

CKAD Ders Dökümanı #07 — Secrets

Dil: Türkçe • CKAD Kategorisi: Application Environment, Configuration and Security (25%) • Standart: Konu anlatımı + komutlar + örnekler

1) Konu Anlatımı

Secret, parola/token/cert gibi hassas verileri saklamak ve Pod'lara güvenli şekilde vermek için kullanılır. CKAD'de en sık: generic secret oluşturma, env/envFrom ile kullanma ve volume mount sorulur.

Pratik not: YAML yazarken stringData kullanmak kolaydır; Kubernetes bunu kaydederken data alanına encode eder.

2) En Sık Karşılaşılan YAML Örnekleri

Opaque Secret (stringData ile)

```
apiVersion: v1
kind: Secret
metadata:
  name: db-secret
type: Opaque
stringData:
  username: admin
  password: "S3cr3t!"
```

Pod'a env olarak verme (tek key)

```
apiVersion: v1
kind: Pod
metadata:
  name: api
spec:
  containers:
    - name: api
      image: busybox:1.36
      env:
        - name: DB_PASSWORD
          valueFrom:
            secretKeyRef:
              name: db-secret
              key: password
      command: ["sh", "-c"]
      args: ["echo $DB_PASSWORD; sleep 3600"]
```

Volume olarak mount etme

```
apiVersion: v1
kind: Pod
metadata:
  name: app
spec:
  volumes:
    - name: secret-vol
      secret:
        secretName: db-secret
  containers:
```

```
- name: app
  image: busybox:1.36
  command: ["sh","-c"]
  args: ["ls -la /etc/secret; cat /etc/secret/password; sleep 3600"]
  volumeMounts:
    - name: secret-vol
      mountPath: /etc/secret
      readOnly: true
```

3) Sık Kullanılan Alanlar (Kısa Açıklamalar)

Alan	Ne işe yarar?
type	Opaque en yaygın; ayrıca kubernetes.io/tls, dockerconfigjson vb.
stringData	Düz metin input (apply için pratik).
data	Base64 encode edilmiş içerik.
env.valueFrom.secretKeyRef	Tek key'i env'e map eder.
envFrom.secretRef	Tüm key'leri env yapar (dikkatli).
volumes.secret	Secret'i dosya olarak mount eder.

4) En Sık Kullanılan Komutlar ve Kullanımları

Aşağıdaki komutlar CKAD pratiklerinde en çok kullanılanlardır.

4.1 Oluşturma (generic / tls / docker)

Sınavda genelde generic kullanılır.

```
kubectl create secret generic db-secret --from-literal=username=admin --from-literal=password='S  
kubectl create secret tls tls-secret --cert=tls.crt --key=tls.key  
kubectl create secret docker-registry regcred --docker-server=REG --docker-username=U --docker-p
```

4.2 Görüntüleme

Secret anahtarlarını görür, YAML'ı incellersiniz.

```
kubectl get secret  
kubectl describe secret db-secret  
kubectl get secret db-secret -o yaml
```

4.3 İçerik decode (debug)

Sınavda bazen decode gerekebilir (base64).

```
kubectl get secret db-secret -o jsonpath='{.data.password}' | base64 -d; echo
```

4.4 Pod içinde doğrulama

Env ve mount edilen dosyayı kontrol.

```
kubectl exec -it api -- env | grep DB_PASSWORD  
kubectl exec -it app -- ls -la /etc/secret
```

5) Troubleshooting Hızlı Rehber

Key not found: secretKeyRef.key yanlış olabilir; kubectl describe secret ile key'leri doğrula. Image pull hatası: docker-registry secret doğru mu ve Pod/SA üzerinde imagePullSecrets var mı? Dosya izinleri: Secret mount readOnly olmalı; uygulama yazmaya çalışıyorsa hata alabilir.

6) CKAD İpuçları

YAML ile yazarken en rahat: stringDataTek key env için: secretKeyRef (adı + key) Decode gerekirse: base64 -d