Charlotte Som

JVM bytecode specialist, reverse engineer, videogame cheat developer, security researcher, undergraduate computer science student.

charlotte@som.codes | she/they | November 2000 | England | som.codes | videogame-hacker

Education

BSc: Computer Science

Loughborough University

2019 – 2023 (ongoing)

A standard Computer Science course, covering programming, software engineering, logic, and algorithms; with electives focusing on computer security and computer graphics.

Competitions

Google Hash Code 2020

Google

20th February 2020

Worldwide intractable problem optimization competition. Completed as part of a global team of three.

- UK: 6th place out of 406 teams.
- USA: 13th place out of 619 teams.
- Canada: 3rd place out of 135 teams.
- Worldwide: 252nd place out of 10724 teams.

Deloitte UK CTF

Deloitte

December 2020

UK-wide infosec (hacking) competition, organised by Deloitte. Participated as a Loughborough University-representative team of six.

- Placed 10th place in remote qualifiers,
- Placed 7th place in London finals.

BLÅHAJ CTF Team

2021 – Present

Casual capture-the-flag infosec team. Since I joined:

- **Hack-A-Sat 2 CTF**: Organised by the *US Air Force* & *US Space Force*. Placed 15th out of 697 teams.
- **corCTF 2021**: Placed 11th out of 904 teams.

Experience

Commercial obfuscator for JVM programs

Self-employed (paramorphism.dev)

2017 – Present

Involved writing the product itself and internal tooling around JVM bytecode to aid debugging. Working primarily with **Kotlin** and Java.

- *Paramorphism*: Bytecode obfuscator for JVM programs written in **Kotlin**.
- *libparamorphism*: Optional native runtime opaque library for programs obfuscated by Paramorphism written in **Zig** and **Rust**.
- *Koffee*: Domain-specific language for **Kotlin** for Java classfile generation.
- *Aksara* (internal): Bytecode assembly language and (dis-)assembly toolchain written in **Kotlin**.
- *Katon* (internal): Bytecode viewer and editor with a GUI written in **Rust** and interfacing via **Java Native Interface** to **Kotlin** (*Aksara*) and Java (*Fernflower*).

Average yearly USD revenue from Paramorphism sales is 4 figures.

Force Software ("Splashforce")

splashforce.io

2020 – Present

At Force Software, I am currently a "Software developer and strategy coordinator". The flagship "Splashforce" product is an end-user e-commerce automation suite for various sneaker sites. My current duties include:

- `cronet`: Custom patches for & isolation of Chromium's HTTP stack.
- Marrying an Electron (node.js / web **JavaScript**) frontend to a **Go** backend.
- Anti-piracy, reverse-engineer-deterrent releases for the Go backend. (Code virtualization, `cgo` FFI bindings, etc)
- Reverse engineering & circumventing bot-prevention measures on websites. (**WebAssembly**, **JavaScript**, more custom browser patches, etc).
- Reverse engineering & circumventing bot-prevention measures on Android apps. (**Java**, Smali, Android, etc).

Overall, I get to work with: Rust, Go, Java, Elixir, C++, Chromium (cronet), Firefox (necko, SpiderMonkey), WebAssembly, TypeScript, and JavaScript.

Novelty custom-mouse-based game-hacking

Self-employed (nullptr.pro)

Mid-2018

Fun, one-off hardware project (now-defunct site). Used **Thunderbolt 3**'s PCIe direct memory access capability to cheat in games without needing custom software to run on the (Windows) target computer, but *did* have Linux configuration software. Sold 10 mice for 10 BTC each.

- Firmware: **C**, C++, Duktape Engine (JavaScript), all on ARMv7.
- Config tool: **Vala**, JavaScript
- Hacking: **USB, PCIe**, Windows NT physical memory layout
- Individual game scripts: JavaScript

Writing

Circumventing the JVM Classfile Verifier

som.codes/jvm-force-no-verify

Reverse Engineering GTA V's Stunt Jumps

hackery.site/writing/gta-v-stunt-jumps

CyberDiscovery's "Valhalla" Challenge

hackery.site/writing/cyberdiscovery-valhalla

Circumventing Cisco Duo's Authenticator App

som.codes/cisco-duo-bypass

Languages

English

Native language.

French

Second familial language. **A*** at GCSE.
Self assessment: **C1**.

Korean

Hobby language, conversational.
Self assessment: **B1**.

Spanish

Studied in school. **A** at GCSE.
Self assessment: **B2**.

Experience (cont.)

Feather.IO

f3ather.io

2020

Feather.IO is an "all-in-one" end-user e-commerce automation application. I was contracted to create a secure (reverse-engineer-deterrent) release strategy for their Java application.

In the end, I wrote a **Rust** wrapper around a custom JVM (using **Java Native Interface**) with various security enhancements. This allowed the shipped application to be one fat executable while still running Java under-the-hood, but still being resistant to off-the-shelf Java reverse engineering tools. The wrapper and custom JVM were cross-compiled to **Windows**, **macOS**, and **Linux**.

Game modding / game-hacking

Self-employed

2016 – Present

- GTA Online: **C++**, C#, Rust, Zig
- Minecraft: **Java**, Kotlin, Scala, JVM Bytecode
- 2007 RuneScape: **Java**, Kotlin, JVM Bytecode
- RuneScape 3: **C++**, Rust
- Various Unity3D: **C#**, .NET MSIL, C++ (for `il2cpp` games)

Professionally, I used to sell a custom client for Minecraft (5-figure total USD revenue to date), and a subscription trainer menu for GTA Online (4-figure yearly recurring USD revenue until 2020). Nowdays, I do bespoke cheat development on commission.

Projects

alloc: Software Distribution & DRM

alloc.tech

2017 – 2020

(Defunct) Software license allocation and management system. Provides application developers with user fingerprinting, Digital Rights Management schemes, and auto-update. Written with **Rust**, **Python**, and **Java**.

Phoebe

git.lavender.software/charlotte/phoebe

Double-puppeting Matrix ↔ Discord chat platform bridge. Written in **Rust** using `matrix-rust-sdk` and `serenity-rs`. Contains actual parsers to translate between Matrix's HTML and Discord's custom Markdown-based message format.

SSH Lockbox

videogame-hacker/ssh-lockbox

Self-hostable centralised store for personal SSH keys. Supports:

- OpenSSH `sshd` automatic key retrieval
- GitHub.com key push via OAuth

Written in **Python**.

uim-status

videogame-hacker/uim-status

A tiny (40-line) utility written in **C** to get the status of the `uim` input method editor. Since I am a daily Linux user and multilingual, I need something on my status bar that displays whether my keyboard is inputting Korean, Cyrillic, or Latin characters.