Deploying a Basic Vault Server

Ned Bellavance
Founder, Ned in the Cloud LLC

@ned1313 | nedinthecloud.com

Overview

Vault server configuration

Storage backend

Vault server initialization

Key management

Vault Server Configuration

Parameter Categories

General

High Availability

Listener

Seal

Telemetry

Storage

Vault-Config.hcl

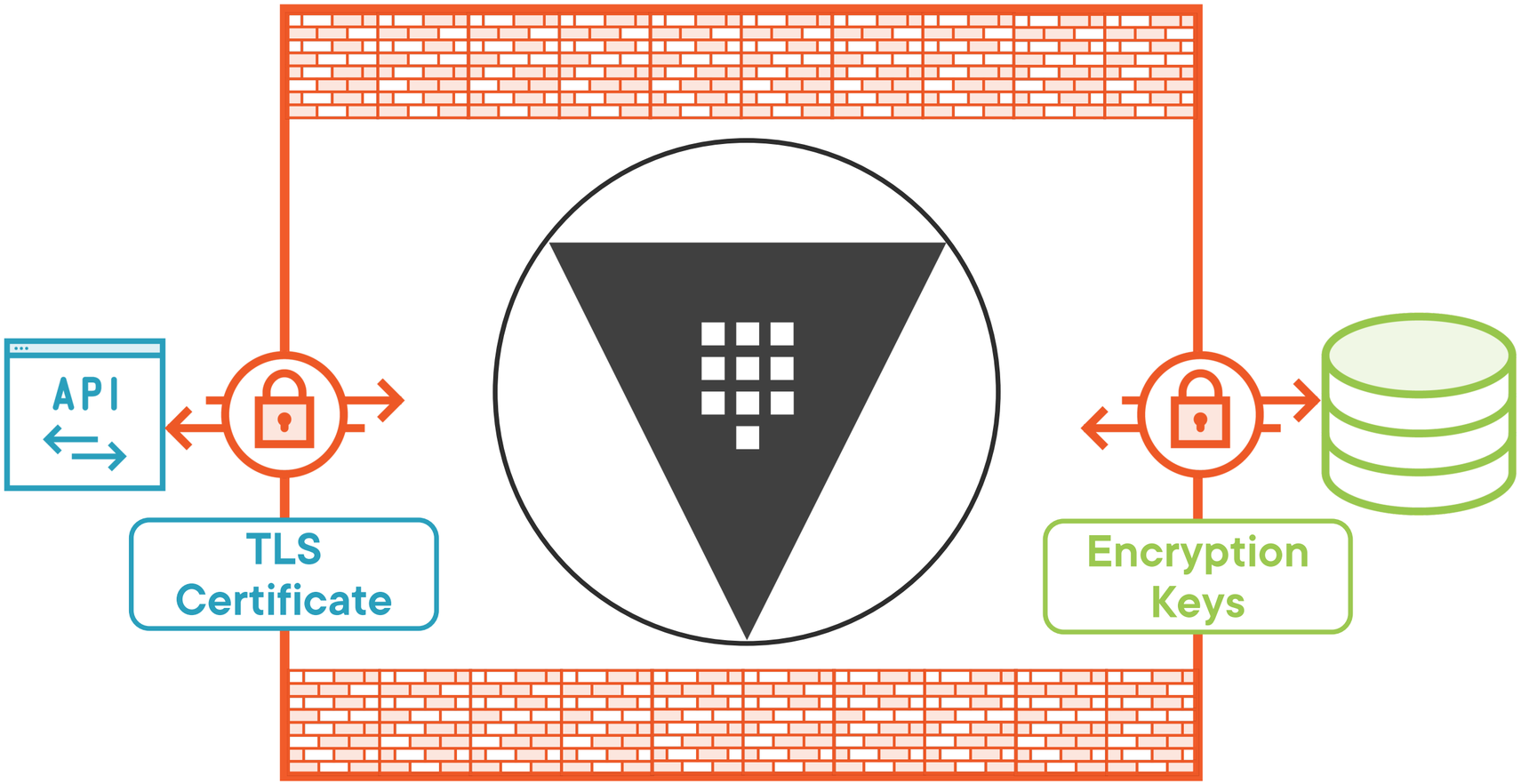
```
listener "tcp" {}

seal "type_name" {}

storage "type_name" {}

service_registration "type" {}

telemetry {}

ui = [true | false]

disable_mlock = [true | false]

cluster_addr = "https://address:port"

api_addr = "https://address:port"
```

Vault-Config.hcl

```
listener "tcp" {
  address = "127.0.0.1:8200"
  cluster_address = "127.0.0.1:8201"
  tls_cert_file = "path/to/public/cert.crt"
  tls_key_file  = "path/to/private/cert.key"
}
```

Vault Storage

Vault Logical Architecture



Storage Backend

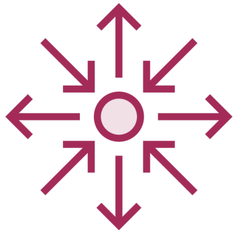
Storage types

- Object
- Database
- Key/Value
- File
- Memory

Integrated Storage (Raft)

- Local storage
- Highly available
- Replicated

Support



High availability

Storage configuration

Globomantics Scenario

Use Case

- Vault services need to be highly available
- Storage backend must be HashiCorp supported
- Minimize external dependencies

Solution

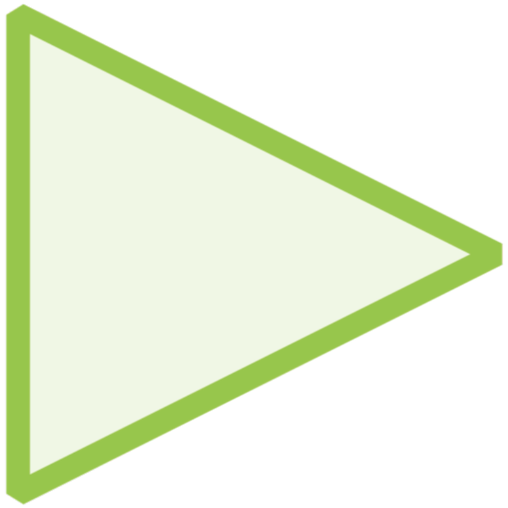
- Select Integrated Storage as the backend
- Deploy at least three nodes for high availability

Vault-Config.hcl

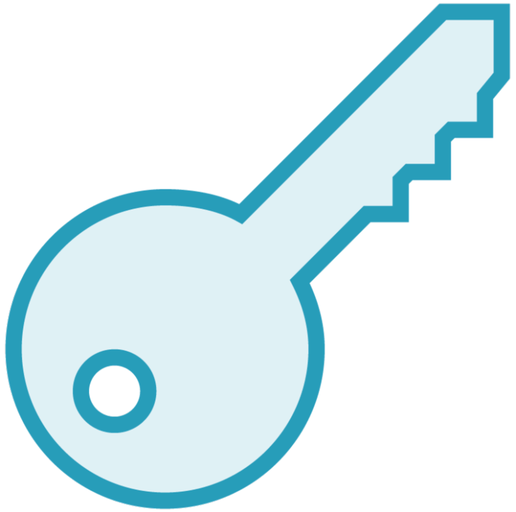
```
storage "raft" {

  path = "/path/to/data/directory"

  node_id = "unique_node_identifier"

  retry_join {

    leader_api_addr = "https://node1.vault.local:8200"
    leader_ca_cert_file = "/path/to/cert/file"

  }

  retry_join {

    leader_api_addr = "https://node2.vault.local:8200"
    leader_ca_cert_file = "/path/to/cert/file"

  }

}
```

Vault Server Deployment

Vault Startup



Start



Initialize

Unseal

Demo

Tasks:

- Inspect Vault server config
- Deploy Vault server with Docker
- Initialize and unseal Vault
- Rotate and rekey Vault

Pre-requisites:

- Exercise files
- Internet connection
- Code editor
- Docker Desktop

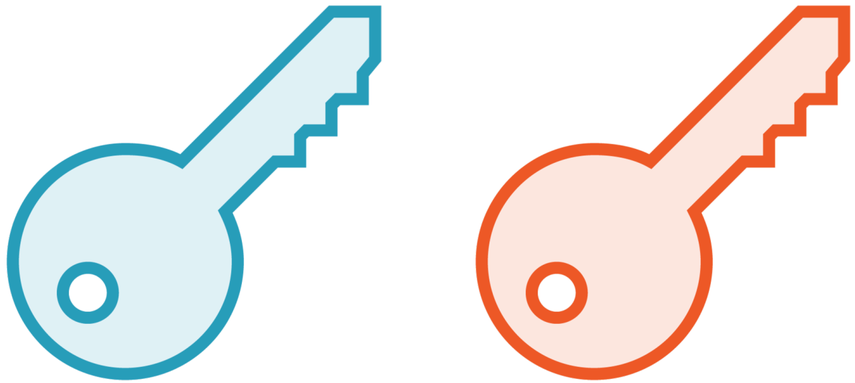
Initializing Vault

```
# Get Vault server status

vault status


# Initialize Vault server

vault operator init [options]

vault operator init –key-shares=5 –key-threshold=3

vault operator init –recovery-shares=5 –recovery-threshold=3
```

Unseal Vault

```
# Start unseal process

vault operator unseal [options] [KEY]


# Seal an unsealed Vault server

vault operator seal [options]
```

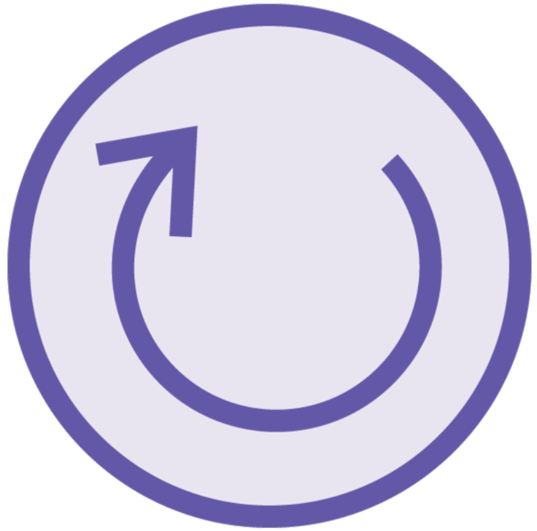
Key Management



Rekey

Update Unseal and Master keys

Change seal settings



Rotate

Update Encryption keyring

Previous versions saved

Manage Keys

```
# Rekey unseal and master keys

vault operator rekey [options] [KEY]

vault operator rekey –init –key-shares=7 –key-threshold=5


# Check the encryption key status

vault operator key-status [options]


# Rotate the encryption key

vault operator rotate [options]
```

Key Takeaways

Vault server configuration is defined by an HCL or JSON file and environment variables.

Storage backends can be HashiCorp or Community supported, and may support high-availability.

Vault server must be initialized on first startup and unsealed after every startup.

The Master and Unseal keys can be updated using the rekey operation.

The Encryption keys can be changed using the rotate operation.

Up Next: Planning for High Availability