Munich, Germany
elm@mail.ederlm.de

Eder Leão Moosmann

linkedin.com/in/ederlf
github.com/ederlf

Work Experience

Engineering Manager   **Platform.sh**   **Jul 2022–Present**
Images delivery team   Remote, Germany

- Managing a team of software and devops engineers, setting clear goals, priorities and areas for improvement. Our team is responsible for building new image versions, adding request packages, building and maintaining debian packages, bug fixes, implementing new features, unit and integration testing and keeping images up to date and free of vulnerabilities. We serve container images to 1000s of customers running a diverse range of tech stacks.
- Coordinated and implemented automated service container releases, transitioning from sporadic releases to weekly updates of hundreds of images. The regular updates reduced the number of customers running outdated images containing CVEs.
- Created proof-of-concept images to run docker inside LXC containers. It was not trivial because the build and execution of the docker containers are done in read-only images.
- Automated issue request creation on Linear for new versions of images using public apis and by scrapping public information from websites, using python, GraphQL API and running regular scheduled pipelines in the gitlab CI.
- Eliminated information and communication friction between company sectors and my team, by creating an easy to use Slack bot, implemented in Go. The bot publishes releases, deployments to production and development status of images and allows the retrieval of package versions and changelogs. While my team also provides the data via metabase, the product and marketing teams find the bot easier to use.
- Delivered Clickhouse as a high-availability platform service for the observability team, which enabled log monitoring and improved analysis capabilities for Platform.sh's observability stack. Implementing high-availability in our platform is tricky and requires complex knowledge about how to set up and coordinate clickhouse container nodes' clustering with our orchestration tool.
- Established a culture of incident data analysis. I classified root cause and mitigation methods according to previous incidents. I generated reports with statistics for the most common causes and mitigations.

Cloud Software Engineer   **Platform.sh**   **2020–2022**
Core and images delivery team   Remote, Germany

- In my first year at Platform.sh, I wrote a large report about the state of our images and our processes. My conclusion advocated for the creation of a dedicated team for images. I was one of the first engineers of the images delivery team and quickly became a manager.
- Improved pipelines and automation for Continuous Integration (CI) on Gitlab. I have used YAML, Gitlab CI, Python, bash and Go. Achieved speed up of CI jobs for image releases by 75%.
- Built and maintained multiple Debian packages for services and application container images. I created packages for software not available upstream and packages with the need for quick patches. Thanks to the diversity of packages we maintained, I was exposed to many build systems, of languages such as C, Go, Rust, Java and many more.
- Implemented integration tests in continuous integration for internal image's packages, reducing by 80% the presence of faulty packages in our container images.
- Reduced the time to identify and fix broken image builds, by introducing CI pipeline health monitoring with Python, Influxdb and Grafana.
- Implemented an emulator for testing the components of the company's network edge layer solution using Python, Mininet, Go and GoBGP. The emulator allowed lighter end-to-end tests of the network software without the need to bootstrap the whole orchestrator.
- Worked on the port from Python 2 to Python 3 of the platform git server. The git server contains tens of thousands of lines of code, with multiple internal and external dependencies that also required porting.

Researcher   **Queen Mary University**   **2015–2019**
                                            London, UK

- Developed a fast network simulator for OpenFlow and routing protocols (BGP, OSPF) using C, Cython,Quagga and Linux network namespaces.
- Developed network applications to scale SDN for large Internet Exchange Points using Python, Shell script, Flask, InfluxDB, Grafana and MongoDB.
- Worked on NS3 and Mininet simulations of data center solutions to reduce congestion. I have played with DCTCP, BGP, Python and C++.
- Performed Internet measurements, data analysis and visualization of DNS, Netflow and BGP records
- using Python (numpy, matplotlib, scypy, networkx) Java and C.

| **Software Developer** | **Lenovo** | **2014-2015** |
| | Campinas, Brazil | |

- Created, built and documented an open source version of a proprietary Operating System for storage devices (Unfortunately, it was never released.)

| **Research Developer** | **CPqD** | **2012-2014** |
| | Campinas, Brazil | |

- Developed open source routing and switching software for Software Defined Networks. At the time, my open source contribution was used by thousands of users in the industry and academia. I dealt, since the early days of the projects, with Github issues and pull requests.

Education and Certifications

- **PhD Computer Science,** Queen Mary University of London.   **2015-2019**
- **M.Sc. Electrical Engineering,** University of Campinas, Brazil.   **2013–2014**
- **B.Sc. Computer Science,** University of Sao Paulo, Brazil.   **2010–2013**

Languages, Technologies and Skills

- Languages: Python, Go, Rust, C, C++, Java
- Technologies: Git, Linux, Debian, Computer Networks, MySQL, Redis, MongoDB, InfluxDB, AWS, Podman, LXC, Docker, Gitlab CI
- Skills: Self-driven, goal oriented, problem solving, organization, time management, team work

Interests

- I am fond of sci-fi and fantasy novels and I admire many parts of the Japanese culture and language