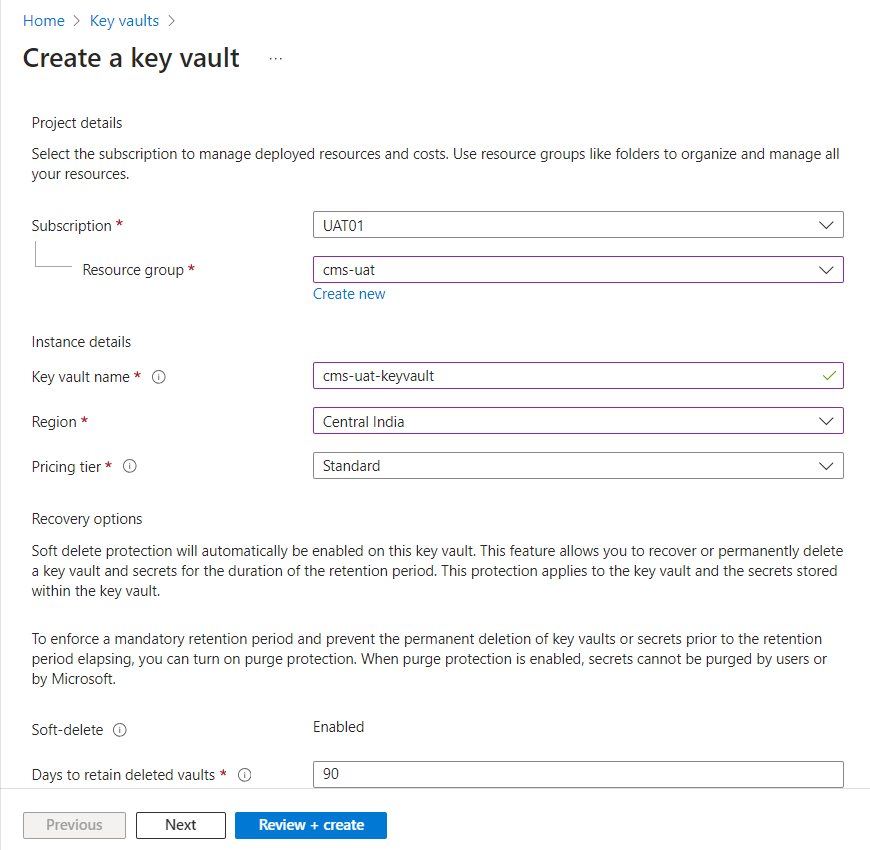
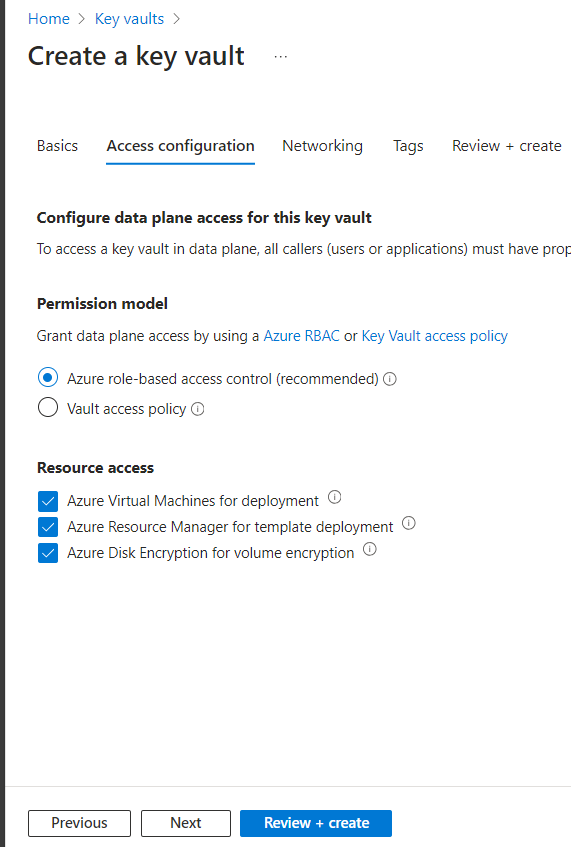
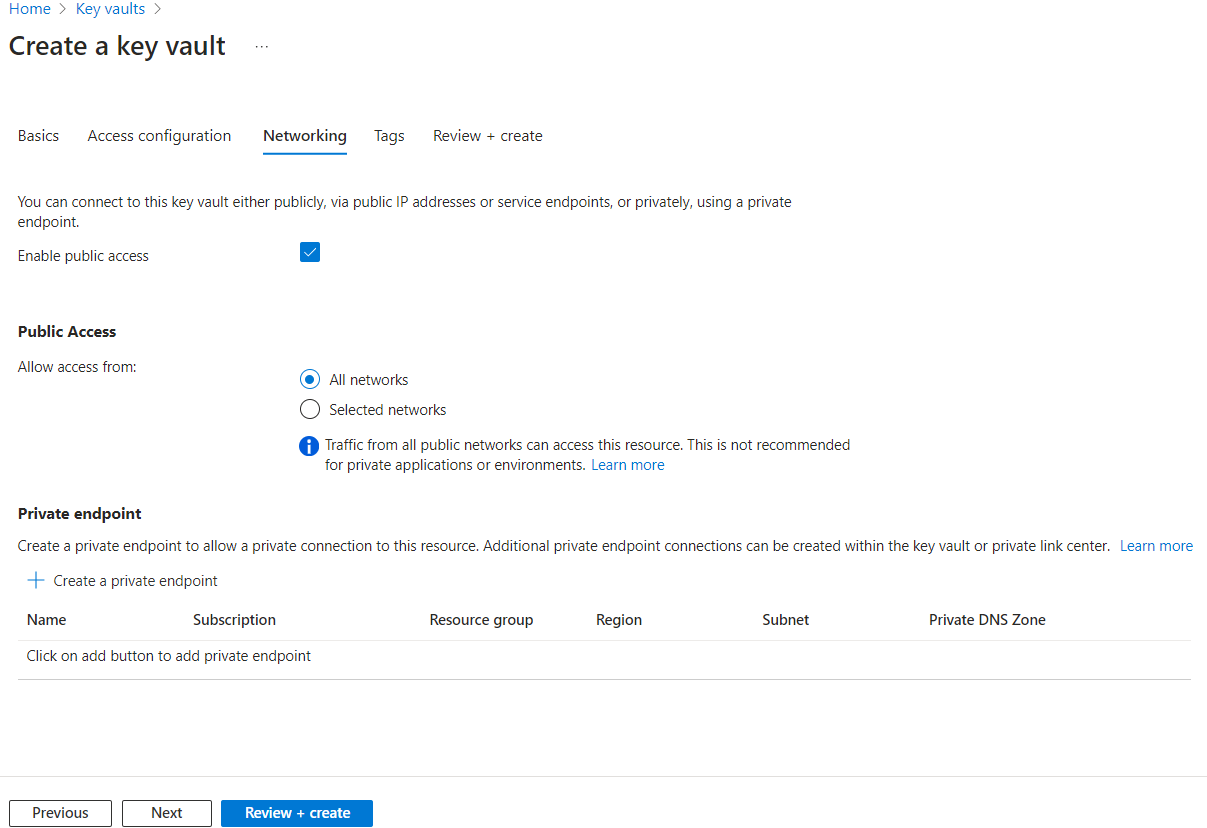
**Creation Of Key Vault:**

**Step 1: login azure portal 🡪 select key vaults form all resources 🡪 click on create**

****

****

****

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Step 2:** After creating key vaults give proper permissions to user like key vault administrator

**installing Azure Key Vault provider for Secrets Store CSI Driver in an Azure Kubernetes Service (AKS) cluster**

**Step:1** we must remove outdated Azure CLI and need to update Azure CLI version with latest.

sudo apt update

Azure CLI remove Command: sudo apt-get remove -y azure-cli

Azure CLI remove Command installation Command:

curl -L https://aka.ms/InstallAzureCli | bash

az login

**Step2:** After installing the latest version of CLI. Start installing Azure Key Vault provider for Secrets Store CSI Driver in an Azure Kubernetes Service (AKS) cluster

URL: [Use the Azure Key Vault provider for Secrets Store CSI Driver for Azure Kubernetes Service (AKS) secrets - Azure Kubernetes Service | Microsoft Learn](https://learn.microsoft.com/en-us/azure/aks/csi-secrets-store-driver)

az aks update --name cms-kshema-uat-aks --resource-group cms-uat --enable-oidc-issuer

az aks enable-addons --addons azure-keyvault-secrets-provider --name cms-kshema-uat-aks --resource-group cms-uat

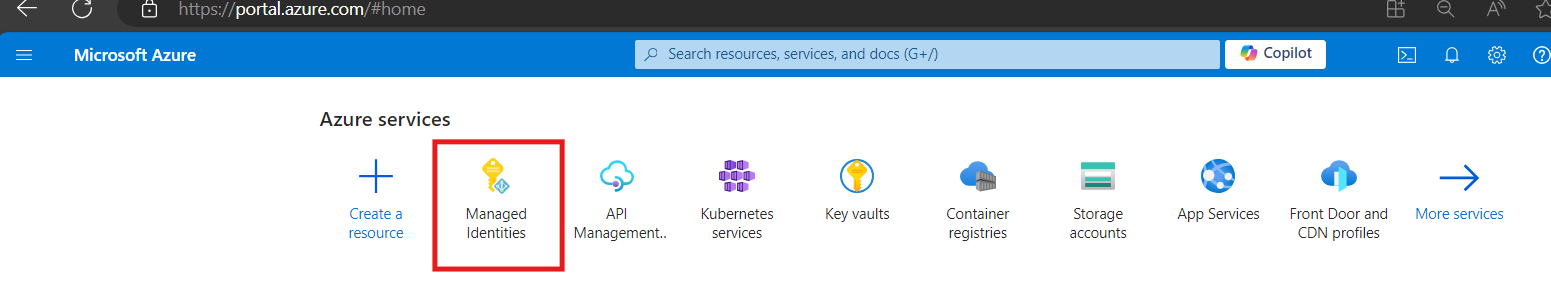
az aks enable-addons --name cms-kshema-uat-aks --resource-group cms-uat --addons azure-keyvault-secrets-provider

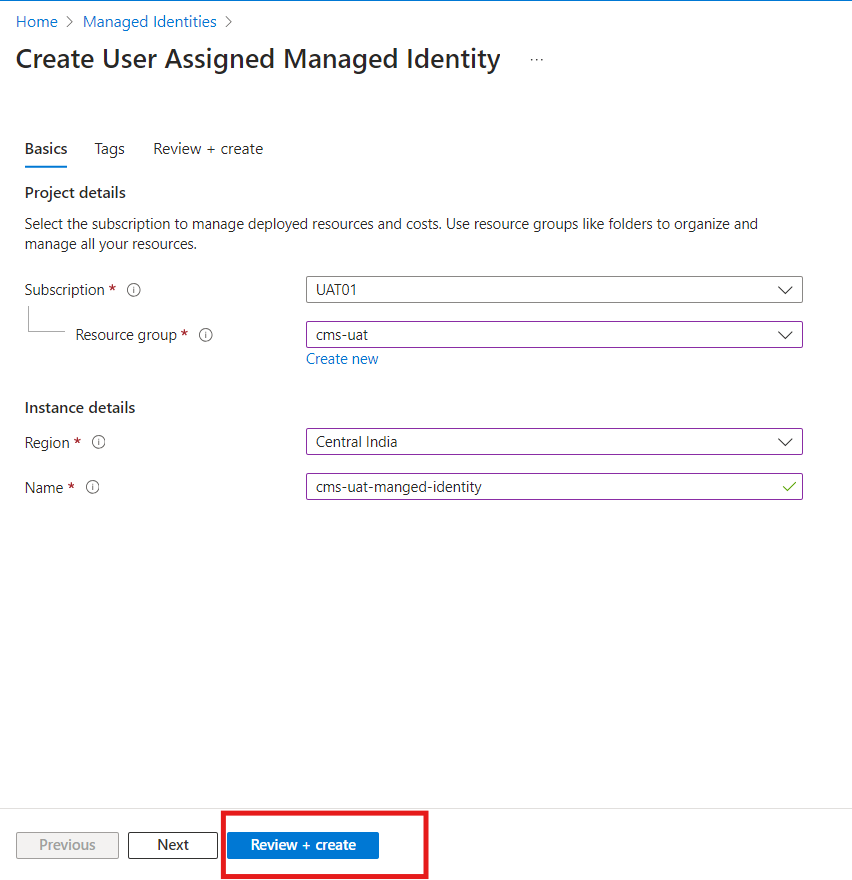
kubectl get pods -n kube-system -l 'app in (secrets-store-csi-driver,secrets-store-provider-azure)'

**Step 3:** Connect your Azure identity provider to the Azure Key Vault Secrets Store CSI Driver in Azure Kubernetes Service (AKS)

**Creation of managed Identity (UAMI):**

Login to to azure portal 🡪 search for managed identity 🡪 create managed identity





A white background with black and blue text

Description automatically generated

A screenshot of a computer

Description automatically generated

Step4: **Configure workload identity**

**URL:** [**Access Azure Key Vault with the CSI Driver Identity Provider - Azure Kubernetes Service | Microsoft Learn**](https://learn.microsoft.com/en-us/azure/aks/csi-secrets-store-identity-access#code-try-2)

export SUBSCRIPTION\_ID=24635f37-d8af-411b-80ee-45c26a30d005

export RESOURCE\_GROUP=cms-uat

export UAMI=cms-uat-ak-uami

export KEYVAULT\_NAME=cms-uat-kv-aks

export CLUSTER\_NAME=cms-kshema-uat-aks

az account set --subscription $SUBSCRIPTION\_ID

az identity create --name $UAMI --resource-group $RESOURCE\_GROUP

export USER\_ASSIGNED\_CLIENT\_ID="$(az identity show --resource-group $RESOURCE\_GROUP --name $UAMI --query 'clientId' -o tsv)"

export IDENTITY\_TENANT=$(az aks show --name $CLUSTER\_NAME --resource-group $RESOURCE\_GROUP --query identity.tenantId -o tsv)

export IDENTITY\_OBJECT\_ID="$(az identity show --resource-group <resource-group> --name <identity-name> --query 'principalId' -o tsv)"

export IDENTITY\_OBJECT\_ID="$(az identity show --resource-group $RESOURCE\_GROUP --name $UAMI --query 'principalId' -o tsv)"

echo $IDENTITY\_OBJECT\_ID

az keyvault set-policy --name $KEYVAULT\_NAME --key-permissions get --object-id $IDENTITY\_OBJECT\_ID

export KEYVAULT\_SCOPE=$(az keyvault show --name $KEYVAULT\_NAME --query id -o tsv)

export SERVICE\_ACCOUNT\_NAME="workload-identity-sa"

export SERVICE\_ACCOUNT\_NAMESPACE="ingress-basic"

echo ${USER\_ASSIGNED\_CLIENT\_ID}

cat <<EOF | kubectl apply -f -

apiVersion: v1

kind: ServiceAccount

metadata:

annotations:

azure.workload.identity/client-id: ${USER\_ASSIGNED\_CLIENT\_ID}

name: ${SERVICE\_ACCOUNT\_NAME}

namespace: ${SERVICE\_ACCOUNT\_NAMESPACE}

EOF

export FEDERATED\_IDENTITY\_NAME="aksfederatedidentity"

export AKS\_OIDC\_ISSUER="$(az aks show --resource-group $RESOURCE\_GROUP --name $CLUSTER\_NAME --query "oidcIssuerProfile.issuerUrl" -o tsv)"

echo ${AKS\_OIDC\_ISSUER}

az identity federated-credential create --name $FEDERATED\_IDENTITY\_NAME --identity-name $UAMI --resource-group $RESOURCE\_GROUP --issuer ${AKS\_OIDC\_ISSUER} --subject system:serviceaccount:${SERVICE\_ACCOUNT\_NAMESPACE}:${SERVICE\_ACCOUNT\_NAME}

**Step 4:** Create SecretProviderClass:

apiVersion: secrets-store.csi.x-k8s.io/v1

kind: SecretProviderClass

metadata:

name: azure-kvname-wi

namespace: ingress-basic

spec:

provider: azure

secretObjects:

- secretName: my-k8s-secret

type: Opaque

data:

- objectName: azure-connection-string

key: azure-connection-string

- objectName: azure-storage-sas-token

key: azure-storage-sas-token

- objectName: camunda-first-name

key: camunda-first-name

- objectName: camunda-jdbc-password

key: camunda-jdbc-password

- objectName: camunda-jdbc-url

key: camunda-jdbc-url

- objectName: camunda-jdbc-username

key: camunda-jdbc-username

- objectName: camunda-user-id

key: camunda-user-id

- objectName: camunda-user-password

key: camunda-user-password

- objectName: cms-client-id

key: cms-client-id

- objectName: cms-client-secret

key: cms-client-secret

- objectName: cms-tenant-id

key: cms-tenant-id

- objectName: kshema-cms-jdbc-password

key: kshema-cms-jdbc-password

- objectName: kshema-cms-jdbc-url

key: kshema-cms-jdbc-url

- objectName: kshema-cms-jdbc-username

key: kshema-cms-jdbc-username

- objectName: omdc-assertingparty-metadata-uri

key: omdc-assertingparty-metadata-uri

- objectName: omdc-graph-url

key: omdc-graph-url

- objectName: omdc-saml-client-secret

key: omdc-saml-client-secret

- objectName: omdc-saml-clientid

key: omdc-saml-clientid

- objectName: omdc-saml-tenatntid

key: omdc-saml-tenatntid

- objectName: spring-datasource-staging-jdbc-url

key: spring-datasource-staging-jdbc-url

- objectName: spring-datasource-staging-password

key: spring-datasource-staging-password

- objectName: spring-datasource-staging-username

key: spring-datasource-staging-username

- objectName: camunda-parent-jdbc-url

key: camunda-parent-jdbc-url

- objectName: token-client-secret

key: token-client-secret

- objectName: token-scope

key: token-scope

- objectName: token-tenantID

key: token-tenantID

parameters:

usePodIdentity: "false"

clientID: c75c93a0-7999-4817-90a7-8ce0bd5efd45

#userAssignedIdentityID: cms-dev-ak-uami

resourceGroup: cms-uat

keyvaultName: cms-uat-kv-aks

tenantId: 225a33a3-a252-427b-a09a-5ae946b50ce4

objects: |

array:

- |

objectName: azure-connection-string

objectType: secret

objectVersion: ""

- |

objectName: azure-storage-sas-token

objectType: secret

objectVersion: ""

- |

objectName: camunda-first-name

objectType: secret

objectVersion: ""

- |

objectName: camunda-jdbc-password

objectType: secret

objectVersion: ""

- |

objectName: camunda-jdbc-url

objectType: secret

objectVersion: ""

- |

objectName: camunda-jdbc-username

objectType: secret

objectVersion: ""

- |

objectName: camunda-user-id

objectType: secret

objectVersion: ""

- |

objectName: camunda-user-password

objectType: secret

objectVersion: ""

- |

objectName: cms-client-id

objectType: secret

objectVersion: ""

- |

objectName: cms-client-secret

objectType: secret

objectVersion: ""

- |

objectName: cms-tenant-id

objectType: secret

objectVersion: ""

- |

objectName: kshema-cms-jdbc-password

objectType: secret

objectVersion: ""

- |

objectName: kshema-cms-jdbc-url

objectType: secret

objectVersion: ""

- |

objectName: kshema-cms-jdbc-username

objectType: secret

objectVersion: ""

- |

objectName: omdc-assertingparty-metadata-uri

objectType: secret

objectVersion: ""

- |

objectName: omdc-graph-url

objectType: secret

objectVersion: ""

- |

objectName: omdc-saml-client-secret

objectType: secret

objectVersion: ""

- |

objectName: omdc-saml-clientid

objectType: secret

objectVersion: ""

- |

objectName: omdc-saml-tenatntid

objectType: secret

objectVersion: ""

- |

objectName: spring-datasource-staging-jdbc-url

objectType: secret

objectVersion: ""

- |

objectName: spring-datasource-staging-password

objectType: secret

objectVersion: ""

- |

objectName: spring-datasource-staging-username

objectType: secret

objectVersion: ""

- |

objectName: camunda-parent-jdbc-url

objectType: secret

objectVersion: ""

- |

objectName: token-client-secret

objectType: secret

objectVersion: ""

- |

objectName: token-scope

objectType: secret

objectVersion: ""

- |

objectName: token-tenantID

objectType: secret

objectVersion: ""

kubectl apply -f secprovidercls\_uat.yaml -n ingress-basic

kubectl get secret -n ingress-basic

**Step 5:** Create deployment.yaml

---

apiVersion: v1

kind: Service

metadata:

  name: kaudit-svc

  namespace: ingress-basic

spec:

  ports:

  - port: 80

    targetPort: 8088

    protocol: TCP

    name: http

  selector:

    app: kaudit

---

apiVersion: apps/v1

kind: Deployment

metadata:

  name: kaudit-deployment

  namespace: ingress-basic

  labels:

    azure.workload.identity/use: "true"

spec:

  replicas: 1

  selector:

    matchLabels:

      app: kaudit

  strategy:

    rollingUpdate:

      maxSurge: 25%

      maxUnavailable: 25%

    type: RollingUpdate

  template:

    metadata:

      labels:

        app: kaudit

    spec:

      serviceAccountName: "workload-identity-sa"

      containers:

      - name: kaudit

        image: cmskshemauatacr.azurecr.io/auditservice:1165

        imagePullPolicy: Always

        resources:

          requests:

            memory: "1024Mi"

            cpu: "0.5"

          limits:

            memory: "2048Mi"

            cpu: "1"

        ports:

        - containerPort: 8088

        env:

        - name: azure\_connection\_string

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: azure-connection-string

        - name: azure\_storage\_sas\_token

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: azure-storage-sas-token

        - name: camunda\_first\_name

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-first-name

        - name: camunda\_jdbc\_password

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-jdbc-password

        - name: camunda\_jdbc\_url

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-jdbc-url

        - name: camunda\_jdbc\_username

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-jdbc-username

        - name: camunda\_user\_id

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-user-id

        - name: camunda\_user\_password

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-user-password

        - name: cms\_client\_id

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: cms-client-id

        - name: cms\_client\_secret

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: cms-client-secret

        - name: cms\_tenant\_id

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: cms-tenant-id

        - name: kshema\_cms\_jdbc\_password

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: kshema-cms-jdbc-password

        - name: kshema\_cms\_jdbc\_url

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: kshema-cms-jdbc-url

        - name: kshema\_cms\_jdbc\_username

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: kshema-cms-jdbc-username

        - name: omdc\_assertingparty\_metadata\_uri

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: omdc-assertingparty-metadata-uri

        - name: omdc\_graph\_url

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: omdc-graph-url

        - name: omdc\_saml\_client\_secret

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: omdc-saml-client-secret

        - name: omdc\_saml\_clientid

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: omdc-saml-clientid

        - name: omdc\_saml\_tenatntid

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: omdc-saml-tenatntid

        - name: spring\_datasource\_staging\_jdbc\_url

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: spring-datasource-staging-jdbc-url

        - name: spring\_datasource\_staging\_password

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: spring-datasource-staging-password

        - name: spring\_datasource\_staging\_username

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: spring-datasource-staging-username

        - name: camunda\_parent\_jdbc\_url

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: camunda-parent-jdbc-url

        - name: token\_client\_secret

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: token-client-secret

        - name: token\_scope

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: token-scope

        - name: token\_tenantID

          valueFrom:

            secretKeyRef:

              name: my-k8s-secret

              key: token-tenantID

        volumeMounts:

        - name: secrets-store

          mountPath: "/mnt/secrets"

      volumes:

      - name: secrets-store

        csi:

          driver: secrets-store.csi.k8s.io

          readOnly: true

          volumeAttributes:

            secretProviderClass: "azure-kvname-wi"

---

apiVersion: autoscaling/v2

kind: HorizontalPodAutoscaler

metadata:

  name: kaudit-dev-hpa

  namespace: ingress-basic

spec:

  scaleTargetRef:

    apiVersion: apps/v1

    kind: Deployment

    name: kaudit-deployment

  minReplicas: 1

  maxReplicas: 2

  metrics:

  - type: Resource

    resource:

      name: cpu

      target:

        type: Utilization

        averageUtilization: 70