

Emmett Collings

(587) 227-2882 | emmett.collings@gmail.com | [LinkedIn](#) | [GitHub](#)

EDUCATION

University of Calgary

Bachelor of Science, Major in Computer Science

Calgary, AB

Graduating December 2022

- GPA: 3.9/4.0
- Concentration in theoretical computer science
- Elective Highlights: Computer Security, Distributed and Randomized Algorithms, Compiler Construction, Database Design, Networks

EXPERIENCE

Software Developer Intern

May 2021 – August 2021

Neo Financial

Calgary, AB

- Worked with developer experience (DX) to create internal tools to aid developer efficiency
- Created private code coverage tool (github action + web app) that tracked test code coverage of repositories
- Created developer contribution tracking web app, which kept leaderboards of code reviews, PRs merged, etc.
- Technologies used: Typescript, Node.js, MongoDB, Next.js

Private Tutor

September 2017 – June 2018

Self Employed

Calgary, AB

- Specialized in high school physics and mathematics

Engineering Student

May 2021 – August 2021

Forge Oil & Gas

Calgary, AB

- Reviewed and maintained well files, inflow performance relationship curves, generated well bore diagrams
- Occasional field trips to examine well operations
- Assisted with general non-engineering office activities

TECHNICAL SKILLS

Languages: C, C++, Typescript/Javascript, Java, Python

Libraries/Frameworks: React.js, Next.js

Developer Tools: bash, git, github actions, Linux, Windows

Databases: MongoDB, MySQL

TECHNICAL PROJECTS/ASSIGNMENTS

J-- to .wat Compiler | C, flex, bison

January 2020 – April 2020

- Term project for compilers course
- Compiled from J-- (a toy subset of Java) to WebAssembly text format
- Used the standard compiler building tools, flex (for lexing), bison (for parsing)

Web Proxy | C, socket programming

January 2022

- Intercepted web traffic and replaced all jpgs from incoming HTTP responses to pictures of clowns
- Swapped text to various joke phrases as well

Hacking... | C

November 2021

- Security course assignment, less of a big project but it was really cool so included
- Exploit several insecure bash tools to spawn a root shell, as well create your own insecure tool. Exploits used included buffer overflows, format string vulnerabilities, timing attacks, environment manipulation.
- The vulnerable tool I made counted the number of lines in the file. The vulnerability exploits a call to `strlen()` that would store its output in an `int`.
- Since the return type of `strlen()` is `size_t`, this allows the `int` to be overflowed, allowing an attacker to bypass a length check and subsequently overflow a buffer to modify return addresses etc.

AWARDS & INTERESTS

Awards

- National AP Scholar (2015)
- National Swimming Championship Qualifier (2013-16)
- High School Div 3 City Champion – Volleyball (2014)

Interests

- Reading – Classical literature, non-fiction
- Sports – volleyball, badminton, swimming, climbing
- Outdoor – backpacking, scrambling, canoeing