



# Aurelian Shuttleworth

Site Reliability Engineer

## CONTACT

✉ aurelian@shuttleworth.tech

☎ +31 619840600

🌐 linkedin.com/Aurelian

🐙 github.com/aurelian-shuttleworth

## LANGUAGES

English Native

## PROFESSIONAL SUMMARY

I am a Senior Site Reliability Engineer focused on making infrastructure invisible to the teams that use it. I drove a 5-person SRE team to ensure the stability and deployment of 30 microservices across **8+ Kubernetes (GKE) clusters** scaling beyond **1,000+ nodes**. I built the company's **HashiCorp Vault** deployment from scratch – **PKI-based mTLS**, zero static credentials, all powering a custom service mesh that ensures every secret is audited, traced, and encrypted.

The engineering skills I bring are grounded in my **Personal Cloud Platform**, a distributed systems R&D environment where I operate closer to the metal. I run a highly available **Talos Linux Kubernetes** cluster on **Proxmox QEMU/KVM**. The platform is backed by **Ceph CSI** distributed storage, a **Dual-Stack IPv6** three-tier network fabric utilizing **FRRouting (FRR)**, and **Cilium eBPF** networking. It is a production-grade stack managed entirely through **ArgoCD GitOps**, built from the kernel up and enshrined as code.

## WORK EXPERIENCE

- EvBox | Amsterdam** 2022 -
- Built the core GitOps platform managing the full lifecycle for 30 microservices. By centralizing container patching, scaling, and ingress via **ArgoCD**, we reduced developer interaction to purely pushing code.
  - Wrote the **Terraform Infrastructure as Code (IaC)** modules used to bootstrap our **GCP** projects from scratch. This automated **IAM, DNS, and HashiCorp Vault** authentication across **8+ GKE clusters** and **1,000+ nodes**.
  - Built the **HashiCorp Vault** deployment from scratch, eliminating static credentials across 3 environments. This PKI-based mTLS and KV v2 secrets architecture handled 500+ requests per second from all 30 microservices.
  - Designed a Vault-backed zero-trust architecture. Every secret is now audited, traced, and encrypted at rest and in transit.
  - Maintained the **GitLab CI/CD** pipelines for 30 product teams. To enforce quality standards, I built shared pipeline templates and 24 custom base images incorporating strict **Snyk** and **SonarQube** quality gates.
  - Owned the **Prometheus** and **VictoriaMetrics** observability stack. I designed a central, multi-tenant metrics cluster with HA storage and auto-scaling ingestion, secured via the Vault mTLS service mesh.
  - Engineered an internal, mTLS-based alternative to **Google Identity-Aware Proxy (IAP)**. This removed the burden of custom Java authentication libraries from developers, centralizing auth into the SRE service mesh.
  - Worked within a 5-person SRE team following Google's SRE methodology. I actively participated in quarterly PI planning to drive infrastructure initiatives from stakeholder requirements to production.

## WORK EXPERIENCE (CONTINUED)

### | SprintHive | Cape Town

2019 - 2022

- Refactored duplicated **Terraform** infrastructure into dedicated, reusable modules. This established a scalable IaC foundation across environments and reduced deployment cycles from days to a single day.
- Took ownership of a failing **AWS** cloud infrastructure project for a financial client. I designed and delivered the complete Terraform-based environment in two months, replacing an outsourced team.
- Maintained highly available **Kubernetes** clusters on **Google Cloud** serving production workloads, integrated with AWS storage. I implemented custom **Horizontal Pod Autoscaling (HPA)** driven by Prometheus latency metrics.
- Developed a **Go-based** repository management CLI to enforce code quality standards. This tool automated Git hook installation and Terraform formatting prior to modern Nix/direnv adoption.
- Automated **Kong Ingress** control and API key provisioning via custom Terraform modules. I also migrated Elasticsearch configuration from manual click-ops to fully codified deployments.
- Automated **Prometheus** provisioning via custom Terraform modules that dynamically calculated resource limits. I implemented query throttling to prevent **Grafana**-induced OOM crashes, and defined latency and availability SLOs.
- Authored blameless Correction of Error (COE) documents post-incident. I also participated in Red/Blue team exercises, supporting developers in security preparation.

### | Amazon Web Services | Remote

2019 - 2019

- Contributed patches to legacy **Java** case management applications, directly resolving workflow bottlenecks. This code contribution led to my transition into this engineering role.
- Proposed and prototyped a **Machine Learning** system to automate abuse case classification by learning from support agent corrections, a precursor to modern AI-driven case management.
- Developed automated, legally compliant archiving tooling for sensitive customer service content. This provided full audit trails supporting high-stakes legal dispute resolution.

### | Amazon Web Services | Cape Town

2016 - 2019

- Developed **Python-based** heuristic scanning tools targeting **EC2** drive I/O metrics to detect malware installations, alerting customers and preventing unauthorized infrastructure charges.
- Engineered Python network traffic analysis tooling to identify botnet signatures and malicious communication patterns, producing standardized reports for multiple AWS security teams.
- Built **Java-based** automation that autonomously processed **14%** of the weekly global abuse email volume, eliminating manual triage and absorbing the workload of five senior support agents.
- I was chosen by AWS leadership to be flown to Seattle headquarters multiple times to collaborate directly with the EC2 Core Team.

### | HX Systems | Somerset West

2013 - 2016

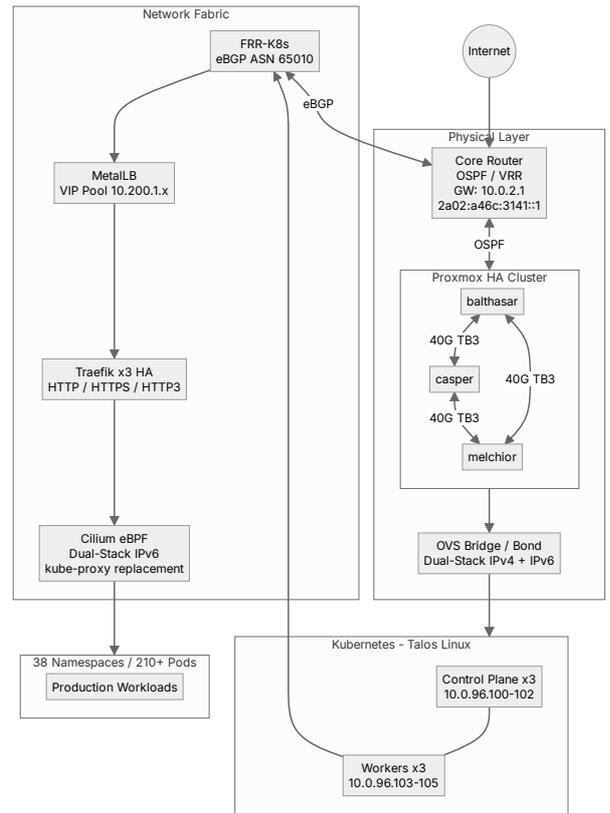
- Engineered and maintained the physical and network infrastructure for a regional Wireless Service Provider (WISP), deploying multi-site routing topologies and administering core endpoints using **OSPF/BGP**. Architected a migration from localized consumer hardware to a highly-available **VMware HA hypervisor stack**, centralizing all critical ISP services, core routing, and authentication systems onto resilient bare-metal virtualization.

## PROJECTS

### Personal Cloud Platform & Distributed Systems R&D

[gitlab.com/shuttlewoth-tech/magi/kustomize/public/-/tree/dev?ref\\_type=heads](https://gitlab.com/shuttlewoth-tech/magi/kustomize/public/-/tree/dev?ref_type=heads)

- **Platform Architecture:** Architected and operate a **6-node, highly-available Talos Linux Kubernetes cluster** (3 control plane, 3 worker) on bare-metal Proxmox QEMU/KVM hypervisors. The platform orchestrates 210+ pods and 150+ services across 35+ namespaces, governed by strict **ArgoCD GitOps** for declared state reproducibility.
- **Distributed Storage & Data:** Engineered a resilient, high-throughput software-defined storage backend using **Ceph CSI** (RBD, CephFS, NFS), optimizing replication performance by isolating bulk storage I/O on a dedicated 40 Gbps Thunderbolt interconnect. Operate HA PostgreSQL topologies via **CloudNativePG (CNPG)**.
- **eBPF Networking & Ingress:** Designed a robust **Dual-Stack IPv6** three-tier network fabric utilizing **FRRouting (FRR)** and OSPF peering. Replaced kube-proxy entirely with **Cilium eBPF** for kernel-level routing and observability, maintaining a production-grade ingress pipeline (MetalLB, 3-replica **Traefik**, HTTP/3, Gateway API) to evaluate emerging CNCF networking standards.



## TECHNICAL SKILLS

Platform Architecture (Kubernetes, Talos Linux, GKE)



Security Architecture (HashiCorp Vault, PKI, mTLS, Zero Trust)



Storage Architecture (Ceph CSI, CNPG, PostgreSQL HA)



CI/CD Pipelines (GitLab CI, GitHub Actions)



Infrastructure as Code & GitOps (Terraform, ArgoCD, Kustomize)



Networking Design (Cilium eBPF, FRR, IPv6, Traefik, Gateway API)



Automation & Scripting (Python, Shell/Bash, Nix)



Systems Programming (Go, Java)

