Cloud-Native DevOps Checklist: CI/CD, Kubernetes and Infrastructure-as-Code

Author: Kamakshaiah Musunuru | **Published:** 11 September 2026 | **Category:** Resource | **Read time:** 2 min read

Tags: DevOps, CI/CD, Kubernetes, Terraform, AWS, cloud, SRE

Cloud-native development is as much about culture as technology. This checklist distils what a mature DevOps practice looks like — use it to audit your own environment before or during a modernisation programme.

CI/CD Pipelines

- Every commit triggers automated linting, unit tests and security scanning (SAST/secret detection) - Builds are reproducible from committed source — no "it works on my machine" - Deployment is automated with rollback as a first-class capability - Environments (dev, staging, prod) are consistent and promoted through the pipeline - Secrets never live in code; inject them from a secrets manager at runtime

Containerisation and Orchestration

- Applications ship as immutable container images with pinned, scanable base images - Containers run as non-root and are free of build tools and unnecessary packages - Kubernetes clusters (or managed EKS/AKS/GKE) apply network policies and pod security - Resource requests and limits are set; health checks (liveness/readiness) are defined - Image tags are pinned and reviewed, with vulnerable images blocked in CI

Infrastructure-as-Code

- All infrastructure is declarative (Terraform/Pulumi/CloudFormation) and version-controlled - Changes follow pull-request review and plan review before apply - State is stored remotely and locked against concurrent modification - Reproducible landing zones exist for multi-account or multi-environment setups - Drift detection and automated remediation are enabled

Observability and Reliability (SRE)

- Golden signals are collected: latency, traffic, errors and saturation - Distributed tracing is available across services - SLIs and SLOs are defined with error budgets that guide release decisions - Incident response has runbooks, an on-call rotation and blameless postmortems - Logs are retained, searchable and protected against tampering

Security in the Pipeline

- Dependency scanning (SCA) and container scanning run on every build - Infrastructure is scanned for misconfiguration before apply - Secrets scans and SAST run in CI; DAST runs against staging - Regular penetration testing validates the whole posture

Closing Thoughts

You do not need to adopt everything at once. Prioritise the items that reduce your biggest risks, then mature incrementally. Codingfigs helps teams build and audit DevOps and platform engineering practices end to end — reach out for an assessment.

Read online: https://codingfigs.com/knowledge/cloud-native-devops-checklist-cicd-kubernetes-and-infrastructure-as-code/