HashiCorp Certified Vault Associate: Vault Management

Understanding Vault Architecture

Ned Bellavance
Founder, Ned in the Cloud LLC

@ned1313 | nedinthecloud.com

Overview

Exam overview

Vault architecture

Data flow and encryption

Seal options

Exam Overview

Compare authentication methods

Create Vault policies

Assess Vault tokens

Manage Vault leases

Compare and configure Vault secrets engines

Utilize Vault CLI

Utilize Vault UI

Be aware of the Vault API

Explain Vault architecture

Explain encryption as a service

Explain Vault architecture

Explain encryption as a service

Describe the encryption of data stored by Vault

Describe cluster strategy

Describe storage backends

Describe the Vault agent

Describe secrets caching

Be aware of identities and groups

Describe Shamir secret sharing and unsealing

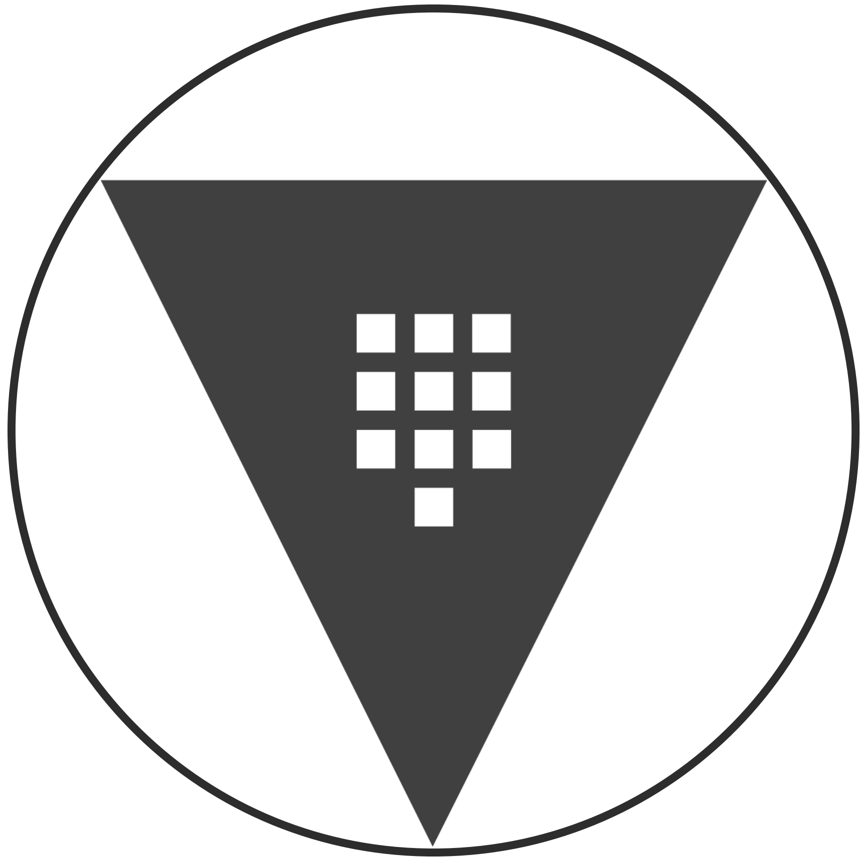
Be aware of replication

Describe seal/unseal

Explain response wrapping

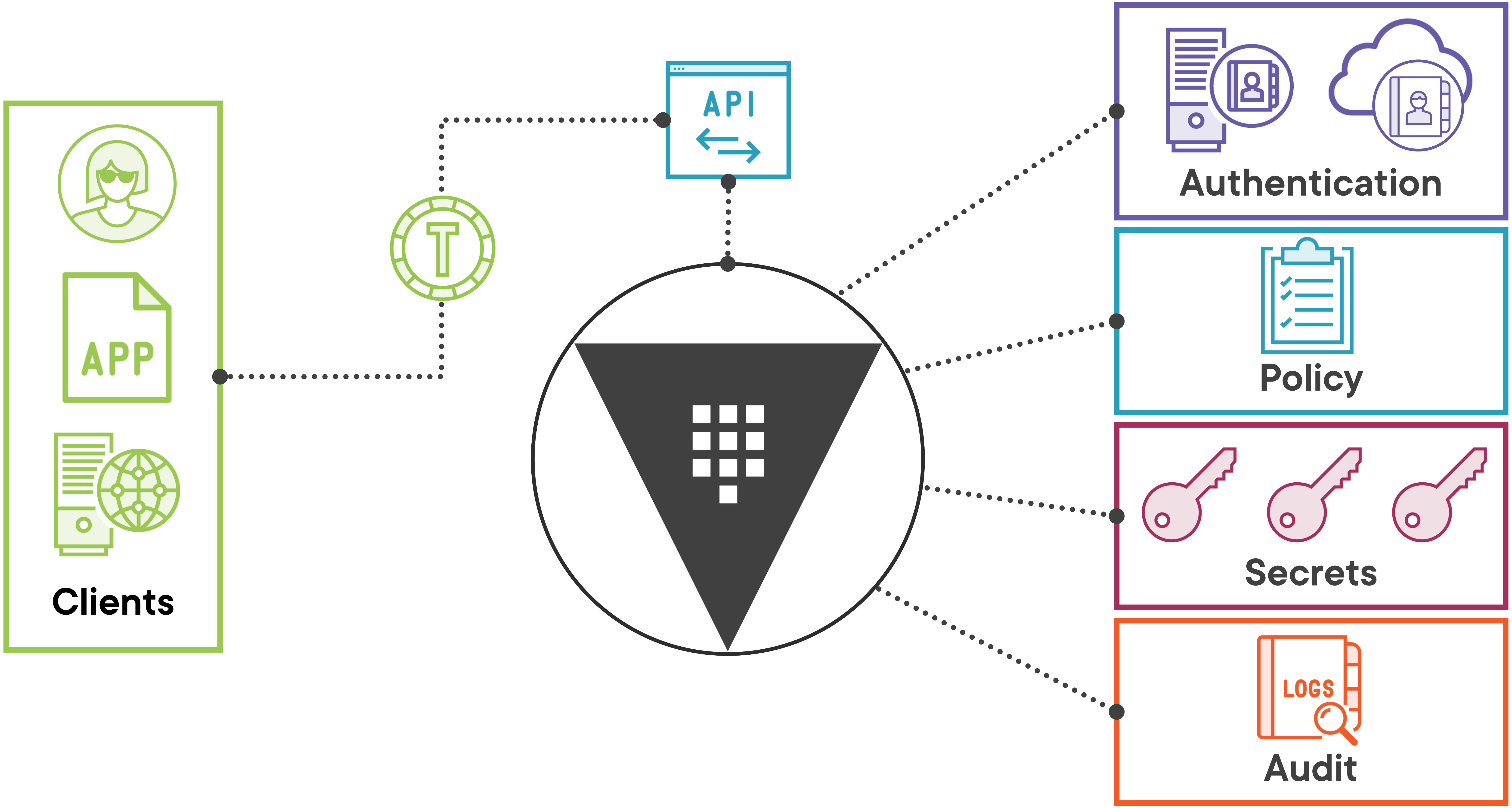
Explain the value of short-lived, dynamically generated secrets

Globomantics Scenario

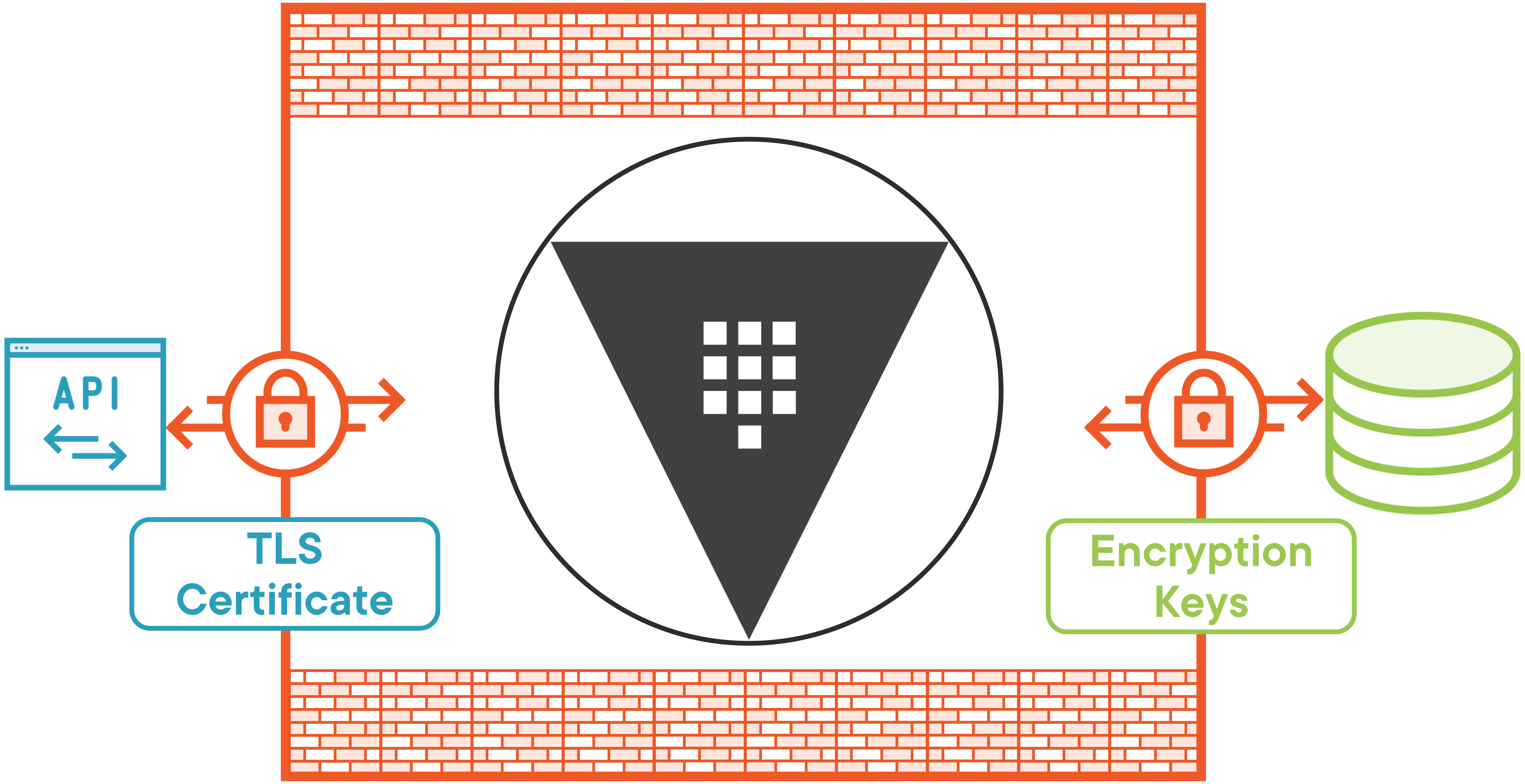


Vault Architecture

Vault Conceptual Architecture



Vault Logical Architecture



Encryption Keys

Encryption keys

Protect data written to storage

Stored on disk

Master key

Protects encryption keys

Stored on disk

Unseal key

Protects master key

Stored as shares or externally

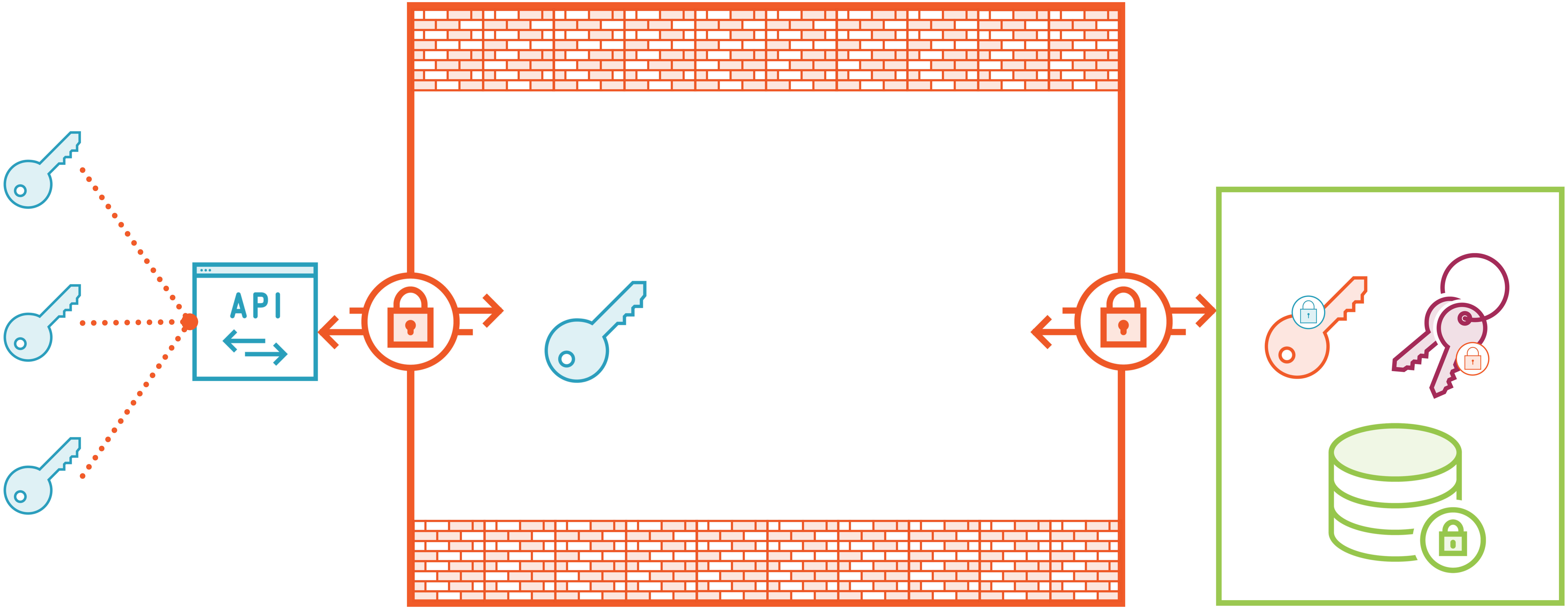
Seal Options

Shamir secret sharing
- Key shares
- Required threshold
- Configured at initialization
- Used for sensitive operations

Auto-unseal
- External service
- Recovery key shares
- Set by Vault server configuration

Seal Migration

Unsealing Vault



Globomantics Scenario

Use Case

- Vault startup should not require human intervention
- Privileged operations should require three people
- Vault will have access to an HSM

Solution

- Set seal type to pkcs11 with HSM values
- Create recovery keys with at least four shares and a threshold of three

Key Takeaways

- All data that leaves the barrier will be encrypted.

- Data leaving the front-end API is encrypted using TLS. Data written to back-end storage is encrypted using the Encryption keys.

- The Encryption keys are protected by the Master key which is protected by the Unseal key.

- The Unseal key is never stored in Vault and can be broken into Shamir key shares or stored on a Cloud/HSM service.

- The Vault must be unsealed before use with the Unseal key.

Up Next: Deploying a Basic Vault Server