

KEVIN 'TYLER' COX

Senior Software Engineer — CI/CD (GitHub Actions & Platform)

📞 (843) 718-4024 ✉ kevincox103@gmail.com 📍 Charleston, SC

🌐 github.com/Ty-lerCox

🌐 linkedin.com/in/tyler-cox-1715065a/

🌐 ty-lercox.github.io/portfolio/

Summary

Platform-leaning engineer specializing in CI/CD at scale with GitHub Actions and GitLab CI. Shipped public Actions workflows (GitHub Pages/Jekyll, Angular → Firebase Hosting & Functions, CityRPG mod artifact packaging) with cache strategy and artifact retention. Strong in containerized deployments (OCI images, HashiCorp Nomad), observability (Grafana/Prometheus/Loki/Tempo), and secure-by-default services (Keycloak/OIDC, PKI/TLS). Production services in Java/C#/TypeScript; thorough ADRs/runbooks.

Education

Computer Science

Trident Technical College

CI/CD & Platform

GitHub Actions (matrix, cache, artifact retention), GitLab CI / Runners, Azure DevOps, Ansible (provisioning), ADR/Runbooks & incident response, Feature flags, testing & quality gates

Containers & Orchestration

OCI container images (Docker/OCI), HashiCorp Nomad (deployments, job specs), Kubernetes (familiar; able to adapt patterns)

Services & Data

Java (Spring Boot), C# (.NET Web API), TypeScript, REST API design, CDC → Kafka, medallion layers

Security & Observability

Keycloak (OIDC/SAML), Kerberos, PKI/Certificate Authority, TLS/mTLS, Grafana, Prometheus, Loki, Tempo, SLO dashboards & alerting

Front-End (supporting)

Work Experience

Expeditors International (Expeditors)

Aug, 2014 - Present

Software Engineer (Angular & Platform)

Design and delivery of front-ends, services, and platform tooling; emphasis on CI/CD, observability, security, and repeatable containerized deployments.

- Standardized CI/CD with **GitHub Actions** and GitLab CI: authored workflows for static sites (Jekyll GitHub Pages), Angular builds with Firebase Hosting & Functions deploy, and artifact packaging (CityRPG mod); implemented caching (actions/cache) and artifact retention to reduce build time and storage.
- Built and maintained **OCI container images** (build/runtime) used by GitLab Runners; versioned base images and documented image hardening and SBOM generation.
- Operated **containerized deployments on HashiCorp Nomad** (our orchestrator); authored job specs, health checks, and deployment runbooks; adaptable to Kubernetes patterns.
- Delivered RESTful services in **Java Spring Boot** and **C#/.NET** supporting logistics workflows; documented API contracts and failure modes.
- Implemented observability with **Grafana/Prometheus/Loki/Tempo**, SLO dashboards, actionable alerting; created incident **runbooks** and supported on-call response.
- Hardened identity and transport security: **Keycloak** (OIDC/SAML) SSO, **Kerberos** integrations, internal **PKI/CA**, and **TLS/mTLS**.
- Modernized data flows from ad-hoc Kafka jobs to **CDC → Kafka** with medallion layers (bronze/silver/gold) to improve reliability, lineage, and reprocessing.
- Produced ADRs/architecture notes to create clarity; led code reviews and cross-team collaboration on risk/mitigation and deployment strategy.

Projects

GitHub Actions Demos

<https://github.com/Ty-lerCox>

Small, no-proprietary examples that exercise Actions features commonly used in hosted CI/CD.

- Jekyll → **GitHub Pages** publish with cache + artifact retention
- **Angular** build → **Firebase Hosting & Functions** deploy
- **CityRPG mod** packaging pipeline that produces versioned build artifacts
- Job **matrix** for Node.js versions; dependency caching via actions/cache
- Workflow docs explaining trade-offs (speed vs. retention) and secret handling

Languages

English

Nomad CI/CD Patterns

Internal pattern library for containerized deployments on HashiCorp Nomad, portable to other orchestrators.

- Job specs with health checks; blue/green rollouts; reproducible **OCI images**; runner image hardening; ADRs and runbooks for ops handoff.

Observability Enablement

Service health/SLO dashboards with Prometheus metrics, Loki logs, Tempo traces; alerting integrated with deployment telemetry.

- Trace-led debugging; consistent dashboards; runbooks and escalation paths; MTTR reduction.

CI/CD Blueprint

Reusable pipelines (GitHub Actions/GitLab CI) and Ansible roles for consistent build/test/deploy across on-prem and cloud targets.

- Cut lead time to change; standardized quality gates and environment promotion; SBOM & vulnerability scans on images.

Data Flow Modernization

Database-level CDC streaming into Kafka with bronze/silver/gold layers.

- Improved reliability, lineage, and downstream transformations; clearer SLAs and reprocessing.

ODEYA.app

Feb, 2025 - May, 2025

ty-lercox.github.io/portfolio/posts/odeya-overview/

Personal web app that auto-curates YouTube playlists from collections of channels so new uploads flow into a single playlist.

- SSR for SEO; componentized TS code; reduced manual upkeep via aggregation logic.