

Домашнее задание для к уроку 4 - Хранение данных и ресурсы

Напишите deployment для запуска сервера базы данных PostgreSQL.

Приложение должно запускаться из образа postgres:10.13

Должен быть описан порт:

- 5432 TCP

В деплоimente должна быть одна реплика, при этом при обновлении образа

НЕ ДОЛЖНО одновременно работать несколько реплик.

(то есть сначала должна удаляться старая реплика и только после этого подниматься новая).

Это можно сделать или с помощью maxSurge/maxUnavailable или указав стратегию деплоя Recreate.

В базе данных при запуске должен автоматически создаваться пользователь testuser с паролем testpassword. А также база testdatabase.

Для этого нужно указать переменные окружения POSTGRES_PASSWORD, POSTGRES_USER, POSTGRES_DB в деплоimente.

При этом значение переменной POSTGRES_PASSWORD должно браться из секрета.

Так же нужно указать переменную PGDATA со значением /var/lib/postgresql/data/pgdata

См. документацию к образу https://hub.docker.com/_/postgres раздел PGDATA

База данных должна хранить данные в PVC с размером диска в 10Gi, замонтированном в pod по пути /var/lib/postgresql/data

Проверка

Для проверки работоспособности базы данных:

1. Узнайте IP пода postgresql

```
kubectl get pod -o wide
```

1. Запустите рядом тестовый под

```
kubectl run -t -i --rm --image postgres:10.13 test bash
```

1. Внутри тестового пода выполните команду для подключения к БД

```
psql -h <postgresql pod IP из п.1> -U testuser testdatabase
```

Введите пароль - testpassword

1. Все в том же тестовом поде, после подключения к экземпляру БД выполните команду для создания таблицы

```
CREATE TABLE testtable (testcolumn VARCHAR (50) );
```

1. Проверьте что таблица создалась. Для этого все в том же тестовом поде выполните команду

```
\dt
```

1. Выйдите из тестового пода. Попробуйте удалить под с postgresql.

2. После его пересоздания повторите все с п.1, кроме п.4

Проверьте что созданная ранее таблица никуда не делась.

Cat deployments_postgres2.yaml

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: deployment-postgres2
spec:
  replicas: 1
  selector:
    matchLabels:
      app: postgres2
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: postgres2
    spec:
      containers:
      - image: postgres:12.4-alpine
        name: postgres2
        ports:
          - containerPort: 5432
            protocol: TCP
        env:
          - name: POSTGRES_DB
            value: testdatabase
          - name: POSTGRES_USER
            value: testuser
          - name: POSTGRES_PASSWORD
            valueFrom:
              secretKeyRef:
                name: postgres
                key: POSTGRES_PASSWORD
          - name: PGDATA
            value: "/var/lib/postgresql/data/pgdata"
        volumeMounts:
          - mountPath: "/var/lib/postgresql/data"
            name: ps-pgdata
      volumes:
      - name: ps-pgdata
        persistentVolumeClaim:
          claimName: ps-pgdata
```

```
cat pvc.yaml
```

```
---
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: ps-pgdata
  annotations:
    volume.beta.kubernetes.io/mount-options:
"dir_mode=0777,file_mode=0777,uid=1000,gid=1000"
spec:
  accessModes:
    - ReadWriteMany
  resources:
    requests:
      storage: 10Gi
  storageClassName: csi-ceph-hdd-dp1
```

Создадим секрет:

```
kubectl create secret generic postgres --from-
literal=POSTGRES_PASSWORD="testpassword"
```

```
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k get secret postgres -oyaml
apiVersion: v1
data:
  POSTGRES_PASSWORD: dGVzdHBhc3N3b3Jk
kind: Secret
metadata:
  creationTimestamp: "2021-07-08T14:38:46Z"
  name: postgres
  namespace: default
  resourceVersion: "139977"
  selfLink: /api/v1/namespaces/default/secrets/postgres
  uid: 5e780ce1-b104-4dd3-868e-35510c41b6ab
type: Opaque
```

Результат:
Pod стартует, вход по секрету работает, БД не стирается при перезапуске.

```
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k get pod -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE                                NOMINATED NODE   READINESS GATES
deployment-postgres2-7d54c74f8-w4wjn 1/1     Running   0           75s   10.100.180.181  kubernetes-cluster-5268-default-group-0  <none>           <none>
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k run -it --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 10.100.180.181 -U testuser testdatabase
Password for user testuser:
psql (10.13 (Debian 10.13-1.pgdg90+1), server 12.4)
WARNING: psql major version 10, server major version 12.
         Some psql features might not work.
Type "help" for help.

testdatabase=# CREATE TABLE testtable (testcolumn VARCHAR (50) );
CREATE TABLE
testdatabase=# \dt
          List of relations
+-----+
| Schema | Name      | Type  | Owner  |
+-----+
| public | testtable | table | testuser
(1 row)

testdatabase=#
```

```
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k delete pod deployment-postgres2-7d54c74f8-w4wjn
pod "deployment-postgres2-7d54c74f8-w4wjn" deleted
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k get po
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE                                NOMINATED NODE   READINESS GATES
deployment-postgres2-7d54c74f8-fhnll 1/1     Running   0           12s   10.100.180.183  kubernetes-cluster-5268-default-group-0  <none>           <none>
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k run -it --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# exit
Session ended, resume using 'kubectl attach test -c test -i -t' command when the pod is running
pod "test" deleted
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k get pod -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE                                NOMINATED NODE   READINESS GATES
deployment-postgres2-7d54c74f8-fhnll 1/1     Running   0           97s   10.100.180.183  kubernetes-cluster-5268-default-group-0  <none>           <none>
lamo@lamo-K501LB:~/nout/GEEKBRAINS/Microservice_architecture_and_containerization/HW4$ k run -it --rm --image postgres:10.13 test bash
If you don't see a command prompt, try pressing enter.
root@test:/# psql -h 10.100.180.183 -U testuser testdatabase
Password for user testuser:
psql (10.13 (Debian 10.13-1.pgdg90+1), server 12.4)
WARNING: psql major version 10, server major version 12.
         Some psql features might not work.
Type "help" for help.

testdatabase=# \dt
          List of relations
+-----+
| Schema | Name      | Type  | Owner  |
+-----+
| public | testtable | table | testuser
(1 row)

testdatabase=#
```