Michael Boterf

LinkedIn | GitLab | GitHub

PROFILE

Lead Platform Engineer / Technical Lead with 10+ years building secure, scalable systems and leading engineering delivery across federal and regulated environments. Specializes in platform engineering, CI/CD systems, infrastructure-as-code (Terraform, NixOS), and large-scale healthcare data platforms (FHIR R4, 100M+ records) in AWS GovCloud and Azure. Proven track record setting technical direction, hiring and mentoring engineers, building reproducible environments, integrating security-first practices (NIST 800-53), and moving teams from ambiguous requirements to reliable delivery. USMC veteran with strong focus on reliability, automation, and operational excellence.

TECHNICAL SKILLS

CI/CD & DevOps: GitLab CI/CD (pipelines, reusable templates, runners), GitHub Actions, Jenkins, Azure Pipelines, Backstage

Containers & Orchestration: Kubernetes (EKS, AKS, k3d), Docker / Podman, Helm / Kustomize, Azure Container Apps, KEDA, Kafka / Flink

Security & Compliance: Trivy / Semgrep / Gitleaks / govulncheck, SBOM (sbomnix) / SARIF, AquaSec / KICS, Fortify / Blackduck, OpenSCAP / Evaluate-STIG / MITRE SAF, sops-nix / HashiCorp Vault, NIST 800-53, Platform One Big Bang

Data & Healthcare: PySpark / Polars / Pandas, Delta Lake / DuckDB / PostgreSQL, FHIR R4 / HL7 / US Core Profiles, EHR Integration (VistA/RPMS, Oracle Health), Medallion Architecture (Bronze/Silver/Gold), Azure Synapse Analytics, Parquet / CSV (RFC 4180)

Infrastructure & IaC: Terraform / OpenTofu (AzureRM), Nix / NixOS / flake-parts, Azure Container Apps / Event Grid / KEDA, Colmena, Ansible, Puppet

Cloud Platforms: AWS (EC2, IAM, RDS, S3, EBS, GovCloud), Azure (Synapse, Container Apps, Blob, Pipelines)

Languages: Python, Go, Nix, Shell / Bash, JavaScript / TypeScript, Julia, SQL, Java, C#, Groovy

Systems & Platforms: NixOS (production multi-host), Linux (RHEL, Ubuntu, NixOS), Windows / WSL, Nginx / Traefik, Authentik SSO / Netbird VPN, Grafana / Prometheus

EXPERIENCE

Senior Platform Engineer / Technical Lead, Data Platform | ATA-LLC (GDIT Federal Contract — Indian Health Service)

Technical Leadership, Hiring & Team Enablement

- Served as technical lead for a six-person mixed team of data engineers, platform engineers, and data analysts, setting engineering direction, assigning work, running standups, reviewing PRs, and turning unclear customer requirements into executable delivery plans.
- Wrote technical job requirements, led technical interviews as the primary engineering evaluator, and provided candidate assessments that shaped team hiring decisions.
- Mentored junior engineers in Terraform, Docker, Kubernetes, CI/CD, data engineering, and project architecture, growing them from new hires into independent contributors capable of owning delivery areas within 6-8

months.

- Stood up the project's engineering foundation from nothing: reproducible Nix developer environments, templates, reference documentation, coding standards, architecture docs, pairing practices, and a Go-based devshell/documentation search tool.
- Reduced first-day onboarding friction so new engineers could start productive project work within 30 minutes, then reach meaningful project productivity around the three-month mark.

Healthcare Data Transformation & Migration Platform

- Led platform and data engineering for federal healthcare system supporting IHS migration from VistA/RPMS to Oracle Health, transforming legacy data into FHIR R4 resources and Oracle Cerner CSV outputs.
- Architected medallion-pattern (Bronze/Silver/Gold) ETL pipelines processing 100M+ patient records across 1,000+ tables using Python, PySpark, and a Nix-based build system on Azure Synapse Analytics.
- Built config-driven data platform generating 880+ Nix packages from declarative table configurations, enabling standardized transformations across 1,000+ datasets with 98.4% coverage.
- Used Nix as the execution and packaging layer for PySpark/Python healthcare transformations, enabling deterministic builds, isolated package entrypoints, and repeatable validation for large clinical domains including labs, vitals, medications, encounters, procedures, and documents.
- Achieved 100% FHIR R4 validation across all clinical domains (demographics, vitals, labs, conditions, medications, immunizations, procedures, encounters, orders, documents) with dual tribal coding for IHS-specific requirements.
- Developed streaming-mode processing for large fact tables (9.7M+ rows per table), and dual-output architecture generating both CSV and FHIR Bundle outputs from single pipeline runs.
- Owned architecture for the Cerner Data Syndication pipeline, including Go bundle processing, Azure landing and extracted containers, Event Grid, Storage Queue, Container Apps Jobs, KEDA scaling, and Terraform-managed deployment modules.
- Created FHIR validation tooling processing ~40,000 JSON records in ~15 minutes; built Python fuzzy-matching field mappers, US Core profile skeleton generators, and Pydantic model extensions.
- Built reproducible Nix-based developer platform reducing onboarding time to under 10 minutes and ensuring parity between local development and CI environments.
- Moved the effort from blocked requirements, missing environments, and undefined architecture into sustained delivery by creating the technical foundation and upskilling the team around it.

Nix, CI/CD & Platform Automation

- Used Nix as the common platform interface for build, test, scan, package, deploy, and operations workflows, making local developer commands match CI behavior across healthcare data, cloud infrastructure, Kubernetes, and security tooling.
- Built a config-driven Nix package generation system that produced 880+ reproducible ETL package entrypoints from declarative table configs, standardizing local and Synapse execution across 1,000+ healthcare datasets.
- Created Nix/flakes operational shells wrapping OpenTofu, Azure CLI, GitHub CLI, ACR publishing, Container Apps Job operations, Log Analytics queries, and Storage Queue inspection into reproducible nix run entrypoints.
- Built Nix-packaged Go OCI artifacts for data-syndication bundle processing, including reproducible container images, Azure Container Apps Job deployment, and versioned Azure ACR publishing flows.
- Replaced Docker-daemon image publishing with Nix + Skopeo workflows, allowing CI and developer machines to build and push OCI artifacts through nix run commands without requiring local Docker state.
- Built a Nix-flake security scanning framework exposing reproducible apps for Trivy, Semgrep, Gitleaks, govulncheck, SBOM generation, SARIF output, and attestation reporting.

- Built NixOS audit modules for scheduled Vulnix and Lynis hardening reports using systemd timers, producing recurring host-level vulnerability and compliance evidence.
- Packaged Azure Pipelines agents as Nix-built OCI images using dockerTools, standardizing runner dependencies, startup behavior, and non-root execution.
- Packaged local Kubernetes platform deployment as Nix apps, generating Helm values and wrapping K3D, kubectl, Helm dependency builds, CRD application, and chart installation into repeatable developer workflows.
- Created reusable Nix devshells for AWS GovCloud Terraform, Platform One Big Bang, Kubernetes labs, and project onboarding workflows, bundling pinned CLIs and project-specific helper commands.

Infrastructure, Security & Platform Engineering

- Built comprehensive security scanning suite: Trivy, govulncheck, Semgrep, Gitleaks, sbomnix SBOM generation — with SARIF output for GitHub Security integration and automated compliance reporting.
- Designed and maintained GitLab CI/CD pipelines with reusable templates, automated STIG scanning, and standardized deployment patterns across multiple teams.
- Designed and operated internal platform engineering capabilities including CI/CD systems, reproducible development environments (Nix), and standardized deployment patterns across teams.
- Built AWS GovCloud infrastructure using Terraform (EC2, IAM, EBS, RDS, AMIs, ACM); created automated OpenVPN deployment with Ansible.
- Implemented HashiCorp Vault for secrets management including disaster recovery tooling (Python export/comparison scripts); deployed containerized applications in Kubernetes using Helm.
- Built Nix devshell for Platform One Big Bang (DoD Kubernetes distribution) wrapping bbctl, Helm, k3d, kubectl, and kustomize into a reproducible environment.
- Contributed to Cyclops platform (ATA's streaming analytics / MLOps / data fabric): Nix dev environment, k3d local Kubernetes, Traefik routing, Keycloak/Grafana ingress.
- Built missile launch data generators and AIS vessel data pipelines for MDA Spot-Lyt defense simulation platform using Kafka and Flink streaming analytics.
- Developed NIST 800-53 Rev 5 compliance analysis tool in Python — cross-references control catalogs with program-specific requirements for automated assessment.
- Mentored team members on DevOps best practices; created documentation, reference implementations, and Nix flake templates to support org-wide adoption.

DevOps / Software Engineer | National Lloyds (now National Summit Insurance Co.)

- Built CI/CD pipeline from zero for a software development team that had been building their enterprise application off a developer's laptop — designed Azure DevOps pipelines, eliminating regressions and disappearing features caused by ad-hoc branch management and manual builds.
- Consolidated three version control systems (SVN, Mercurial, Git) into Azure Repos with standardized branching strategy, merge request workflows, and formal release process — all documented from scratch as the organization's first engineering standards.
- Authored comprehensive release and branching strategy documentation, establishing the team's first formalized development practices; documentation became the reference for all future development workflows.
- Architected Java-based solutions transitioning from IBM AS/400 legacy systems to modern on-prem infrastructure, supporting claims processing, policy administration, and agent portal systems across the property and casualty insurance platform.
- Supported nightly database rebuild processes for production insurance data — participated in on-call rotations to diagnose failures, manually adjust data integrity issues, and restore system operations during overnight maintenance windows.

- Drove cultural transformation in a change-resistant organization — introduced automated builds, code review practices, and agile methodologies to a team with no prior CI/CD experience, fundamentally modernizing their software delivery capability.

Software Engineer / DevOps Engineer | JP Morgan Chase (Global Identity & Access Management — JANUS)

- Built the entire CI/CD pipeline from zero for the JANUS identity platform — designed shared Jenkins libraries, automated build/deploy workflows, and integrated on-prem artifact management, reducing deployment setup from 6 months to minutes across 20+ repositories.
- Authored CI/CD documentation and standards adopted firm-wide — referenced by JP Morgan's internal Jenkins platform team (Jules) and sought out by engineers across global offices for deployment guidance.
- Integrated automated security scanning (HP Fortify, SonarQube, Blackduck) as pipeline gates, establishing shift-left security practices for the legacy Java/Maven application running on RHEL.
- Served as Security Champion — audited the JANUS codebase and uncovered critical encryption standard violations missed by the standing team; escalated findings to management, coordinated remediation with security leads, and drove rapid resolution of severe vulnerabilities.
- Mentored team members and drove adoption of standardized branching strategy, automated unit testing frameworks, and modern development practices for a legacy Java/Tomcat enterprise application.

Software Engineer | L3 Technologies

Gorgon Stare ISR Platform

- Developed web-based geospatial intelligence application (.NET MVC, jQuery/JavaScript) for the Gorgon Stare wide-area persistent surveillance program — aggregated multi-source intelligence data with live video streaming from MQ-9 Reaper UAV sensor pods onto a unified web display.
- Primary Puppet developer — automated provisioning of Windows Server 2008 and Linux environments in air-gapped SCIFs, managed versioned map layer deployments to Google Maps-based GIS application.
- Engineered versioned map layer deployment system enabling incremental GIS data updates instead of full-layer copies, with date-time versioning that caught missed updates and dramatically accelerated deployment cycles.
- Served as Security Champion — ran SCAP and Nessus vulnerability scans, resolved critical/high findings via Puppet automation, documented attestations for exceptions, and performed NIST 800-53/800-171 compliance remediation across the Gorgon Stare server fleet.
- Virtualized SCIF test environments — created VM replicas of production servers so developers could test changes locally instead of manual deployment, significantly improving development velocity.
- Engineered cross-hardware compatibility solution — used Puppet to force Dell RAID controller drivers across hardware generations (R420/R430), enabling backup/restore portability despite vendor RAID controller changes.
- Diagnosed and resolved PostgreSQL garbage collection issue on the production database — identified root cause from documentation and applied configuration fix, restoring database performance.

NCCT

- Built custom Apache NiFi processors (Java) to simulate real-time sensor and targeting data feeds for the Network Centric Targeting (NCCT) legacy application, replacing manual QA data creation and accelerating test/feature development cycles.
- Mentored NCCT colleagues on Puppet configuration management, helping them become self-sufficient practitioners and improving infrastructure automation capabilities across teams.

PROJECTS

forge-nexus gitlab.com/michaelboterf/forge-nexus
Dendritic NixOS monorepo managing 8 hosts (4 workstations, 4 lab servers) via flake-parts and import-tree auto-discovery with Colmena parallel deployment. Implements Authentik SSO, Netbird VPN mesh, sops-nix encrypted secrets, Cloudflare tunnels, and Stylix theming.
NixOS | flake-parts | Colmena | sops-nix | Authentik | Netbird

Portfolio Platform gitlab.com/michaelboterf
Cloud-native platform portfolio spanning Azure, AWS GovCloud, and Kubernetes lab environments. Built reproducible Nix / flake-parts developer workflows, OpenTofu-managed cloud infrastructure, Kubernetes lab tooling, and Nix-built OCI artifacts with CI/CD security scanning.
Nix | flake-parts | OpenTofu | Kubernetes | Azure | Go

mkresume gitlab.com/michaelboterf/forge-nexus
Resume-as-code: single TOML metadata file generates PDF (Typst), HTML (dark portfolio theme), JSON, Markdown, and DOCX via a Python build script packaged as a Nix flake. Runs with nix run .#mkresume.
Python | Nix | Typst | Pandoc | TOML

Homelab (4-node cluster)
HP EliteDesk cluster running NixOS with fully declarative configuration, Syncthing file sync hub, Docker workloads, and Cloudflared edge tunnels. Serves as production testbed for infrastructure patterns before cloud deployment.
NixOS | Docker | Cloudflared | Syncthing | Authentik

MILITARY SERVICE

Corporal, Helicopter Mechanic (H-1 Airframes) | United States Marine Corps

Inspected and maintained helicopter airframes and components including flight line operations for UH-1Y/AH-1Z. Instructed and led Marines in career development. Earned BS in Software Engineering while serving full-time. Assisted in C++ rewrite of the Maintenance Support Application for UH/AH-1 maintenance teams at Weapons Systems Support Activity (WSSA), NAWS China Lake — simplifying the interface and preparing the application for future mission data card migration.

Awards: Navy and Marine Corps Achievement Medal | Marine Aviation Detachment Marine of the Year | Marine Aviation Detachment Marine of the Quarter (multiple) | Regional Navy League Joint Service Junior Blue Jacket of the Year

EDUCATION

Master of Business Administration | University of Phoenix

Bachelor of Science in Software Engineering (earned while serving full-time USMC) | University of Phoenix