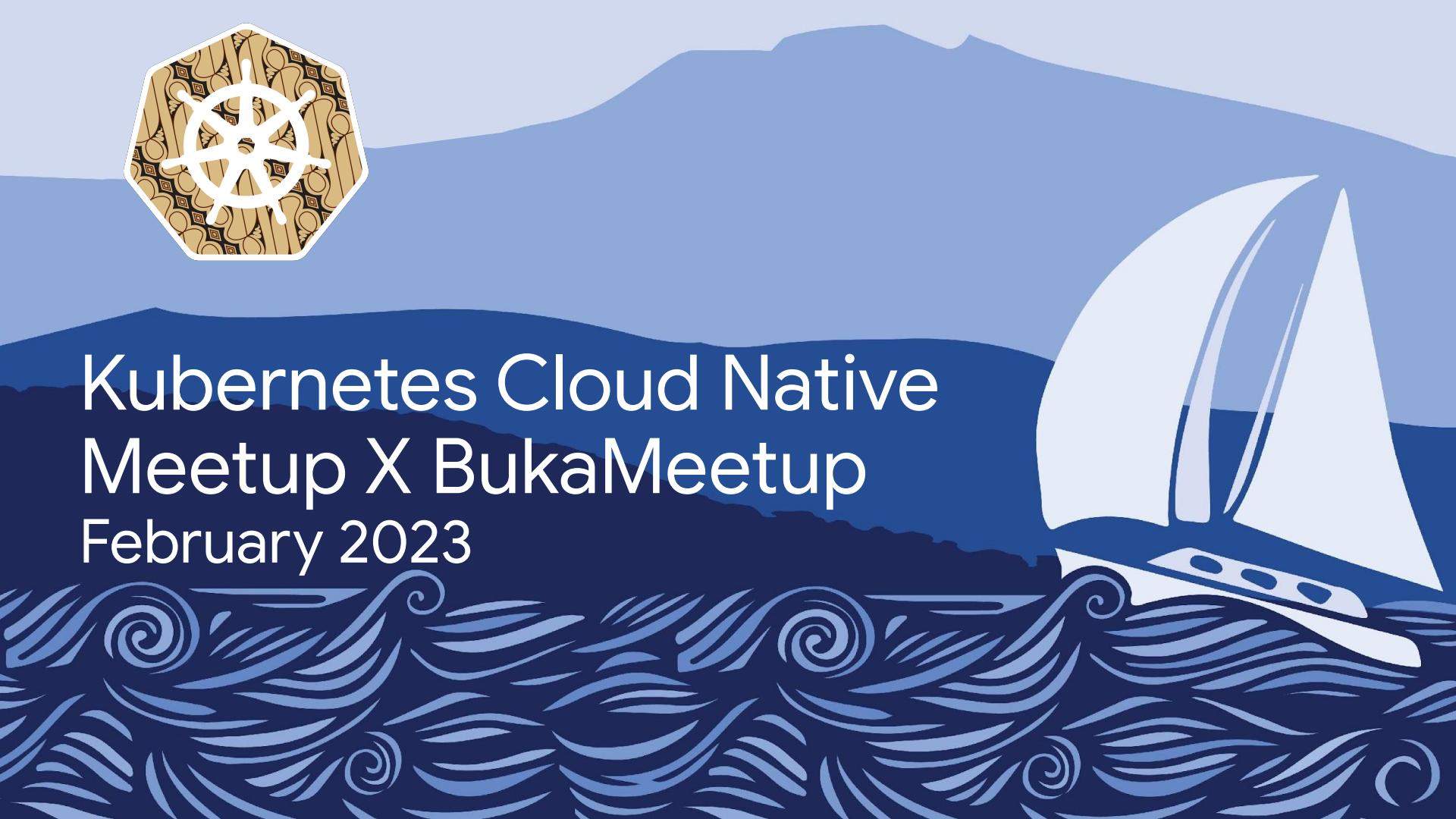




# Kubernetes Cloud Native Meetup X BukaMeetup

## February 2023



# Agenda

- Kubernetes Community Update
- Opening Speech from Bukalapak
- Talk #1 by **Dewangga Alam @ Bukalapak**
- Talk #2 by **Nicolas Julian @ Jubelio**
- Q&A
- Announcements

Update from  
Kubernetes & CloudNative Indonesia

# Last Meetup

August 31, 2022



September 24, 2022

# Last conference: KCD X IOD Indonesia November 26-27 2022



# Last conference: KCD X IOD Indonesia November 26-27 2022

Event Documentation

[bit.ly/kcd-x-iod-indonesia-photos](https://bit.ly/kcd-x-iod-indonesia-photos)



Speakers Talk Playlist

[bit.ly/kcd-x-iod-indonesia-playlist](https://bit.ly/kcd-x-iod-indonesia-playlist)

# Next Conference: KubeCon | CloudNativeCon Europe 2023



April 18-21, 2023 @ Amsterdam

[events.linuxfoundation.org/kubecon-cloudnativecon-europe/](https://events.linuxfoundation.org/kubecon-cloudnativecon-europe/)

## Scholarships

<https://events.linuxfoundation.org/kubecon-cloudnativecon-europe/attend/scholarships/>

A screenshot of the "Scholarships" section of the website. It features a large group photo of diverse attendees. The section title "Scholarships" is at the top left. The event details "KubeCon + CloudNativeCon Europe 2019" are displayed above the photo. Below the photo, text states: "The Dan Kohn Scholarship Fund exists to support individuals who may not otherwise have the opportunity to attend in four categories: DIVERSITY and NEED-BASED." Two boxes describe these categories: "DIVERSITY" for underrepresented groups and "NEED-BASED" for financial assistance.

# Our Speakers

In partnership with:



## Kubernetes Cloud Native Indonesia Meetup Februari 2023

16 Februari 2023 | 19.00 WIB | @Bukalapak HQ  
<https://online.kubernetescommunity.id>



**Dewangga Alam**

Engineering Manager @Bukalapak  
K8s Component Election



**Nicolas Julian**

DevOps Engineer @Jubelio  
Management confidential data with Vault and  
ArgoCD

Hosted by Kubernetes & Cloud Native Indonesia



# Buka Meet Up

February 2023

# Opening Speech:

## Agung Wijayanto

# Let's Join With Us!

 **bukalapak is Hiring**

Together we grow and create a fair economy for all



## DevOps Engineer

### **Responsibilities:**

- Manage infrastructure use CI/CD process and automation tools such as Ansible
- Build script to automate operational and deployment process use programming language like Python, Go, Ruby
- Optimizing Infrastructure Automation thru repetitive refactors and Infra Architecture Review
- Deploying, automating, maintaining, building and managing On-Prem and Cloud-based production systems.
- Develop and integrate monitoring, logging, dashboard, and alert systems to quickly and proactively prevent or detect errors
- Ensure the availability, performance, scalability, and security of Infrastructure.
- Build, release and manage configuration management of infrastructure and application in multiple environments.
- Within a cross-functional team, collaborate with other engineers specializing in backend services, web frontend, mobile apps, and test automation, as well as product design and ideation.

### **Requirements:**

- Bachelor degree in Computer Science or related fields, or equivalent professional experience
- Minimum 2 years of working experience
- Strong understanding of Operating System ( Linux )
- Understand the basic of Security Best Practices
- Solid understanding of DevOps Philosophy, Agile Methods, and Infrastructure as Code
- Humble culture, zero egos, and excellent collaborative spirit. We are all here to learn together and grow together as a team
- Experience in Monitoring Tools like Grafana, Datadog, and Prometheus
- Experience in using Versioning Control tools
- Knowledge & Experience in Go, Python, & Bash
- Experience in one of these following technologies, including Kubernetes, Nginx, Haproxy, Kubernetes Ingress
- Experience in Infrastructure-as-Code and Automation like Ansible or Terraform

[careers.bukalapak.com](http://careers.bukalapak.com)



[bit.ly/DevOpsBL](http://bit.ly/DevOpsBL)

# K8s Component Election

Dewangga Alam



## Dewangga Alam

---

dewanggaba@xtremenitro.org  
t.me/hostmaster

### Working Experience(s) at Bukalapak

- Senior System Engineer (2018-2020)
- Engineering Manager (2020-Now)

### Interest at

- Infrastructure
- Internet Security
- Networking

# Overview

- Start from early 2010 - Now
- Up to **20 Gbps** traffic
- Up to **118 Terabytes** traffic per month
- **90% Infrastructure as a Code (IAC)**
- **3 different cloud provider(s) (2 Singapore & 1 Indonesia)**

# Core Components

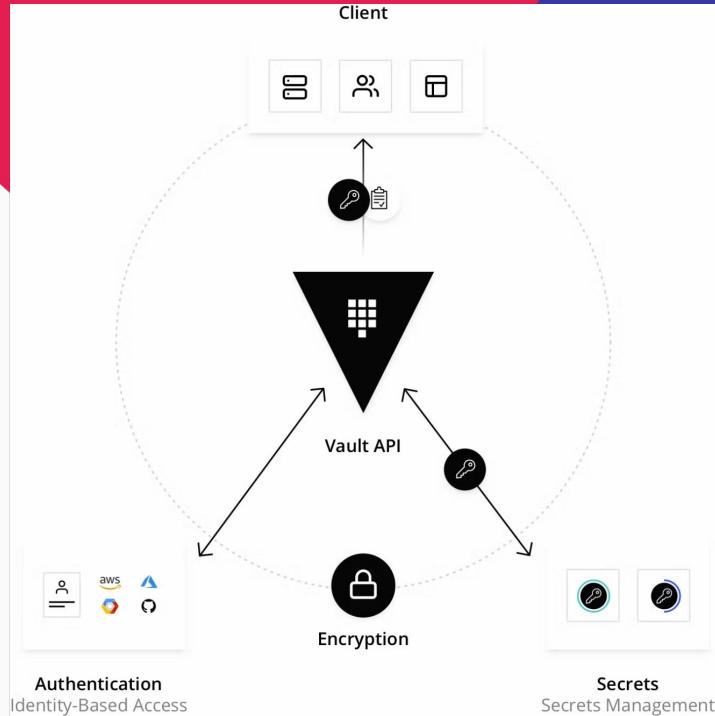
- **Security & Policy**
  - Hashicorp Vault
- **Traffic Management**
  - Ingress Nginx
- **Deployment Strategy**
  - Terraform
  - Helm
  - Gitlab CI
  - ArgoCD

# Security & Policy

## Hashicorp Vault

It's agnostic solution for multi-cloud deployment. We can deploy variety workload across cloud provider.

Ref: AWS Secret Manager, Azure Key Vault



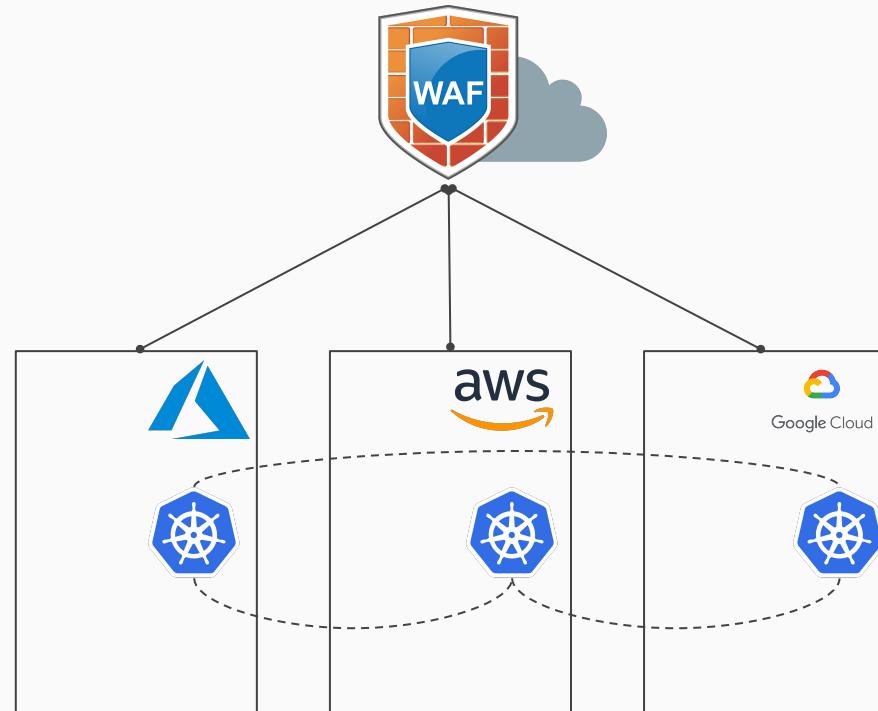
# Why Hashicorp Vault?

- **Agnostics!**
- Allows dynamic secret (ISO 27001 & PCI Standard)
- RBAC Support
- Self-hosted support

# Traffic Management

## Ingress Nginx

Just like other ingress, but we trust in LUA for runtime changes.



# Why Ingress Nginx?

- “Official” Kubernetes Ingress
- Nginx fork with LUA Support (extra features can easily hooked/called)
- Community support (no vendor lock)
- Open sources!

# Deployment Strategy

Terraform

	Source	Cloud	Type	Infrastructure	Language	Agent	Master	Community	Maturity
Chef	Open	All	Config Mgmt	Mutable	Procedural	Yes	Yes	Large	High
Puppet	Open	All	Config Mgmt	Mutable	Declarative	Yes	Yes	Large	High
Ansible	Open	All	Config Mgmt	Mutable	Procedural	No	No	Large	Medium
SaltStack	Open	All	Config Mgmt	Mutable	Declarative	Yes	Yes	Medium	Medium
CloudFormation	Closed	AWS	Provisioning	Immutable	Declarative	No	No	Small	Medium
Heat	Open	All	Provisioning	Immutable	Declarative	No	No	Small	Low
Terraform	Open	All	Provisioning	Immutable	Declarative	No	No	Medium	Low



# Deployment Strategy

## Terraform (cont)

a This is the number of cookbooks in the Chef Supermarket.

b To avoid false positives for the term “chef”, I searched for “chef engineer”.

c Based on the Puppet Labs JIRA account.

d This is the number of modules in the Puppet Forge.

e To avoid false positives for the term “puppet”, I searched for “puppet engineer”.

f This is the number of reusable roles in Ansible Galaxy.

g This is the number of formulas in the Salt Stack Formulas GitHub account.

h This is the number of templates in the awslabs GitHub account.

i Based on the OpenStack bug tracker.

j I could not find any collections of community Heat templates.

k To avoid false positives for the term “heat”, I searched for “openstack”.

l This is the number of modules in the terraform-community-modules repo.

	Source	Cloud	Contributors	Stars	Commits in Sept	Bugs in Sept	Libraries	StackOverflow	Jobs
Chef	Open	All	477	4,439	182	58	3,052 <sup>a</sup>	4,187	5,631 <sup>b</sup>
Puppet	Open	All	432	4,158	79	130 <sup>c</sup>	4,435 <sup>d</sup>	2,639	5,213 <sup>e</sup>
Ansible	Open	All	1,488	18,895	340	315	8,044 <sup>f</sup>	3,633	3,901
SaltStack	Open	All	1,596	6,897	689	347	240 <sup>g</sup>	614	454
CloudFormation	Closed	AWS	?	?	?	?	240 <sup>h</sup>	613	665
Heat	Open	All	283	283	83	36 <sup>i</sup>	0 <sup>j</sup>	52	72 <sup>k</sup>
Terraform	Open	All	653	5,732	440	480	40 <sup>l</sup>	131	392



# Why Terraform?

- Agent-less
- Multi-cloud deployment support in a single workflow
- Community support (no vendor lock)
- Open sources!

# How do we choose?

- Define your goal at the beginning
- Do the PoC(s)
- We prefer leading edge than bleeding edge
- Open Source

# Conclusion

Know your expectations, deliver it on time, panic nicely.



Thanks!



# Our Speakers

In partnership with:



## Kubernetes Cloud Native Indonesia Meetup Februari 2023

16 Februari 2023 | 19.00 WIB | @Bukalapak HQ  
<https://online.kubernetescommunity.id>



**Dewangga Alam**

Engineering Manager @Bukalapak  
K8s Component Election

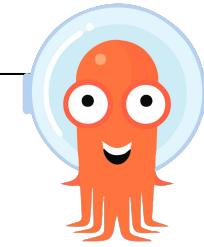


**Nicolas Julian**

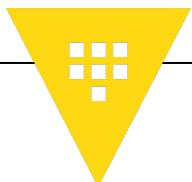
DevOps Engineer @Jubelio  
Management confidential data with Vault and  
ArgoCD

Hosted by Kubernetes & Cloud Native Indonesia

# Management **confidential** data with Vault and ArgoCD



argo



HashiCorp  
Vault



# Kulo Sopo ?

**Kulo Nicolas Julian**

Sekarang bekerja sebagai “Kuli Server” di **Jubelio**.

**Latar belakang**, Seorang warga sipil bisa yang suka ngopi sambil *sebats*, dan *scrolling* twitter ...

1

# Mau Ngomongin Apa ?

Management Secret di Kubernetes dengan bersumber dari Vault ~~di era~~  
**GitOps ini ..**



## Apa yang bakal didapat dari talk ini ?

- Bagaimana caranya membuat **Custom Plugin** di ArgoCD
- Kelebihan dan Kekurangan dari setiap Secret Operator (sependek pengetahuan saya)
- Ideal integrasi secret managemen, “Kondisi di mana semua *confidential data* ter-sentralisasi di satu tempat, dapat di-consume oleh semua deployment, dan dapat di-update dari satu tempat tersebut”



# Kemudahan dan Permasalahan baru yang dibawa GitOps (Yang saya hadapi)

## Kemudahan

- ◉ Single Sources of truth.
- ◉ Make Multi Cluster deployment ezz.
- ◉ All your application is reflected from your declarative yaml file in repo.

## Permasalahan

- ◉ (Sometimes) Hard Coded sensitif value in your repo.



## Bagaimana menghindari “Hardcoded sensitif value”

Choose your secret management flow

1. Integrate your Kubernetes with **ExternalSecretOperator** for pulling secret from your vault.
2. Integrate your ArgoCD with **ArgoCD Vault Plugin** for pulling secret from your vault, right before your yaml deployment applied to Kubernetes Cluster.
3. Both is good too!



## Apa itu ...

### ArgoCD

Continuous Delivery dengan prinsip **GitOps**. (Pull base)

### Hasci Vault

Sebuah Secrets Management tools.

### Secret Operator

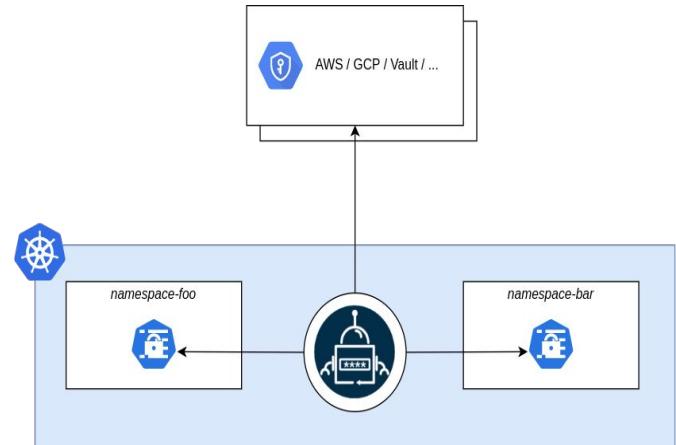
- External Secret Operator
  - Kubernetes CRD for pulling secret from your vault
- ArgoCD Vault Plugin
  - Tools for filling your template yaml with secret value from vault.



# Option 1

## External Secret Operator

```
tmux attach -t blog
→ secret-operator git:(main) kubectl get ns -l eso-test
NAME      STATUS   AGE
kubevirt  Active   82d
prometheus  Active  19d
→ secret-operator git:(main) cat /tmp/testing-rate.yaml
apiVersion: external-secrets.io/v1beta1
kind: ClusterExternalSecret
metadata:
  name: eso-test
spec:
  namespaceSelector:
    matchLabels:
      eso: test
  externalSecretSpec:
    refreshInterval: "5m"
    secretStoreRef:
      name: vault-backend
      kind: ClusterSecretStore
    target:
      name: eso-test
    dataFrom:
      - extract:
          key: secret/test-secret
→ secret-operator git:(main) kubectl get secret/eso-test -n kubevirt
NAME      TYPE     DATA   AGE
eso-test Opaque   5      17m
→ secret-operator git:(main) kubectl get secret/eso-test -n prometheus
NAME      TYPE     DATA   AGE
eso-test Opaque   5      17m
→ secret-operator git:(main)
[blog] 0:nvim- 1:zsh* 2:zsh
"MacBook-Pro-MAC.local" 15:42 06-Feb-23
```



The screenshot shows a terminal window titled "demo-secret". The window contains the following command history:

```
~ /tmp kubectl get ns -l eso-test
NAME      STATUS   AGE
analytic  Active   13h
monitoring Active   27h
~ /tmp nvim test-es0.yaml
~ /tmp kubectl apply -f test-es0.yaml
clusterexternalsecret.external-secrets.io/eso-test created
~ /tmp kubectl get secret/eso-test -n monitoring
NAME      TYPE      DATA   AGE
eso-test  Opaque    1      13s
~ /tmp
```



## Limitasi dari External Secret Operator

### Pros

- There's some schedule **RefreshInterval** by design
- More mature project also more wide community
- Multiple namespace management secret by design

### Cons

- Only support **Kind=Secret** resources
- You need to recreate the **clusterexternalsecret** if you wanna add new namespace



## Contoh Resources Selain Secret Yang Memerlukan Sensitif Data

databaseInitSQL

object

DatabaseInitSQL defines a ConfigMap containing custom SQL that will be run after the cluster is initialized. This ConfigMap must be in the same namespace as the cluster.

To fulfil these kind of resources, we can go to  
**Option 2**

<https://access.crunchydata.com/documentation/postgres-operator/v5/references/crd/#postgresclusterspecdatabaseinitsql>



## Option 2

### ArgoCD (Vault) Plugin

#### Why use this plugin?

This plugin is aimed at helping to solve the issue of secret management with GitOps and Argo CD. We wanted to find a simple way to utilize Vault without having to rely on an operator or custom resource definition. This plugin can be used not just for secrets but also for deployments, configMaps or any other Kubernetes resource.





# Installation ArgoCD Vault Plugin – Add Backed Vault and InitContainer

```
# Download tools
initContainers:
- name: download-tools
image: registry.access.redhat.com/ubi8
env:
- name: AVP_VERSION
  value: 1.11.0
command: [sh, -c]
args:
- >-
  curl -L https://github.com/argoproj-labs/argocd-vault-plugin
  chmod +x argocd-vault-plugin &&
  mv argocd-vault-plugin /custom-tools/
volumeMounts:
- mountPath: /custom-tools
  name: custom-tools
volumes:
- configMap:
    name: cmp-plugin
  name: cmp-plugin
- name: custom-tools
emptyDir: {}
```

```
apiVersion: v1
stringData:
  VAULT_ADDR: http://vault.myvaultserver.com
  AVP_AUTH_TYPE: github
  AVP_GITHUB_TOKEN: t0ke3nsecret
  AVP_TYPE: vault
kind: Secret
metadata:
  name: argocd-vault-plugin-credentials
  namespace: argocd
type: Opaque
```

```
containers:
- name: argocd-repo-server
volumeMounts:
- name: custom-tools
  mountPath: /usr/local/bin/argocd-vault-plugin
  subPath: argocd-vault-plugin
envFrom:
- secretRef:
    name: argocd-vault-plugin-credentials
```



# Installation ArgoCD Vault Plugin – Add Cluster Role and Mount SA-Token

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: argocd-repo-server
spec:
  template:
    spec:
      serviceAccount: argocd-repo-server
      serviceAccountName: argocd-repo-server
      automountServiceAccountToken: true
```

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  # "namespace" omitted since ClusterRoles are not namespaced
  name: argocd-vault
rules:
- apiGroups: [""]
  #
  # at the HTTP level, the name of the resource for accessing Se
  # objects is "secrets"
  resources: ["secrets"]
  verbs: ["get", "watch", "list"]
---
apiVersion: rbac.authorization.k8s.io/v1
# This cluster role binding allows anyone in the "manager" group
kind: ClusterRoleBinding
metadata:
  name: read-secrets-argocd-repo-server
subjects:
- kind: ServiceAccount
  name: argocd-repo-server
  namespace: argocd
roleRef:
  kind: ClusterRole
  name: argocd-vault
  apiGroup: rbac.authorization.k8s.io
```



# Installation ArgoCD Vault Plugin – Config Plugin

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: cmp-plugin
data:
  avp-kustomize.yaml: |
    ---
    apiVersion: argoproj.io/v1alpha1
    kind: ConfigManagementPlugin
    metadata:
      name: argocd-vault-plugin-kustomize
    spec:
      allowConcurrency: true

      # Note: this command is run _before_ anything is done, therefore the logic is to check
      # if this looks like a Kustomize bundle
      discover:
        fileName: "./kustomization.yaml"
      generate:
        command:
          - sh
          - "-c"
          - "kustomize build . | argocd-vault-plugin generate --secret-name argocd:argocd-vault-plugin-credentials -"
      lockRepo: false
---
```

<https://nicolas.my.id/sebuah-cerita-implementasi-gitops-part-2/>

```
# argocd-vault-plugin with Kustomize
- name: avp-kustomize
  command: [/var/run/argocd/argocd-cmp-server]
  image: quay.io/argoproj/argocd:v2.4.0
  securityContext:
    runAsNonRoot: true
    runAsUser: 999
  volumeMounts:
    - mountPath: /var/run/argocd
      name: var-files
    - mountPath: /home/argocd/cmp-server/plugins
      name: plugins
    - mountPath: /tmp
      name: tmp
  # Register plugins into sidecar
  - mountPath: /home/argocd/cmp-server/config/plugin.yaml
    subPath: avp-kustomize.yaml
    name: cmp-plugin
  # Important: Mount tools into $PATH
  - name: custom-tools
    subPath: argocd-vault-plugin
    mountPath: /usr/local/bin/argocd-vault-plugin
```

The screenshot shows the Argo CD interface with an application named "CI-supper-secret-test". The application status is "Healthy" and "Synced". The sync status is "Sync OK". The terminal window below shows the command-line process of updating a Kubernetes secret named "supper-secret" in the "testing-vault" namespace.

```
testing-vault git:(main) ls
argocd-test.yaml  kustomization.yaml  supper-secret.yaml
testing-vault git:(main) nvim supper-secret.yaml
testing-vault git:(main)
18179 git add
18180 git commit -m "Update argocd secret path testing"
18181 git pull --rebase && git push origin
18182 kubectl get secret/supper-secret-kubesecret -o yaml | yq .data."s3.com"
18183 kubectl get secret/supper-secret-kubesecret -o yaml | yq .data."myKey"
18184 kubectl get secret/supper-secret-kubesecret -o yaml | yq .data.myKey
18185 kubectl get secret/supper-secret-kubesecret -o yaml | yq .data.myKey|base64 --d
18186 ls
18187 nvim supper-secret.yaml
18053/18053 -
```



## Limitasi dari ArgoCD Vault Plugin

### Pros

- Handle all resources k8s
- ArgoCD Ecosystem

### Cons

- **Required hard refresh the application**, if we want to pull new values from vault
- Required third party tools like reflector or create multi application in all wanted namespace where resources contain sensitive data going to deploy



## Only choose what you need!

**Pilih Solusi Sesuai  
Kebutuhan**, Apabila tidak ada  
keperluan konfidensial data di  
luar *Kind=Secret* maka  
Externalsecret.io operator  
adalah pilihan paling cocok!



# Thanks!

Any *questions* ?

You can find me at

- [nicolas.my.id](http://nicolas.my.id)
- [me@nicolas.my.id](mailto:me@nicolas.my.id)

# QnA

# Photo Session

# Let's Join With Us!

 **bukalapak is Hiring**

Together we grow and create a fair economy for all



## DevOps Engineer

### **Responsibilities:**

- Manage infrastructure use CI/CD process and automation tools such as Ansible
- Build script to automate operational and deployment process use programming language like Python, Go, Ruby
- Optimizing Infrastructure Automation thru repetitive refactors and Infra Architecture Review
- Deploying, automating, maintaining, building and managing On-Prem and Cloud-based production systems.
- Develop and integrate monitoring, logging, dashboard, and alert systems to quickly and proactively prevent or detect errors
- Ensure the availability, performance, scalability, and security of Infrastructure.
- Build, release and manage configuration management of infrastructure and application in multiple environments.
- Within a cross-functional team, collaborate with other engineers specializing in backend services, web frontend, mobile apps, and test automation, as well as product design and ideation.

### **Requirements:**

- Bachelor degree in Computer Science or related fields, or equivalent professional experience
- Minimum 2 years of working experience
- Strong understanding of Operating System ( Linux )
- Understand the basic of Security Best Practices
- Solid understanding of DevOps Philosophy, Agile Methods, and Infrastructure as Code
- Humble culture, zero egos, and excellent collaborative spirit. We are all here to learn together and grow together as a team
- Experience in Monitoring Tools like Grafana, Datadog, and Prometheus
- Experience in using Versioning Control tools
- Knowledge & Experience in Go, Python, & Bash
- Experience in one of these following technologies, including Kubernetes, Nginx, Haproxy, Kubernetes Ingress
- Experience in Infrastructure-as-Code and Automation like Ansible or Terraform

[careers.bukalapak.com](http://careers.bukalapak.com)



[bit.ly/DevOpsBL](http://bit.ly/DevOpsBL)

# Feedback Form

We love to hear your  
feedbacks and thoughts  
for the event by fill out or  
scan the feedback form!

[bit.ly/BLKubernetes\\_Feedback](https://bit.ly/BLKubernetes_Feedback)



**Thank  
You**