JONATHAN BOULIGNY

PLATFORM / INFRASTRUCTURE ENGINEER

Austin, TX • 713-371-6810 • jonathanbouligny@gmail.com • linkedin.com/in/jonathan-j-bouligny • github.com/JonathanBouligny • bouligny.dev

PROFESSIONAL SUMMARY

Infrastructure engineer with 6+ years automating enterprise Linux environments at Red Hat and Lockheed Martin, working in Ansible, Terraform, Python, and containers. Red Hat Certified Architect in Ansible. Currently building a GitOps-managed Kubernetes platform with SRE practices (SLOs, burn-rate alerting) from scratch.

TECHNICAL SKILLS

Languages: Python, Bash, JavaScript, C++

Infrastructure: Linux (RHEL, Ubuntu), Proxmox VE, VMware vSphere, Networking

Automation: Ansible, Terraform, Red Hat Ansible Automation Platform 2.4, Red Hat Satellite 6.15, CI/CD, Infrastructure as Code (IaC), YAML

Kubernetes & GitOps: Kubernetes, k3s, Red Hat OpenShift 4.14, Argo CD, GitOps, Helm, cert-manager, HAProxy, Containers (Docker/Podman), cloud-init, Git, AWS (S3, IAM)

Observability & SRE: Prometheus, Grafana, PromQL, Alertmanager, blackbox exporter, SLIs/SLOs, burn-rate alerting

PROJECTS

Kubernetes Platform on a Private Cloud — 2026

- Built a Kubernetes (k3s) platform on a three-node Proxmox cluster (40 cores, 157 GB RAM): Terraform-provisioned VMs from cloud-init templates, Ansible-configured nodes, Argo CD bootstrapped by playbook.
- Manage all cluster state via GitOps with Argo CD (app-of-apps), sourced from a self-hosted Forgejo server provisioned as code in a separate repo, so the source of truth doesn't depend on the platform it rebuilds.
- Applied SRE practices to the platform, monitoring it with Prometheus and Grafana through the same GitOps workflow, with a 99% availability SLO and multiwindow burn-rate alerts (page on fast burns, ticket on slow).
- Store Terraform state in an AWS S3 backend with locking, so the platform is reproducible from any machine and safe for concurrent and automated runs.
- Treat full teardown and rebuild as the acceptance test: one script recreates the whole platform — VMs, cluster, GitOps engine, synced apps — from empty hypervisors in 3–7 minutes.

PROFESSIONAL EXPERIENCE

Consultant — 2022 – Present

Red Hat — Automation consulting across 7 federal and healthcare engagements, including air-gapped networks; authored Ansible roles and playbooks and trained customer teams

- Built the Python dynamic inventory plugin that let Ansible Automation Platform consume a customer's Ivanti CMDB instead of a Red Hat Satellite source that covered only a subset of hosts: ~1,300 hosts across two datacenters, filtered by owning team, environment, and lifecycle status so automation never targeted decommissioned systems, and documented for customer handoff.
- Deployed and automated Ansible Automation Platform across three engagements as multi-node installs (controller, hub, database) on VMware vSphere, converting manual install runbooks into version-controlled Ansible so environments rebuilt identically.
- Converted Ansible execution environment image builds into infrastructure-as-code in a CI/CD pipeline, cutting build time from 10 minutes to 2 and replacing the platform default so every job in a closed network ran on the versioned image.
- Stood up containerized GitLab on an air-gapped network using Podman and systemd, the environment's first source-control system, to version-control the automation synced into AAP.

Software Engineer — 2020 – 2022

Lockheed Martin

- Resolved an OpenShift deployment blocker by investigating container failures, finding the root cause in pod logs, and correcting the deployment manifest to restore service.

CERTIFICATIONS & EDUCATION

Red Hat Certified Architect (RHCA) in Ansible • Red Hat Certified Engineer (RHCE), Enterprise Linux and Ansible • RHCSA (EX200) • Red Hat Certified Specialist: Managing Automation (EX467), Developing Automation (EX374), Ansible Network Automation (EX457)

B.S. Computer Science, *Magna Cum Laude* • Texas Tech University — 2015 – 2019