KEEPING & SHARING SECRETS

AT ZENDESK

ASH MCKENZIE

TECH LEAD DEVOPS ENGINEER

ZENDESK

TOPICS

SECRETS?

ZENDESK: PRE VAULT

VAULT VS. VAULT

ZENDESK'S VAULT

HASHICORP'S VAULT

SECRETS?

SECRETS?

- Pieces of information that are confidential, private or sensitive in nature
- Examples
  - Usernames and/or passwords
  - API keys
  - SSH private keys
  - Base64 encoded binary license files

ZENDESK

PRE VAULT

SECRETS IN YAML FILE

- A single YAML file
- Environment keys at root
- Used by both Chef & applications
- `sync-secrets`
    - Ruby command line tool
    - Relied upon Chef & `scp`
    - Processed in parallel

SECRETS.YML

```
---
staging:
  project_x:
    admin_email: "do-not-reply@zendesk.com"
  new_relic_api_key: "40d6b575e0fed78ee8483c71b4b2ce0e"

production:
  new_relic_api_key: "4a6e4ec5d55a7c38d8a07e1acc2c52a0"
```

STAGING SECRETS.YML

```
---
project_x:
  admin_email: "do-not-reply@zendesk.com"
new_relic_api_key: "40d6b575e0fed78ee8483c71b4b2ce0e"
```

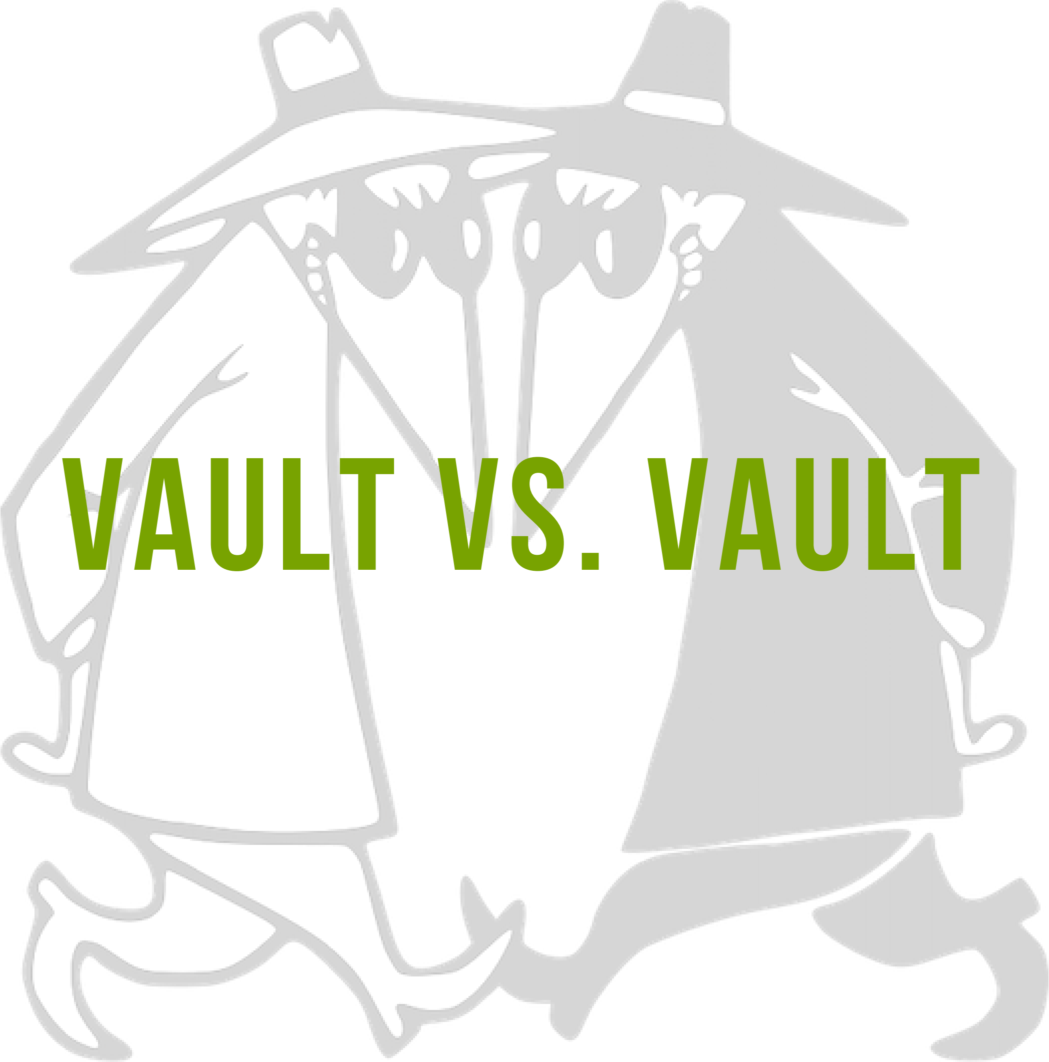
PROS

- Easy to see all secrets 😄
- Single source of truth
- Simple to update, just text!
- Did not rely on an internal / external service
- Access control used Unix permissions
- Managed using Git

CONS

- Data at rest was not encrypted 😞 😞
- Easy to see all secrets 😞
- Single source of truth
- Increasingly slow to push out updates
- Window where some servers could be out-of-sync
- New or previously offline nodes come online and could be out-of-sync
- Difficult to quickly update / revoke a secret
- Not possible for non Ops staff to maintain

VAULT VS. VAULT



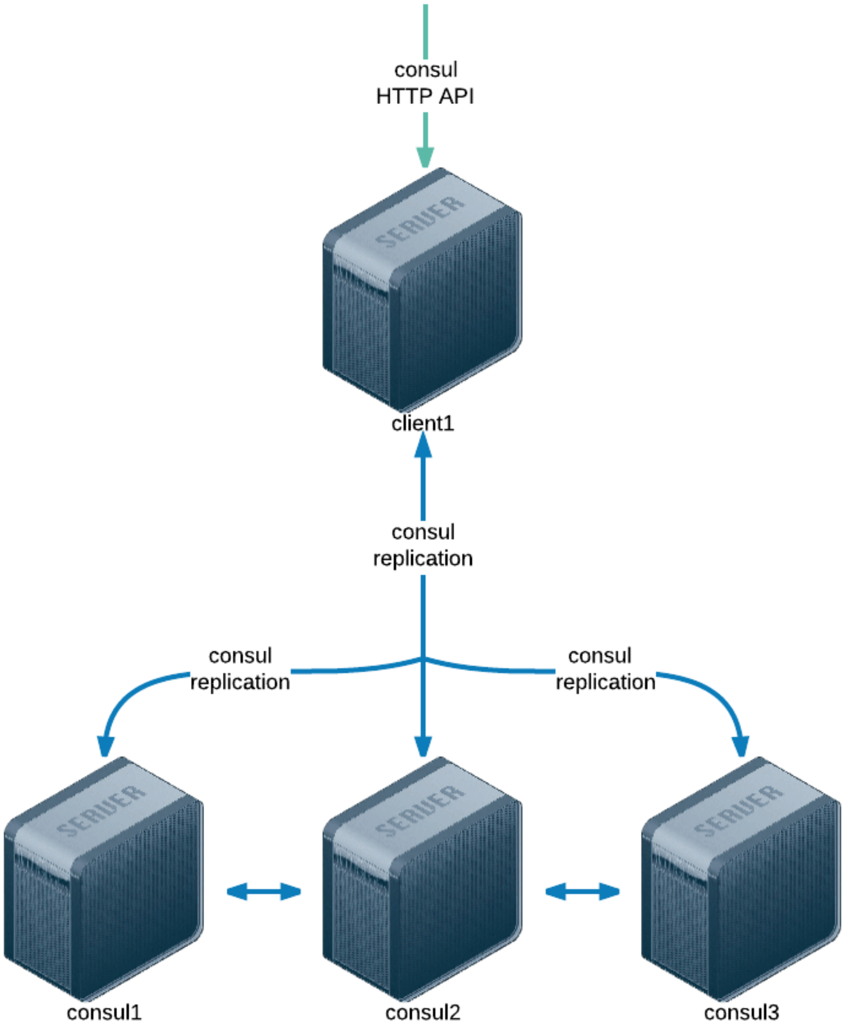
ZENDESK'S VAULT

OVERVIEW

- Utilises Hashicorp's (awesome) Consul
  - key -> value store
  - HTTP API
- Ruby library & command line tools
  - List / Get / Add / Update / Remove
- Used by both Chef & applications
- In production ~ 18 months
- Created pre Hashicorp's Vault

ZENDESK: VAULT

```
$ list_secret '/project_x/admin_email'
do-not-reply@zendesk.com
```



PROS

- Automatically replicated (with Consul) 😄 😄
- Centraliased storage 😄
- Extremely easy to quickly update / revoke a secret
- Environment only secrets exposed
- Add / Update / Remove from any node (for given environment)
- Potential to allow non Ops staff to manage

CONS

- Data at rest is not encrypted 😞 😞
- No Git style commit history 😞
- Requires Consul up and running (we use it anyway)
- All secrets for an environment exposed
- Manual step of granting access required

HASHICORP'S VAULT

OVERVIEW

- Provides secure storage & retrieval of secrets
  - via command line tool & HTTP API
- Supports many secret, auth & audit backends
- Seal/Unseal concept
- Supports key rotation 😂
- Has a super convenient `-dev` mode

PROS

- HTTP API 😂 😂
- Dynamic secrets 😂
- Access Control Policies 😂
- Multiple secret backends
  - Consul, generic, MySQL, AWS
- Multiple auth backends
  - TLS certs, token, GitHub, user/pass
- Multiple audit backends
  - File, syslog

CONS

- Requires vault service running

MIGRATING TO HASHICORP'S VAULT

BENEFITS

- Maintained by a world class team
- Full featured HTTP based API
- Multiple backend types (and future support)
- Solves the following concerns with our Vault:
    - Data at rest is encrypted 😂 😂
    - Comprehensive audit logging (file / syslog)
    - Ability to have multiple vaults

INTERACTIVE DEMO

www.vaultproject.io/#/demo/0

THANKS!

WE'RE HIRING, SO COME AND TALK TO ME!

OR

AMCKENZIE@ZENDESK.COM