

# Secret LAB

07 January 2021 12:23

Secret is used to store confidential data

Creation of Secret and inject in Pod as environment variable and volume.

Two ways to create secret

## 1) Imperative

```
# kubectl create secret generic app-secret --from-literal=DB_Host=mysql --from-literal=DB_User=root --from-literal=DB_Password=password
```

```
[root@master ~]# kubectl create secret generic app-secret --from-literal=DB_Host=mysql --from-literal=DB_User=root --from-literal=DB_Password=password
secret/app-secret created
```

```
# kubectl get secret
```

```
[root@master ~]# kubectl get secret
NAME                                TYPE                                DATA  AGE
app-secret                          Opaque                              3      16s
default-token-htjnm                 kubernetes.io/service-account-token 3      3d14h
```

```
# kubectl describe secret app-secret
```

```
[root@master ~]# kubectl describe secret app-secret
Name:          app-secret
Namespace:     default
Labels:        <none>
Annotations:   <none>

Type: Opaque

Data
====
DB_Host:      5 bytes
DB_Password:  8 bytes
DB_User:      4 bytes
```

```
[root@master ~]# cat appsecret.properties
DB_Host=mysql
DB_User=root
DB_Password=password
```

```
# kubectl create secret generic app-secret-file --from-file=appsecret.properties
```

```
[root@master ~]# kubectl create secret generic app-secret-file --from-file=appsecret.properties
secret/app-secret-file created
```

```
# kubectl get secret
```

```
[root@master ~]# kubectl get secret
NAME                                TYPE                                DATA  AGE
app-secret                          Opaque                              3      5m32s
app-secret-file                     Opaque                              1      9s
default-token-htjnm                 kubernetes.io/service-account-token 3      3d14h
```

```
# kubectl describe secret app-secret-file
```

```
[root@master ~]# kubectl describe secret app-secret-file
Name:          app-secret-file
Namespace:     default
Labels:        <none>
Annotations:   <none>

Type: Opaque

Data
====
appsecret.properties: 48 bytes
```

## 2) Declarative

```
# echo -n 'mysql' | base64
# echo -n 'root' | base64
```

```
# echo -n 'password' | base64
```

```
[root@master ~]# echo -n 'mysql' | base64
bXlzcWw=
[root@master ~]#
[root@master ~]# echo -n 'root' | base64
cm9vdA==
[root@master ~]# echo -n 'password' | base64
cGFzc3dvcmQ=
```

\*\*\*\*\* encode your keys and then update values in below file\*\*\*\*\*

```
[root@master ~]# cat secret.yml
apiVersion: v1
kind: Secret
metadata:
  name: my-secret
data:
  DB_Host: bXlzcWw=
  DB_User: cm9vdA==
  DB_Password: cGFzc3dvcmQ=
```

```
# kubectl create -f secret.yml
```

```
[root@master ~]# kubectl create -f secret.yml
secret/my-secret created
```

```
# kubectl get secret
```

```
[root@master ~]# kubectl get secret
NAME                TYPE                      DATA  AGE
app-secret          Opaque                    3      16m
app-secret-file     Opaque                    1      11m
default-token-htjnm kubernetes.io/service-account-token 3      3d14h
my-secret           Opaque                    3       7s
```

```
# kubectl describe secret my-secret
```

```
[root@master ~]# kubectl describe secret my-secret
Name:          my-secret
Namespace:     default
Labels:        <none>
Annotations:   <none>

Type: Opaque

Data
====
DB_Host:      5 bytes
DB_Password:  8 bytes
DB_User:      4 bytes
```

View Secrets values

```
# kubectl get secret my-secret -o yaml
```

```
[root@master ~]# kubectl get secret my-secret -o yaml
apiVersion: v1
data:
  DB_Host: bXlzcWw=
  DB_Password: cGFzc3dvcmQ=
  DB_User: cm9vdA==
kind: Secret
metadata:
  creationTimestamp: "2021-01-06T19:54:21Z"
  managedFields:
  - apiVersion: v1
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:DB_Host: {}
        f:DB_Password: {}
        f:DB_User: {}
        f:type: {}
    manager: kubectl-create
    operation: Update
    time: "2021-01-06T19:54:21Z"
  name: my-secret
  namespace: default
  resourceVersion: "409636"
```

```
#echo -n 'bXlzcWw=' | base64 --decode
#echo -n 'cGFzc3dvcmQ=' | base64 --decode
#echo -n 'cm9vdA==' | base64 --decode
```

```
[root@master ~]# echo -n 'bXlzcWw=' | base64 --decode
mysql[root@master ~]#
[root@master ~]#
[root@master ~]#
[root@master ~]# echo -n 'cGFzc3dvcmQ=' | base64 --decode
password[root@master ~]#
[root@master ~]#
[root@master ~]# echo -n 'cm9vdA==' | base64 --decode
root[root@master ~]#
```

## Inject Secret in Pod as environment variable

```
[root@master ~]# cat pod_secret.yml
apiVersion: v1
kind: Pod
metadata:
  name: pod-secret
spec:
  containers:
    - name: test-container
      image: k8s.gcr.io/busybox
      command: [ "/bin/sh", "-c", "env" ]
      envFrom:
        - secretRef:
            name: my-secret
      restartPolicy: Never
```

```
[root@master ~]# kubectl apply -f pod_secret.yml
pod/pod-secret created
```

```
[root@master ~]# kubectl logs pod-secret
KUBERNETES_SERVICE_PORT=443
KUBERNETES_PORT=tcp://10.96.0.1:443
HOSTNAME=pod-secret
SHLV=1
HOME=/root
DB_Host=mysql
DB_User=root
KUBERNETES_PORT_443_TCP_ADDR=10.96.0.1
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
KUBERNETES_PORT_443_TCP_PORT=443
KUBERNETES_PORT_443_TCP_PROTO=tcp
DB_Password=password
KUBERNETES_SERVICE_PORT_HTTPS=443
KUBERNETES_PORT_443_TCP=tcp://10.96.0.1:443
PWD=/
KUBERNETES_SERVICE_HOST=10.96.0.1
```

## Inject secret in Pod as volume

```
[root@master ~]# cat pod-secret-volume.yml
apiVersion: v1
kind: Pod
metadata:
  name: secret-vol
spec:
  containers:
    - name: test-container
      image: k8s.gcr.io/busybox
      command: [ "/bin/sh", "-c", "ls /etc/config/" ]
      volumeMounts:
        - name: config-secret
          mountPath: /etc/config
  volumes:
    - name: config-secret
      secret:
        secretName: my-secret
      restartPolicy: Never
```

```
# kubectl apply -f pod-secret-volume.yml
```

```
[root@master ~]# kubectl apply -f pod-secret-volume.yml  
pod/secret-vol created
```

```
# kubectl logs secret-vol
```

```
[root@master ~]# kubectl logs secret-vol  
DB_Host  
DB_Password  
DB_User
```