

---

# **vault-secrets-operator**

**vault-secrets-operator development team**

**Aug 24, 2022**



CONTENTS:

<b>1</b>	<b>vaultsecrets-operator installation</b>	<b>1</b>
1.1	Requirements . . . . .	1
1.2	Install to K8S cluster . . . . .	1
1.3	EnvVariables . . . . .	1
<b>2</b>	<b>FAQ</b>	<b>3</b>
<b>3</b>	<b>VaultSecret CRD</b>	<b>5</b>
3.1	Spec . . . . .	5
3.2	SecretsPaths . . . . .	5
<b>4</b>	<b>Indices and tables</b>	<b>7</b>



## VAULTSECRETS-OPERATOR INSTALLATION

The contains details of how to install and uninstall vaultsecrets-operator

### 1.1 Requirements

vaultsecrets-operator runs on K8S cluster 1.16 and up. To install it you would need:

1. Admin access to cluster
2. `kubectl` which is configured to access your cluster and is in your execution path
3. GNU or \*NIX Make which is in your execution path
4. Vault role and AWS IAM role which grants access to Vault

### 1.2 Install to K8S cluster

The vaultsecrets-operator docker image is located at [DockerHub](#).

To install it to your K8S cluster:

1. edit `deploy/operator.yaml` and add your environment variables.
2. Install using `make` and `kubectl`:

```
cd deploy
make install
```

### 1.3 EnvVariables

Environment variables, which allow to configure operator:



---

## CHAPTER TWO

---

### FAQ

Any questions related to vault-secrets-operator:





## VAULTSECRET CRD

Example:

```
apiVersion: xo.90poe.io/v1alpha1
kind: VaultSecret
metadata:
  name: example-vaultsecret
  labels:
    app: example-vaultsecret-app
    owner: DevOps
spec:
  name: default-v1-test
  reread_intervals: 300 # 10 minutes
  type: kubernetes.io/dockerconfigjson
  secrets_paths:
    .dockerconfigjson: shared/nexus_dockerconfigjson
```

### 3.1 Spec

You will have to amend spec section according to your requirements.

Spec section:

### 3.2 SecretsPaths

This map contains keys and values. Keys would be used in Secrets as data keys. And values would be fetched from Vault on path, specified by values in this structure.

Full path, on which this operator is going to read secret from Vault is constructed as follows:  
**VAULT\_SECRETS\_PREFIX** / + value

NOTE: **VAULT\_SECRETS\_PREFIX** is environment variable.

Example:

```
spec:
  . . . .
  secrets_paths:
    SOME_DATA: shared/very_secure_password
```

In here, `SOME_DATA` will be put to K8S secret as data key and value for it would be fetched from `$VAULT_SECRETS_PREFIX/shared/very_secure_password` in Vault.

Operator is expecting special form for secret in Vault. You must have secret `shared/very_secure_password` hold key `value` or `base64_value` and your secret. If `value` is used for key in Vault, secret will be encoded with base64 before putting into K8S Secret object. If `base64_value` value is used (for binary or JSON objects), then operator expects that value is already encoded in Vault and will not perform additional encoding before putting to K8S Secret object.

## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`