## classic **“Leaf Certificate Automation at Scale”** use case

**Vault + Consul Template (or Vault Agent) on each server**

**Each server will:**

* Connect securely to Vault
* Authenticate (via **AppRole**, **TLS**, or **Token**)
* Request its **own TLS certificate** (using its hostname)
* Write it to disk (e.g., /etc/nginx/certs/)
* Restart the app (e.g., nginx, grafana, apache, etc.) if the cert is updated
* Auto-renew on expiration

A diagram of a server

AI-generated content may be incorrect.

Solutioning:

1. On Vault: Create a Role:

* This role will be used by **every server** to get its cert.

vault write pki\_int/roles/server-certs \

allowed\_domains="corp.example.com" \

allow\_subdomains=true \

allow\_any\_name=false \

enforce\_hostnames=true \

ttl="24h"

1. On Each Server: Install consul-template (or Vault Agent)

wget https://releases.hashicorp.com/consul-template/0.32.0/consul-template\_0.32.0\_linux\_amd64.zip

unzip -o consul-template\_0.32.0\_linux\_amd64.zip

sudo mv consul-template /usr/local/bin/

1. Each Server Gets a Unique Cert Template

Next: follow all the other templates on document 5