Clustering HashiCorp Vault

Here are special instructions for setting up a vault cluster.

> NOTE: See the regular setup page for other details: Hashicorp Vault Setup

DNS Resolution

Since the vault services will communicate with eachother over TLS, they will need certificates. And as such, the certs will include hostnames.

So, open the /etc/hosts file of each vault host, and add entries, at the bottom of the file, for each instance and API host.

Here's an example list of entries for a cluster:

```
192.168.75.10 vault02api
192.168.75.21 vault0201
192.168.75.22 vault0202
192.168.75.23 vault0203
192.168.75.24 vault0204
192.168.75.25 vault0205
192.168.75.26 vault0206
```

Filesystem Changes

Raft Folder

The Vault service will be running Raft. So, it will need a folder for the Raft backend.

> NOTE: This may mean that the folder /opt/vault/data is obsolete.
> But, we will not worry about that, for now.
> The 'data' folder was created by the installer as the FS location for a storage = 'file' backend.

Create the raft folder with these:

```
sudo mkdir /opt/vault/raft
```

```
sudo chmod 700 /opt/vault/raft
```

```
sudo chown vault:vault /opt/vault/raft
```

TLS Folder

The installer already created the TLS folder, to store certificates.
It is at: /opt/vault/tls

We will leave it as is.

Config Folder

The installer created a config folder at: /etc/vault.d

We need to bolster its permissions, as it may contain seal stanzas.

Update permissions of the config folder with these:

```
sudo chmod 0750 /etc/vault.d
```

```
sudo chown root:vault /etc/vault.d
```

Firewall Rules

Update the local firewall rules for each vault host, to allow 8200 and 8201 access.

```
sudo ufw allow 8200
```

```
sudo ufw allow 8201
```

Certificates

We will create certificates for each vault instance, and put them in the tls folder at: /opt/vault/tls

Follow instructions, here, to generate certificates for each host: Generate Certificates with Hashicorp Vault

> NOTE: Be sure to do the following:
> - Set the common name to the fully qualified name: ex: vault0204.ogsofttech.lan.
> - Set the expiry to two years (17520 hours).
> - Set set the SAN IP to the address of the host: 192.168.75.24

Copy the CA certificate bundle (CA chain of issuer and root CA) into: /opt/vault/tls/ca.crt

> NOTE: The ca.crt file should be the certificate bundle of issuer CA cert and root CA certificate.
> These should be bundled (root + intermediate) in the ca.crt file, as a concatenated PEM.

Concatenate the vault service certificate with the issuer CA cert as a concatenated PEM.

Paste the certificate file (vault service + issuer CA cert) into: /opt/vault/tls/vault.crt

> NOTE: The vault.crt file should include the leaf certificate (of the node) plus the signing intermediate.

Copy the vault service private key into: /opt/vault/tls/vault.key

Once key and certs are stored, we need to set permissions on the files, with these:

```
# Ownership: keep vault:vault
sudo chown vault:vault /opt/vault/tls/vault.crt /opt/vault/tls/vault.key /opt/vault/tls/ca.crt

# Permissions:
# private key: only vault access
sudo chmod 0600 /opt/vault/tls/vault.key
# server cert usually fine as world-readable
sudo chmod 0644 /opt/vault/tls/vault.crt
# CA cert usually fine as world-readable
sudo chmod 0644 /opt/vault/tls/ca.crt
```

Here's a quick sanity check, to verify the certificates on a host:

```
# Verify the server chain file against your trust bundle
openssl verify -CAfile /opt/vault/tls/ca.crt /opt/vault/tls/vault.crt

# See what the gateway/clients will see
openssl s_client -connect vault0204:8200 -showcerts -verify_return_error \
  -CAfile /opt/vault/tls/ca.crt </dev/null
```

Vault.HCL Changes

Now, we need to create a config file for each vault service (vault.hcl).

There will be some tailoring required for each host.
So, pay attention to the notes.

Open the vault config file at: /etc/vault.d/vault.hcl

```
sudo nano /etc/vault.d/vault.hcl
```

Below is a config file for a single node in a vault cluster.

> NOTE: You will need to change the following lines for each host:
> api_addr
> cluster_addr
> node_id

```
# Vault Configuration for Clustering.
# Created on 20250901

ui            = true
cluster_name  = "vaultcluster2"
# recommended with Integrated Storage
disable_mlock = true

# Advertise addresses (MUST be correct & reachable)
# Define the external API address that clients will use to communicate with this Vault node.
# set per node
api_addr     = "https://vault0204.ogsofttech.lan:8200"
# Define the internal address used by Vault nodes to communicate with this Vault node.
# Vault ignores the scheme of this URL, so it doesn't matter if http or https.
# set per node (node's intra-cluster IP:8201)
cluster_addr = "https://192.168.75.24:8201"

listener "tcp" {
  address                = "0.0.0.0:8200"
  tls_disable            = 0
  tls_cert_file          = "/opt/vault/tls/vault.crt"
  tls_key_file           = "/opt/vault/tls/vault.key"
  tls_client_ca_file       = "/opt/vault/tls/ca.crt"
  # optional: require client certs from automation:
  # tls_require_and_verify_client_cert = "true"
}
```

```
# Updated to use raft backend (Integrated Storage).
storage "raft" {
  path    = "/opt/vault/raft"
# unique on each node
  node_id = "vault0204"

  # Auto-join peers (preferred over manual joins)
  retry_join {
    leader_api_addr      = "https://vault0204.ogsofttech.lan:8200"
    leader_ca_cert_file  = "/opt/vault/tls/ca.crt"
  }
  retry_join {
    leader_api_addr      = "https://vault0205.ogsofttech.lan:8200"
    leader_ca_cert_file  = "/opt/vault/tls/ca.crt"
  }
  retry_join {
    leader_api_addr      = "https://vault0206.ogsofttech.lan:8200"
    leader_ca_cert_file  = "/opt/vault/tls/ca.crt"
  }
}

# Recommended health endpoint behavior for LBs
# (defaults are fine; LB should treat 200 as active, 429 as standby, 503 sealed)
# See /v1/sys/health docs for details.
```

Service File (Systemd Unit)

With folders fixed, certs and config defined, we need to configure the vault service for operation.

To do so, we need to identify the service file that the installer created, with this:

```
sudo systemctl cat vault.service
```

As of 20250803, the installer puts the systemd unit file, here:

```
/usr/lib/systemd/system/vault.service
```

We need to open it and update it:

```
sudo nano /usr/lib/systemd/system/vault.service
```

Once updated, save and close the systemd file.

Return to the generic vault setup page: https://wiki.galaxydump.com/link/434#bkmrk-continue-setup

Revision #20
Created 1 September 2025 23:27:31 by glwhite
Updated 4 September 2025 03:12:39 by glwhite