the cost tradeoffs
  - Detect, if not prevent
  - Design in-depth defense
  - Provide least privilege
    - Do not grant unnecessary permissions
    - Devise methods to control access