Chris Cohen

20 Littleton Street, Apt. 12, West Lafayette, IN 47906
(636) 675-9358
chriscohen@chriscohen.dev
https://www.linkedin.com/in/chris-cohen-purdue/
https://www.chriscohen.dev
https://github.com/cohenchris

EDUCATION

Aug. 2017 – Present

Bachelor of Science at Purdue University

- Software Engineering and Cybersecurity
- 3.85 GPA
- 6x Dean's List 5x Semester Honors

EMPLOYMENT

May 2020 – Present

Qualcomm, QGOV Division

Software Engineering Intern

- Developed an Android app for a Qualcomm chipset feature that ensures secure wireless connection, and communicates the location of malicious access points to the user.
- Innovated an AI-powered system that processes raw media from non-AI devices, plotting important objects onto a map for the user to view.

May 2019 – Aug. 2019

Naval Surface Warface Center, Crane Division

Software Engineering Intern

- Improved US Navy missile sustainment efforts by upgrading an existing natural language processing algorithm to process failure databases.
- Held a valid 'secret' level security clearance given by the US Government.

EXPERTISE

Languages C C++ Python ARM/x86 Assembly Bash Javascript

Memory Management

- Paging, Virtualization
- Cache Memory Hierarchy
- Stack and Heap Management for ARM/x86

OS and Systems Programming

- Software/Hardware Interrupts and Device Management
- Asynchronous Inter-Process Communication (IPCa
- Return-Oriented Programming (ROP)
- Concurrency and Parallelism (Semaphores, Locks, Forking, Threading, Scheduling)

OSI/ISO 7-Layer Model

- TCP, UDP, HTTP
- IP addressing/routing, DHCP, DNS translation
- MAC addressing/routing, ARP
- Basic cryptography and security approaches

PROJECTS

April 2020

Web Server Honeypot (Extracurricular)

- Hosted an HTTPS Honeypot Server to lure attackers and collect information
- Automatic blacklisting for clients sending excessive requests in a short period of time
- Analyzed logs and learned about different types of attacks on web servers

March 2020

Process Hijacking in XINU (Operating Systems)

- Manipulated a victim process by locating and modifying return addresses and local variables in the runtime stack
- Learned about protection against this sort of attack (i.e. stack canaries)
- Studied how x86 interrupts, system calls, and function calls affect the runtime stack

Sept. 2019 – Oct. 2019

Shell Interpreter in C (Systems Programming)

- Parsing and execution of commands
- Signal handling and inter-process communication
- Subshell execution via forking