

# Using Substrate Vault Secrets with Codefresh

---

Downloaded from Epic Games Confluence

Date: 2025-07-12 04:07:23

Original URL: <https://confluence-epicgames.atlassian.net/wiki/spaces/CDE/pages/81068351>

Document Level Classification

[200](#)

- [Using a Build Step](#)
- [Fetching a Vault token](#)

## Introduction

Codefresh has native support for Vault. Each Codefresh account has a unique AppRole and policy which provide access to your team's secrets. There are two ways to use the pre-configured Vault credentials. First, you can use a build step to fetch secrets and populate environment variables in your build. Second, you can use the AppRole credentials to fetch a Vault token which can be passed to other tools.

## Using a Build Step

Build steps are available on the right side of the pipeline configuration page. Search for "HashiCorp vault" to see your options, and click through any given integration to see a usage example. **vault-next-gen** or **eos/**

**vault-next-gen** are the recommended steps to use, but you are free to experiment.

STEPS


hashicorp|vault

×


⋮

MARKETPLACE

MY STEPS


**Github Actions**

Run existing github actions

**Fetch secrets &\_0\_0\_> from Hashicorp Vault**


deniscodefresh/vault-step-3

Read secrets from Hashicorp Vault inside a Codefresh pipeline.

**Fetch specific secrets from Hashicorp Vault**


eos/vault-next-gen

Read specific secrets from multiple paths in Hashicorp Vault inside a Codefresh pipeline.

**portshift-security/vault**


portshift-security/vault

The plugin exports KV pairs from Hashicorp Vault to Codefresh pipeline ENV variables

**Fetch secrets from Hashicorp Vault**

vault

Read secrets from Hashicorp Vault inside a Codefresh pipeline.

**Fetch specific secrets from Hashicorp Vault**

vault-next-gen

Read specific secrets from multiple paths in Hashicorp Vault inside a Codefresh pipeline.

STEPS

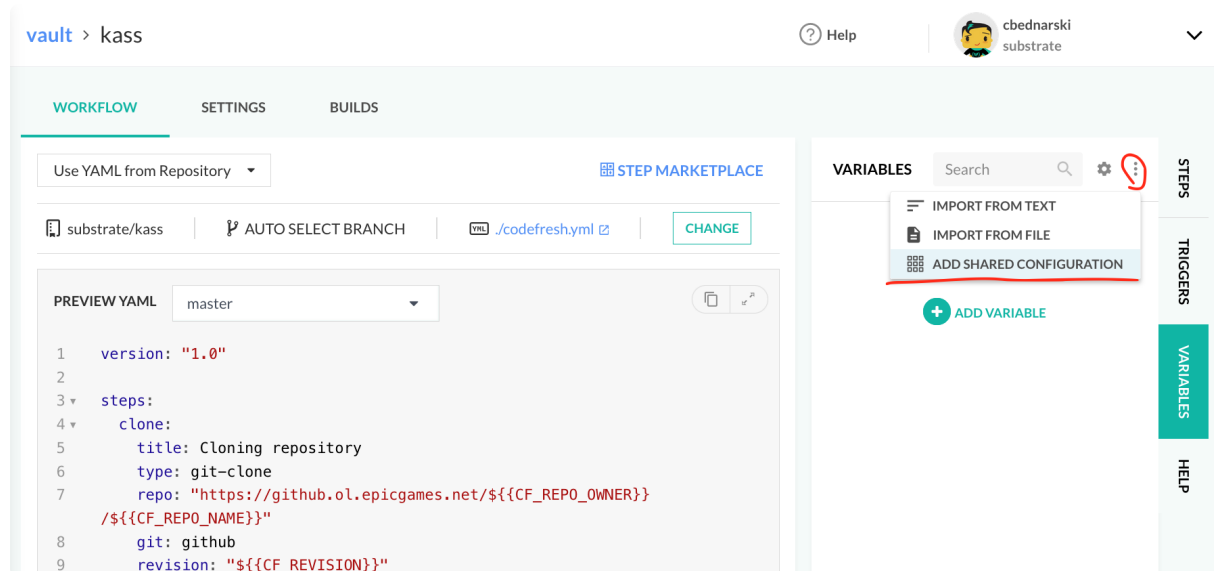
TRIGGERS

VARIABLES

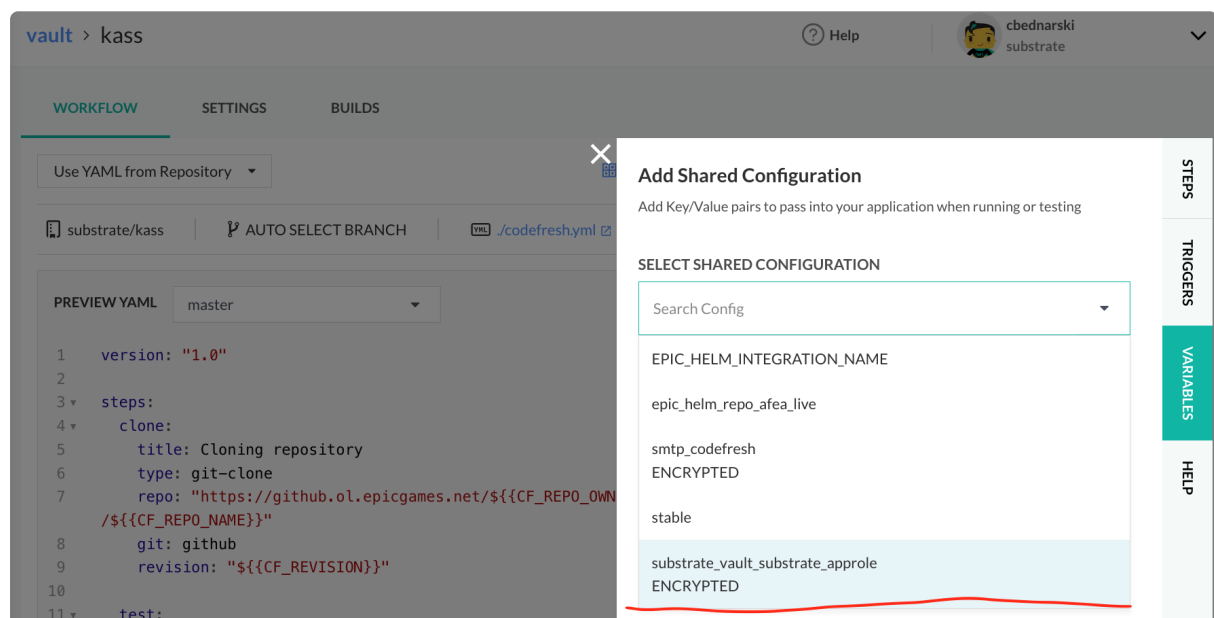
HELP

# Fetching a Vault token

First, we need to share the AppRole configuration with our build pipeline. This is achieved through the [Codefresh UI](#). We'll start by adding the Shared Configuration to our pipeline.




Next, select the `substrate_vault_[yourteam]_approle`. Make sure you click the green **Add** button at the bottom of this screen.





When you complete this you'll see that your build pipeline includes these environment variables:

VARIABLES

Search










✓ SUBSTRATE\_VAULT\_SUBSTRATE\_APPROLE 3

substrate\_vault\_address = \*\*\*\*\*

substrate\_vault\_approle\_id = \*\*\*\*\*

substrate\_vault\_secret\_id = \*\*\*\*\*



 ADD VARIABLE

STEPS

TRIGGERS

VARIABLES

HELP

The AppRole configuration variables *are* environment variables although they do not follow the traditional `APP_CAPS` naming convention.

To fetch a Vault token in your build steps you will need `castle` or the `vault` CLI in your build to run the `vault login` command. You can also do this using `curl` [if you are feeling adventurous](#). As an example, I use the `golang-build` docker image which includes the Vault CLI and a [helper script](#) to fetch the Vault token.

Read more about [CodeFresh Steps](#) and the [Vault Next Gen step](#).

---

#### Page Information:

Page ID: 81068351

Space: Cloud Developer Platform

Downloaded: 2025-07-12 04:07:23