

Multimode architecture

Today we shall see about multimode cluster in K8s and also get to know the environment variables.

For that, we launch a pod of mysql image.

```
C:\Users\lenovo>kubectl run mydb --image=mysql:5.7
pod/mydb created
```

After the pod gets created, check its status

```
C:\Users\lenovo>kubectl get pods
NAME          READY   STATUS             RESTARTS   AGE
mydb          0/1     ContainerCreating   0           22s
```

We notice that the pod won't get launched and crash. Let us look at the details and reason behind this failure.

```
C:\Users\lenovo>kubectl get pods
NAME          READY   STATUS              RESTARTS   AGE
mydb          0/1     CrashLoopBackOff    1           2m25s
```

Why did this happen? For that, we shall check the logs.

Here we find that some required passwords are not initialized. It means that we need to set up user name and password as environment variables. This is not a problem related to kubernetes.

```
C:\Users\lenovo>kubectl logs mydb
2021-01-19 16:19:01+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.33-1debian10 started.
2021-01-19 16:19:01+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2021-01-19 16:19:01+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.33-1debian10 started.
2021-01-19 16:19:01+00:00 [ERROR] [Entrypoint]: Database is uninitialized and password option is not specified
You need to specify one of MYSQL_ROOT_PASSWORD, MYSQL_ALLOW_EMPTY_PASSWORD and MYSQL_RANDOM_ROOT_PASSWORD
```

Let us run one of the available pods.

```
C:\Users\lenovo>kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
mydb          0/1     CrashLoopBackOff   6          12m
rc1-jld2v     1/1     Running    3          4d5h
rc1-jwbvs     1/1     Running    3          4d5h
rc1-m985v     1/1     Running    3          4d5h
rc1-q8t2f     1/1     Running    3          4d5h
skpod1        1/1     Running    3          4d5h
skpod2        1/1     Running    1          4d
skpod3        1/1     Running    1          4d
```

We shall get into the pod using `kubectl exec -it skpod1 bash`

```
C:\Users\lenovo>kubectl exec -it skpod1 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec
c [POD] -- [COMMAND] instead.
[root@skpod1 /]#
```

Linux command is running. It means that we have successfully entered into the container.

```
[root@skpod1 /]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.17.0.5 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:ac:11:00:05 txqueuelen 0 (Ethernet)
    RX packets 3 bytes 126 (126.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@skpod1 /]#
```

We shall get into the file and set some variable.

```
[root@skpod1 /]# vi /root/.bashrc
```

```
@skpod1:/  
# .bashrc  
  
# User specific aliases and functions  
  
alias rm='rm -i'  
alias cp='cp -i'  
alias mv='mv -i'  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi  
  
x=18  
~  
~
```

```
[root@skpod1 /]# vi /root/.bashrc  
[root@skpod1 /]# exit  
exit  
  
C:\Users\lenovo>kubectl exec -it skpod1 bash  
kubectl exec [POD] [COMMAND] is DEPRECATED and  
c [POD] -- [COMMAND] instead.
```

```
[root@skpod1 /]# echo $x  
18
```

But if this or crashes or gets deleted, we will lose this variable.

To make the variable permanent, we use the concept of environment variable.

```
C:\Users\lenovo>kubect1 run skpod4 --image=vimal13/apache-webserver-php --env=x=10
pod/skpod4 created

C:\Users\lenovo>
```

Now lets start:

We need the following environment variable for mysql image.

itself respects (especially variables like `MYSQL_HOST` , which is k

`MYSQL_ROOT_PASSWORD`

This variable is mandatory and specifies the password that will
set to `my-secret-pw` .

`MYSQL_DATABASE`

This variable is optional and allows you to specify the name of
(see below) then that user will be granted superuser access (cc

`MYSQL_USER` , `MYSQL_PASSWORD`

These variables are optional, used in conjunction to create a n

To begin, we delete all the existing pods.

```
C:\Users\lenovo>kubect1 delete all --all
pod "mydb" deleted
pod "rc1-jld2v" deleted
pod "rc1-jwbvs" deleted
pod "rc1-m985v" deleted
pod "rc1-q8t2f" deleted
pod "skpod1" deleted
pod "skpod2" deleted
pod "skpod3" deleted
pod "skpod4" deleted
replicationcontroller "rc1" deleted
service "kubernetes" deleted
service "lb" deleted
service "rc1" deleted
```

Check if all the pods are successfully deleted.

```
C:\Users\lenovo>kubectl get pods
No resources found in default namespace.
```

For launching the pod using mysql image, we use the following command:

```
Kubectl run mydb --image=mysql:5.7 --env=MYSQL_ROOT_PASSWORD=shreya --
env=MYSQL_DATABASE=wpdb --env=MYSQL_USER=shreya --env=MYSQL_PASSWORD=shreya
```

```
C:\Users\lenovo>kubectl run mydb --image=mysql:5.7 --env=MYSQL_ROOT_PASSWORD=shreya --env=MYSQL_DATABASE=wpdb --env=MYSQL_USER=shreya --env=MYSQL_PASSWORD=shreya
pod/mydb created

C:\Users\lenovo>kubectl get pods
NAME      READY   STATUS    RESTARTS   AGE
mydb      1/1     Running   0           35s
```

Now, launch another pod using wordpress which will act as the frontend.

```
C:\Users\lenovo>kubectl run mywp1 --image=wordpress:5.1.1-php7.3-apache
pod/mywp1 created
```

```
C:\Users\lenovo>kubectl get pods
NAME      READY   STATUS    RESTARTS   AGE
mydb      1/1     Running   0           8m22s
mywp1     1/1     Running   0           6m35s
```

Now, expose the pods and get the IP and port number of wordpress.

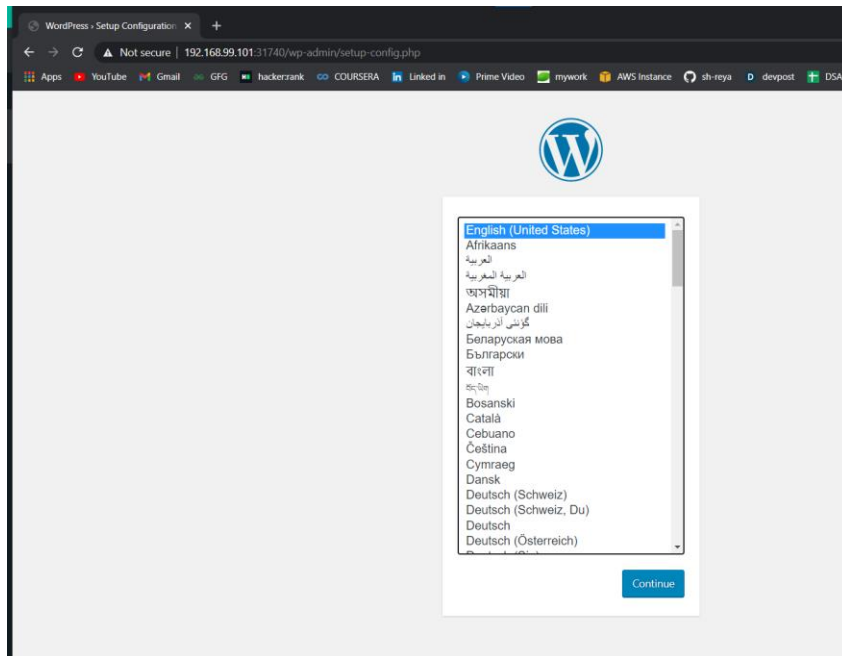
```
C:\Users\lenovo>kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP    9m56s

C:\Users\lenovo>kubectl expose pod mywp1 --type=NodePort --port=80
service/mywp1 exposed

C:\Users\lenovo>kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP    10m
mywp1        NodePort    10.103.91.103 <none>        80:31740/TCP 15s

C:\Users\lenovo>
```

Use the IP and the port number to get into the homepage of wordpress.



Give the details of the database and you will get successfully connected to the wordpress account.

A screenshot of the WordPress database connection details form. At the top is the WordPress logo. Below it, a message states: "Below you should enter your database connection details. If you're not sure about these, contact your host." The form contains five input fields, each with a label and a description: "Database Name" with value "wpdb" (description: "The name of the database you want to use with WordPress."), "Username" with value "shreya" (description: "Your database username."), "Password" with value "shreya" (description: "Your database password."), "Database Host" with value "10.103.91.103" (description: "You should be able to get this info from your web host, if localhost doesn't work."), and "Table Prefix" with value "wp_" (description: "If you want to run multiple WordPress installations in a single database, change this."). A blue "Submit" button is located at the bottom left of the form.