Marius Follow
DevOps Engineer @car2go who loves music, tech & code. Opinions are my own.
Jan 28 · 2 min read

Automating a Private PKI with vaultbot

With new microservices surfacing every day, securing the communication between the different entities, becomes more and more important. Using Transport Layer Security (TLS) communication will ensure the integrity and privacy of exchanged messages as well as the authenticity of involved parties.

When using TLS at scale, a fitting Public Key Infrastructure (PKI) is essential. Hashicorp Vault is a tool that secures, stores, and tightly controls access to tokens, passwords, certificates and other secrets of a modern architecture. Vaults PKI Backend fits perfectly into an automated infrastructure. It can be used to generate chains of Certificate Authorities and issue certificates for every use-case.

A good way to establish a PKI powered by Vault is by following the official setup guide. It will walk you through setting up a private Root-CA, generating intermediate CAs and issuing your first certificates.

While using Vault's API to request certificates works great, it is not really optimised for services where you can not integrate vaults API directly. Vault does not provide a client for simply requesting and updating specific certificates in an automated way. When looking at public CAs, there are services like letsencrypt and certbot, which automate these processes. Having a similar tool that integrates with Vault, would therefore be a huge benefit.

Meet **vaultbot**.

> *A lightweight Hashicorp Vault PKI client, built for infrastructure automation, which can automatically request and renew certificates generated inside Vault's PKI backend.*

Vaultbot inherits resilience by design. It is built to be run at fixed intervals to request and renew certificates as they reach the end of their lifetime. By default vaultbot only renews certificates that are near their expiry date, so it can be scheduled at a high frequency.

Requesting and renewing a certificate with a specific hostname, alternative names and IPs is as easy as running the following command:

```
./vaultbot --vault_addr=http://localhost:1234 --
vault_token=myroot --pki_mount=pki --pki_role_name=example-
dot-com  --pki_common_name=mydomain.com --pki_ttl=24h --
pki_renew_time=4h --
pki_alt_names=otherdomain.com,testing.com --
pki_ip_sans=127.0.0.1
```

This will issue a certificate with a time to live of 24 hours, which will get renewed 4 hours before it will expire. Running this as a scheduled task every 15 minutes will ensure that the certificate is renewed in time, even if there are short interruptions to vault. Additionally a renew-hook can be specified, which can for example restart your application after the certificate has been renewed.

If you want to try it yourself, check out the artifacts section to download builds for all major platforms, or pull the official container from dockerhub.

Vaultbot aims to provide an easy and lightweight way to keep your certificates up-to-date. Head over to the gitlab page and give it a try now!