ILLINOIS TECH

College of Computing

Introduction to Software Security

Yue Duan

CS 527: Software Security

- Instructor:
  - Yue Duan, Assistant Professor
    - https://yueduan.github.io/
    - yduan12@iit.edu
  - PhD in Computer Science from UC Riverside (2019)
  - Postdoctoral training at Cornell University and University of Utah
  - Specialized in Computer Security, software engineering, AI security and blockchain
- TA:
  - Sajad Meisami, second-year PhD student
  - smeisami@hawk.iit.edu
- Course webpage:
  - https://yueduan.github.io/cs527.html

CS 527: Software Security

- Course overview
  - Binary analysis: code search, malware analysis, vulnerability detection, etc
  - Mobile security: Android app analysis, Android framework analysis
  - Program testing: most effective way to find bugs
  - IoT security: firmware analysis
  - Blockchain security: smart contract analysis
- Textbook
  - No textbook needed
  - Focus on research papers from top venues in computer security

CS 527: Software Security

- Prerequisite
  - Basic knowledge about OS and compiler
  - Programming skills
  - No prior security knowledge required
- Goal
  - Learn basic concepts in software security
  - Obtain hands-on experience with state-of-the-art analysis techniques
  - Develop the ability for analyzing and solving real-world security problems
  - Gain interest to conduct further research in this exciting field

CS 527: Software Security

- Course format and gradings
  - Paper presentation: 10%
  - Paper review: 10%
  - Labs: 50%
    - Lab1: Static Analysis on Android Applications (25%)
    - Lab2: Symbolic Execution (25%)
    - Lab3 (optional): Blockchain Smart Contract Analysis (15% bonus)
  - Final exam: 30%
  - Class participation: 5% bonus

CS 527: Software Security

- Paper presentation
  - Each student needs to present one paper in the class
    - 10-15 min presentation
      - Hint: google the slides of the paper. You may find it but don't directly use it
    - Lead the discussion
      - 5 - 10 mins
      - What are the pros and cons?
      - Why the authors do research the way it is?
      - Any thought for improvement?

CS 527: Software Security

- Paper review
  - Each student needs to write one review for papers from the reading list
    - At least 300 words
    - Summarize the paper
      - Content: What's this paper about?
      - Motivation: Why do the authors want to conduct this research?
      - Contribution: How is the paper different from its peers?
      - Technique: How do the authors achieve their goal?
      - Evaluation: How is the work evaluated?
    - Read critically:
      - You should not assume that the authors are always correct. Instead, be suspicious
      - Any limitations?

CS 527: Software Security

- Labs
  - Students can
    - either form groups (no more than 2 students)
    - or work individually
  - Need to demonstrate your code to the TA and submit a report
    - with well described contributions for each team member (in case of teamwork)
  - 3 labs
    - lab1: static analysis on Android applications (9/15 - 10/14)
      - use a static analyzer Soot to:
        - generate control-flow graph of a given Android app
        - collect Android API usage information
    - lab2: symbolic execution (10/14 - 11/11)
      - use Angr to automatically find a vulnerability in a given binary
    - lab3: smart contract analysis (optional) (11/11 - 12/02)

CS 527: Software Security

- Tentative course schedule
  - **8.23 - 9.15** introductions to different topics
  - **8.23** start looking for collaborator if you decide to work as a group
  - **9.15** start working on project topics
  - **9.20 - 10.4** binary analysis
  - **10.6 - 10.11** mobile security
  - **10.13 - 10.20** program testing
  - **10.25 - 10.27** blockchain security
  - **11.1 - 11.3** Iot security
  - **11.1 - 11.29** paper presentation
  - **12.5 - 12.10** final exam

What is Software Security?

- From traditional PCs, mobile devices to IoT devices, software is literally ubiquitous in our everyday life.

What is Software Security?

- Protecting software is essential for us.
  - Huge impact
  - Malicious software is designed to cause damages
  - Normal software can and **will** contain vulnerabilities
    - Microsoft Applications: 10 - 20 defects per 1000 LOC during in-house testing
    - Industry Average: about 15 - 50 errors per 1000 LOC

What is Software Security?

- Heartbleed vulnerability
  - In popular OpenSSL library
  - Result in potential private keys leakage

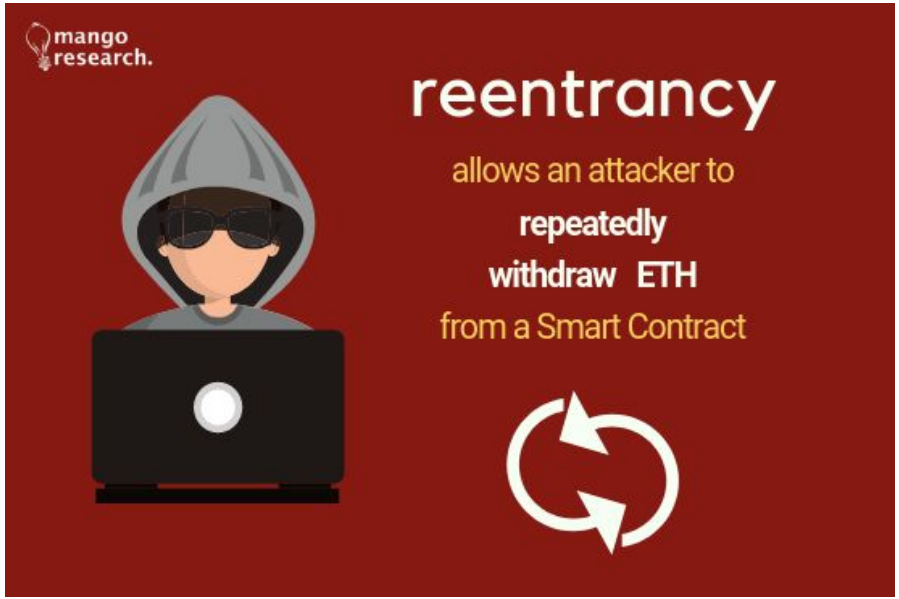
Reference: The Heartbleed Bug, explained
https://www.vox.com/2014/6/19/18076318/heartbleed

What is Software Security?

- Marriott Data Breach 2020
  - On March 31st, 2020, Marriott disclosed a security breach that impacted the data of more than 5.2 million hotel guests who used their company's loyalty application.

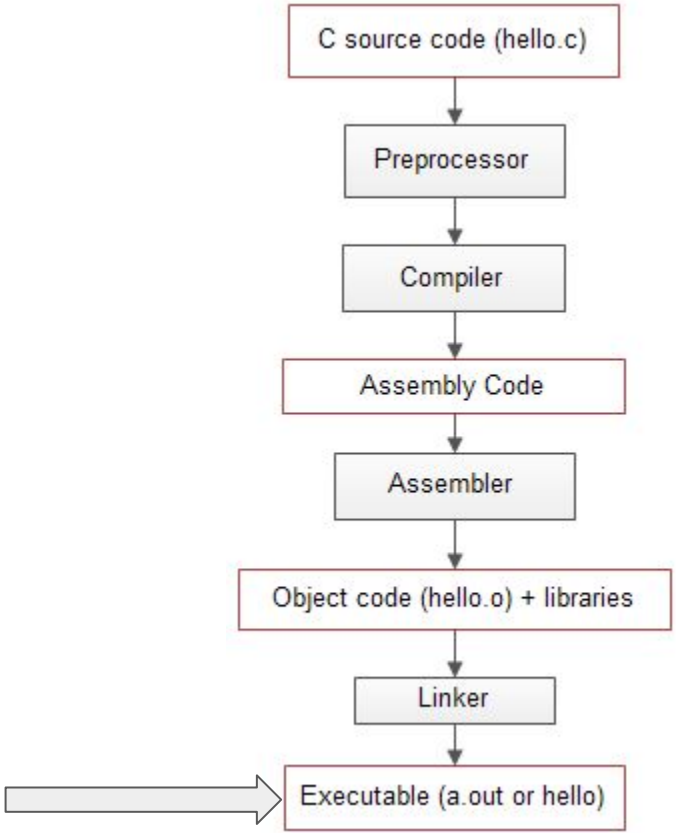
What is Software Security?

- The DAO attack
  - On 16 June 2016, the attacker managed to retrieve approximately 3.6 million Ether (1 Either = 410 USD) from the DAO fund abusing this loophole.



Topics covered

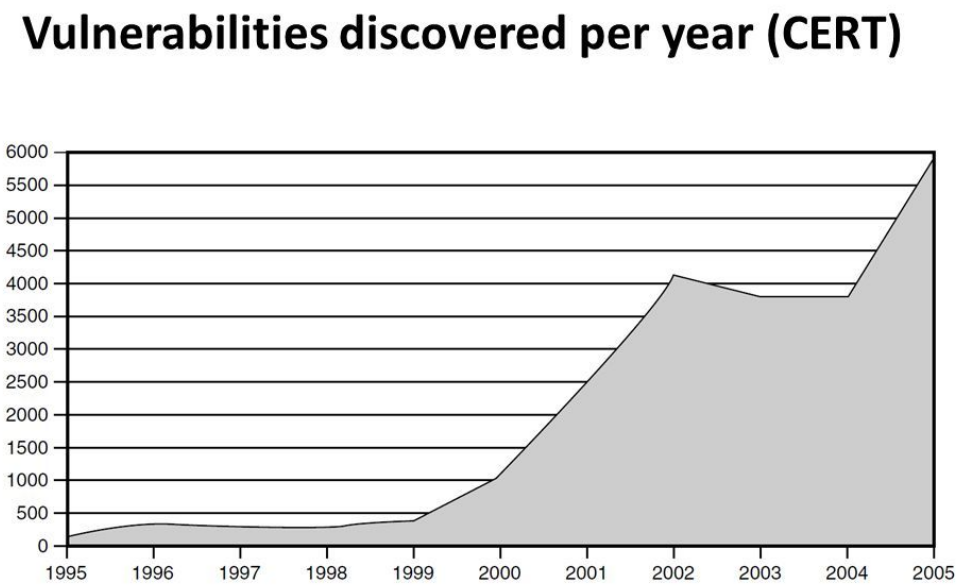
- Binary analysis



Binaries
No source code
Maybe no debug symbol

Topics covered

- Binary analysis
  - Common vulnerabilities
    - Buffer overflow
    - Format string
    - Integer overflow
    - Race condition
    - Dangling pointer
    - etc
  - Malware analysis
  - Defense mechanisms

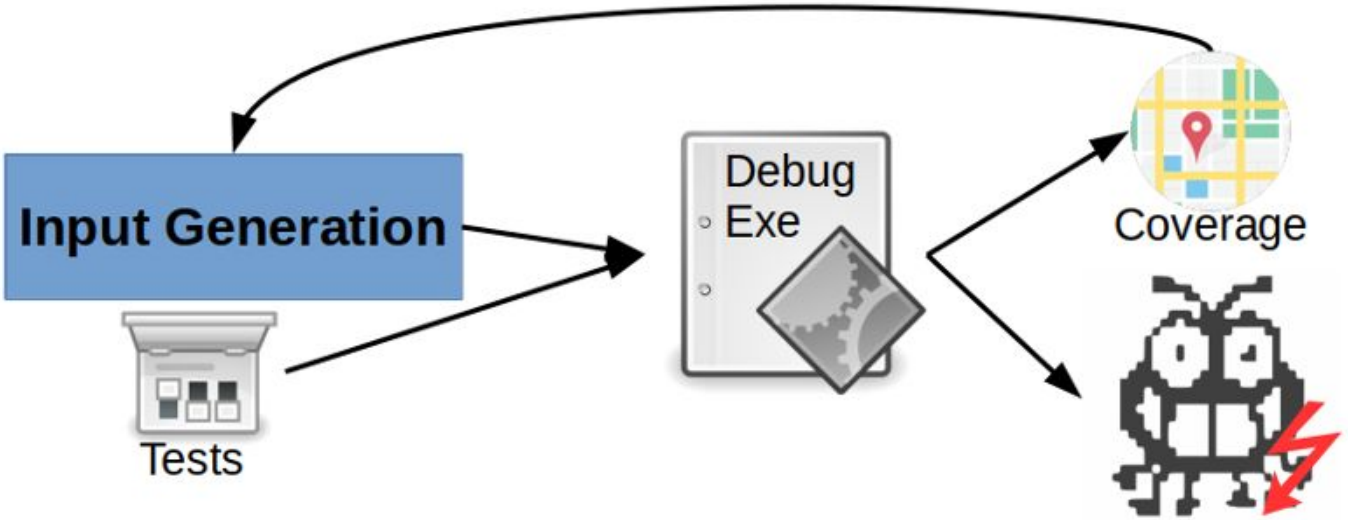


Topics covered

- Mobile Security
  - Is your phone secure?
    - Mobile system analysis
  - Are the apps on your phone secure?
    - Mobile app analysis
  - If no, how to fix?
    - System and app patching

Topics covered

- Program testing
  - Part of binary analysis
  - Dynamic approaches to detect vulnerabilities
  - Fuzzing, symbolic execution, hybrid approaches



Topics covered

- IoT Security
  - smart watch, smart TV, smart router, self-driving car, etc
  - Are they secure?
  - How are they different from traditional binary and mobile?

Topics covered

- Blockchain security
  - Smart contracts
    - piece of software running on blockchain
  - Attacks and vulnerabilities
  - Anonymity

Question?