**White Paper: Modern Software Development Frameworks**

Enabling Agile, Scalable, and Cloud-Native Solutions (VA Context)

White Paper: Modern Software Development Frameworks – Enabling Agile, Scalable, and Cloud-Native Solutions (VA Context)

## Executive Summary

Software delivery velocity in federal environments, particularly at the Department of Veterans Affairs (VA), requires new practices that embed security, automation, and governance throughout development. This white paper provides a comprehensive breakdown of VA’s approach to DevSecOps, infrastructure as code (IaC), continuous ATO compliance, and secure CI/CD pipelines. Using DoD DevSecOps patterns and tools aligned with VA's TRM and FedRAMP standards, it explains how modern frameworks enable scalable, cloud-native software delivery.

1. DevSecOps Toolchains

## 1.1 Jenkins

- What it is: Jenkins is an open-source CI/CD automation server.

- Key Features:

- Declarative pipeline-as-code using Groovy (Jenkinsfiles).

- Integrates with Git repositories, Kubernetes, Docker, and SonarQube.

- Rich plugin ecosystem for security tools, build agents, notifications.

- VA Use Case:

- Jenkins orchestrates full CI/CD pipelines for VA.gov and My Education Benefits.

- Pipelines embed SAST (SonarQube), DAST (OWASP ZAP), artifact signing, and image scanning.

## 1.2 GitHub Actions

- What it is: CI/CD engine built into GitHub repositories.

- Key Features:

- Triggered by code events (PRs, commits).

- Supports workflows with matrix builds and conditional logic.

- Secrets management and integration with Dependabot.

- VA Use Case:

- Enforce pre-merge security scans.

- Enable pipeline-as-code across VA GitHub organizations.

## 1.3 GitLab CI/CD

- What it is: An all-in-one Git platform with CI/CD and registry support.

- Key Features:

- GitOps-ready pipelines with container scanning, auto-deployments.

- Built-in issue tracking and static/dynamic scanning (GitLab Ultimate).

- VA Use Case:

- Used by VA teams adopting GitOps for Kubernetes deployments.

- Enables end-to-end pipelines with traceability across repositories.

2. Infrastructure as Code (IaC)

## 2.1 Terraform

- What it is: HashiCorp's declarative IaC tool for provisioning infrastructure.

- Benefits:

- Reusable modules across environments.

- Infrastructure state tracking and drift detection.

- Broad support for AWS, Azure, GCP, and Kubernetes.

- VA Use Case:

- Used to provision secure environments (EC2, IAM roles, VPC) for VA.gov applications.

- Security Best Practices:

- Secrets managed with Vault.

- Policies enforced via OPA and Sentinel for FedRAMP alignment.

## 2.2 AWS CloudFormation

- What it is: AWS-native IaC tool using JSON/YAML templates.

- Features:

- Rollback protection and nested stacks.

- Integrated with AWS Config and CloudWatch.

- VA Use Case:

- Templates validated with CFN Lint and CFN Guard.

- IaC scanned in CI pipelines for security misconfigurations.

3. ATO Integration and Continuous Compliance

## 3.1 Continuous Compliance in Pipelines

- Purpose: Automates STIG checks and documentation as part of software delivery.

- Steps:

- Run OpenSCAP and InSpec to verify STIG compliance.

- Validate Terraform and CloudFormation templates against TRM.

- Sign and hash build artifacts using Cosign/Sigstore.

- Outputs:

- SSP (System Security Plan)

- CIS (Control Implementation Summary)

- Immutable audit logs (stored in S3, timestamped)

## 3.2 DoD DevSecOps Reference Design

- Components:

- Iron Bank: Source for hardened container images.

- Platform One: DevSecOps pipeline service with compliance baked in.

- Pipeline Phases:

1. Plan & Code

2. Build & Test

3. Release

4. Deploy

5. Operate & Monitor

- VA Usage:

- Adopted in VEO and Lighthouse programs for secure CI/CD.

- Integrated into Kubernetes deployments for mission systems.

4. Secure CI/CD Pipeline Example

## Scenario: Deploying a new VA.gov backend service to AWS GovCloud.

- Pipeline Stages:

1. Code committed to GitHub.

2. Jenkins or GitHub Actions runs unit tests, code scans.

3. Container image pulled from Iron Bank and scanned with Anchore.

4. Terraform provisions cloud infrastructure.

5. InSpec validates STIGs.

6. Logs are streamed to Splunk.

## Security Controls:

- IAM policies reviewed using OPA/Rego policies.

- All build artifacts are signed and encrypted.

- CloudTrail logs retained per compliance requirements.

## 5. Tools and Resources

| Function | Tools |

|------------------------|---------------------------------------------------|

| CI/CD | Jenkins, GitHub Actions, GitLab |

| Security Scanning | SonarQube, Trivy, Anchore, OWASP ZAP |

| Infrastructure as Code| Terraform, AWS CloudFormation |

| Compliance Validation | OpenSCAP, InSpec, CFN Guard |

| Artifact Security | Sigstore, Cosign, SOPS |

| DoD Tools | Iron Bank, Platform One |

| Documentation | DoD DSOP Docs, Terraform Registry |

## Conclusion

By integrating DevSecOps, IaC, and secure CI/CD pipelines, the VA accelerates delivery of compliant, secure, and scalable digital services. These frameworks transform the development lifecycle and support VA’s mission to provide high-quality experiences to Veterans and stakeholders.

## Next Steps:

- Standardize frameworks and language support across development teams.

- Align Agile practices to modular, cloud-native deployments.

- Apply 12-Factor App and OpenTelemetry for quality and observability.

- Scale Kubernetes and GitOps practices across portfolios.