Corey Jones

INFRASTRUCTURE RELIABILITY ARCHITECT · BIRMINGHAM, ALABAMA

I build the reliability practice before you need one.

hello@coreyjones.dev · coreyjones.dev · github.com/unorig-username · linkedin.com/in/coreycjones

I'm Corey Jones. **Twelve years in production systems, eight of them in SRE.** I join early, stand up observability, Kubernetes, and incident response from zero, and make systems tell the truth about themselves. Right now I'm the sole SRE at Linq. I also build complete products, from Slack tooling to a self-hosted e-commerce platform.

Metrics

1,243	415	22	8.4M
pull requests merged across 38 repos	teammates' pull requests reviewed	repos started, still the top contributor	active metric series on the stack I built

Incident log

INC-2026-01
Jan 2026 – now

Linq

Infrastructure Reliability Architect · sole SRE · reports to the CTO

- **Observability.** Stood up the observability platform in my first two weeks: self-hosted Grafana, Mimir, Loki, Tempo, and Pyroscope on EKS, all in Terraform, fed by a two-tier OpenTelemetry Collector with tail sampling. It now carries about 8.4M active series, with roughly 140 alert rules and 560 recording rules in code. Moved the company's Grafana off Grafana Cloud and cut 55 datasources to 14.
- **Analytics platform.** Built the ClickHouse platform end to end: signed HTTP ingest into NATS JetStream, then ClickHouse (three replicas on Graviton) and a Parquet archive in S3, on its own EKS cluster managed by Argo CD. Nightly backups, a staging mirror, and 10 runbooks. In production since September 2026.
- **Core platform.** 332 merged pull requests in the core messaging platform (Go): Postgres vacuum and WAL tuning, about 123 GB of redundant indexes dropped, NATS JetStream capacity and placement work, send-path caching and fetch fixes, and the removal of the Kong API gateway with zero partner errors at cutover.
- **Device management.** Designed and built a self-hosted Apple MDM (NanoMDM, zero-touch Apple Business enrollment, step-ca) plus its operator UI in the internal Tech Ops platform. Zero-touch enrollment proven in October; production pilot running in November.
- **Platform foundations.** A self-hosted GitHub Actions runner fleet on EKS (ARC and Karpenter) that cut lint from 127s to 31s and queue time from 67s to 3s. An AWS Organizations landing zone in OpenTofu: account vending, SCP guardrails, Identity Center with SCIM, org-wide CloudTrail and GuardDuty, and GitHub OIDC with no static keys.
- **Incident response.** Configured incident.io as code (severities, service catalog, alert routing, Linear and Notion workflows). Built the partner paging bot, a status-page-to-incident bridge, a heartbeat monitoring service, and Megaphone for broadcasts and partner alerts.
- **Prototypes.** TraceLinq, a Rollbar replacement built in seven days (Go API, React dashboard, four SDKs, Helm chart). Siren, a Slack incident app with Claude-drafted status updates. Specter, a live map of edge traffic.

INC-2020-12
Dec 2020 – Aug 2025

Fetch

Czar of Site Reliability Engineering · first SRE hire

- Joined as the first SRE and built the SRE team from there.

- Built, ran, and migrated the company onto self-hosted Grafana clusters, Loki (replacing several log aggregators), and Mimir for Prometheus metrics, running at 22.5M active series.

INC-2018-01
Jan 2018 – Dec 2020

Shipt

QA Engineer → Site Reliability Engineer → Senior Site Reliability Engineer

- Built and ran custom incident-management software integrated with Slack.
- Moved observability from hosted Graphite to a self-hosted Graphite cluster on Kubernetes with a carbon-relay tier, then to self-hosted Prometheus. Ran Grafana in Kubernetes.
- Architected the load-testing platform, built Jenkins CI for integration tests, and ran a Selenium grid in Kubernetes.
- Kept retail partners' S3 and SFTP integrations reliable, and taught classes in Shipt Tech University.

INC-2014-11
Nov 2014 – Jan 2018

Influence Health (formerly MEDSEEK)

Systems Integration Engineer

- Built integrations that normalized HL7, C-CDA, and FHIR messages.
- Deployed and upgraded the patient-portal platform for hospital systems, with on-site work for sales, architecture, and support.

Things I've shipped

OBSERVABILITY · LINQ

LGTM+P observability platform

Metrics, logs, traces, and profiles in one self-hosted stack, fed by OpenTelemetry and run entirely from Terraform and code-reviewed rules.

DATA PLATFORM · LINQ

ClickHouse analytics platform

Signed ingest, a JetStream buffer, ClickHouse for queries, and a Parquet archive for the long tail. Built, documented, and handed to on-call with runbooks.

DEVICE MANAGEMENT · LINQ

Self-hosted Apple MDM

Zero-touch enrollment and declarative configuration for company devices, with a device API and an operator UI that refuses writes unless it owns the device.

SLACK TOOLING · LINQ

Megaphone

Company-wide broadcasts that can be edited everywhere after sending, partner alerts for flagged and recovered lines, status-page fan-out, and micro-incidents.

CI · LINQ

Self-hosted CI runner fleet

Org-wide GitHub Actions runners on EKS with arm64 and x64 pools, scale to zero, and warm minimums during the work week.

ON-CALL · LINQ

Paging and on-call plumbing

incident.io as code, a Slack bot that lets partners page on-call, a bridge from status-page webhooks to incident alerts, and a heartbeat monitoring service.

VISUALIZATION · LINQ

Specter

A live world map of HTTP traffic: each request is an arc from client to edge location to origin, colored by outcome. It runs on the office wall displays.

ERROR TRACKING · LINQ

TraceLinq

A self-hosted Rollbar replacement: fingerprint grouping, releases and deploy markers, Slack, PagerDuty, Linear, and GitHub integrations, and Claude root-cause notes.

DEVELOPER TOOLING · LINQ

Xavier

A versioned source system for agent-driven development: it bootstraps repos with review lenses, deterministic gates, and human approval, and tracks loop health across every repo.

E-COMMERCE PLATFORM

Naptime Candles

A custom storefront and admin portal for a handcrafted soy candle business, replacing Shopify end to end: Stripe checkout with server-side price checks, Shippo labels, gift cards, local delivery, and pickup.

PERFORMANCE

Server challenges

Public load-test contests: implement one API, get benchmarked on a fixed box. I entered in C, Rust, Go, Zig, x86-64 assembly, Java, and, for fun, COBOL, Emacs Lisp, and Neovim Lua.

COMPUTABILITY

Busy Beaver tooling

Go tooling for the open BB(6) holdouts: a machine parser, a simulator checked against the BB(5) champion, a run-length accelerated simulator, and cycler deciders.

LOCAL AI

Local-first AI lab

A private RAG knowledge base on Gemma 4, Whisper transcription with speaker labels on Apple Silicon, and a blind-graded comparison of shared agent-memory systems over MCP.

Load tests

CHALLENGE	BOX	RESULT	RANK
Million users c-epoll · PR #104	16 vCPU, Postgres on its own box	1,000,000 users (max)	#8 of 85
Multicore server c-epoll · PR #203	8 vCPU, Postgres on the same box	210,000 users · 52k req/s	#5 of 127
$12 server c-iouring · PR #290	1 vCPU, 2 GB, SQLite	280,000 users · 27k req/s	#10 of 237

Dependencies

Observability	OpenTelemetry · Grafana · Mimir · Loki · Tempo · Pyroscope · Alloy · Prometheus · Graphite
Platform	Kubernetes (EKS) · Karpenter · KEDA · Helm · Argo CD · Terraform · OpenTofu · AWS · Cloudflare Workers
Data	PostgreSQL · ClickHouse · NATS JetStream · Valkey/Redis · SQLite and D1 · S3 and Parquet
Languages	Go · Python · TypeScript · C · Rust · Zig · SQL · shell
Incident response	incident.io · PagerDuty · Slack apps · Linear · status pages · blameless postmortems
AI engineering	Claude and Bedrock · MCP servers · agent loops with review gates · local models (Ollama, Gemma, Whisper)