# Kubernetes - Secrets

Secrets can be defined as Kubernetes objects used to store sensitive data such as user name and passwords with encryption.

There are multiple ways of creating secrets in Kubernetes.

* Creating from txt files.
* Creating from yaml file.

### Creating From Text File

In order to create secrets from a text file such as user name and password, we first need to store them in a txt file and use the following command.

$ kubectl create secret generic tomcat-passwd –-from-file = ./username.txt –fromfile = ./.

password.txt

### Creating From Yaml File

apiVersion: v1

kind: Secret

metadata:

name: tomcat-pass

type: Opaque

data:

password: <User Password>

username: <User Name>

### Creating the Secret

$ kubectl create –f Secret.yaml

secrets/tomcat-pass

## Using Secrets

Once we have created the secrets, it can be consumed in a pod or the replication controller as −

* Environment Variable
* Volume

### As Environment Variable

In order to use the secret as environment variable, we will use **env** under the spec section of pod yaml file.

env:

- name: SECRET\_USERNAME

valueFrom:

secretKeyRef:

name: mysecret

key: tomcat-pass

### As Volume

spec:

volumes:

- name: "secretstest"

secret:

secretName: tomcat-pass

containers:

- image: tomcat:7.0

name: awebserver

volumeMounts:

- mountPath: "/tmp/mysec"

name: "secretstest"

### Secret Configuration As Environment Variable

apiVersion: v1

kind: ReplicationController

metadata:

name: appname

spec:

replicas: replica\_count

template:

metadata:

name: appname

spec:

nodeSelector:

resource-group:

containers:

- name: appname

image:

imagePullPolicy: Always

ports:

- containerPort: 3000

env: -----------------------------> 1

- name: ENV

valueFrom:

configMapKeyRef:

name: appname

key: tomcat-secrets

In the above code, under the **env** definition, we are using secrets as environment variable in the replication controller.

### Secrets As Volume Mount

apiVersion: v1

kind: pod

metadata:

name: appname

spec:

metadata:

name: appname

spec:

volumes:

- name: "secretstest"

secret:

secretName: tomcat-pass

containers:

- image: tomcat: 8.0

name: awebserver

volumeMounts:

- mountPath: "/tmp/mysec"

name: "secretstest"