Gabriel Brown

315 West Comstock St, Seattle, WA *gabriel.brown@columbia.edu; GitHub.com/GabeIB // GabrielBrown.net*

EDUCATION

Columbia University – New York, NY
Bachelor of Arts, Computer Science, Concentration in Mathematics, May 2020
GPA: 3.8/4.0
Honors: Dean's List (Fall 2016, Spring 2017, Fall 2017, Spring 2018, Fall 2018, Fall 2019)

Recurse Center – New York, NY
Self-directed programming retreat, Summer 2020

Lakeside High School – Seattle, WA
May 2016

EXPERIENCE

Columbia Internet Real-Time Lab, Researcher — *August 2019 - February 2020*
As a researcher at the IRT Lab, I worked on a project to improve Internet of Things (IoT) security in the wake of the Mirai botnet. My major contribution was writing software, running on a home router, to automate IoT firewalling using IPTables. I also developed a test tool using Scapy to imitate the behavior of a compromised IoT device. My code and a full summary of my work is available at github.com/GabeIB/IoT-Security-MUDfile-to-IPTables

- Wrote and managed a large Python codebase running on a DD-WRT router
- Used MongoDB, iptables, packet sniffing, packet forgery, and DNS spoofing

General Antiparticle Spectrometer Lab (GAPS), Researcher — *January 2017 - May 2018*
GAPS is a project focused on identifying antiparticle energy signatures that are useful to understanding the nature of dark matter. I worked as a lab technician, handling a range of programming tasks. More information about the project is available at gaps1.astro.ucla.edu/gaps/

- Improved legacy C code for manufacturing of x-ray detectors, **increasing production by 50%**
- Wrote data analysis and particle simulation applications in Python for internal lab use

PROJECTS

Additive Synthesizer VST – github.com/GabeIB/additive_synth
During my time at Recurse Center, I built a real-time additive synthesizer in C++ using the JUCE framework. The synthesizer features per-partial amplitude envelopes and a python interface for computational sound design.

- Proficiency working within a large C++ codebase
- Proficiency developing interactive GUI applications with JUCE

MiniC to YUL Transpiler – github.com/GabeIB/YUL-transpiler
Under the direction of Vilhelm Sjöberg, I wrote a MiniC to YUL transpiler in OCaml that will form the basis of YUL language support for the DeepSEA compiler. DeepSea is a language developed by CertiK for generating formally verified smart-contracts for the Ethereum blockchain. Because of this contribution, I am listed as a co-author for the DeepSEA Compiler.

- lexing and parsing with OCaml
- Strong practical understanding of compiler design and implementation

Windows X86 Shellcode – gabrielbrown.net/blog/shellcode3/
As a personal project, I wrote a payload in x86 assembly that would spawn a reverse shell to a command and control server listening on an arbitrary port and IP address.

- Understanding of undocumented Windows OS internals
- Proficiency in disassembly and assembly analysis using IDA, Ollydbg, and WinDbg

SKILLS

Programming Languages: Python, C, C++, Java, x86 Assembly, OCaml
Programming Concepts: Networking, Sockets, Threads, Process Management
Frameworks and Tools: MongoDB, JUCE, UNIX/Windows internals, Bash, IDA, Ollydbg, Wireshark, IPTables, Scapy
Relevant Coursework: Malware Analysis and Reverse Engineering, Operating Systems, Computer Networks, Data Structures, Cryptography, Modern Algebra, Linear Algebra, Ordinary Differential Equations