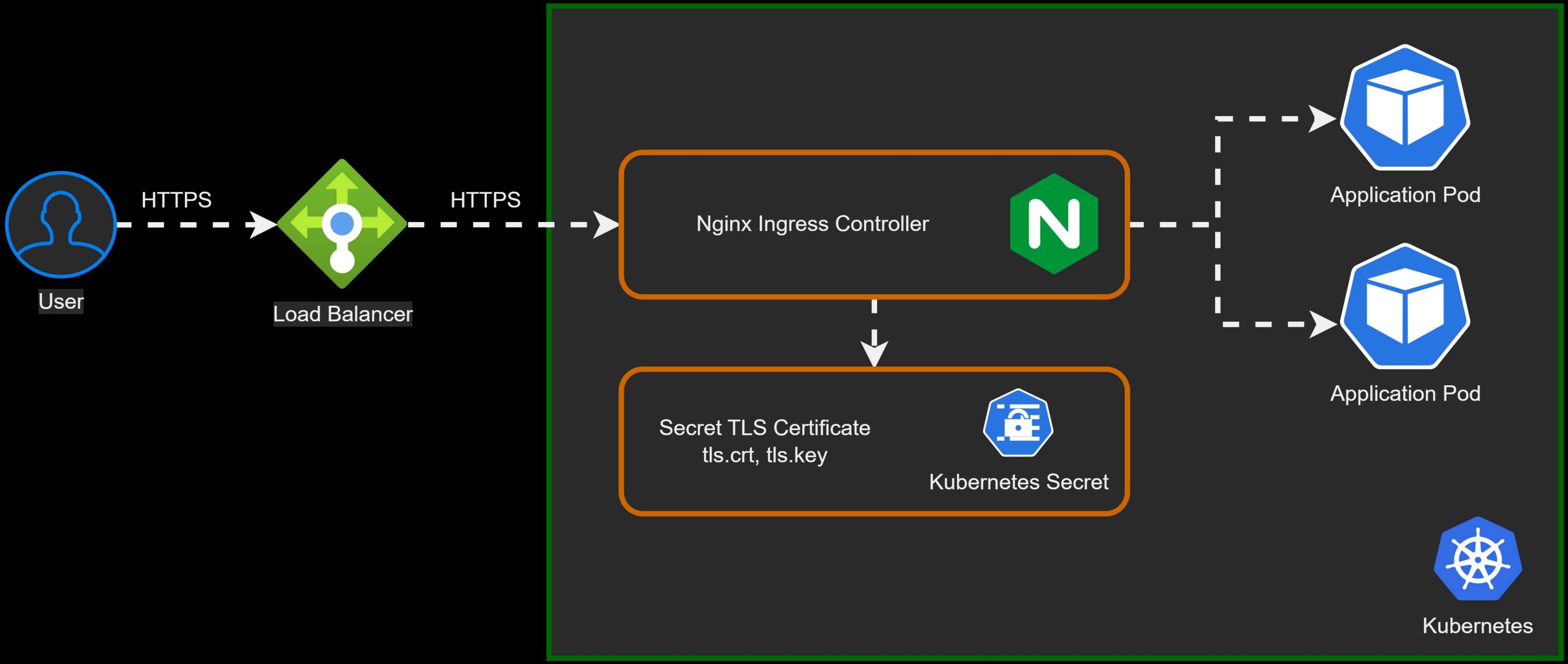
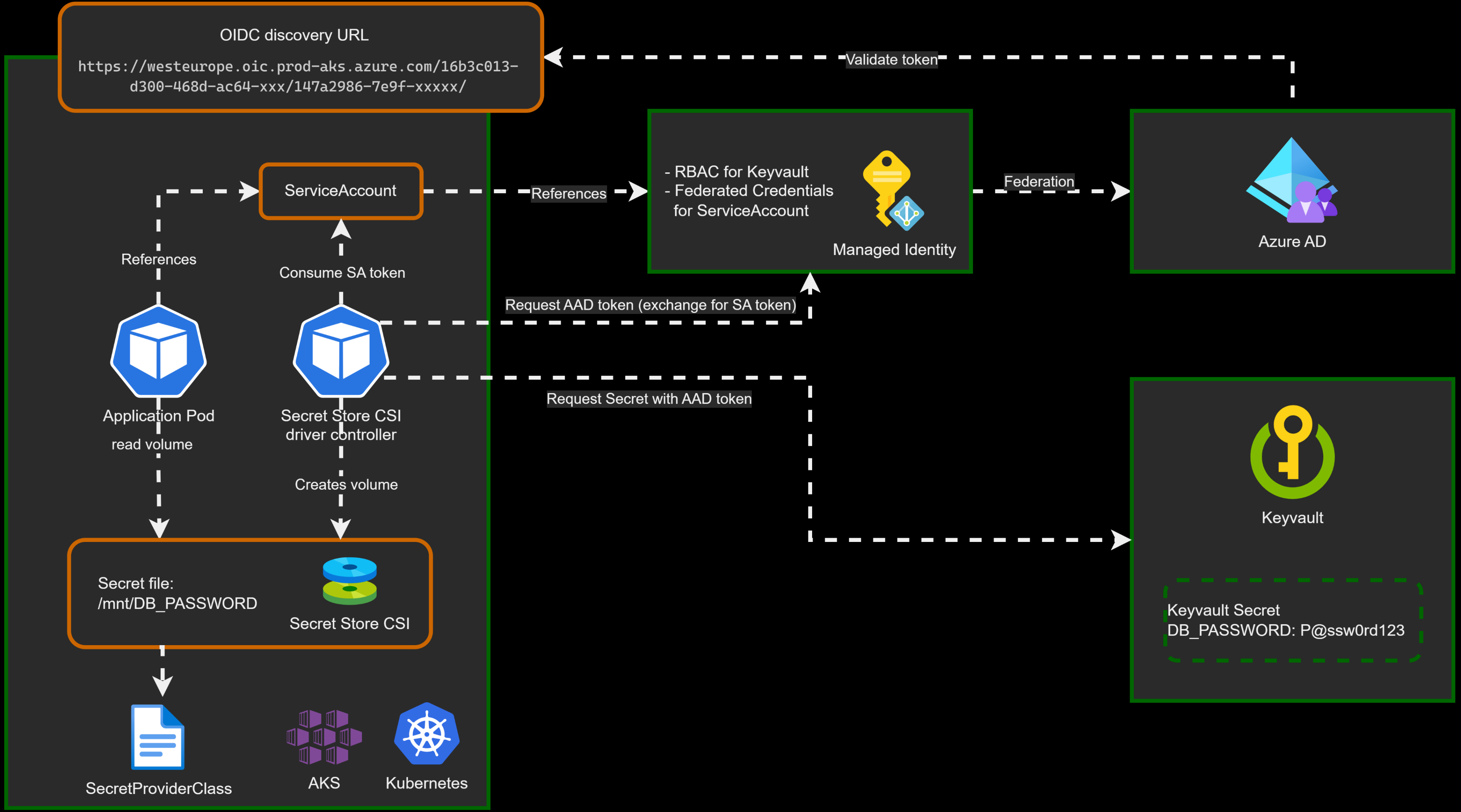
Securing Ingress with TLS/HTTPS and Keyvault

Houssem Dellai

Ingress Controller using TLS certificate from Secret



Secrets Store CSI & Workload Identity



OIDC discovery URL

https://westeurope.oic.prod-aks.azure.com/16b3c013-d300-468d-ac64-xxx/147a2986-7e9f-xxxxx/

Validate token

ServiceAccount

References

- RBAC for Keyvault
- Federated Credentials for ServiceAccount

Managed Identity

Federation

Azure AD

References

Consume SA token

Request AAD token (exchange for SA token)

Application Pod

Secret Store CSI driver controller

Request Secret with AAD token

read volume

Creates volume

Keyvault

Secret file: /mnt/DB_PASSWORD

Secret Store CSI

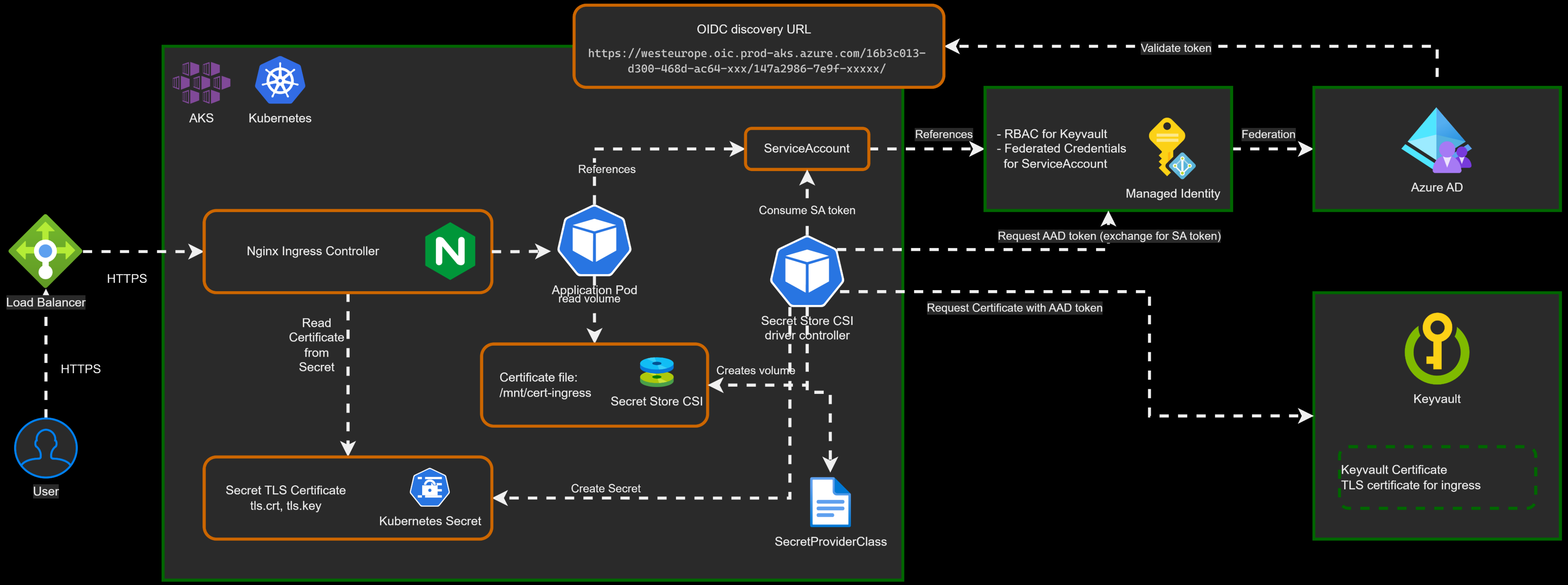
Keyvault Secret
DB_PASSWORD: P@ssw0rd123

SecretProviderClass

AKS

Kubernetes

Ingress Controller using TLS certificate from Keyvault



OIDC discovery URL

https://westeurope.oic.prod-aks.azure.com/16b3c013-d300-468d-ac64-xxx/147a2986-7e9f-xxxxx/

Validate token

AKS

Kubernetes

References

ServiceAccount

References

- RBAC for Keyvault
- Federated Credentials for ServiceAccount

Managed Identity

Federation

Azure AD

Consume SA token

Request AAD token (exchange for SA token)

Load Balancer

HTTPS

Nginx Ingress Controller

Application Pod

read volume

Secret Store CSI driver controller

Request Certificate with AAD token

HTTPS

User

Read Certificate from Secret

Certificate file:
/mnt/cert-ingress

Secret Store CSI

Creates volume

Keyvault

Keyvault Certificate
TLS certificate for ingress

Secret TLS Certificate
tls.crt, tls.key

Kubernetes Secret

Create Secret

SecretProviderClass

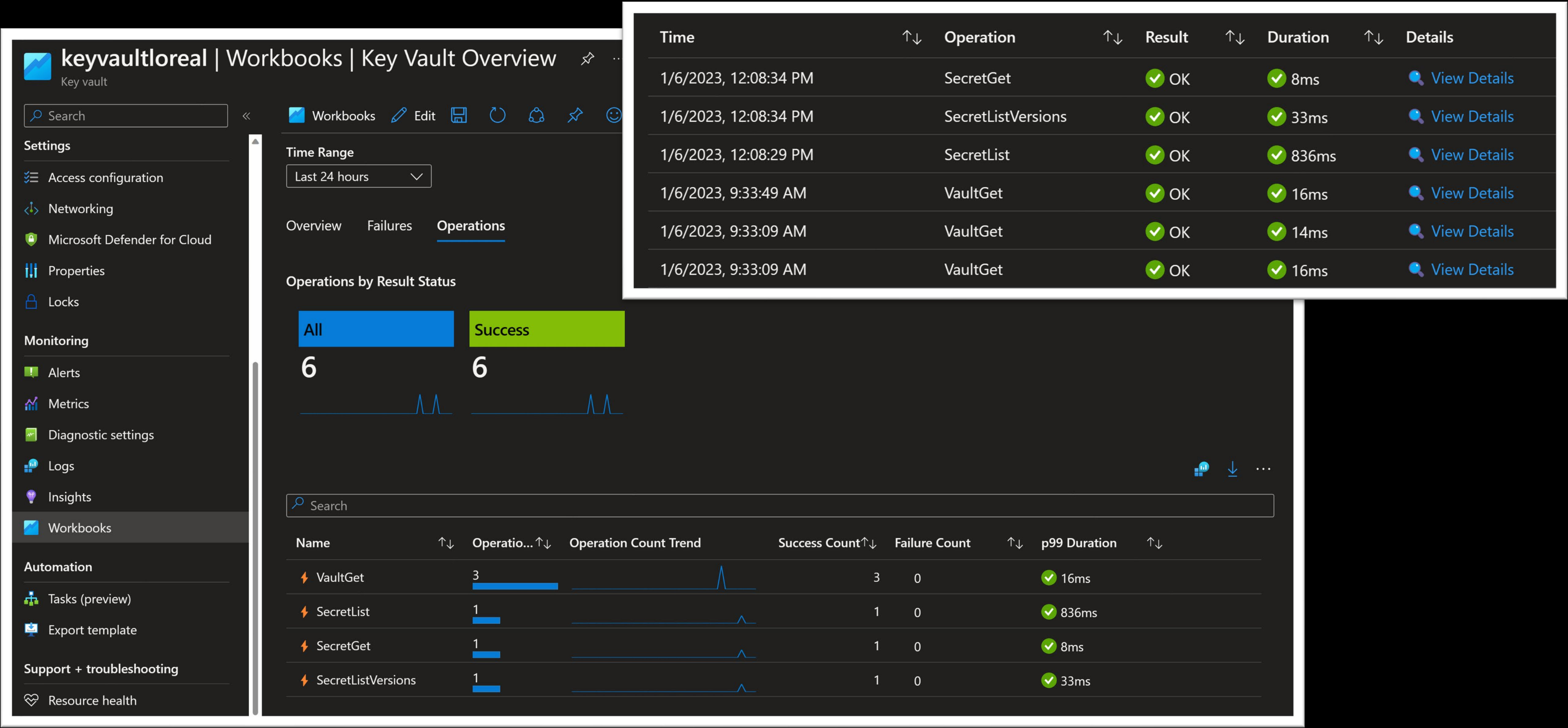
How does it work ?

1. Operator creates TLS certificate and stores it in Keyvault.
2. Secret Store CSI provider pull TLS certificate from Keyvault.
3. Secret Store CSI driver sync TLS certificate into k8s Secret.
4. Ingress controller uses the TLS cert from the k8s secret.
5. Operator can rotate the TLS certificate in Keyvault.
6. Secret Store CSI driver sync TLS certificate into k8s Secret.
7. Ingress controller will ‘hot reload’ the cert (no reboot).

Related topics

1. Keyvault supports private link and service endpoint.
2. Keyvault can auto-rotate certificates (KV self-signed, DigiCert & GlobalSign).
3. Auto-rotation capability is not applicable for certificates created with CAs that are not partnered with Key Vault.
4. Use Azure RBAC instead of Access Policy.
5. Use User Managed Identity.
6. Use Soft-delete secrets/certificates.

Logs and metrics from Key vault



Time	Operation	Result	Duration	Details
1/6/2023, 12:08:34 PM	SecretGet	✓ OK	✓ 8ms	View Details
1/6/2023, 12:08:34 PM	SecretListVersions	✓ OK	✓ 33ms	View Details
1/6/2023, 12:08:29 PM	SecretList	✓ OK	✓ 836ms	View Details
1/6/2023, 9:33:49 AM	VaultGet	✓ OK	✓ 16ms	View Details
1/6/2023, 9:33:09 AM	VaultGet	✓ OK	✓ 14ms	View Details
1/6/2023, 9:33:09 AM	VaultGet	✓ OK	✓ 16ms	View Details

Time Range: Last 24 hours

Operations by Result Status

All 6

Success 6

Name	Operatio...	Operation Count Trend	Success Count	Failure Count	p99 Duration
VaultGet	3		3	0	✓ 16ms
SecretList	1		1	0	✓ 836ms
SecretGet	1		1	0	✓ 8ms
SecretListVersions	1		1	0	✓ 33ms

Best practices for Key vault

1. Start with 1 Key vault per Application per environment.

1. Key vault supports RBAC roles per Secret. Could be used to leverage multi-tenancy.

2. Key vault for secrets and certificates.

3. Azure App Configuration for ‘non sensitive’ data like
app/user settings and feature flags.

Demo

https://github.com/HoussemDellai/docker-kubernetes-course/tree/main/31_https_ingress_pods_kv_oidc