weaveworks

YOUR 16-POINT CHECKLIST FOR GITOPS SUCCESS

Team culture

- [ ] We have documented a clear workflow between Application Development teams and the Platform team
- [ ] We have trained the teams on the new workflows and tooling
- [ ] We have identified which changes can be automatically deployed to production, and which require a manual pull request

Git management

- [ ] We have declared everything in Git (this includes applications, infrastructure, networking, configuration)
- [ ] We have decided the structure for our Git repositories. This includes whether to use Kustomize or branches for different environments e.g. dev, staging and prod. Does each microservice have its own Git repository or are they grouped in the same repository by application.
- [ ] We have connected GitOps tooling like Flux, Helm, and Kustomize to our Git repositories

GitOps pipeline

- [ ] We have selected the appropriate tooling that makes up our GitOps pipeline (Flux, Helm, Flagger, etc)
- [ ] Configured Git webhook for build trigger
- [ ] Completely automated the workload through GitOps Pipelines so the clusters are "always kept reconciled" with changes made in the Git repository.
- [ ] We have automated a majority of testing on code
- [ ] We have made test runs to automatically deploy changes to different environments using the new GitOps pipeline

Kubernetes

- [ ] We have adopted Kubernetes for container and infrastructure management
- [ ] We have decided where we would host our Kubernetes clusters (AWS EKS, OpenShift, etc)

Security & Policies

- [ ] We have set up policies to run security checks end-to-end from Git to pipeline tooling to Kubernetes clusters. (For example, leveraging a policy engine like Weave GitOps Trusted Delivery)
- [ ] We use a dedicated secrets management tooling to manage sensitive data
- [ ] We have ensured that only the Admin has direct access to production Kubernetes clusters