We are changing
the way the world moves

Critical
TechWorks

Secrets management with Hashicorp Vault

Focus

- What is secrets management
- Why do we need it
- What is Hashicorp Vault and how can it help you secure your infra/processes
- Vault features and possible use cases

Goal

- Raise awareness regarding security best practices
- Think about your approach to secrets management in your project
- Inspect and adapt your security design
- Go and play with Hashicorp Vault **(or any other tool / process)**

Secrets management 101

"**Secrets management** refers to the tools and methods for **managing** digital authentication credentials (**secrets**), including passwords, keys, APIs, and tokens for use in applications, services, privileged accounts and other sensitive parts of the IT ecosystem."

Common misconceptions

- "We only have 2 or 3 credentials, it's not worth the trouble"
- "Hashicorp Vault is difficult to deploy and manage"
  - Easy deployment – Service / Container / Kubernetes / Cloud Managed Solution
  - Easy update – Single binary
  - Built for high avilability – Raft storage backend / Consul / etc
  - Easy maintenance – Backups / restore with a single command (High availability) or a disk snapshot (standalone)
- "I do not need auditing"
- Keepass is secure / Git is secure

Secrets management 101

- **Don't let your authentication secrets live forever** > Limit of uses, short ttl
- **Distribute auth secrets securely** > Vault with HTTPS / leveradge already implemented infra (Jenkins / Orchestrator / etc)
- **Limit exposure if auth secrets are disclosed** > Use principle of least priviledge in your roles
- **Have a break-glass procedure in case of auth secrets are stolen/exposed** > Use audit logs and revoke API
- **Detect unauthorized access to auth secrets** > APP should alert if secret is absent/no good

Who am I and why talk about this

- Carlos Cunha

  Past :
  - Windows Sysadmin and Ops guy for more than 20 years

  Present :
  - DevOps Engineer in the CTW ITOps Team

Team Goal

Advertise best practices and tooling for development teams @ CTW

Why Hashicorp Vault

- Unmatch feature set
- Open Source
  - mostly !! Some closed source features aimed for specific scenarios
    - Multi-Datacenter replication, 2FA, FIPS compliance, etc.
- Not vendor or framework specific
- Single binary
- Enterprise support is available if this is a requirement

Vault

HashiCorp Vault

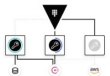
Joy in Motion

Auth-n + Auth-z

(Authentication + Authorization)

Identity-based Access

Authenticate and access different clouds, systems, and endpoints using trusted identities



Secrets Management

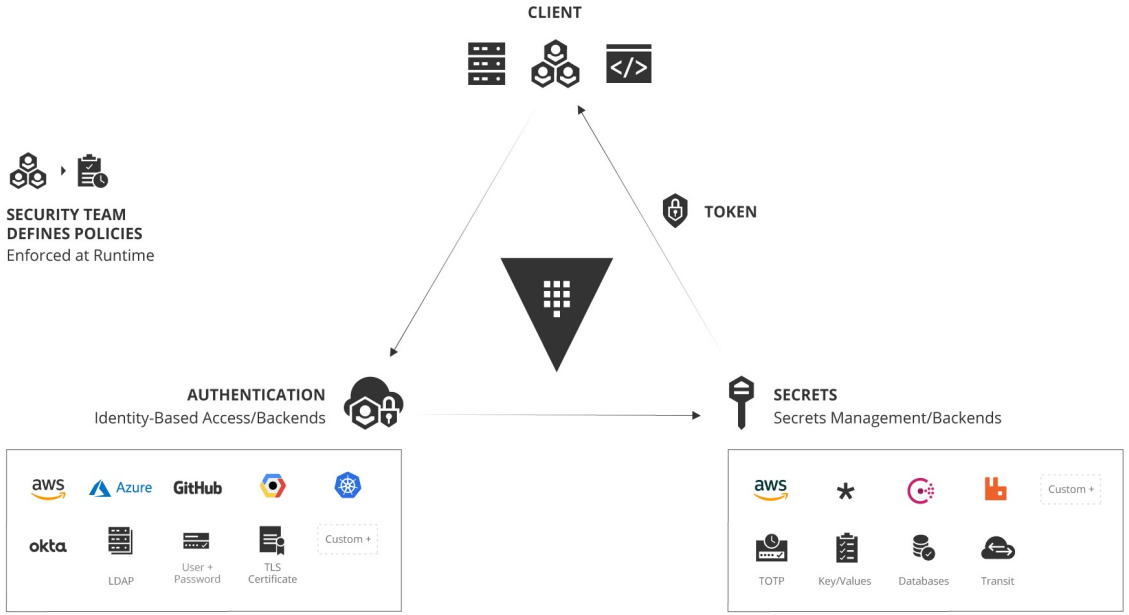
Audit access, automatically Centrally store, access, and deploy secrets across applications, systems, and infrastructure

Data Encryption

Keep secrets and application data secure with one centralized workflow to encrypt data in flight and at rest

Vault

HashiCorp Vault



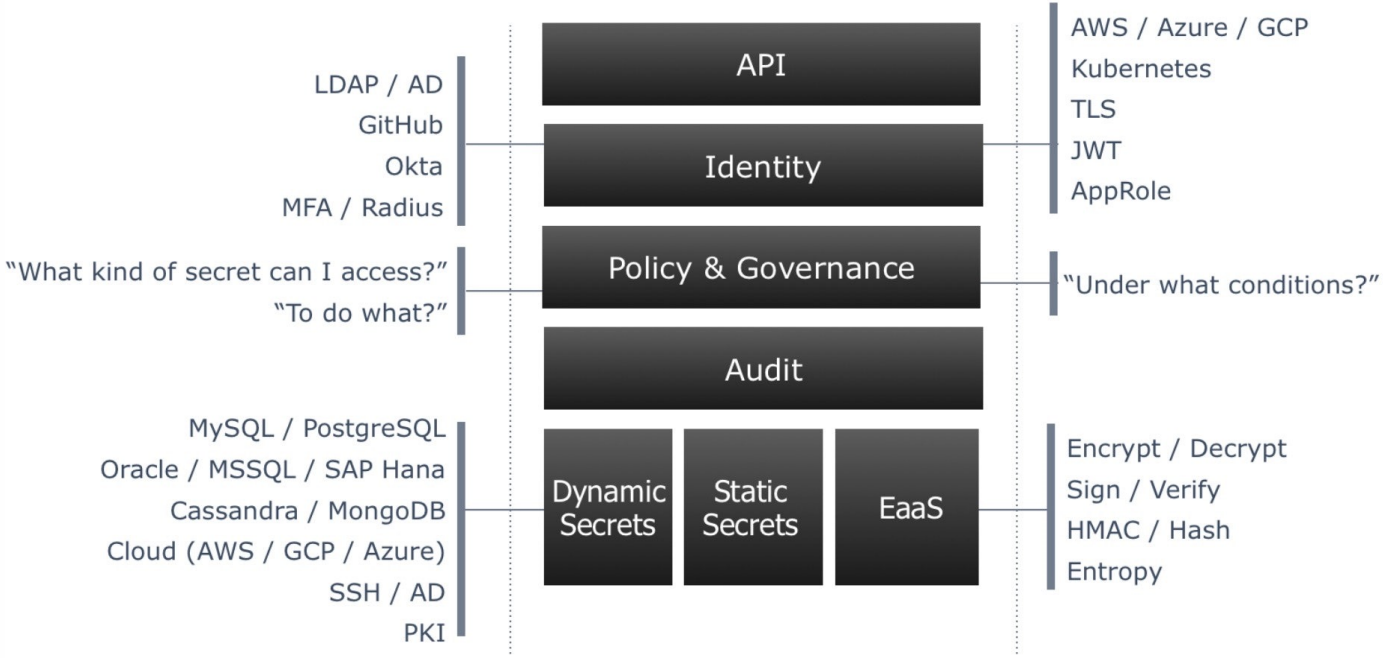
Joy in Motion

Vault

- **Secures, Stores and tightly controls:**
  - Tokens
  - Passwords
  - API Keys
  - Other secrets
- **Handles:**
  - Leasing
  - Key revocation
  - Key rolling
  - Certificates
  - Auditing

All vault functions are build around its API

Vault



Vault

Authentication Backends	
Token	GitHub
AliCloud	OCI
Cloud Foundry	Okta
AWS	Tokens
Oracle Cloud Infrastructure	RADIUS
Google Cloud	TLS Certificates
Azure	Username & Password
LDAP	AppRole
JWT / OIDC	Kerberos
Kubernetes	

Vault

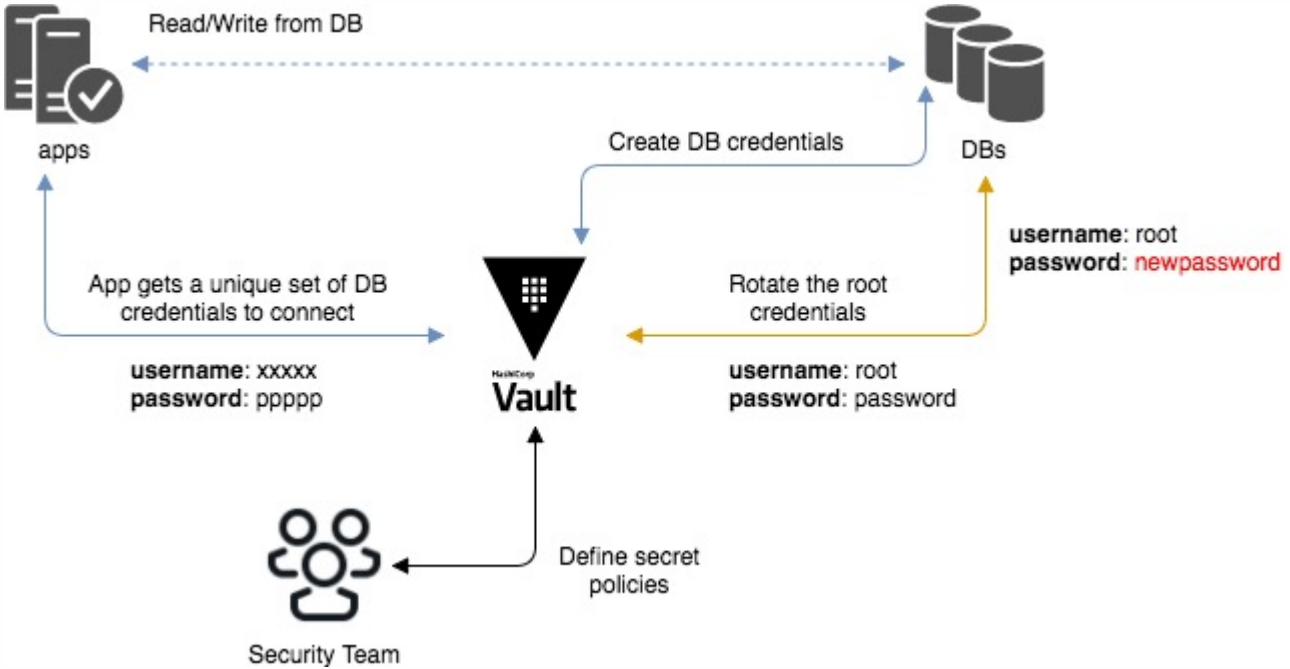
Authorization Engines	
Active Directory	AWS
AliCloud	Nomad
Azure	PKI (certificates)
Consul	RabbitMQ
CubbyHole	SSH
Google Cloud	TOTP
Google Cloud KMS	Transit
Identity	OpenLDAP
Static Secrets (Versioned Key-Value store)	**Databases**

Vault

Databases	
Cassandra	ElasticSearch
InfluxDB	HanaDB
MongoDB	MSSQL
MySQL / MariaDB	PostgreSQL
Oracle	Custom

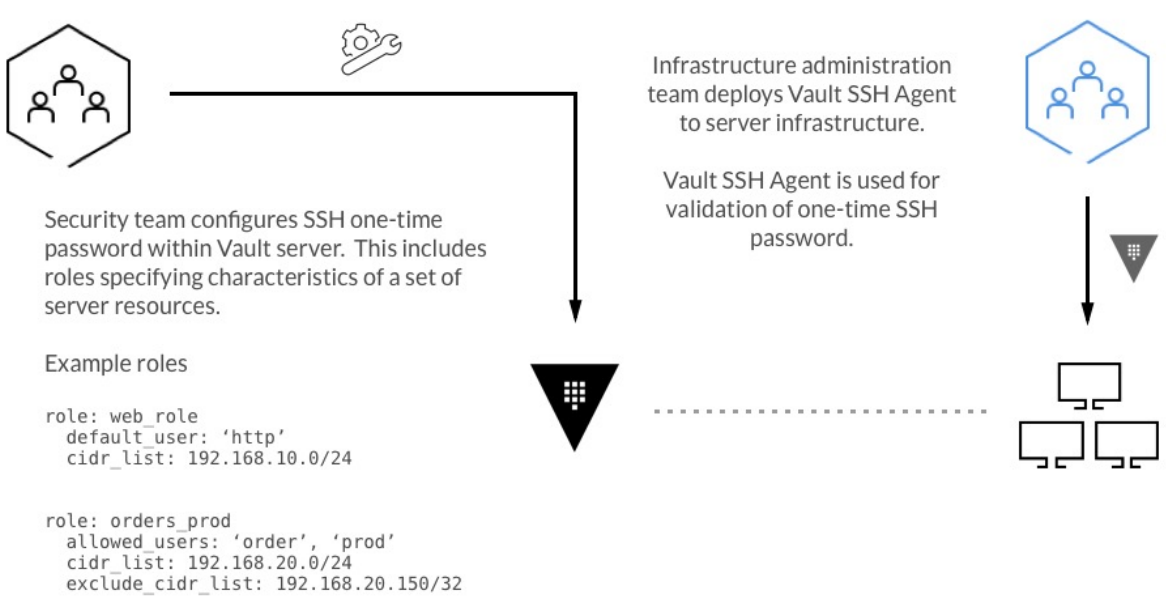
Vault – Database engine

Credential generation and rotation



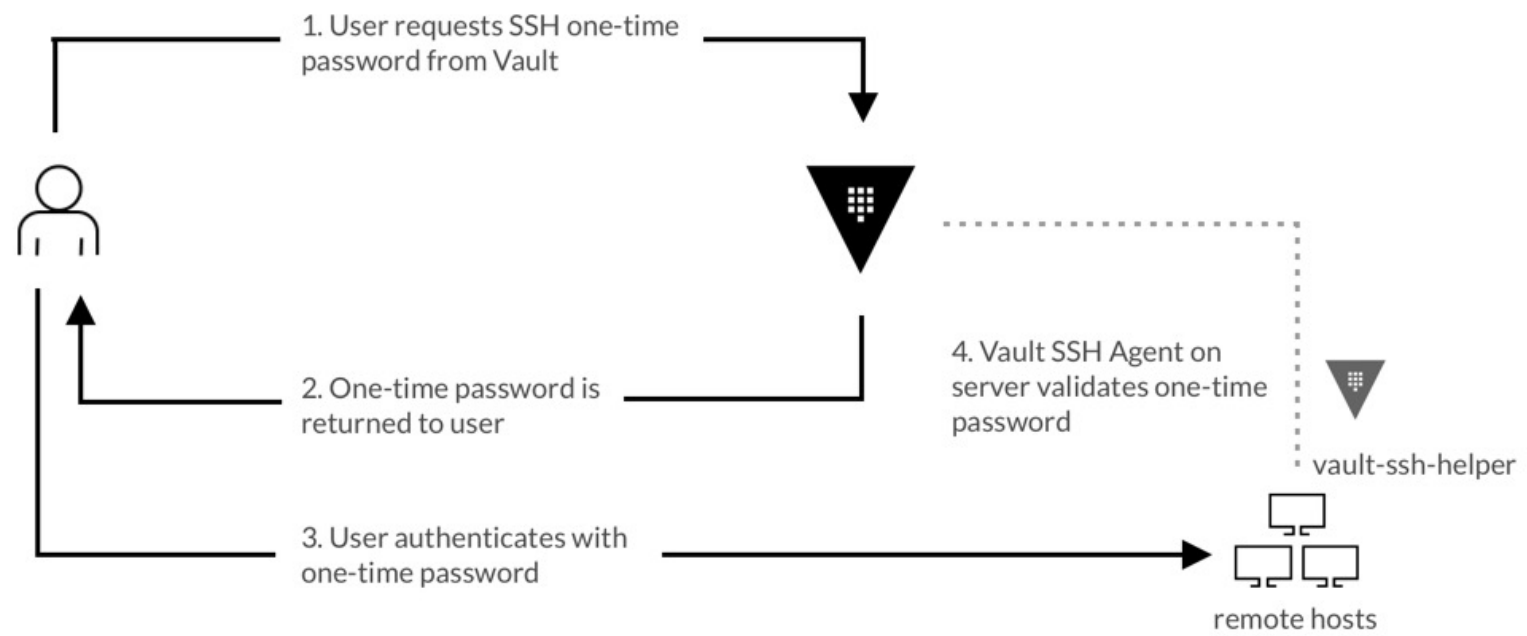
Vault – SSH engine

OTP (implementation)



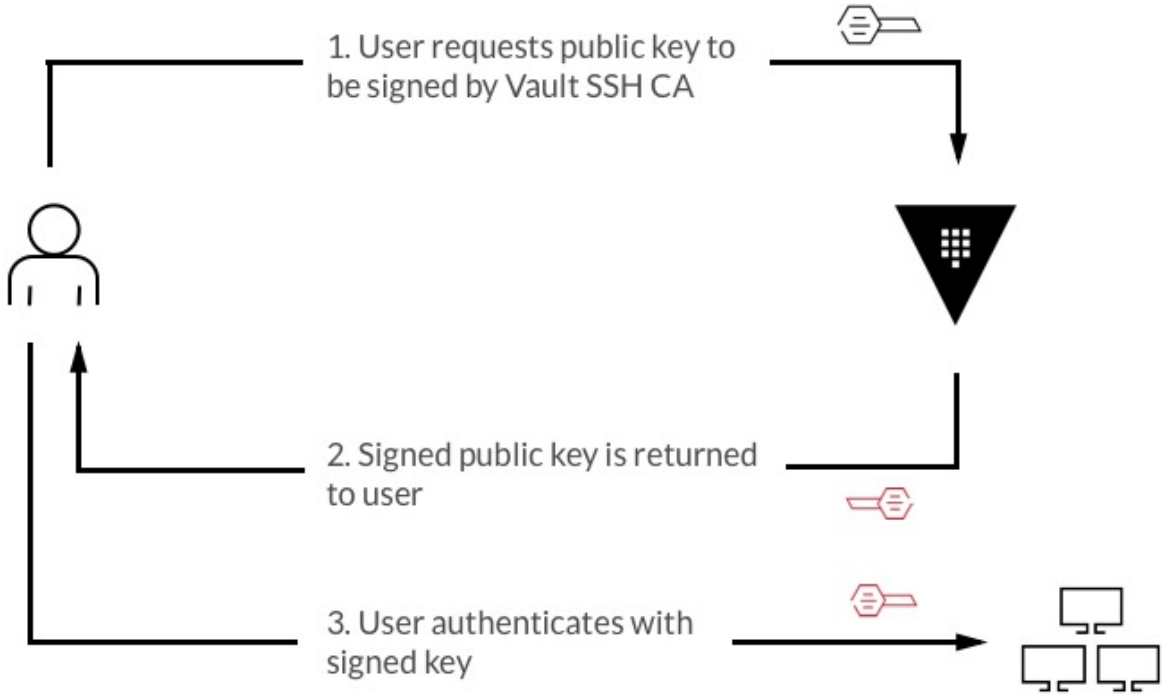
Vault – SSH engine

OTP (usage scenario)



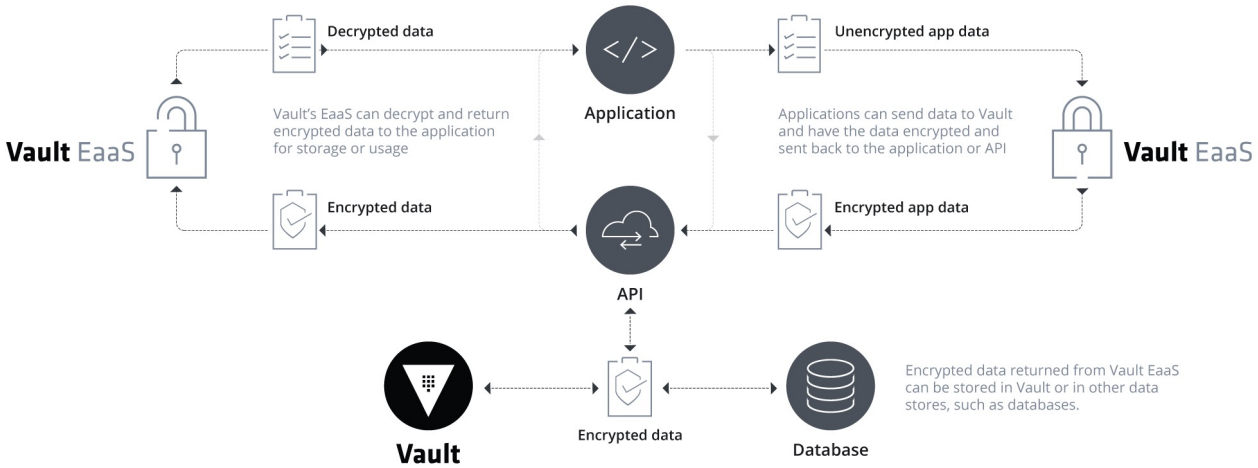
Vault – SSH engine

Public Key Signing



Vault – Transit engine

Encryption as a Service

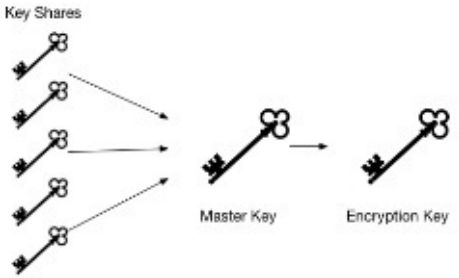


Vault – How the magic happens

Vault Initialization and operation

SHAMIR SECRET SHARING

- Protect Encrypt Key with Master Key
- Split Master Key into N shares
- T shares to recompute Master
- Quorum of key holders required to unseal
  - Default N:5, T:3



HashiCorp

We are changing
the way the world moves

Critical TechWorks

Demo Time

Vault – Information and tutorials

www.vaultproject.io

learn.hashicorp.com/vault

github.com/carlosrbcunha

Q & A

Live simply,

Joy in Motion.

Critical TechWorks

A BMW GROUP & CRITICAL SOFTWARE COMPANY