**SSH to your AWS Workstation**

**ssh devops@<public-ip-addr**> of your Workstation  
Password is : **Dev0p$!!/**

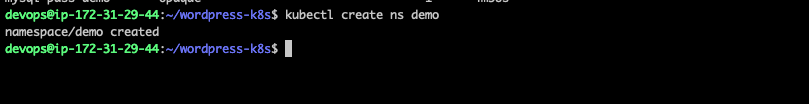
**Replace <your-name> with your name throughout the lab.**

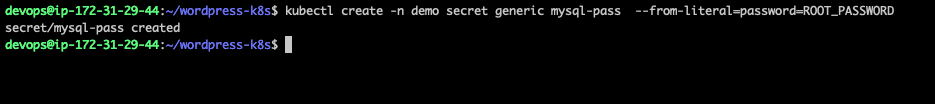
## **Configure a MySQL database**

1. Start with creating a secret password for the MySQL root user:

|  |
| --- |
| $ mkdir /home/devops/wordpress-k8s  $ cd /home/devops/wordpress-k8s  $ kubectl create ns <your-name>  $ kubectl create -n <your-name> secret generic mysql-pass --from-literal=password=ROOT\_PASSWORD |

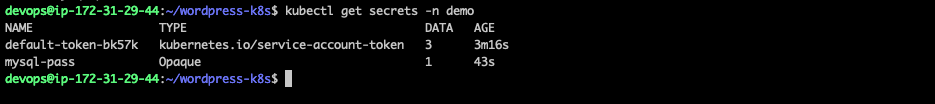
Update **ROOT\_PASSWORD** with the password you want to set for the Mysql POD





2. You can check if the password was properly configured by running

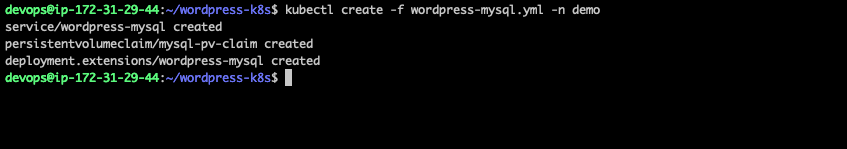
|  |
| --- |
| $ kubectl get secrets -n <your-name> |



Secrets in K8s are hidden and cannot be displayed. This means there’s no risk of exposing them in config files in public repositories.

3. Deploy the MySql application.

|  |
| --- |
| $ curl -k -f https://pastebin.com/raw/hav5k7Em > wordpress-mysql.yml  $ kubectl create -f wordpress-mysql.yml -n <your-name> |



4. Check the Status of the POD.

|  |
| --- |
| $ kubectl get po -n <your-name> |



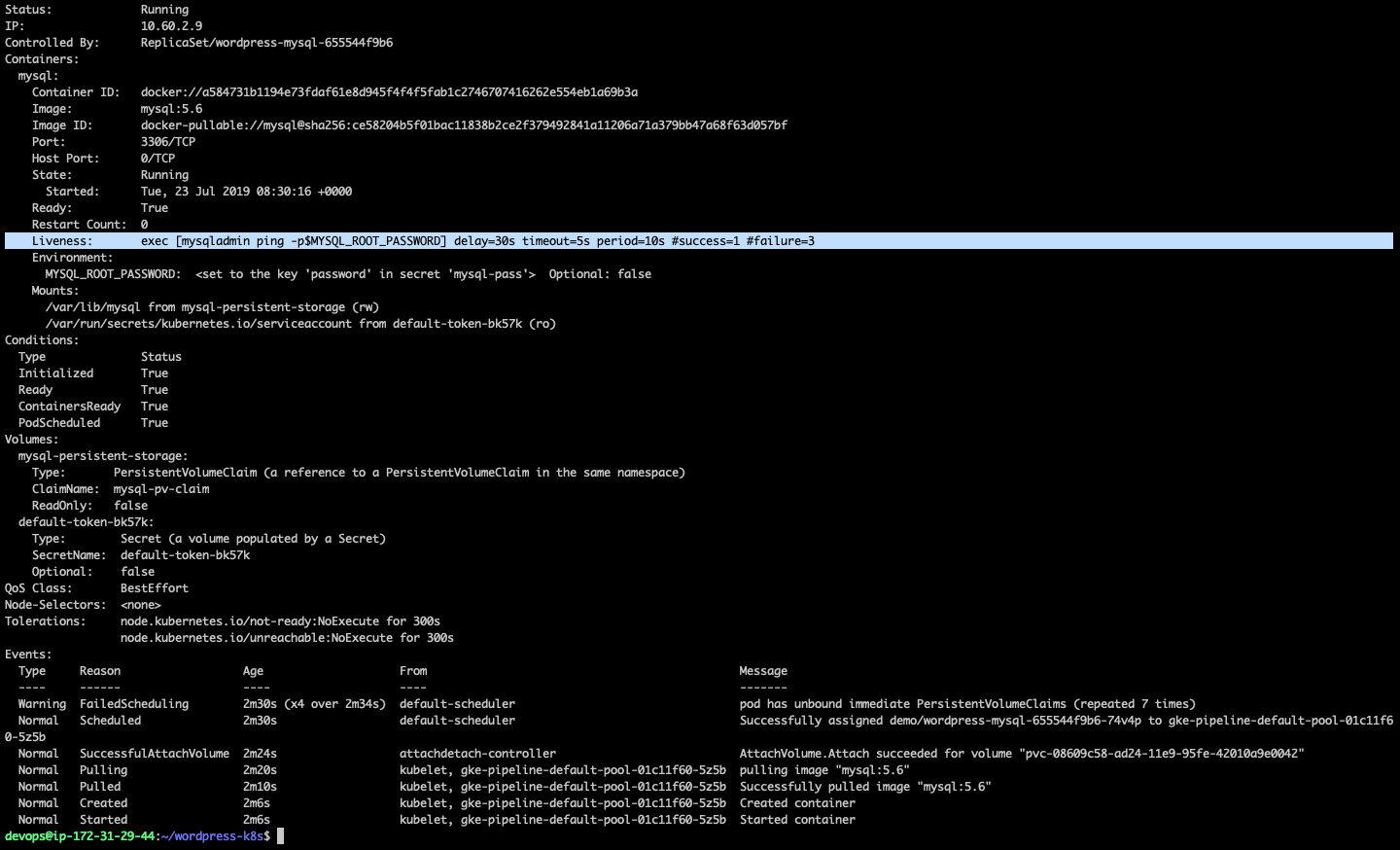
5. Copy the POD name of the MySql POD and run the below command.

|  |
| --- |
| $ kubectl describe po <POD-NAME> -n <your-name> |

Ex.

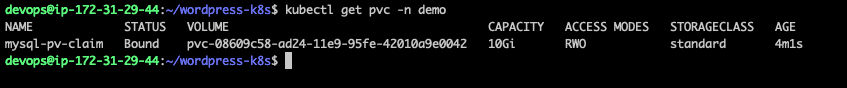


POD is created after the Heath Check “Liveness Probe” Passed. As shown below.



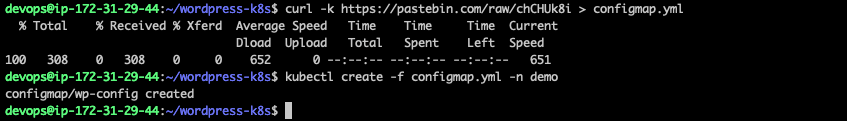
6. Check the status of the Persistent Volume Claims

|  |
| --- |
| $ kubectl get pvc -n <your-name> |



7. Now, create ConfigMaps for the wordpress deployment.

|  |
| --- |
| $ curl -k <https://pastebin.com/raw/chCHUk8i> > configmap.yml  $ kubectl create -f configmap.yml -n <your-name> |



8. Check all the components created in your namespace.

|  |
| --- |
| $ kubectl get all -n <your-name> |

