vault

vault by hashicorp

basics

vault is a secrets management engine by hashicorp.

some cli usage

after installing `vault`:

login to your instance: `vault login -address https://vault.arul.io`

using concourse as an example:

enable kv store: `vault secrets enable -version=2 -path=concourse kv`

`concourse-policy.hcl`:

```
path "concourse/*" {
  capabilities = ["read"]
}
```

save the policy: `vault policy write concourse ./concourse-policy.hcl`

create a token with the above policy: `vault token create --policy concourse --period 1h`

deployment

I deploy vault with docker compose.

caveats

when vault or docker restart, the vault is sealed. this can be problematic when other programs have short-lived access tokens, since they will be unable to renew the tokens, therefore being left with expired tokens.

configuration

`docker-compose.yml`:

```
version: '3'

services:
    vault:
      image: vault:latest
      volumes:
        - ./config:/vault/config
        - ~/data/vault/file:/vault/file
      cap_add:
        - IPC_LOCK
      networks:
        - web
      labels:
        traefik.enable: true
        traefik.http.routers.vault.entrypoints: https
        traefik.http.routers.vault.rule: Host(`vault.arul.io`)
        traefik.http.services.vault.loadbalancer.server.port: 8200
      command: vault server -config=/vault/config/vault.json
      restart: unless-stopped

networks:
  web:
    external: true
```

`config/vault.json`:

```
{
  "ui": "true",
  "listener":  {
    "tcp":  {
      "address":  "0.0.0.0:8200",
      "tls_disable":  "true",
      "proxy_protocol_behavior": "use_always"
    }
  },
  "backend": {
    "file": {
      "path": "/vault/file"
    }
  },
  "default_lease_ttl": "168h",
  "max_lease_ttl": "720h",
  "api_addr": "http://0.0.0.0:8200"
}
```