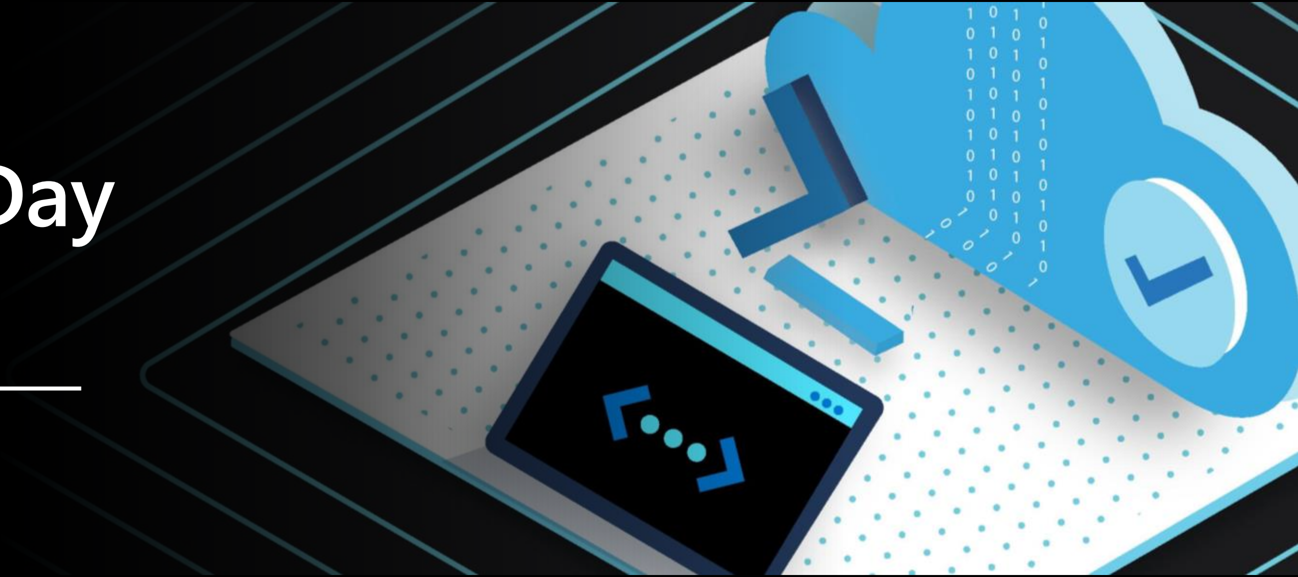
Microsoft

# Open Azure Day Canada



**Webinar:** Secure together – Apps on AKS with HashiCorp Vault Secrets
**Speaker: Marek Anderson (Arctiq)**

# //Agenda

- Requirements for Secrets Management (Ideal)
- Introduction to HashiCorp Vault
- Architecture for Azure Kubernetes Service (AKS) and HashiCorp Vault
- Apps on AKS consuming secrets from Vault
- Demo Time
- Recap and Outlook
- Q&A

# //about me

**Marek Anderson**
@ArctiqMarek /
GitHub manderson-it

Consultant @ Arctiq
At Arctiq, I focus on automation, keeping secrets secret with Vault, and shipping containers.

# //Requirements for Secrets Management (Ideal)

1. Works on-premise and on any cloud
2. Integrates with tech stacks by REST API
3. Future-proof solution (OSS, community, scales)
4. Truly secure (at rest, in transit, access revocable, pre-researched vulnerabilities)
5. Granular Access Control Lists, and self-service options

# //Introduction to HashiCorp Vault

- Tool for securely storing and accessing secrets
- Key features include:
    - Secure secret storage
    - Dynamic secret generation
    - Data encryption
    - Automatic revocation
    - Detailed audit logs
    - Policy-based secrets controls

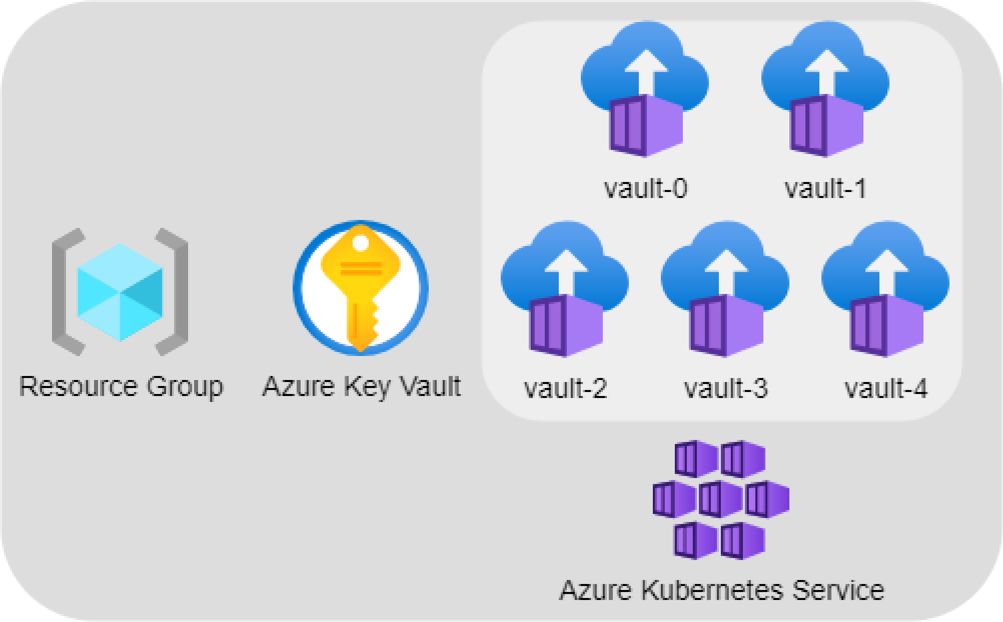
# //Introduction to HashiCorp Vault

# //Architecture for AKS and Vault

Azure
Kubernetes
Service

HashiCorp
Vault

# //Architecture for AKS and Vault



# //Benefits of Cloud KMS

Opinionated advancements in cryptography:

1. Symmetric (Shared long password)
2. Asymmetric (Public-Private Key Pair)
3. Hardware Security Modules (HSM)
4. Cloud Key Management Services (KMS)

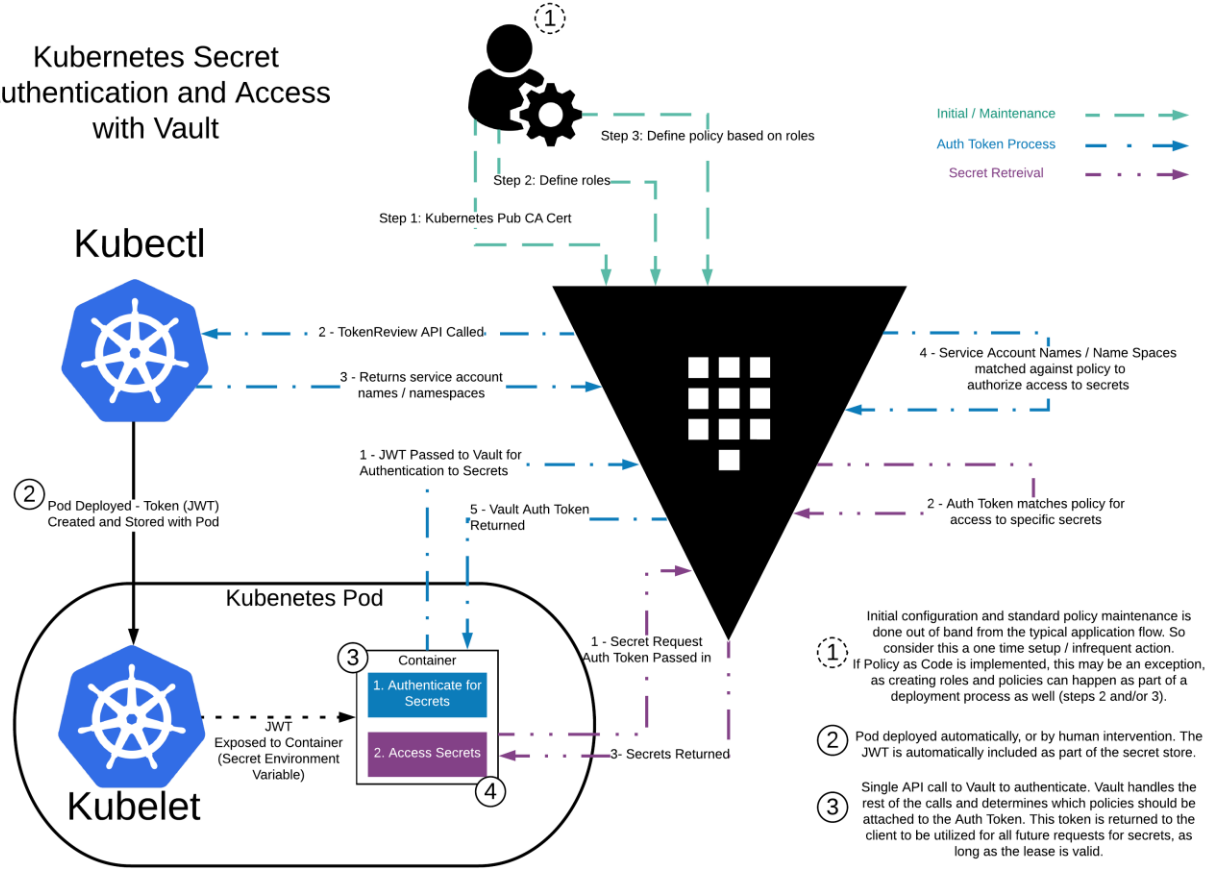
# //Benefits of Cloud KMS (continued)

4. Cloud Key Management Services (KMS)
    1. using identity instead of encryption/decryption keys
    2. KMS are cheap
    3. KMS expose REST API -> user/automation friendly
    4. KMS are extremely secure
    5. Identity credentials leaked -> revoke
    6. Trivial task with centralized Access Control List (ACL)
    7. Crypto Anchoring: (a) Network ACLs for KMS, and (b) monitor KMS decryption rates for baseline

# //Apps on AKS consuming secrets from Vault

- Vault Agent Templating
- Mutating Webhook Controller
- Annotation-driven (minimal configuration)
- Auto-injection of secrets via init and sidecar containers to authenticate with Vault and get secrets

# Kubernetes Secret Authentication and Access with Vault



1
Step 3: Define policy based on roles
Step 2: Define roles
Step 1: Kubernetes Pub CA Cert
Initial / Maintenance
Auth Token Process
Secret Retreival
Kubectl
2 - TokenReview API Called
3 - Returns service account names / namespaces
4 - Service Account Names / Name Spaces matched against policy to authorize access to secrets
1 - JWT Passed to Vault for Authentication to Secrets
5 - Vault Auth Token Returned
2 - Auth Token matches policy for access to specific secrets
2
Pod Deployed - Token (JWT) Created and Stored with Pod
Kubenetes Pod
1 - Secret Request Auth Token Passed in
3
Container
1. Authenticate for Secrets
2. Access Secrets
4
JWT
Exposed to Container (Secret Environment Variable)
3- Secrets Returned
Kubelet

1 Initial configuration and standard policy maintenance is done out of band from the typical application flow. So consider this a one time setup / infrequent action. If Policy as Code is implemented, this may be an exception, as creating roles and policies can happen as part of a deployment process as well (steps 2 and/or 3).

2 Pod deployed automatically, or by human intervention. The JWT is automatically included as part of the secret store.

3 Single API call to Vault to authenticate. Vault handles the rest of the calls and determines which policies should be attached to the Auth Token. This token is returned to the client to be utilized for all future requests for secrets, as long as the lease is valid.

4 Single API call to Vault, including the Auth Token. Vault returns appropriate secrets.

https://learn.hashicorp.com/tutorials/vault/agent-kubernetes

# Demo Time!

Arctiq
INTELLIGENT ARCHITECTURE

# //Recap and Outlook

- Azure Kubernetes Service and HashiCorp Vault build a strong team
- Benefits of dynamic secrets versus static
- Further increase security with Azure Pod Identity

# //Thank you

Thank you for your time!

Happy to continue the discussion!

Reach out on Twitter

Twitter: @ArctiqMarek

LinkedIn: marekanderson

Arctiq
INTELLIGENT ARCHITECTURE

Microsoft

# Q&A