Matthew Olenik

matt@olenik.me | olenikm@gmail.com
https://github.com/mattolenik
https://olenik.io

B.S. Software Engineering
Rochester Institute of Technology

Summary

- 14+ years of experience as a generalist programmer and engineer with a code-heavy focus on devops, backend services, infrastructure, build and CI/CD pipelines, and developer tools
- Deep programming experience across many languages with expertise in Go, TypeScript, JavaScript, C#, C++, C, Python, Ruby, Groovy, Scala, bash/shell, and others
- 8+ years building infrastructure within AWS, especially with building secure complex network and VPC topologies
- Expertise with Docker, especially containerizing existing applications and building dev environments using Docker
- Deep expertise with Terraform, including writing new Terraform providers and custom tooling
- Consistently identifies engineering problems and solutions, owning projects from spec/design phase, build, test, documentation, deployment and support
- Comfortable with high pressure on-call support e.g. live debugging, incident response, root cause analysis
- Works well with a team or autonomously, whatever is best suitable for the organization

Okta — *DevOps Engineer (Contract)*

July 2024 — May 2025 (Contract)

- Building and supporting Okta’s incident management Slack bot used by all of Auth0
  - Built with Tines, a low-code tool that was chosen to allow our customers to edit business logic on their own
  - Custom feature development built on APIs of Jira, Confluence, PagerDuty, StatusPage, Slack, and incident.io
  - Most programming done with Tines’ own functional language, with complex features requiring considerable code
- Led development of the TypeScript rewrite of Okta’s incident Slack bot, a BoltJS Node app running on Kubernetes in AWS
  - Reimplemented existing incident logic in TypeScript, integrating with Slack, Jira, DynamoDB, Statuspage, PagerDuty
  - Tutored teammates not familiar with TypeScript or JavaScript to onboard them to the project
- Prototyped a tool in Go for exporting applications from Tines (a low-code tool) and compiling them into standalone applications (targetting Go or any other language)
  - Utilized PEG (parsing expression grammar) for parsing Tines’ functional formula language into an in-memory Go representation, generated working Go code from exported Tines stories

Harpoon — *Principal Software Engineer*

No-code Kubernetes Platform

May 2023 — November 2023

- Built out complete infrastructure-as-code solution for deploying the product into Amazon EKS
  - Responsible for technical direction of all staging and production infrastructure
  - Base of Terraform for all essential AWS infra, such as VPC configuration, security groups
  - Terraform for creating and managing the EKS cluster, including all the extra work for deploying EKS plugin Helm charts, integrating with ALBs, Secrets Manager integration
  - Helm chart for Harpoon itself, allowing us to deploy product infra changes in an atomic manner
  - Full CI/CD with instant deployments using GitHub Actions and Keel
  - All IaC following Terraform and HCL2 best practices, clean modularization, easy management of AWS/k8s secret sync, just add a few lines of config in TF to get secrets synced/imported
- Product Development
  - Built autodeploy functionality allowing customers to push deployments directly to Harpoon, using a from-scratch Docker Registry proxy in Go
  - Rewrote container build service to use Buildkit with Kubernetes jobs, vastly improving scale and reliability from old method of shelling out to Docker
  - Converted JavaScript codebase to TypeScript, refactoring and repairing problematic JavaScript
  - Nodejs service modernization, rewritten to use async/await, removing promises and complex callback chains
  - Miscellaneous frontend work with AngularJS
- Developer Productivity
  - Built new developer experience using Devbox, creating a local dev environment that is isolated and reproducible
  - Improved build workflow so that all services could be built with a single `make` instead of many by-hand individual builds in each project

Pulumi — *Senior Software Engineer*

Developer Tools — May 2022 — October 2022 (Contract)

- Modernized crd2pulumi, a Go CLI tool for generating Pulumi code from Kubernetes Custom Resource Definitions
- Refactor3ng from hackathon project to a standardized CLI application with full E2E tests and code coverage

HashiCorp — *Senior Software Engineer*

Terraform Enterprise — November 2021 — April 2022 (Contract)

- E2E test framework development, rewriting Ruby RSpec tests into a custom Go test framework
- Implemented code coverage for E2E tests, providing developers with coverage of their Rails app even when run from within Go tests

eSentire Inc — *Software Engineer*

DevOps/Infrastructure Team — April 2018 — October 2021

Developer Tools

- Designed, implemented, and tested escli, a swiss army knife for engineering tasks
- Git repo composition that works as a replacement for submodules, simpler for devs to understand and use, with a focus on CLI usability
- Enforces deprecation of repos to encourage keeping dependencies up to date
- Templating system for creating new projects using Go's text/template engine
- Custom, simple package manager for internal tools, cross platform, powered by Artifactory
- Used by all of engineering for composing repos of any complexity

Terraform and Automation

- Created solution for multi-account VPC networking CIDR block management with NetBox
  - Wrote Terraform provider for NetBox, focus on e2e tests against real NetBox
  - Made upstream contributions to NetBox for race conditions
  - Automated deployment and update of NetBox into AWS with Terraform and Jenkins
- Owned design and implementation of infrastructure for esINSIDER
- Automation for generating Terraform code from existing prod resources then importing the state
- Created Terraform state management CLI tool to work around state bugs in Terraform 0.11
- Jenkins scripted pipeline work, such as refactoring old multi-stage Jenkins jobs into scripted pipelines
- Provided support for the engineering org in their day-to-day use of our development pipeline
- Made upstream contributions to NetBox

Curalate, Core Services — *DevOps Engineer*

April 2017 — Febuary 2018

- Overhauled ECS cluster management with blue/green cluster deployments and autoscaling
- Automated ops tasks (e.g. backups, syncs, migrations) with Terraform and AWS Lambda
- Internal dev tools for working with Terraform, internal CLI development

VMware, Photon Controller — *Software Engineer, MTS III*

Open source multi-tenant IaaS for on-premise management of VMs and containers — January 2014 — May 2016

- Wrote new CI process for building VM images using Jenkins, Packer, and Vagrant
- Designed the API for and implemented the Photon Controller Go client SDK, used in the project's CLI
- Wrote the plugin for Bosh, Cloud Foundary's platform-as-a-service

Microsoft, Xbox Live Commercial Services — *Software Development Engineer*

June 2011 — January 2014

- Refactored legacy Xbox 360 services to run on modern CI/CD with Autopilot (internal Microsoft IaaS)
- Implemented, tested, and deployed features in services for: user accounts, billing and offering, in-game DLC, TV-on-demand, and others
- Owned user email preference services, consisting of a REST API, and background jobs for exporting and migrating bulk data