

# Sean Gatewood

hello@sgatewood.dev |  [sean-t-gatewood](#) |  [sgatewood](#)

## Summary

Senior-leaning DevOps / Platform Engineer with a focus on the **dev** side of DevOps. I build scalable, reproducible, and reliable production environments with Kubernetes ecosystems, Infrastructure as Code, and CI/CD pipelines. I bring strong development skills to the infra world, elevating production environments with seamless automation, custom microservices, proactive application-level fixes, and spikes that push the boundaries of what is possible.

## Experience

**DevOps Engineer**, Digital Asset (New York, NY / remote) April 2023 - present

- Developed infrastructure and automation around critical kubernetes services, applying SRE practices to ensure reliability and minimize operational toil.
- Played an instrumental role in the launch of our blockchain network by owning the greenfield deployment and management of a core validator node.
- Utilized Terraform, Jsonnet, Bash, and Helm to automate deployments in a flexible manner, enabling the success of more than 70 weekly upgrades.
- Orchestrated a complex blockchain migration process in the deployment automation, reducing required live-on-zoom code changes down to 0.
- Migrated our node's cluster across GCP regions with only 2 minutes of downtime.
- Instrumented monitoring/alerts with DataDog to maintain 99.5% uptime and tune our configuration to minimize cloud costs.
- Communicated deployment designs with partner companies, helping them overcome similar hurdles and accelerating the launch of our blockchain network.
- Mentored 2 teammates on how the validator deployment worked so they could contribute to the codebase and participate in support rotations.
- Authored a Prometheus metrics exporter in Go, providing a programmatic framework for custom metrics. Closed 8 critical gaps in production observability.
- Authored a Kubernetes operator in Go to streamline our backup/restore process, eliminating operational tasks on a product entering maintenance mode.
- Migrated Kubernetes manifests from a legacy system to public helm charts, allowing external customers to deploy our product.
- Orchestrated our internal helm installations using Helmfile to reduce our deployment process from 35 commands to 1 and pave the way for using GitOps.
- Built isolated blockchain test environments in both k3d (for local dev) and vClusters, allowing developers to effortlessly spin up fully independent networks.
- Developed CI/CD pipelines using Google Cloud Build and CircleCI.
- Participated in on-call rotations, triaging a multitude of production alerts and maintaining the reliability of our services.

**Software Engineer II**, Appian Corporation (McLean, VA / remote) August 2020 - April 2023

- Worked on a Development Experience team that owned a variety of tooling and infrastructure to accelerate and modernize development workflows.
- Contributed to a remote development platform allowing devs to build/develop/test on dedicated EC2s, improving cycle time and stability by at least 50%.
- Built a Gitlab pipeline to precook warm builds that devs could take and use instantly, reducing startup time from ~12m to ~5s for many common test cases.
- Architected a separate mode using [Mutagen](#) to enable bi-directional file syncing and enhance stability.
- Deployed a LaunchDarkly [relay proxy](#) in Kubernetes, satisfying a FedRAMP security requirement for our modern feature toggles initiative.
- Contributed to a team effort to integrate LaunchDarkly with a legacy product, including developing our own Java client.
- Regularly initiated & led architectural design discussions, and spiked a wide variety of prototypes.
- Regularly unblocked developers by assisting in troubleshooting technical issues, improving our local build stability.
- Utilized an ELK stack to collect diagnostics from our development tool, greatly enhancing visibility into command utilization and user flows.
- Enhanced our documentation by injecting Matplotlib graphs (refreshed daily via Gitlab deployment pipeline) to show metrics on our codebase over time.
- Treasured high code quality standards and contributed to our data-driven goal to lead the department in code quality.
- Mentored two interns as a Summer Intern Manager, meeting with them weekly to optimize their internship experience.
- Gained a high standard for usability that continues to shape my tooling designs today.

**Teaching Assistant - Introduction to Programming**, University of Virginia (Charlottesville, VA) Fall 2017 - May 2020

- Assisted hundreds of students with introductory programming concepts in Python.
- Created [review videos](#) that have received over 13,000 views.
- Helped manage the course as one of four "Head TAs."

## Selected Skills

<b>Languages</b> Python, Bash, Go, Java, TypeScript, C++	<b>Kubernetes</b> Helm, Helmfile, FluxCD, Istio, Custom Operators	<b>Monitoring</b> Grafana, Prometheus, Datadog
<b>Cloud Platforms</b> GCP, AWS	<b>Infrastructure As Code</b> Terraform, Pulumi	<b>CI / CD</b> GitLab, Github Actions, CircleCI, GCB, Jenkins
<b>Cloud Resources ("I've terraformed it!")</b> Compute (GKE Cluster, Node Pools, Persistent Disks, Snapshot Schedules, VMs), Networking (VPC, Subnet, Cloud NAT, L7 Application Load Balancer, MCI, Cloud DNS, Google-Managed Certificates), Security (IAM, RBAC, Secrets Manager, Cloud Armor, Service Accounts, Workload Identity)		<b>Other</b> Docker, Docker Compose, SQL, LaunchDarkly, MermaidJS, Nix, Jsonnet, Auth0, Kotlin, Gradle, Linux, jq, Justfile, Git, GitHub

## Side Projects

- [This Resume](#): Written in YAML and templated into the document you are reading. :-)
- [JobDeployment controller](#): Spiked an idea I had using [kubebuilder](#) to solve a DX problem with deploying Job objects in helm charts.
- [gcloud-fzf](#): Wrote a Go CLI I wanted. The interesting part is I actually distributed it via homebrew.
- [Portfolio site](#): Used an old Jekyll theme I liked to serve as my website. Includes some interesting dark mode & MermaidJS hacks.

## Education

**University of Virginia** (May 2020) Rodman Scholar, Raven Society

- B.S. Computer Science. GPA 3.95

**Notable Courses**: Computer Graphics, Algorithms, Introduction to Cybersecurity, Software Testing