Olivier Guandalini

EXPERIENCE · 4 YEARS IN PRODUCTION

What I run in production

DevOps and SRE engineer at a software publisher, in the in-house infrastructure team. I run more than 400 on-prem Linux servers, and since 2026 I have been contributing to the company's Kubernetes platform: creating its clusters on vSphere, as code, and deploying their components through GitOps.

400+ on-prem Linux servers, configured by Ansible	**~15 min** to create a Kubernetes cluster	**16** projects on one Ansible role for Java	**2** apprentices trained up to full autonomy

AT WORK

Problems solved

Each time: the problem, what I did, and what it changed.

01

Kubernetes clusters in 15 minutes

`Terraform · Rancher · vSphere · Kubernetes · Argo CD`

> THE PROBLEM On vSphere, a cluster went through servers prepared on their own, then registered one by one in Rancher.

> WHAT I DID I added a native method to the team's Terraform project. Rancher drives vCenter: it clones the VM templates, creates the node pools (roles, labels, taints, storage nodes) and runs the rolling updates. A cluster's whole topology fits in one variables file. That is how I created the cluster of the software factory, and I deploy the components of every cluster through GitOps: Envoy Gateway, External Secrets wired to Vault, kube-prometheus-stack, Longhorn, Datadog.

> THE RESULT **A new cluster in 10 to 15 minutes with `make apply`, documented for the team: the architecture, how it works, and a step-by-step tutorial.**

02

A Java repository about to shut down

`Ansible · Eclipse Temurin · Debian · CentOS`

> THE PROBLEM The daily package update failed across the fleet: the JDK repository had been abandoned, 8 security updates behind, and its shutdown was scheduled. At the same time, the developers wanted to move to Java 21.

> WHAT I DID My first Ansible role: it installs Eclipse Temurin in the version a project asks for, JDK or JRE, on Debian and CentOS. Then the projects moved over, and the back-office applications went to Java 21.

> THE RESULT **One role, used today by 16 projects, from Java 8 to Java 25.**

03

Knowing before the users do

`Prometheus · Grafana · Datadog · Ansible · Jira`

> THE PROBLEM Across more than 400 servers, an incident has to show before anyone reports it.

> WHAT I DID Monitoring as code: Prometheus and Alertmanager jobs and alerting rules, exporters rolled out to the whole fleet by Ansible, Grafana dashboards for the teams. SLOs and SLAs in Datadog, which alert in Slack and in Jira.

> THE RESULT **Incidents are caught earlier, from the alert rather than from a report.**

04

Secrets out of the code

`Vault · External Secrets · Kubernetes · Ansible`

> THE PROBLEM API gateway and database credentials, to keep out of the repositories, on the servers as in Kubernetes.

> WHAT I DID I set up HashiCorp Vault to keep them in one place, and External Secrets to hand them to the applications on Kubernetes. On the servers, the secrets of the Ansible projects are encrypted with Ansible Vault.

> THE RESULT **Two mechanisms for one rule: no secret in clear text in a repository.**

05

Logs the developers read on their own

`ELK · Logstash · Kibana`

> THE PROBLEM The developers have no access to the servers, yet they have to debug what runs there.

> WHAT I DID The application logs, gathered with ELK: Filebeat on the servers, a Logstash pipeline and an index mapping for each log format, Kibana to search them.

> THE RESULT **The developers debug on their own, without asking for access.**

06

An AI assistant to take to production

`Python · Mistral · Qdrant · Langfuse · MCP · OpenTelemetry`

> THE PROBLEM A conversational assistant to take from code to production, with new needs: several LLM providers, a vector database, long-lived WebSocket connections.

> WHAT I DID Since 2025, its deployment: a Python API, several LLM providers (Mistral, Cohere), the Qdrant vector database and Langfuse to follow every request, all deployed by a two-phase Ansible pipeline with its secrets encrypted by Vault; an OpenResty gateway for the WebSockets (TLS, long-lived connections, CORS); an MCP server that opens data services to AI agents; and since September, traces and logs with OpenTelemetry, Tempo and Loki.

> THE RESULT **A beta in production.**

07

Training the next ones

`Ansible · Prometheus · ELK`

> THE PROBLEM Two DevOps apprentices to bring up to speed, on a critical production.

> WHAT I DID Mentoring them for two years: Ansible, Prometheus, the ELK stack and running a critical production.

> THE RESULT **Both became autonomous.**

Also

- `In progress` Moving a critical API from its servers to Kubernetes, behind a 3scale and OpenResty gateway.
- Jenkins on the software factory's cluster (Helm, Configuration as Code), on nodes of its own so that builds never slow the other workloads; the legacy pipelines kept running with Rundeck and Ansible.
- Incidents from start to finish: the live response, post-mortems, communication with the people affected, architecture documents and runbooks.
- vSphere, internal and in a DMZ: Debian VMs, routing, firewalls, and a PHP application put in production on isolated servers.

The tools, and where I use them

AT WORK, IN PRODUCTION

Linux · Debian · CentOS · vSphere · vCloud Director · Ansible · Terraform · Kubernetes (RKE2) · Rancher · Argo CD · Helm · Kustomize · Docker · Jenkins · Rundeck · GitLab · Vault · External Secrets · Envoy Gateway · Longhorn · Prometheus · Alertmanager · Grafana · Datadog · ELK · OpenSearch · OpenTelemetry · OpenResty · 3scale · Qdrant · Langfuse · OVH · Bash · Python

IN MY PROJECTS, ON MY OWN TIME

AWS EKS · VPC, IAM, S3 · Secrets Manager · k3d · cert-manager · Sloth · GitHub Actions · Trivy

Path

SINCE 2023

DevOps and SRE engineer

A software publisher, in-house infrastructure team. An apprentice first, permanent since October 2024.

2022 TO 2023

DevOps engineer, apprentice

An adtech SME: applications moved to Docker and Kubernetes, a first Prometheus and Grafana monitoring.

2021 TO 2022

Automation engineer, apprentice

The energy branch of a large construction group: the electrical distribution of a metro line's structures.

Education

2022 TO 2024

MSc-level degree in DevOps and Cloud, Sup de Vinci

Thesis: monitoring and diagnosing problems in real time, to resolve incidents fast.

2021 TO 2022

Professional bachelor's degree in industrial IT, IUT de Ville-d'Avray

Industrial computing, automation and production.

2019 TO 2021

Two-year degree in electrical engineering and industrial IT, IUT de Ville-d'Avray

A French technical degree, the DUT GEII.

LANGUAGES

French, native. English, professional and technical.

And on my own time

The public projects on this site take the same subjects end to end, on AWS and on a laptop: cloned in minutes, documented, checked on every change.

CONTACT

Write to me on LinkedIn. My code is on GitHub.

LinkedIn linkedin.com/in/ogu GitHub github.com/Olivg92