

Anthony Ruhier

INFRASTRUCTURE SOFTWARE ENGINEER

17 rue de la Libération
25 410 Pouilley-Français, France

+33 688 162 558

✉ anthony.ruhier@gmail.com

🌐 aruhier.fr

<https://github.com/aruhier>

25 years old (born on 10/22/1993), French

Work experience

- 2019 **Network Engineer, Software and Automation on CorpNet**, Google, Zürich (Switzerland).
As a Network Engineer focused on software and automation, I work on the Corporate and Enterprise infrastructure team to deliver Google's next generation network and corporate data centers. I build software for distributed services, abstractions and the components of the system that operates and powers Google.
- 2018 **Network DevOps Engineer**, Online/Scaleway, Paris (France).
Working with the Backbone SREs on maintaining and improving the entire network, by designing and coding tools — mostly in Python — to allow them to parallelize and automate their actions on a larger scale, to target a hundred of thousands of users.
- 2014–2017 **Apprentice Software Engineer for a local Telecom Operator**, Trinaps, Belfort (France).
Lead developer, designing micro-services focused on the automation of redundant processes, linked to the conception of an Information System allowing the company to efficiently store and use its data.

Qualifications

- 2014–2017 **Master's Degree in C.S**, UTBM, Apprenticeship Program in 3 years, Belfort (France).

Projects

- Personal project **Personal infrastructure, to satisfy a will of self-hosting**
Web: Nginx as ingress controller in Kubernetes.
Databases: PostgreSQL as main DBMS, MariaDB for applications requiring it.
Hypervisor: 3 Kubernetes nodes over KVM (Libvirt as hypervisor) on Debian, with automated backups to Btrfs and ZFS Glusterfs NAS, using virt-backup and btrbk.
Management: More than 30 pods and hosts running on ArchLinux, uniformly managed by Ansible from the network configuration to the databases, X.509 certificates and Nginx setup.
DNS: Authoritative servers use Knot with automatic DNSSEC signing, resolvers use Unbound.
Network: Hosts are spread into 6 VLAN, via OpenVSwitch and a Cisco switch SG300. Traffic is subject to QoS policies, using PyQoS. IPv4/IPv6 external traffic is dynamically routed using BGP and Wireguard. DNS resolvers in anycast.
- Online/Scaleway **Rate limiting the entire server farm uplinks**
Python
Celery
Dynamically set rate limiting on 3K Online's custom switches, depending on the customer offer, with a computed burst size large enough to be the least restrictive possible. Impacted more than 50K servers, saved several 100gbps of bandwidth.
- Personal project **Python framework to setup a QoS on Linux** (<https://github.com/aruhier/pyqos>)
Python
Traffic Control
Framework to build complex QoS rules for Linux, in a hierarchical way with built-in tweaks. Rules can make use of the Python OOP system to be as much clear as possible. Imported and used at TRINAPS for events setups, to manage the QoS of more than 5K simultaneous users, with a really tight WAN bandwidth (200mbps).
- TRINAPS **Captive portal used by TRINAPS in events installations**
Python (Flask)
Rust
Javascript
Mainly used for festivals and important events. Used in the Eurockeennes festival since 2014, with 3K connected users. The portal communicates with a router running on Debian, managing a hashmap of MAC addresses efficiently stored in ipset.

Programming

- Imperative Programming Python (TDD), Rust, Java (TDD), C, C++
Web development Django, Flask, Javascript
Others Bash, RabbitMQ, Docker

Language skills

English **C1 Level**, BULATS – 85/100 (2016)

French **Native speaker**