Carnegie Mellon

LG Software Security Class

Introduction

Session Outline

- Software Security Engineering
- Program Overview
- Program Faculty
- CMU Overview
- Program Structure
- Project
- Mentors
- Course Policies

Instructors – 1

- Jeffrey Gennari
- Senior Member of the Technical Staff at CMU Software Engineering Institute
  - 10+ years experience as software engineer and security analyst
    - CERT Program
    - Areas of expertise include
      - Program analysis and reverse engineering
      - Quality assurance
      - Vulnerability analysis
      - Secure coding
  - Graduate of CMU's Software Engineering Program

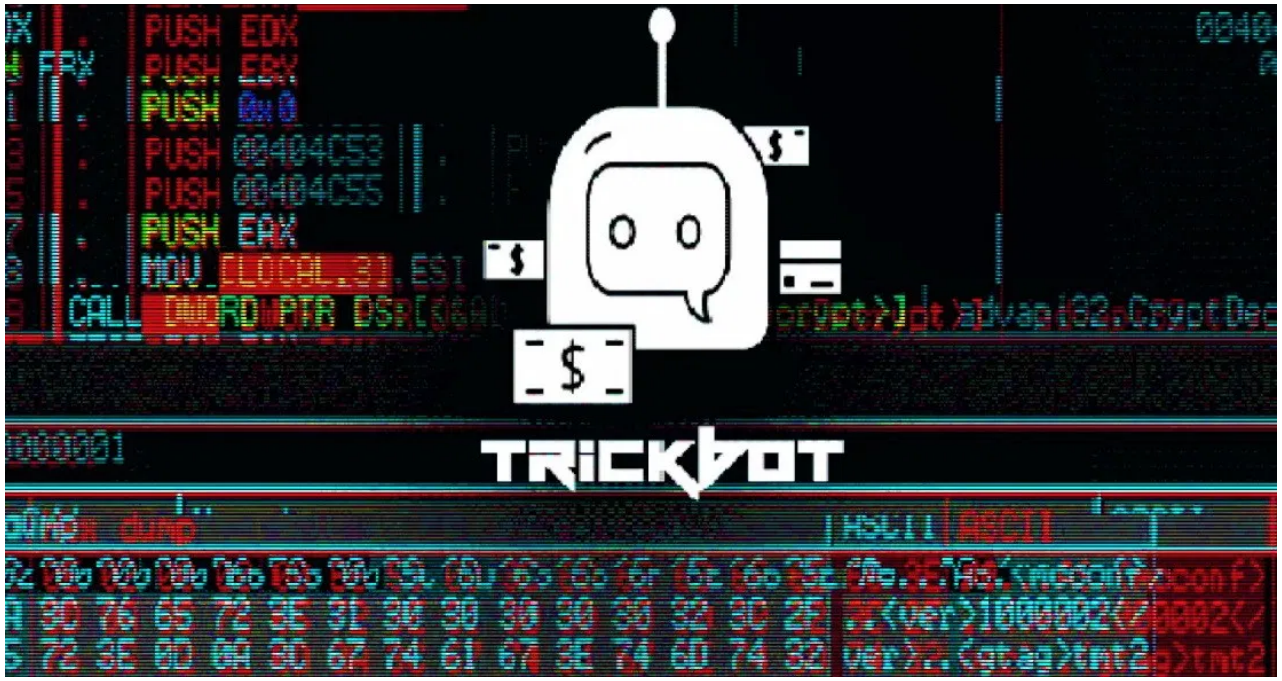
Need for Security

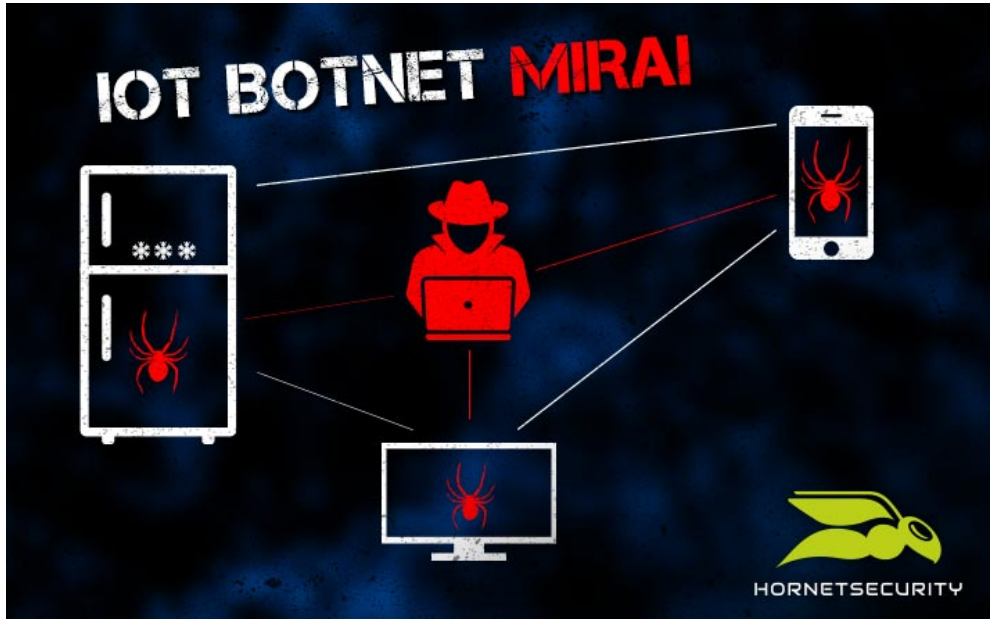
solarwinds

SPECTRE

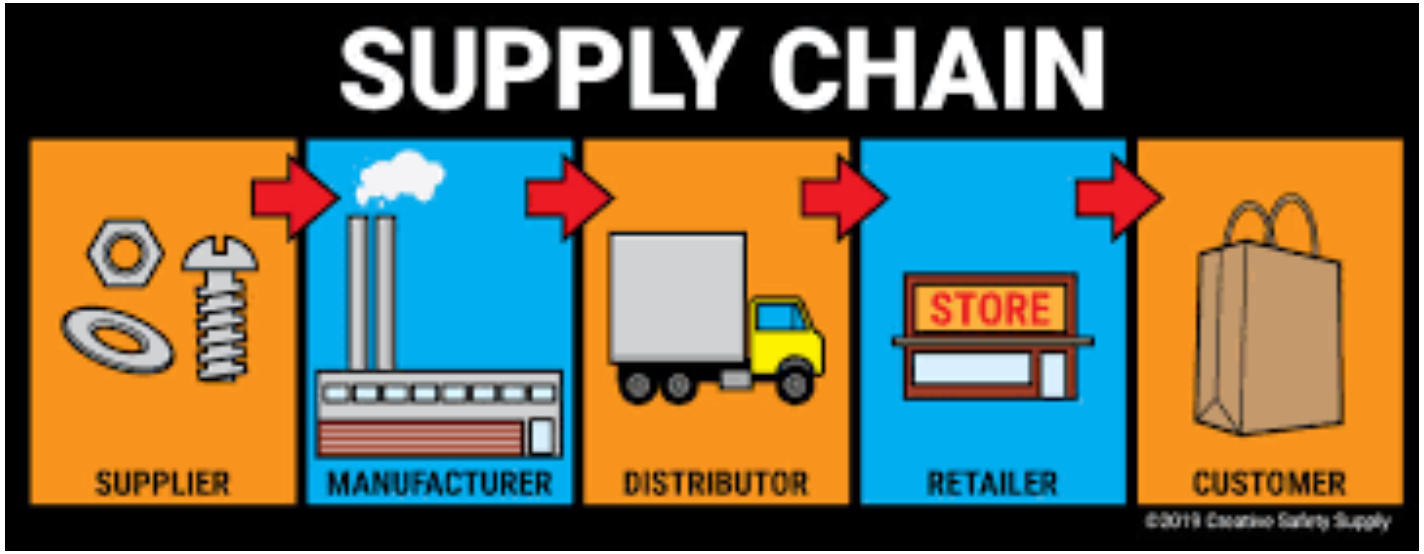
Need for Security

Need for Security





Need for Security



Insider Threat

https://www.eweek.com/security/research-half-of-enterprises-suffered-insider-attacks-in-last-12-months

Software Security

- Security (or lack thereof) can impact all phases of software development
- Software engineers
  - Must account for threats during all phases of product development
  - Requirements, design, construction, transition
- Software analysts
  - Search for and evaluate issues that can impact security
    - For both altruistic and malicious reasons
- Managers
  - Must have plans to respond to security concerns

Security and your system

- Software systems are rarely created in a vacuum
  - We use off-the-shelf components
  - We copy code ☺
  - We interact with archaic, legacy systems
  - Our systems often outlive their intended environments and purposes
  - We extend and adapt systems with varying degrees of rigor
  - Our users continually surprise us

The Definition of Security is not straight forward

- The ability of a system to resist attack
- The ability of a system to recover from an attack
- The ability of a system to detect an attack
- Damage assessment
- Retribution (controversial)
- Compliance?

Thought exercise: Denial of Service Attack?

Thought exercise: Denial of Service Attack?

- Fundamentally a performance issue
  - But different than a legitimate traffic surge
  - Should we tolerate the surge in traffic or attempt to stop it?

Thought exercise: Denial of Service Attack?

- Fundamentally a performance issue
  - But different than a legitimate traffic surge
  - Should we tolerate the surge in traffic or attempt to stop it?
- Also a reliability and robustness concern
  - Our system may need to continue operation

Thought exercise: Denial of Service Attack?

- Fundamentally a performance issue
  - But different than a legitimate traffic surge
  - Should we tolerate the surge in traffic or attempt to stop it?
- Also a reliability and robustness concern
  - Our system may need to continue operation
- What about detection, situational awareness and attribution?
  - Should we contract an external consultancy to monitor our systems or build our own capability?
  - What if it happens again?

Thought exercise: Denial of Service Attack?

- The specifics of the attack matter in response
  - Pure volume based attack
  - Manipulation of protocol or software vulnerability
  - Application-layer attack
- Intent matters
  - Why are we being targeted?

How we will operate

- Our focus will be on practice
- We will complete exercises meant to simulate real-world security scenarios
  - From various perspectives
- The course project will be our capstone
  - Gain realistic experience building and assessing the security of a software system

Course Topics

- From the developer perspective
  - Security requirements
  - Secure design
  - Secure coding
  - Secure development operations
  - Security review, testing and analysis
- From the analyst perspective
  - Vulnerability analysis
  - Malware analysis
  - Penetration testing
- From the manager perspective
  - Planning
  - Incident response
  - Personnel management

Instructors – 2

- Jonathan Woytek

Instructors – 3

- Professor David Belasco
  - Senior Researcher Staff at CMU Software Engineering Institute
  - Research scientist and Assistant Professor in CMU School of Computer Science
  - Expert in program analysis and verification
    - Focus on the development of tools and techniques that enable the verification and certification of the safety, security and reliability of component-based software systems

Instructors – 4

- Dan Plakosh

History of CMU

- Carnegie Institute of Technology (Carnegie Tech) was founded by Andrew Carnegie in 1900
  - Created the school to support his needs for chemical, electrical, and mechanical engineers
  - Focused on finding real solutions to the problems facing society
- The Mellon Institute for Industrial Research was founded in 1913 by Andrew and Richard Mellon
  - Conducted research for firms on a contractual basis

Richard Mellon

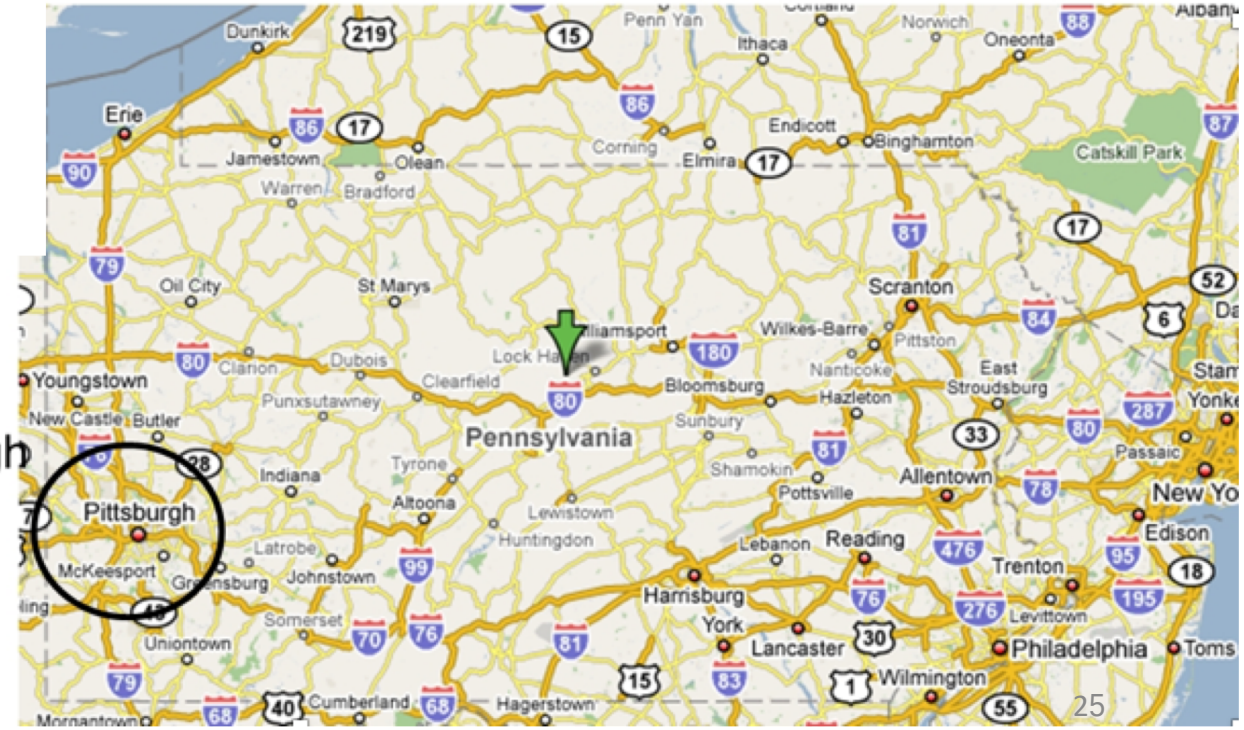
Carnegie Tech and the **Mellon Institute** merged in 1967 to become **Carnegie Mellon University**

Andrew Carnegie

CMU Location

United States

Pittsburgh



Software Security at CMU

- CMU is a very exciting place to be if you are a software engineer
  - Department of Computer Science founded in ‘65 as one of the first such programs in the world
  - CERT program at Software Engineering Institute

The CERT Division is the birthplace of cybersecurity. For nearly 30 years, the CERT Division of the SEI has partnered with government, industry, law enforcement, and academia to advance cybersecurity and improve the security and resilience of computer systems and networks.

Originally focused on incident response, we have expanded into cybersecurity areas such as network situational awareness, malicious code analysis, secure coding, resilience management, insider threats, digital investigations and intelligence, workforce development, DevOps, forensics, software assurance, vulnerability discovery and analysis, and risk management.

https://www.sei.cmu.edu/about/divisions/cert/index.cfm#history

Program Structure

- Your program is focused around three key subject areas:
  - Software security engineering
  - Software security analysis
  - Software security management
- This is program is designed to be very intense and enable you to successfully fulfill various security-related roles

Specific Program Objectives

- Our goals are
  1. Introduce set of tools and techniques to evaluate software security
  2. Instill an understanding of how security verification activities fit into software-intensive projects
  3. Build a “toolbox” of security tools for use on the job
  4. Learn to approach software security the same way as our adversaries

Program Structure: Week-to-Week

- Each topic will have short (15-20min) video lectures
- Each week will have three in-person sessions where will will answer questions and complete exercises
  - Approximately one exercise per subject
- There will be an examination every Friday on the week's material
  - The remaining time will be dedicated to project work
- You will be evaluated on your weekly test scores and in your project performance

Program Philosophy

- Provide practical instruction
  - The topics, tools and techniques presented are immediately applicable
    - You can use them in a quality assurance role at LG
- But stress the underlying principles
    - Enable success in the future
- The best way to learn is to do
  - Give you experience with quality tools/techniques on a real-*ish* software-intensive system

Course Themes to Keep in Mind

- Every security attack/defense technique has strengths and weaknesses
  - Sophistication
  - Technical advantage
  - Potential damage
  - Defense effectiveness

- You will learn to evaluate both defenses and attacks for a given situation with consideration for project *goals* and *available* resources

Assignments

- Weekly readings and tests
  - There will be weekly reading assignments
  - Each week there will be an open-book test based on the reading and lecture materials
  - Questions will be multi-part, open-ended, essay style questions
  - Tests will have approximately five questions
    - Tests will also be open-book

Readings

- Software security includes a broad range of topics
  - Readings will generally be a collection of papers and select book chapters
- Readings will be posted to the course Canvas website

Course Project

- Cornerstone of the class
- To be completed in teams of four or five
- Given the length of this program it is impossible to have a real project, but we can still provide a substantial team project with the same benefits
  - Create a quality plan
  - Evaluate the quality of a relatively complex software system
  - Work in teams

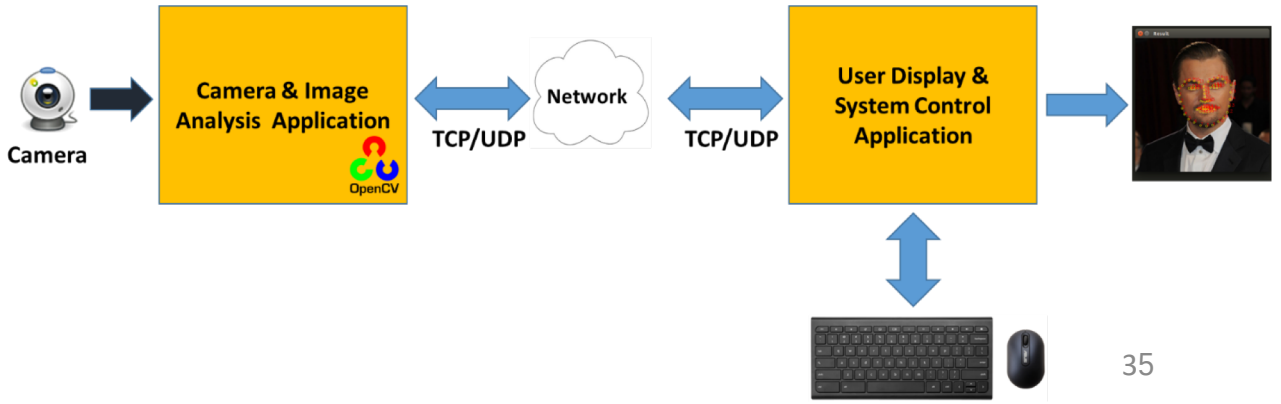
Image Analysis System

Phase 1: Development

- Identify security requirements
- Construct an extension in a secure way
- Evaluate the security of your extension

Phase 2: Defense

- Evaluate the security of a third party product
- Gain familiarity with tools/techniques
- Learn to think like an attacker



Studio Mentors

- Mentors are practitioners and faculty members with extensive industry experience, and are familiar with course materials
- Each team will be assigned a mentor
  - Teams will meet with mentors once a week for a status update and feedback
  - Mentors can help you apply course materials to the project
  - Mentors give teams feedback on course artifacts and progress
  - Mentors will help to develop students' skills as SDETs

Role of Mentors

- Mentors will …
  - Give hints, provide advice, and references to help team make progress
  - Ask a lot of questions to foster team reflection
  - Challenge teams to see if they agree or disagree with team operations, progress, approaches
  - Push teams to try new techniques and change the way that students think about quality in software intensive systems

Cheating Policy

- Cheating is not acceptable behavior!
- Examples of cheating include (but are not limited to)
  - Asking other students for answers or artifacts
  - Copying or stealing work from other classmates
  - Using laptops or cell phones to share answers
  - Having other students do your work for you
- We will report instances of cheating to LG program directors

Cooperation Policy

- We encourage you to discuss anything with other students that is presented throughout the week
- We encourage you to study together and share ideas and discussion
- Tests and assignments must be your own work!
- Team assignments and course project: share everything and work together on everything
- Ask when in doubt

Get Ready!

- This program is a lot of work, but I think it will be well worth the effort
  - Try to stay ahead of the workload and proactively manage your time
  - Your life will be miserable if you do not manage your time
- Focus on team work and communication
  - Trying to complete the project yourself is impractical and will make your life miserable
  - Embrace your team
    - Discuss, compromise, delegate, honor commitments, behave in a professional way
    - This is a fantastic opportunity to grow

Welcome!


CARNEGIE MELLON UNIVERSITY
PITTSBURGH PENNSYLVANIA 1900