IVAN MOROZKO

+7 (***) *** **** ◇ ✉ *********@gmail.com ◇ ol-imorozko ◇ foobar

EDUCATION

Saint Petersburg State University Sep 2019 - Jul 2023
Bachelor of Software and Administration of Information Systems *St. Petersburg, Russia*
Overall GPA: 4.78 out of 5

TECHNICAL SKILLS

Programming languages	C (proficient), Bash (intermediate), C++/Python/SQL (basic)
Technologies	Git, Fuzzing, Make

EXPERIENCE

Oktet Labs Oct 2019 - Present
Embedded Software Engineer, Software Engineer *St. Petersburg, Russia*

- Ported code for loading boot loader image over UART for an auxiliary kernel on a wi-fi router to the kernel thread, resulting in a **65% auxiliary kernel loading speedup**
- Developed **Linux kernel module** for logging all serial devices, which allowed debugging in case of unexpected kernel crashes
- Collaborated with a 20-man industrial team in improving a proprietary **multi-threaded HW NIC verification system** written in C and Bash
- Extended open-source greybox fuzzer AFLNet by implementing **network namespaces support**, and used it for **fuzzing user-level network stack OpenOnload**, resulting in one **critical bug** found

PROJECTS

Embedded kernel loading speedup Sep 2020 - Oct 2020
C, Linux kernel, Low level serial subsystem

- Managed to send boot loader image over UART for an auxiliary kernel inside a thread of the main kernel
- Designed and implemented API for using the serial port device file before mounting a root file system
- These changes resulted in a **65% auxiliary kernel loading speedup**

Log-All-UARTs Linux kernel module Dec 2021 - May 2021
C, Linux kernel module, Low level serial subsystem, Parallelism

- Designed and implemented a Linux kernel module to store serial devices logs from the start of the system
- Implemented **procfs** interface for configuring module parameters
- The module is constantly used for storing serial devices logs of embedded devices, **allowing debugging in case of unexpected kernel crashes**

Multi-threaded HW NIC verification system Sep 2021 - Mar 2022
C, Bash, Networking, VirtIO, Virtualization, Parallelism

- Taught myself the complex design of a big proprietary industrial system written in C and Bash
- Augmented the expectation subsystem to validate **VirtIO-Net packet receive filtering** features
- The system is constantly used for **real high-performance NICs** verification and development

Exec-on-board tool May 2020 - June 2020
C, Networking

- Designed and implemented single-binary **Telnet client + TFTP server** to ease configuration of the embedded devices