JACOB COLVIN

Platform, Site Reliability, and Software Engineer

Cincinnati, Ohio   @ me@jacobcolvin.com   jacobcolvin.com   @0xMacro   colvinjm   MacroPower

ABOUT

With over seven years of experience in the tech industry, my expertise lies in **Site Reliability Engineering**, enriched by a deep focus in **Software** and **Platform Engineering**. My passion lies in building robust **developer platforms**, crafting maintainable **infrastructure as code**, and architecting systems in a way that is simple and straightforward. My experience as a developer, with a focus on **Go** and **Python**, enables my ability to create platforms that are not only reliable, but also enjoyable to use.

EDUCATION

B.S. Information Technology

University of Cincinnati - CECH

Aug 2015 – April 2020

- Cybersecurity specialization
- Summa Cum Laude

SKILLS

Infrastructure: AKS | K3s | Talos | Azure | Cilium | Terraform | Ansible | Helm | Crossplane | Cloud Foundry

Development: Go | Python | FastAPI | TypeScript | Jsonnet | Profiling | OpenTelemetry | PowerShell | Bash | Make

Observability: Prometheus | Thanos | Grafana | Jaeger | Loki | Vector | Fluentd | OpenTelemetry | Datadog

Azure Managed: Databricks | Datafactory | Postgres | Key Vault | Storage Account

DevOps/GitOps: FluxCD | ArgoCD | GitHub Actions | Azure DevOps

EXPERIENCE

84.51° / The Kroger Company

Remote / Cincinnati, Ohio   May 2016 – Current

Senior Site Reliability Engineer (Kubernetes)   January 2023 – Current

- Redesigned large parts of our **IaC (Terraform, Flux)** for **Kubernetes (AKS)**, reducing manual toil by 200+ hours per year, to support increased scaling demands of new production workloads (driving $300MM+ operating profit).
- Developed **Python** application for scaffolding template repositories, as well as centralized **Helm charts** and related **GitHub Actions**, which collectively saved developers 3000+ hours during the first year in production.
- Led **Datadog** cost-saving efforts which reduced spend by over $400k per year; enforced controls using **Terraform**.
- Participated in an **on-call rotation** for our production **Kubernetes** clusters; ensured compliance with SLAs.
- Empowered developers to efficiently and independently troubleshoot their applications by acting as lead subject-matter expert for observability platforms; headed support and maintenance of all observability tooling, including **Datadog**, **Grafana**, **Prometheus**, **Thanos**, **Jaeger**, **OpenTelemetry Collectors**, and **Vector**.

Senior Site Reliability Engineer (Observability)   April 2021 – December 2022

- Deployed and supported enterprise observability services/tooling, utilizing **Ansible** (on-premises) and **Flux** (Azure Kubernetes), including **Grafana**, **Prometheus**, **Thanos**, **Fluentd**, **Telegraf**, and **Jaeger**.
- Developed **Python** library to centralize instrumentation for **Prometheus** metrics, **OpenTelemetry** tracing, structured logging, and **pprof** profiling, with support for **FastAPI**, **Databricks** notebooks, and more, which both delivered a positive and consistent experience with our observabilitly platforms, and directly saved developers a collective 1000+ hours.
- Led development of multiple **Prometheus exporters** using **Go**, utilizing **Redis** for caching, distributed workers with pub/sub and leader election, **CEL** for custom rule evaluation, **Cue** and custom tooling for validation, and Go text templating.
- Rolled out **Datadog** as a unified logging platform for both **Azure** and on-premises systems, using **Terraform** for configuration and **Fluentd** for log parsing and forwarding; enabled deprecation of multiple disparate logging solutions.
- Created multiple **Jsonnet** libraries for **Prometheus** and **Alertmanager** configuration, saving 100+ hours per year.
- Contributed small fixes to **Grafana**, **Thanos**, **Jaeger**, and other upstream **Git** repositories.

Versatilist Engineer

📅 January 2020 – March 2021

- Created and administered multiple **Azure** environments using **Terraform** in Azure DevOps; said environments frequently included resources such as **Databricks** workspaces, **Datafactory** instances, **Key Vaults**, **Storage Accounts**, and **Postgres** databases.
- Enhanced our **Prometheus** ecosystem by adding high-availability and long-term storage via **Thanos**, thus allowing consumers of Prometheus to track SLOs and KPIs over years instead of days.
- Assisted a mix of seven development and data science teams with **Azure** migration, **Terraform**, **Grafana** dashboards, **Prometheus** alerts, **GitHub Actions**, and **Azure DevOps** pipelines; acted as technical liaison to increase speed and accessibility of support.

ICT Co-op

📅 May 2016 – December 2019

- Designed a **C#** API layer over several legacy systems, and a SPA using **TypeScript** with **React**, to both assist with support, and improve velocity towards deprecation of legacy systems.
- Created a **Prometheus exporter** for **SonarQube** data, along with corresponding dashboards, rules and alerts, to allow security team to gather relevant KPIs.
- Automated **Red Hat Linux** VM deployment through **ServiceNow**, via interactions with **Satellite**, **vCenter**, **SolarWinds** and **Ansible**.
- Designed a custom web framework using **PowerShell** and **Bootstrap** for executing administrative tasks and aggregating events and metrics from many distinct products.
- Automated many miscellaneous tasks using **PowerShell**, **VBA**, **Bash** and **PL/SQL** in Bash; created and documented **Automic** workflows with **Bash** and **SAS ODS**.
- Interfaced with **BOSH** and **Hadoop** to design custom **Pivotal Cloud Foundry** monitoring solution.

KEY PROJECTS

OmegaGraf

jacobcolvin.com/OmegaGraf

- An open-source project that seeks to completely automate vCenter monitoring, by orchestrating a small ecosystem of containers, including Telegraf, Prometheus, and Grafana.
- Paper: https://scholar.uc.edu/concern/student_works/jw827c971
- Technologies: Docker, C# / .NET Core, TypeScript, React

Homelab

MacroPower/homelab

- Infrastructure-as-code for my homelab / personal cloud. Defines multiple interconnected Kubernetes clusters, spanning across bare metal (Talos) and multiple Hetzner cloud environments (k3s-on-MicroOS).
- Technologies: Talos, Cilium, ArgoCD, Helm, Jsonnet, Terraform

Prometheus Video Renderer

MacroPower/prometheus_video_renderer

- Just for fun, completely impractical tool that allows you to encode audio and video as Prometheus metrics.
- Featured on the Grafana blog:
  https://grafana.com/blog/2021/07/30/how-to-use-grafana-and-prometheus-to-rickroll-your-friends-or-enemies
- Technologies: Go, Jsonnet

Analytics Panel Plugin

MacroPower/macropower-analytics-panel

- Grafana panel plugin that injects JavaScript into dashboards, which reports user session information to a backend Go server, which in turn exports Prometheus metrics for display in (you guessed it) Grafana.
- Over 1MM downloads, featured by Giant Swarm:
  https://www.giantswarm.io/blog/grafana-ception-or-how-we-do-grafana-analytics-giant-swarm
- Technologies: TypeScript, React, Go

Wakatime Exporter

MacroPower/wakatime_exporter

- Prometheus exporter and Grafana dashboards for Wakatime coding statistics.
- Over 100k downloads
- Technologies: Go