**SECRETS**

1. We use SECRETS object to handle small amount of sensitive data like username and password of a database.
2. The main aim of SECRET object is to prevent the accidental exposure of sensitive data. Hence, it is strongly advised not to use them as a plain text in our manifest files.
3. To manage these confidential data we use SECRETS concept/object in kubernetes.
4. SECRETS are created outside the POD and containers. SECRETS don’t have any clue about which POD or container will use it, once it is created then it can be deployed on any POD and any number of times.
5. SECRETS are stored inside ETCD database on Kubernetes Master.
6. SECRETS should not be more than 1MB.
7. We can consume SECRETS in two ways – Volumes and Environment variables.
8. SECRETS can be sent only to the target nodes.
9. Once you create a SECRET, you can use them in one or more PODs.
10. Each SECRET is stored in ***tempfs*** volume, so that we can restrict access to the application in the node. (The advantage of *tempfs*, an application which runs in another container can’t access this one)

We can create SECRETS in two ways:

1. Using kubectl command
2. Manually using manifest file.

![image](<https://user-images.githubusercontent.com/26220908/158866844-b73560c2-bf36-451d-bef0-8f0d97f35684.png>)

We can consume SECRETS as Volumes and Environment variables.

* How to create a SECRET.

This is a *syntax* to create a SECRET.

![image](<https://user-images.githubusercontent.com/26220908/158866932-05a001e6-7b9e-4ff0-8a3c-ee2850ea6102.png>)

**CREATING SECRETS USING A *--FROM-FILE* OPTION**

Here, we are passing ***admin*** username to username.txt file and ***password123*** password to a password.txt file.

![image](<https://user-images.githubusercontent.com/26220908/158867020-ebcf788c-6583-40c3-b273-13f747a97e83.png>)

Now, run the below command to create a SECRET. Here, we are passing the files ***username.txt*** and ***password.txt*** which we created in the above step and using ***–from-file*** option to create a ***db-user-pass*** SECRET.

![image](<https://user-images.githubusercontent.com/26220908/158867074-55c2ac31-334f-4b89-8b28-afb06ee50ad6.png>)

We can see that ***db-user-pass*** SECRET is created now.

![image](<https://user-images.githubusercontent.com/26220908/158867116-1830cabe-e1d9-40d6-981e-e15eb138ea55.png>)

If you want to know the details of the SECRET, then you can use the describe command to see the details of ***db-user-pass*** SECRET.

**NOTE**: It won’t display the SECRETS in plain text which is we want.

![image](<https://user-images.githubusercontent.com/26220908/158867156-4622d13c-6b8e-46c2-a04d-737c660918f2.png>)

You can see all the commands at once in the below screenshot.

![image](<https://user-images.githubusercontent.com/26220908/158867184-47c30879-bdb4-42ab-846c-204647f08273.png>)

***CREATEING SECRETS MANUALLY***

To create SECRETS manually, first we need to convert the credentials to base64 format as shown below.

![image](<https://user-images.githubusercontent.com/26220908/158867230-cff6ae55-6a9b-43d3-a379-fd3bc054b1fa.png>)

Then, use those credentials inside the SECRET yaml file like this.

![image](<https://user-images.githubusercontent.com/26220908/158867262-fbbfad4e-c9f7-4047-9839-ea66f7b2212a.png>)

Now, execute the below command to create a SECRET.

![image](<https://user-images.githubusercontent.com/26220908/158867294-b2f9a901-6a56-43e7-b6bb-2b14b064c4ab.png>)

Once, the SECRET is created, we can see the details of the SECRETS by executing the below command. We can see that the credentials are used inside the ***data*** field.

![image](<https://user-images.githubusercontent.com/26220908/158867320-b2c78f10-dd05-47e4-8dec-3a8754db81df.png>)

You can also ***decode*** the base64 credentials if you want as shown below.

![image](<https://user-images.githubusercontent.com/26220908/158867349-465a31b9-41e0-48f8-a57f-466f9ab85850.png>)

Now, it’s time to consume these SECRETS in PODS. We can consume SECRETS inside POD in two ways.

1. Using Volumes
2. Using Environment Variables.

![image](<https://user-images.githubusercontent.com/26220908/158867406-249fb97e-9407-4b5a-97e6-fd40329986fa.png>)

**1st Method: Consume SECRETS using Volumes:**

Here, we can see that we are using SECRET ***mysecret*** under ***volumes*** section.

**NOTE**: *We created* ***mysectet*** *in previous steps*

![image](<https://user-images.githubusercontent.com/26220908/158867435-6f4a57ac-64a5-4d0e-a32e-89f30dd38401.png>)

After updating the SECRET details inside POD, we can execute the below command to create the POD, which will internally use the SECRET *mysecret*.

![image](<https://user-images.githubusercontent.com/26220908/158867463-158bda99-85b7-449c-8a2f-8f2c285701f8.png>)

After the POD is created, we can see that the POD is in Running state which is using SECRET mysecret.

![image](<https://user-images.githubusercontent.com/26220908/158867502-b266f64d-11ea-48f3-8e0a-94e68186d12a.png>)

**2nd Method: Consume SECRETS using Environment Variables:**

Let’s create SECRETS using ***--from-literal*** option as shown below.

![image](<https://user-images.githubusercontent.com/26220908/158867560-ba37b4f9-8bac-4560-84ff-74b88a30040b.png>)

We can see the details of the SECRET by using the below describe command.

![image](<https://user-images.githubusercontent.com/26220908/158867604-acdb0eed-e7f7-4cc6-9ba3-68e6ce8e31b3.png>)

Now, let’s use the above SECRET in our Environment Variable of POD definition file.

We can create a POD using the below definition file by running the below command

“***kubectl create –f pod.yaml***”

![image](<https://user-images.githubusercontent.com/26220908/158867635-bd9eb2e2-6b98-46df-b76b-fb4ce18f49fd.png>)

Here, we can see that both POD and SECRET is in running state after creating the POD.

![image](<https://user-images.githubusercontent.com/26220908/158867663-3b3b25e4-7100-4775-affc-357bc24096ee.png>)

**NOTE**: In this document we shown how to create SECRETS in different ways and how to use SECRETS in our PODs definition file using Volumes and Environment Variables.