Flux CLI GitOps for Kubernetes

CONCEPTS

Flux pulls: a Source (Git, OCI, Helm repo, bucket) is fetched, then a Kustomization or HelmRelease applies it. The CLI mostly inspects and nudges; Git stays the source of truth.

Source GitRepository, OCI, HelmRepository
↓ artifact
Kustomization / HelmRelease what to apply, how often
↓ apply
Cluster live objects, drift corrected

Commands default to the flux-system namespace. Add -n <ns> or -A, or --ns-follows-kube-context to use the kubeconfig context's namespace.

Anything that changes the cluster directly is overwritten by the next sync of the Kustomization that owns it. Change Git instead, and use create --export to generate the YAML.

INSTALL & BOOTSTRAP

flux check --pre
Check cluster meets prerequisites

flux bootstrap github --owner=*<org>* --repository=*<repo>* --path=*<dir>*
Install Flux and sync from GitHub

flux check
Verify controllers are healthy

Bootstrap commits the controller manifests and a sync config under --path, so Flux then manages its own upgrades from Git.

INSPECT

flux get all
Everything in one namespace

flux get all -A
Everything in all namespaces

INSPECT (cont.)

flux get kustomizations
List Kustomizations and readiness

flux get helmreleases -A
List Helm releases

flux get all -A --status-selector ready=false
Only the broken ones

RECONCILE & SUSPEND

flux reconcile kustomization *<name>*
Apply now, skip the interval

flux reconcile kustomization *<name>* --with-source
Refetch source, then apply

flux reconcile helmrelease *<name>*
Re-run a Helm release

flux suspend kustomization *<name>*
Pause one Kustomization

flux resume kustomization *<name>*
Resume and reconcile

flux suspend helmrelease *<name>*
Pause a Helm release

flux resume helmrelease *<name>*
Resume a Helm release

CREATE RESOURCES

flux create source git *<name>* --url=*<url>* --branch=*<b>*
Define a Git source

flux create kustomization *<name>* --source=*<src>* --path=*<dir>* --prune
Apply a path from a source

flux create helmrelease *<name>* --source=HelmRepository/*<repo>* --chart=*<chart>*
Install a chart

flux create helmrelease *<name>* ... --values=*<file>*
Pass local values files

CREATE RESOURCES (cont.)

flux create helmrelease *<name>* ... --create-target-namespace
Install into a new namespace

flux create secret git *<name>* --url=ssh://git@*<host>*/*<repo>*
Generate an SSH deploy key secret

flux create source git *<name>* ... --export > *<file>*.yaml
Write YAML instead of applying

flux export kustomization --all
Dump all Kustomizations as YAML

DEBUG

flux logs -A
Controller logs, all namespaces

flux logs -A --level=error
Only errors

flux logs --kind=Kustomization --name=*<name>*
Logs for one object

flux events
Events for Flux objects

flux events --for Kustomization/*<name>*
Events for one object

flux trace *<kind>* *<name>* -n *<ns>*
Which source manages an object

BUILD & DIFF

flux diff kustomization *<name>* --path *<dir>*
Local changes vs the cluster

GLOBAL FLAGS

--context *<name>* / --kubeconfig *<file>*
Choose cluster or kubeconfig

-n *<ns>* / -A
Namespace or all namespaces

--export
Print YAML instead of applying