INDONESIA LINUX CONFERENCE 2022

GitOps - Intro

Implement Infrastructure as code

Present By: **Estu Fardani | @tuanpembual**

INDONESIA LINUX CONFERENCE 2022

Hello & Welcome

Estu Fardani
DevOps Consultant
openSUSE ID Member

GitOps itu apa?

GitOps adalah cara mengimplementasikan Continuous Deployment pada penyedia layanan awan.

Ide utamanya adalah ada satu lumbung git, yang selalu berisi deskripsi deklarasi kondisi infrastruktur awan dan memiliki proses otomatisasi untuk selalu sinkron antara di lumbung git dan kondisi sebenarnya.

Jika ingin melakukan perubahan infrastruktur cukup dengan melakukan perubahan di lumbung git tersebut kemudian diteruskan menggunakan alur kerja git.

> GitOps: versioned CI/CD on top of declarative infrastructure. Stop scripting and start shipping.
>
> — Kelsey Hightower

Alasan mengadopsi GitOps

Deploy Lebih Cepat, Lebih Sering

Perubahan infrastruktur hanya melewati satu mekanis, git flow. Sesederhana membuat Merge Request (MR).

Alasan mengadopsi GitOps

Mudah dan Cepat dalam Perbaikan Kesalahan

Ketika ada kekeliruan konfigurasi, kita memiliki riwayat lengkap perubahan. Lakukan revert MR untuk perbaikan kesalahan. *PS: tapi kalo kesalahan delete sumber daya tetap saja datanya hilang :D

The Git record is then not just an audit log but also a transaction log. You can roll back & forth to any snapshot.

— Alexis Richardson

Alasan mengadopsi GitOps

Kemudahan dalam Manajemen Kredensial (Credential Management)

GitOps memungkinkan untuk tidak menyimpan username password pada kode. Membatasi akses pada pengembang, karena hanya mekanisme deployment yang membutuhkan hal tersebut.

> kubectl is the new ssh. Limit access and only use it for deployments when better tooling is not available.
>
> — Kelsey Hightower

Alasan mengadopsi GitOps

Dokumentasi Deployment Otomatis

Kita tidak perlu lagi ssh ke setiap mesin untuk tau, ada apa disini. Cukup melihat konfigurasi deskripsi di lumbung git untuk mengetahui hal tersebut.

Kita juga bisa mengetahui riwayat penuh terhadap semua perubahan. Sehingga memudahkan audit pada akhirnya.

Alasan mengadopsi GitOps

Berbagi Pengetahuan dalam Tim

Menggunakan Git untuk menyimpan keseluruhan deskripsi konfigurasi infrastruktur membantu semua orang untuk memeriksa perubahan dalam satuan waktu. Dengan mekanisme Merge Request yang detail dan rapi membantu mereproduksi suatu konfigurasi hingga dapat mengaplikasikannya pada sistem yang baru.

> GitOps is the best thing since configuration as code. Git changed how we collaborate, but declarative configuration is the key to dealing with infrastructure at scale, and sets the stage for the next generation of management tools.
>
> — Kelsey Hightower

Bagaimana GitOps bekerja?

Konfigurasi Lingkungan sebagai Lumbung Git

GitOps mengkoordinir deployment pada infrastruktur. Pada umumnya terdapat dua lumbung: lumbung aplikasi dan lumbung konfigurasi lingkungan.

Lumbung aplikasi berisi kode sumber aplikasi dan manifestasi deployment aplikasi. Lumbung konfigurasi lingkungan berisi semua manifestasi penerapan infrastruktur yang diinginkan. Ini menjelaskan bagaimana seharusnya konfigurasi aplikasi dan infrastruktur diatur.

Implementasi GitOps

Push-based vs. Pull-based Deployments

Ada dua cara untuk implementasi strategi deployment di GitOps: Push-based dan Pull-based deployments.

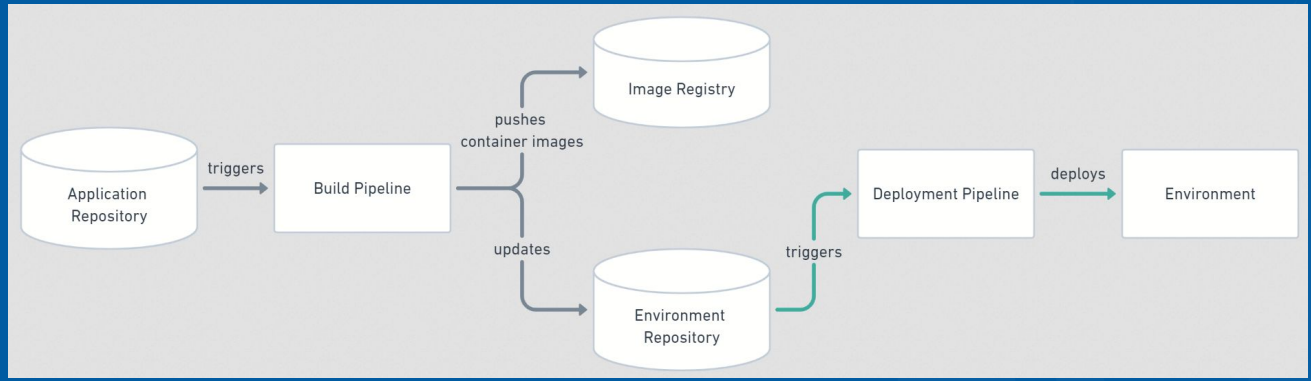
Perbedaan antara keduanya adalah bagaimana memastikan kondisi lingkungan deployment benar-benar menyerupai infrastruktur yang diinginkan.

Pendekan pull-based lebih diutamakan karena dianggap lebih aman dan memiliki praktik lebih baik dalam implementasi GitOps.

Implementasi GitOps

Push-based Deployments

Contoh implementasinya dapat kita lihat pada perkakas seperti Jenkins, CircleCI, or Travis CI. Kode sumber aplikasi berada di dalam lumbung aplikasi bersama dengan YAML Kubernetes yang diperlukan untuk mendeploy aplikasi. Setiap kali kode aplikasi diperbarui, pipeline build dipicu, membangun image container dan akhirnya deploy ke kubernetes.

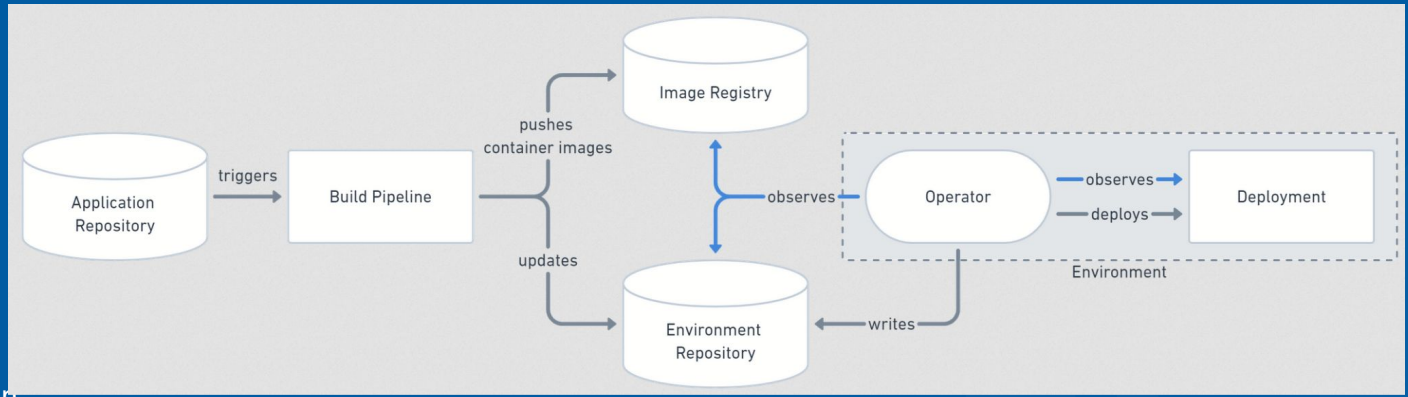


Implementasi GitOps

Pull-based Deployments

konsep strategi Pull-based deployment menggunakan konsep yg sama pada push-based tapi berbeda bagaimana pipeline deployment bekerja. Disini dikenalkan konsep operator.

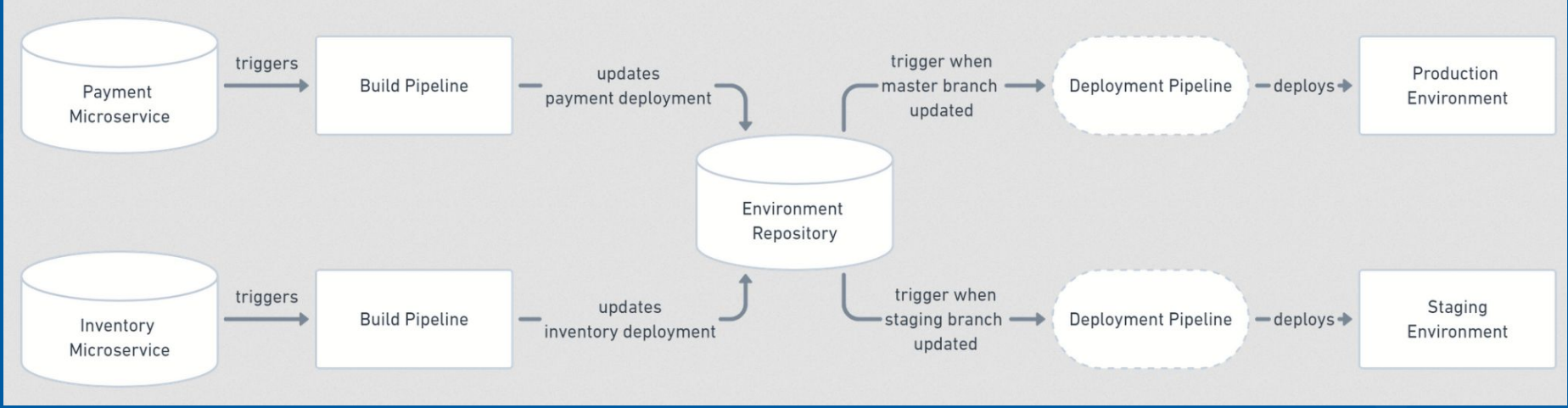
Tugasnya terus menerus melakukan perbandingan kondisi yang diinginkan (pada kode) dengan kondisi saat ini di lingkungan (pada infrastruktur). Ketika ada perbedaan akan dilakukan penyesuaian pada lumbung lingkungan.



Implementasi GitOps

Implementasi pada Multi Lingkungan

Sangat bisa dilakukan. hanya membedakan trigger dan kondisi yang dibutuhkan.



Sampel

Push-base deployment

- Using cloud provider: GCP
- Kubernetes Cluster
- Manage using Terraform
- Semua resource dibuat dari terraform
  - GKE
  - CloudSQL
  - Loadbalancer GCLB
  - Pubsub
  - Redis
  - VM
  - Firewall
  - VPC
  - DNS
- CICD: Gitlab CI + Kubernetes Runner

Sampel

Push-base deployment (2)

- Operator like
  - Gitlab Schedule
  - Check terraform state: repo <> gcp
- Tidak melakukan serupa di sisi lumbung aplikasi
  - Self-create helm chart

Kendala

Banyak:

- Identity Access Management (IAM)
  - Click base deployment
- Flow Git Review
  - MRs - 5-15 a day, approval
  - Green pipeline

Next hops

Perkakas baru:

- ArgoCD
- Flex

INDONESIA LINUX CONFERENCE 2022

Thank you

https://www.ilc.opensuse.id

Biznet GioCloud

SAKTI cooperation for everyone

SUSE

CLOUDKILAT

Valutac

Biznet DataCenter

INGRAM MICRO

eXcellent