KubeCon CloudNativeCon
Europe 2022

# Flux Security Deep Dive

Stefan Prodan
Flux & Flagger Maintainer
Principal Engineer @ Weaveworks

# Flux - Project overview



The Flux project aims to provide a complete **Continuous Delivery** platform on top of Kubernetes, supporting all the common practices and tooling in the field.

**Flux** v2 is powered by the **GitOps Toolkit**, a set of composable APIs and specialized tools for keeping Kubernetes clusters in sync with sources of configuration, and automating updates to configuration when there is new code to deploy.

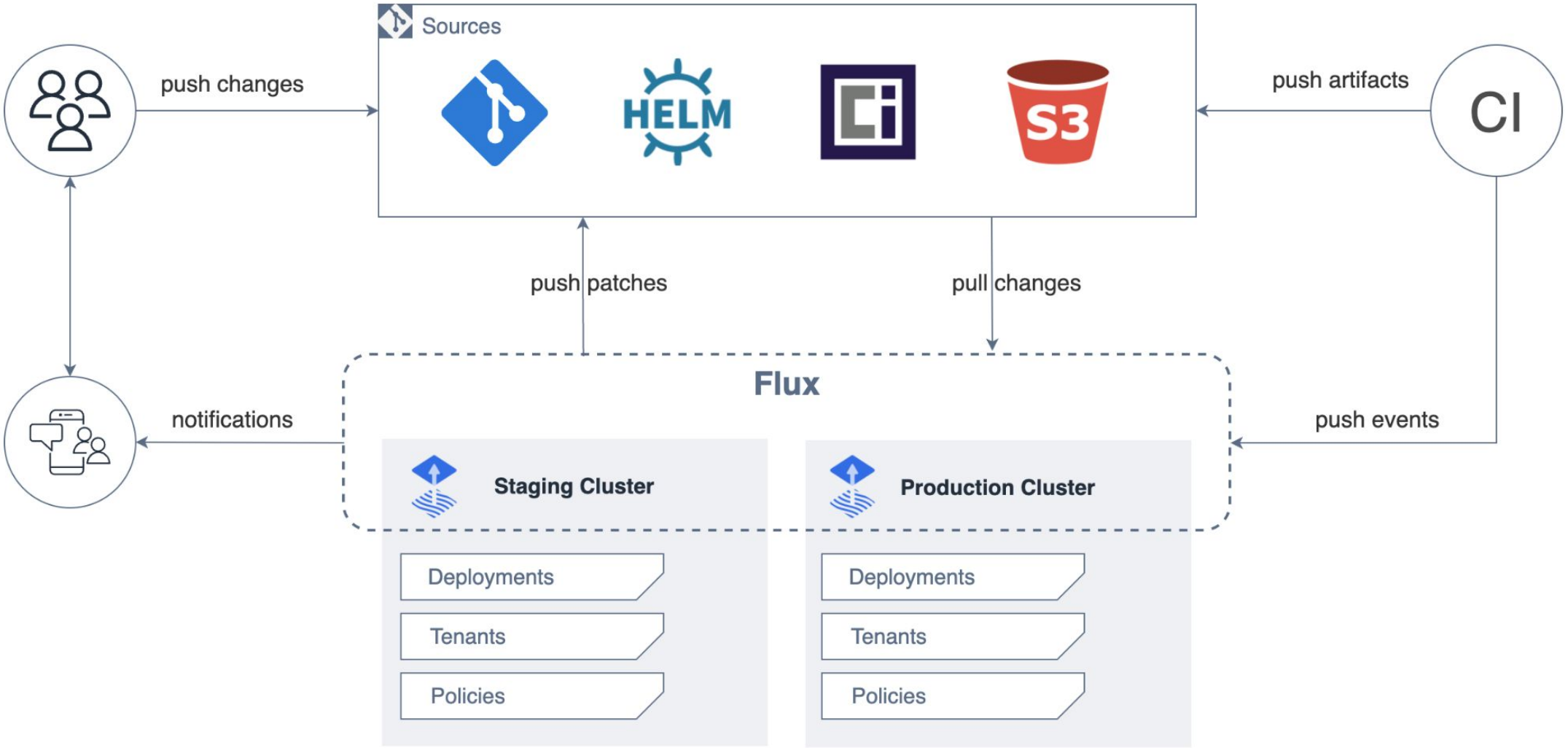
https://github.com/fluxcd/flux2

**Flagger** is a **Progressive Delivery** tool that automates the release process for applications running on Kubernetes. Flagger comes with a declarative model for **decoupling** the deployment of apps on Kubernetes from the release process.

https://github.com/fluxcd/flagger

# Flux - GitOps Continuous Delivery





# Agenda

- How is Flux made
- How secure is Flux
- Are my secrets safe in Git
- Is Kubernetes really multi-tenant
- Flux soft vs hard multi-tenancy
- When will Flux v2 be GA

# What is Flux made of?



Flux is made of many things

- Kubernetes API extensions (CRDs)
- Specialized Kubernetes operators (powered by controller-runtime)
- Flux command-line tool (powered by Kubernetes cli-utils)
- Flux Terraform provider
- Go & C libraries (Go stdlib, Kubernetes client-go, kstatus, go-git, **libgit2**, kustomize, helm, minio, oras, sops, age, aws, azure, gcp, github, gitlab, bitbucket SDKs… and some more)

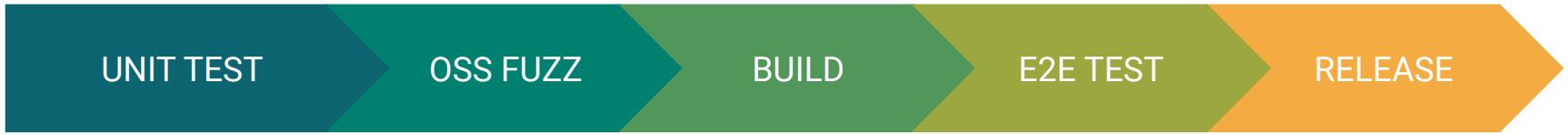
# How can a handful of people maintain such a project?

With care…
A helpful community
And lots of automations

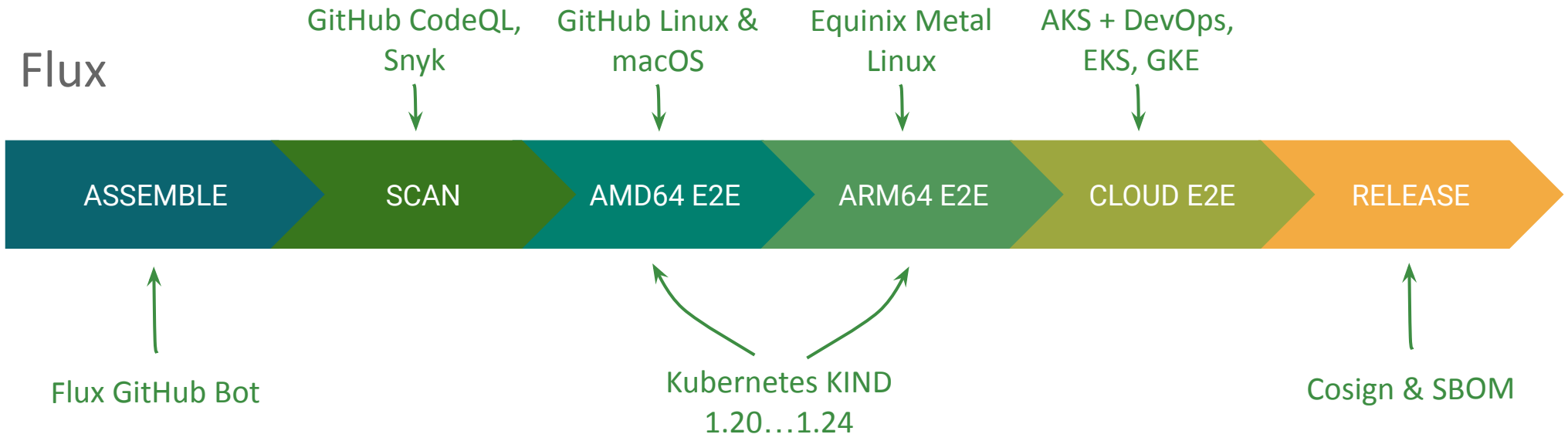
# Flux - release pipeline



Libraries, controllers and CLI




Flux

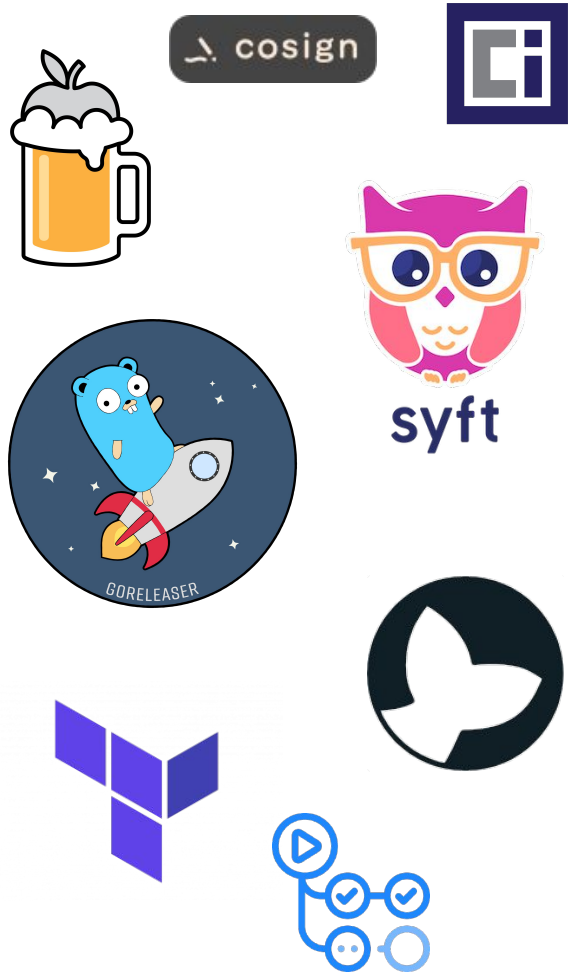


# Flux - release artifacts



A Flux release is comprised of

- Multi-arch container images (GHCR & DockerHub)
- Signed images and checksums (Cosign + GitHub OIDC)
- Software Bill of Materials (SBOM SPDX)
- Deployment manifests (YAML)
- OpenAPI specs (JSON)
- CLI binaries (Linux, macOS & Windows)
- Packages (Homebrew, Arch Aur, NIX, Chocolatey)
- Flux Terraform provider (Terraform Registry)
- Flux GitHub Action (AMD64 & ARM64 runners)



# What makes the Flux controllers secure?

- No shell-out to 3rd party binaries
- All Linux capabilities are dropped
- The root filesystem is set to read-only
- The seccomp profile is runtime default
- Controllers run as non-root
- Uses Kubernetes impersonation API

# Flux vs competition?



Unlike most CD products, Flux has a small attack surface

- Flux controllers are statically built and have no dependencies on OS packages
- No shell-exec to git, kubectl, helm, kustomize, sops, aws, gcloud, etc
- No HTTP APIs, you can control Flux only via Kubernetes API
- All actions performed on the cluster are auditable and subject to Kubernetes RBAC
- Flux execution is predictable, there are no plugins nor scripting
- Flux can only be extended with other controllers that adhere to the GitOps Toolkit std

# Who trusts in Flux?



Flux is embedded in

- Azure Arc
- Amazon EKS Anywhere
- VMware Tanzu
- D2iQ Enterprise Kubernetes Platform
- Platform One (US DoD & US Air Force)
- Deutsche Telekom Das Schiff
- And many more


Amazon EKS

VMware Tanzu

Azure

# How secure is Flux?



- In 2021 Flux has undergone a security audit (OSTIF & ADA Logics)
  - We've addressed all the security issues found in record time
  - We've put in place an RFC process for changes to Flux security posture
- In 2022 the Flux team focused on security hardening
  - We've found and addressed a series of multi-tenancy vulnerabilities
  - We've made secrets decryption safer on multi-tenant environments
  - We've improved the test coverage of sensitive operations
- Flux is scheduled to undergo a security review by CNCF TAG Security

# Is Flux bulletproof?



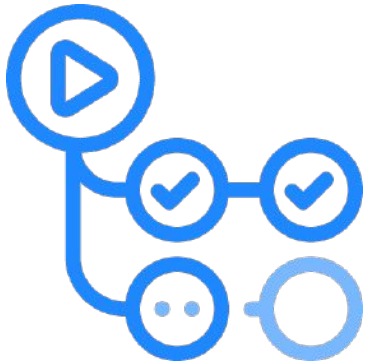
| Date | CVE | Title | Severity | Affected version(s) | Reported by |
|---|---|---|---|---|---|
| 2021-11-10 | CVE-2021-41254 | Privilege escalation to cluster admin on multi-tenant Flux | High | < 0.18.0 | ADA Logics |
| 2022-05-04 | CVE-2022-24817 | Improper kubeconfig validation allows arbitrary code execution | Critical | < 0.29.0 >= v0.1.0 | The Flux Team |
| 2022-05-04 | CVE-2022-24877 | Improper path handling in Kustomization files allows path traversal | Critical | < v0.29.0 | The Flux Team |
| 2022-05-04 | CVE-2022-24878 | Improper path handling in Kustomization files allows for denial of service | High | < v0.29.0 >= v0.19.0 | The Flux Team |

# How to keep Flux up-to-date?



Flux is able to **update itself** from Git.

We offer a **GitHub Action** that checks for new releases and opens a pull request on your bootstrap repository when a newer Flux version is available.



For GitLab, BitBucket, Azure DevOps and other platforms, you can use **Renovate Bot** which offers the same update automation for Flux.

# What security challenges come with GitOps?

- Keeping secrets safe
- Restricting access to sensitive data
- Compromised Git credentials
- Prevent destructive cluster ops

# Are my secrets safe in Git?



Flux comes with built-in secrets management

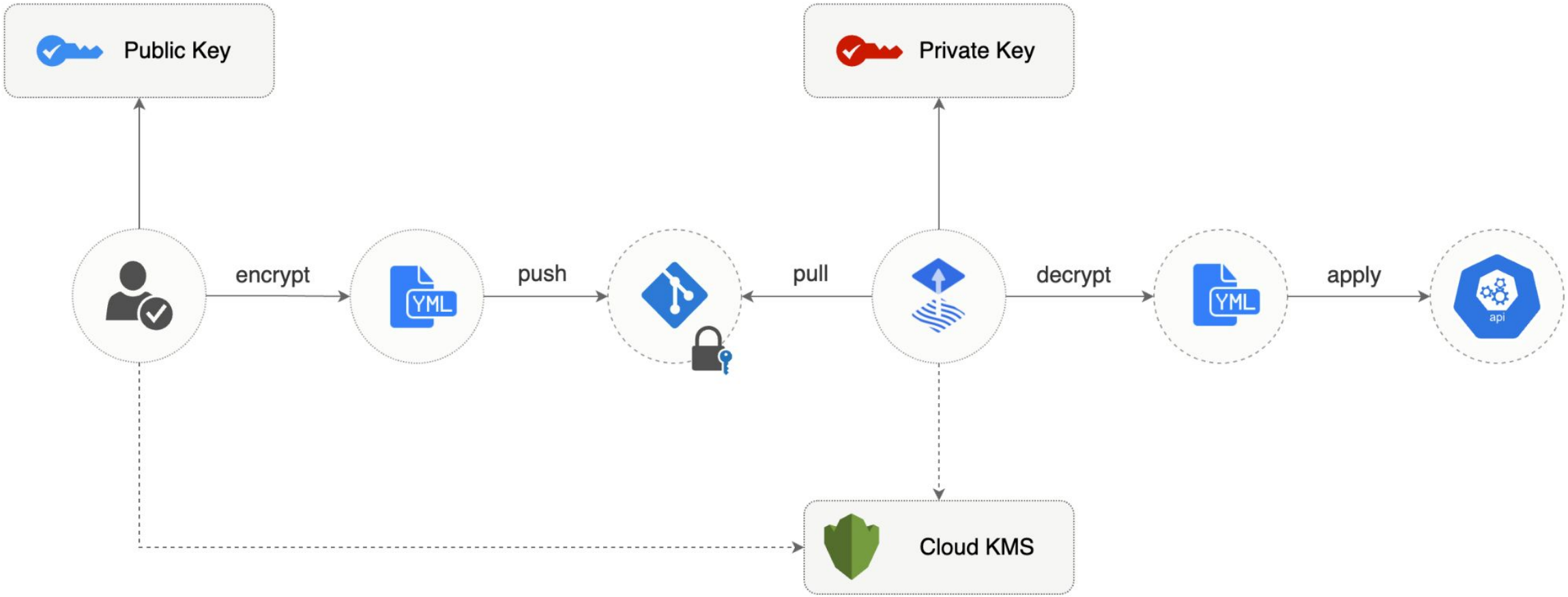
- Client-side encryption with Mozilla SOPS
- Server-side decryption with Flux
- Supported technologies
  - Age Encryption and OpenPGP
  - Hashicorp Vault
  - AWS Key Management Service
  - Azure Key Vault
  - Google Cloud KMS




The Flux team is committed to SOPS' development and maintenance

# Secrets operations





# Is Kubernetes truly multi-tenant?

In some regards **YES** but soft multi-tenancy is difficult to secure while hard multi-tenancy can be easier to reason with but hard to orchestrate.

# Tenant isolation boundaries



Flux bridges the gap between Kubernetes and Git tenancy models.

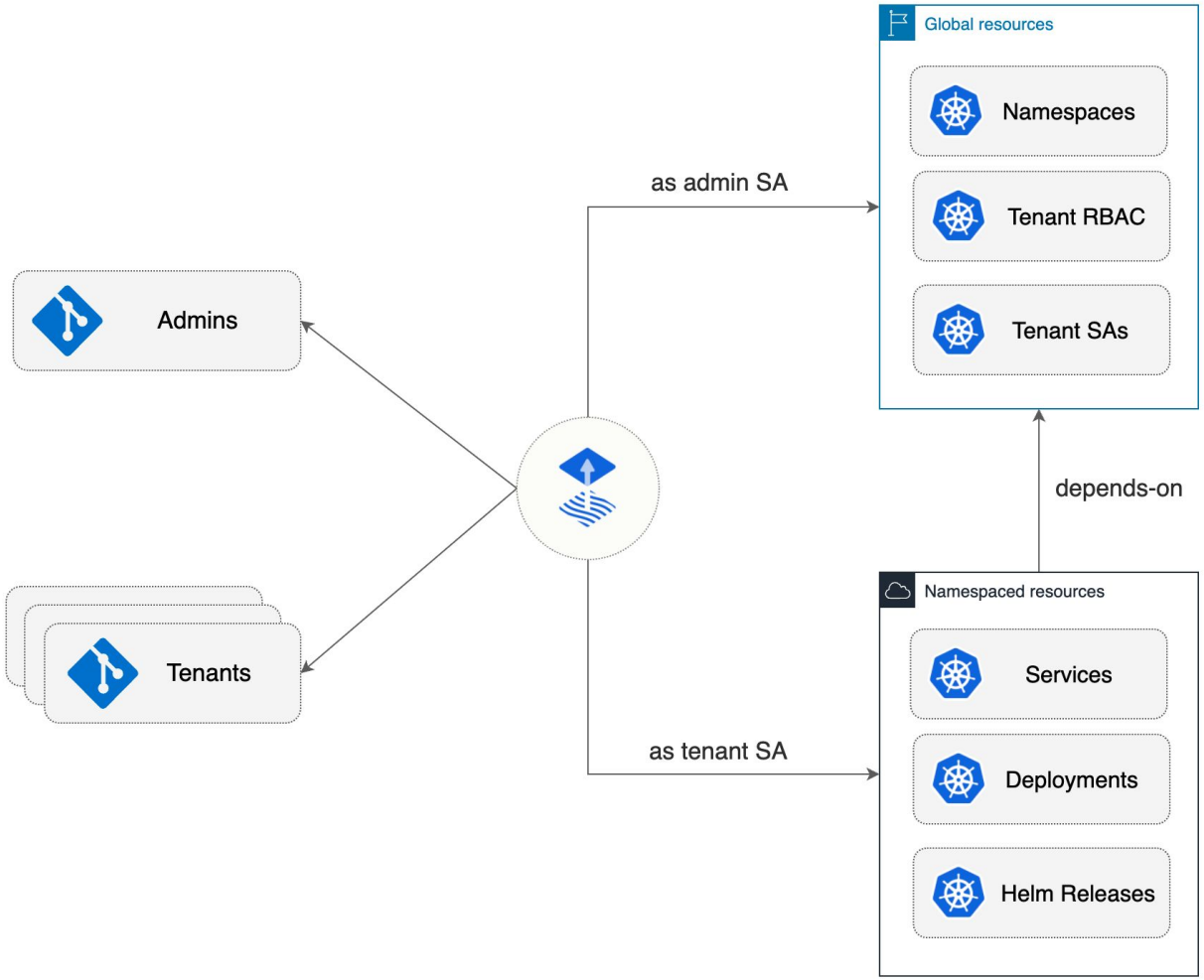
- Kubernetes
  - Dedicated clusters per tenant (hard multi-tenancy)
  - Namespaces and role bindings
  - Node groups, taints and tolerations
  - Resource quotas and network policies
  - 3rd party policies (OPA & Kyverno)
- Git
  - Dedicated repositories per tenant (hard multi-tenancy)
  - Protected branches
  - Team access management (GitHub, GitLab, etc)

# Flux - GitOps Multi-Tenancy



Flux enables multi-tenancy by allowing platform admins to assign restricted Kubernetes accounts to the tenants' sources.

When Flux reconciles the tenant's Kubernetes resources, it does so by impersonating the tenant's account, thus enforcing the isolation boundary as defined by platform admins in their Git repo.



# Demo

Multi-tenancy with Flux and Kubernetes
namespace-as-a-service

# When will Flux v2 reach GA?

https://fluxcd.io/roadmap

## TODOs

- Adopt kstatus for all Flux APIs
- Helm controller refactoring
- Support for Helm OCI
- Notification API improvements
- Documentation refactoring

# Additional Resources

https://fluxcd.io/security

https://fluxcd.io/docs/security

https://github.com/fluxcd/flux2-multi-tenancy

https://fluxcd.io/docs/guides/mozilla-sops/

KubeCon | CloudNativeCon

Europe 2022

THANK YOU!

https://twitter.com/stefanprodan