Hashicorp Vault is a free and open source tool designed for securely storing and accessing secrets. A secret can be a password, API key, certificate, and more. The job of Vault Server is to provide a unified interface to any stored secret while providing tight access control and recording a detailed audit log.

# Steps

# 1: Install and Configure Hashicorp Vault Server

* Create the EC2 instance with the port 8200 enabled.

The precompiled Vault binaries are available at : <https://releases.hashicorp.com/vault/>

# cd /opt

# wget <https://releases.hashicorp.com/vault/1.7.0/vault_1.7.0_linux_amd64.zip>

--2021-04-09 05:30:04-- https://releases.hashicorp.com/vault/1.7.0/vault\_1.7.0\_linux\_amd64.zip

Resolving releases.hashicorp.com (releases.hashicorp.com)... 151.101.249.183, 2a04:4e42:2f::439

Connecting to releases.hashicorp.com (releases.hashicorp.com)|151.101.249.183|:443... connected.

HTTP request sent, awaiting response... 200 OK

Length: 68789357 (66M) [application/zip]

Saving to: ‘vault\_1.7.0\_linux\_amd64.zip’

100%[===============================================================================================>] 68,789,357 107MB/s in 0.6s

2021-04-09 05:30:05 (107 MB/s) - ‘vault\_1.7.0\_linux\_amd64.zip’ saved [68789357/68789357]

# cp vault /usr/local/sbin

# vault --version

Vault v1.7.0 (4e222b85c40a810b74400ee3c54449479e32bb9f)

Enable command autocompletion.

# vault -autocomplete-install

# complete -C /usr/local/sbin/vault vault

## Configure Vault systemd service

Create Vault data directories and vault user

mkdir /etc/vault

mkdir -p /var/lib/vault/data

useradd --system --home /etc/vault --shell /bin/false vault  
chown -R vault:vault /etc/vault /var/lib/vault/

**Now configure the vault to run as a service, for this creates the vault service as below:**

**# cat /etc/systemd/system/vault.service**

[Unit]

Description="HashiCorp Vault - A tool for managing secrets"

Documentation=https://www.vaultproject.io/docs/

Requires=network-online.target

After=network-online.target

ConditionFileNotEmpty=/etc/vault/config.hcl

[Service]

User=vault

Group=vault

ProtectSystem=full

ProtectHome=read-only

PrivateTmp=yes

PrivateDevices=yes

SecureBits=keep-caps

AmbientCapabilities=CAP\_IPC\_LOCK

NoNewPrivileges=yes

ExecStart=/usr/local/sbin/vault server -config=/etc/vault/config.hcl

ExecReload=/bin/kill --signal HUP

KillMode=process

KillSignal=SIGINT

Restart=on-failure

RestartSec=5

TimeoutStopSec=30

StartLimitBurst=3

LimitNOFILE=65536

[Install]

WantedBy=multi-user.target

Create Vault /etc/vault/config.hcl file.

**touch /etc/vault/config.hcl**

Add basic configuration settings for Vault to /etc/vault/config.hcl file.

**# cat /etc/vault/config.hcl**  
disable\_cache = true  
disable\_mlock = true  
ui = true  
listener "tcp" {  
 address = "0.0.0.0:8200"  
 tls\_disable = 1  
}  
storage "file" {  
 path = "/var/lib/vault/data"  
 }  
api\_addr = "http://0.0.0.0:8200"  
max\_lease\_ttl = "10h"  
default\_lease\_ttl = "10h"  
cluster\_name = "vault"  
raw\_storage\_endpoint = true  
disable\_sealwrap = true  
disable\_printable\_check = true

Start and enable vault service to start on system boot.

**# sudo systemctl daemon-reload**

**# sudo systemctl enable --now vault**

Created symlink from /etc/systemd/system/multi-user.target.wants/vault.service to /etc/systemd/system/vault.service.

**# systemctl status vault**

● vault.service - "HashiCorp Vault - A tool for managing secrets"

Loaded: loaded (/etc/systemd/system/vault.service; enabled; vendor preset: disabled)

Active: active (running) since Fri 2021-04-09 05:48:25 UTC; 1min 29s ago

Docs: https://www.vaultproject.io/docs/

Main PID: 3654 (vault)

CGroup: /system.slice/vault.service

└─3654 /usr/local/sbin/vault server -config=/etc/vault/config.hcl

Apr 09 05:48:25 vault\_server vault[3654]: Go Version: go1.15.10

Apr 09 05:48:25 vault\_server vault[3654]: Listener 1: tcp (addr: "0.0.0.0:8200", cluster address: "0.0.0.0:8201", max\_request\_d...abled")

Apr 09 05:48:25 vault\_server vault[3654]: Log Level: info

Apr 09 05:48:25 vault\_server vault[3654]: Mlock: supported: true, enabled: false

Apr 09 05:48:25 vault\_server vault[3654]: Recovery Mode: false

Apr 09 05:48:25 vault\_server vault[3654]: Storage: file

Apr 09 05:48:25 vault\_server vault[3654]: Version: Vault v1.7.0

Apr 09 05:48:25 vault\_server vault[3654]: Version Sha: 4e222b85c40a810b74400ee3c54449479e32bb9f

Apr 09 05:48:25 vault\_server vault[3654]: ==> Vault server started! Log data will stream in below:

Apr 09 05:48:25 vault\_server vault[3654]: 2021-04-09T05:48:25.760Z [INFO] proxy environment: http\_proxy= https\_proxy= no\_proxy=

Hint: Some lines were ellipsized, use -l to show in full.

Start initialization with the default options by running the command below:

**# export VAULT\_ADDR=http://127.0.0.1:8200**

**# vault operator init |tee /etc/vault/init.file** 🡪 This will generate the 5 unseal key and Initial root token which is require to access he gui.

**# cat /etc/vault/init.file**

Unseal Key 1: 8w/vmDxrxTPovyxHz8e0wfixH4IUb08JozU/hvahZpV7

Unseal Key 2: P8TVjb2jUvqwp8d3F306vlEHlJ4LGbfRo4BfFvhNc5xb

Unseal Key 3: xgkP1Yh4hprcgRMpscVEkMAT0q+nOQlFimU2B6uWCEkJ

Unseal Key 4: 1nn1rbAh+2VrEMbLWd2RLsHItSG5tK1STsJDHXQ8dJnN

Unseal Key 5: yZlRakRA1Caxs7HcP7Kq5uM6gP/vcRfdyB8qHl14zRy0

Initial Root Token: s.UQwEqH4ZevMplOBGjdmQo4eS

*Vault initialized with 5 key shares and a key threshold of 3. Please securely*

*distribute the key shares printed above. When the Vault is re-sealed,*

*restarted, or stopped, you must supply at least 3 of these keys to unseal it*

*before it can start servicing requests.*

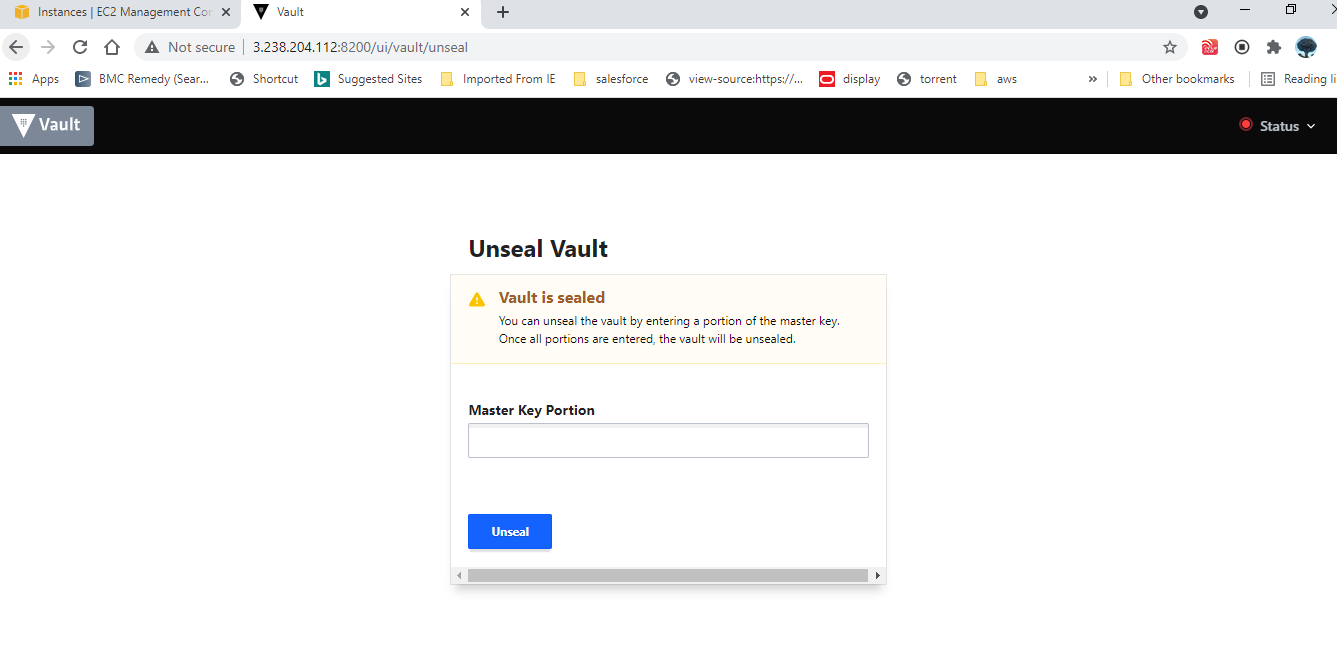
Vault does not store the generated master key. Without at least 3 key to

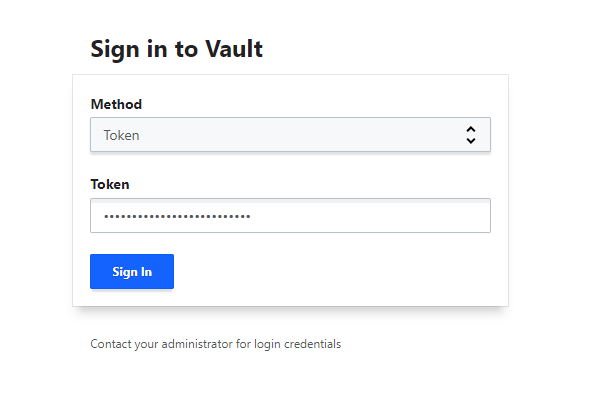
reconstruct the master key, Vault will remain permanently sealed!

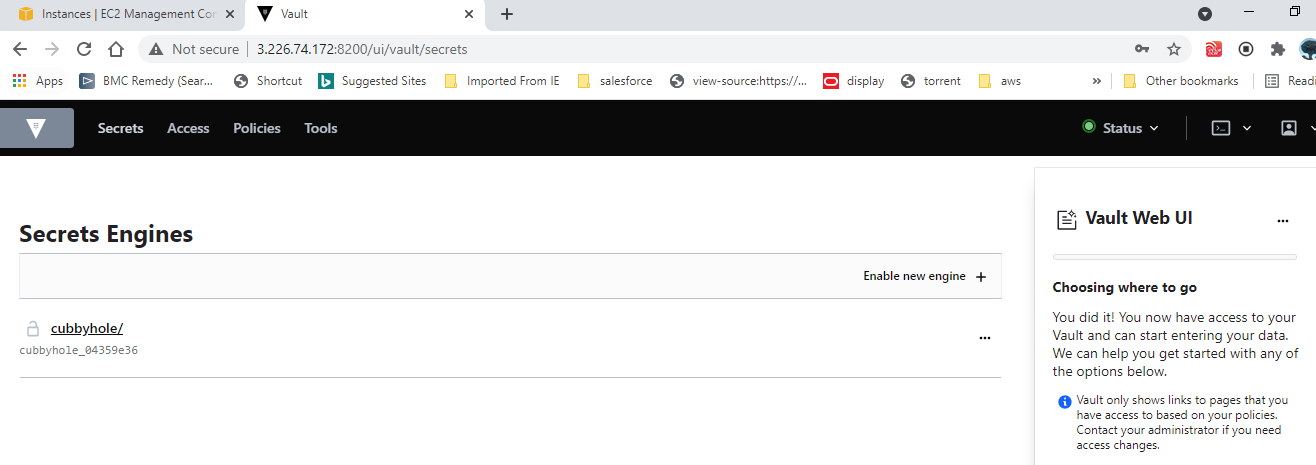
It is possible to generate new unseal keys, provided you have a quorum of

existing unseal keys shares. See "vault operator rekey" for more information.

Now access the UI in :  http://serverip:8200/ui







You can also view Vault status from CLI.

**# vault status**

Key Value

--- -----

Seal Type shamir

Initialized true

Sealed false

Total Shares 5

Threshold 3

Version 1.7.0

Storage Type file

Cluster Name vault

Cluster ID 13e0186b-9cdb-0fc0-95df-722c6d43dab8

HA Enabled false

## Configure Vault Roles And Policies

Export your Vault root token, value store in **/etc/vault/init.file:**

# export VAULT\_TOKEN="s.UQwEqH4ZevMplOBGjdmQo4eS"

Then enable the approle auth method which allows machines or apps to authenticate with Vault-defined roles

# vault auth enable approle

Success! Enabled approle auth method at: approle/

# vault auth list

Path Type Accessor Description

---- ---- -------- -----------

approle/ approle auth\_approle\_7f7b27c7 n/a

token/ token auth\_token\_913c80e1 token based credentials

**# vault auth enable kubernetes**

Success! Enabled kubernetes auth method at: kubernetes/

**# vault auth enable userpass**

Success! Enabled userpass auth method at: userpass/

**# vault auth enable ldap**

Success! Enabled ldap auth method at: ldap/

**# vault auth list**

Path Type Accessor Description

---- ---- -------- -----------

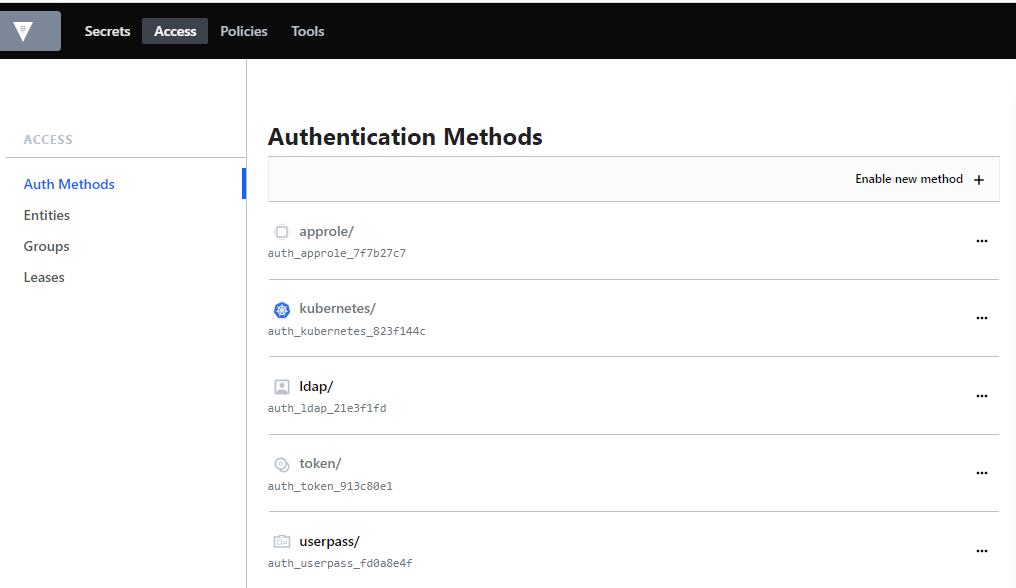
approle/ approle auth\_approle\_7f7b27c7 n/a

kubernetes/ kubernetes auth\_kubernetes\_823f144c n/a

ldap/ ldap auth\_ldap\_21e3f1fd n/a

token/ token auth\_token\_913c80e1 token based credentials

userpass/ userpass auth\_userpass\_fd0a8e4f n/a



**To get started, enable the kv secrets engine. Each path is completely isolated and cannot communicate talk to other paths. For example, a kv secrets engine enabled at foo has no ability to with a kv secrets engine enabled at bar. Now we will enable the secrets.**

**# vault secrets enable -path=secrets kv**

Success! Enabled the kv secrets engine at: secrets/

**# vault secrets list**

Path Type Accessor Description

---- ---- -------- -----------

cubbyhole/ cubbyhole cubbyhole\_04359e36 per-token private secret storage

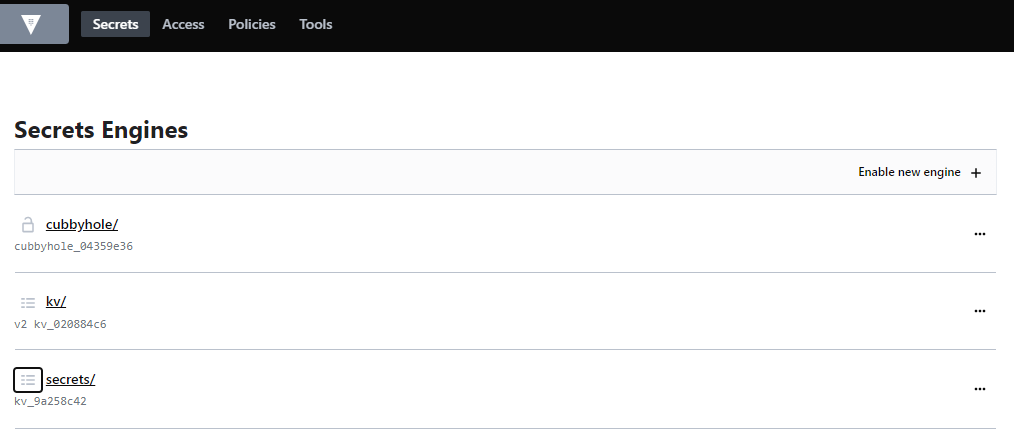
identity/ identity identity\_84ad24cc identity store

secrets/ kv kv\_9a258c42 n/a

sys/ system system\_3839b622 system endpoints used for control, policy and debugging

**# vault secrets enable -version=2 kv**

Success! Enabled the kv secrets engine at: kv/



We have create two secrets one of version one and another of version2 type.

**Create secret value of version2:**

**# vault kv put secrets/my-secret my-value=s3cr3t\_v2**

Success! Data written to: secrets/my-secret

**# vault kv get secrets/my-secret**

====== Data ======

Key Value

--- -----

my-value s3cr3t\_v2

**Create secret value of version1:**

**# vault kv put kv/my-secret my-value=s3cr3t\_v1**

Key Value

--- -----

created\_time 2021-04-09T07:26:00.879514202Z

deletion\_time n/a

destroyed false

version 1

**# vault kv get kv/my-secret**

====== Metadata ======

Key Value

--- -----

created\_time 2021-04-09T07:26:00.879514202Z

deletion\_time n/a

destroyed false

version 1

====== Data ======

Key Value

--- -----

my-value s3cr3t\_v1

To get the valut secret using anible playbook :

# cat get\_secret\_v1\_and\_v2.yml

---

- hosts: localhost

tasks:

- name: Return all secrets from a path

debug:

msg: "{{ lookup('hashi\_vault', 'secret=secrets/my-secret token=s.UQwEqH4ZevMplOBGjdmQo4eS url=http://3.226.74.172:8200')}}"

- name: Return secrets value from path

debug:

msg: "{{ lookup('hashi\_vault', 'secret=secrets/my-secret:my-value token=s.UQwEqH4ZevMplOBGjdmQo4eS url=http://3.226.74.172:8200')}}"

- name: Return all kv v2 secrets from a path

debug:

msg: "{{ lookup('hashi\_vault', 'secret=kv/data/my-secret:data token=s.UQwEqH4ZevMplOBGjdmQo4eS url=http://3.226.74.172:8200')}}"

**# ansible-playbook get\_secret\_v1.yml**

[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [localhost]

TASK [Return all secrets from a path] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [localhost] => {

"msg": {

"my-value": "s3cr3t\_v2"

}

}

TASK [Return all kv v2 secrets from a path] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [localhost] => {

"msg": {

"my-value": "s3cr3t\_v1"

}

}

TASK [Return secrets value from path] \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

ok: [localhost] => {

"msg": "s3cr3t\_v2"

}

PLAY RECAP \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

localhost : ok=3 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

Terraform Script to access the secrets from hashicorp vault:

#https://computingforgeeks.com/install-and-configure-vault-server-linux

provider "vault" {

address = "http://3.226.74.172:8200"

skip\_tls\_verify = true

token = "s.UQwEqH4ZevMplOBGjdmQo4eS"

}

data "vault\_generic\_secret" "secrets" {

  path = "secrets/my-secret"

}

output NameKV1 {

value = "${data.vault\_generic\_secret.secrets.data["my-value"]}"

}

data "vault\_generic\_secret" "kv" {

  path = "kv/my-secret"

}

output Namekv2 {

value = "${data.vault\_generic\_secret.kv.data["my-value"]}"

}

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

NameKV1 = "s3cr3t\_v2"

Namekv2 = "s3cr3t\_v1"