Eugen Ajechiloae

Professional Experience

Capital Markets Trading (CMT) **March 2021 - Present**

Software Engineer **Remote**

I currently work for Panthera Investment GmbH, a venture of CMT. I'm part of a small team and our main responsibility is building server sided applications that transport and process trades and market information to other internal components or to various exchanges. These applications are mainly written in Python. They are split in microservices and deployed to a Kubernetes Cluser, using Ansible. We also maintain Qt C++ applications.

Bitdefender **June 2019 - March 2021**

Software Engineer/Security Research **Cluj-Napoca, Romania**

I was part of the security assessment team for Bitdefender Endpoint Security. Contributed to building programs used for bug hunting and security auditing, such as a *Pipe Monitor*, *IOCTL Fuzzer*, *Code Injection Tools* etc. These tools were built in Python, C++ or ASM, depending on the use case. We found and evaluated numerous vulnerabilities in Bitdefender's products.

One more noteworthy fact to mention is that I first started working for Bitdefender as an intern. During my internship, I learned the basics of the profession, such as source control and writing clean code. While working for the Hypervisor Memory Introspection team, I learned details on how the x64 architecture and Windows kernel operates, by building relevant programs.

Technical Skills

- Languages: Python, C, C++, ASM, Lua, Bash
- Deployment: Docker, Kubernetes, Ansible, Gitlab CI/CD
- Communication: FIX, gRPC, REST
- Technologies/Libraries: Redis, Prometheus, Flask, quickfix, Qt, WinApi, boost, AFL
- Experienced managing asynchronous and multithreaded applications
- Strong security research background
- Solid knowledge of algorithms and data structures

Education

'Babes-Bolyai University', Cluj-Napoca, Romania **Graduated: Summer 2021**

Bachelor of Computer Science.

Open Source Contributions

Poker Table Controller github.com/ajecc/poker-table-controller

Made a program that is capable of scraping and interacting with Desktop poker games. The data can then be passed to a DLL I built in C++. The DLL is capable of making a game decision based on various heuristics and game theory optimal strategies, in a timely manner. A major part of the computation is done concurrently. Multiple tables are supported in parallel.