Name: Amir Mamdouh

Kubernetes Wiki + MySQL Deployment

This project sets up a MySQL StatefulSet and a Wiki application Deployment on Kubernetes. It includes persistent storage, service exposure, configuration, secrets management, and backup/restore functionality.

- StatefulSet for MySQL
- Deployment for Wiki.js
- Persistent Volumes (PV) and Persistent Volume Claims (PVC)
- Internal Services for communication
- ConfigMap and Secret for DB credentials
- Backup and restore Jobs using `mysqldump`

Components

1. **MySQL (StatefulSet)**

- **YAML:** `mysql.yml`
- **Persistent Volume Claim:** `mysql-pvc.yml`
- **Persistent Volume:** `mysql-pv.yml`
- **Service:** `mysql-service.yml`

MySQL is deployed as a **StatefulSet** to ensure stable network identity and persistent storage for database data.

2. **Wiki (Deployment)**

- **YAML:** `wiki.yml`
- **Service:** `wiki-service.yml`
- **ConfigMap:** `wiki-configmap.yml`
- **Secrets:** `wiki-secrets.yml`

The Wiki app connects to MySQL using environment variables defined in the ConfigMap and Secrets.

3. **Storage**

- **MySQL Data Storage**
  - PVC: `mysql-pvc.yml`
  - PV: `mysql-pv.yml`
- **Backup Storage**
  - PVC: `mysql-backup-pvc.yml`
  - PV: `mysql-backup-pv.yml`

4. **Backup & Restore Jobs**

- **Backup Job:** `mysql-backupjob.yml`
  - Periodically backs up MySQL data to the backup volume.

- **Restore Job:** `mysql-restorejob.yml`
  - Used to restore MySQL data from the backup volume.

> All persistent volumes are configured using hostPath (for local development) or suitable alternatives for cloud-based clusters.

Usage

1. Deploy Storage

```
kubectl apply -f mysql-pv.yml
kubectl apply -f mysql-pvc.yml

kubectl apply -f mysql-backup-pv.yml
kubectl apply -f mysql-backup-pvc.yml
```

2. Deploy MySQL

```
kubectl apply -f mysql-service.yml
kubectl apply -f mysql.yml
```

3. Deploy Wiki

```
kubectl apply -f wiki-configmap.yml
kubectl apply -f wiki-secrets.yml
kubectl apply -f wiki-service.yml
kubectl apply -f wiki.yml
```

4. Run Backup Job

```
kubectl apply -f mysql-backupjob.yml
```

5. Run Restore Job

```
kubectl apply -f mysql-restorejob.yml
```

Directory Structure

```
task3/
│
├── mysql.yml                # MySQL StatefulSet
├── mysql-service.yml        # MySQL Service
├── mysql-pvc.yml            # MySQL PersistentVolumeClaim
├── mysql-pv.yml             # MySQL PersistentVolume
│
├── wiki.yml                 # Wiki Deployment
├── wiki-service.yml         # Wiki Service
├── wiki-configmap.yml       # Wiki Configuration
├── wiki-secrets.yml         # Wiki Secrets (e.g., DB password)
│
├── mysql-backupjob.yml      # CronJob or Job for backing up MySQL
├── mysql-restorejob.yml     # Job to restore MySQL from backup
├── mysql-backup-pv.yml      # Backup Volume
├── mysql-backup-pvc.yml     # Backup VolumeClaim
│
├── task3-doc.odt            # Project documentation (ODT)
└── task3-doc.pdf            # Project documentation (PDF)
```

Kubernetes Manifests

`mysql.yml` – MySQL StatefulSet

```yaml
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: mysql
spec:
  selector:
    matchLabels:
      app: mysql
  serviceName: mysql
  replicas: 1
  template:
    metadata:
      labels:
        app: mysql
    spec:
      containers:
        - name: mysql
          image: mysql:5.7
          env:
            - name: MYSQL_ROOT_PASSWORD
              value: rootpassword
          ports:
            - containerPort: 3306
          volumeMounts:
            - name: mysql-storage
```

```
              mountPath: /var/lib/mysql
      volumes:
        - name: mysql-storage
          persistentVolumeClaim:
            claimName: mysql-pvc
```

`mysql-service.yml` – MySQL Headless Service

```
apiVersion: v1
kind: Service
metadata:
  name: mysql
spec:
  ports:
    - port: 3306
  selector:
    app: mysql
  clusterIP: None
```

`mysql-pv.yml` – MySQL PersistentVolume

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: mysql-pv
spec:
  capacity:
    storage: 1Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data/mysql"
```

`mysql-pvc.yml` – MySQL PersistentVolumeClaim

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-pvc
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
```

`wiki.yml` – Wiki.js Deployment

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: wiki
spec:
  replicas: 1
  selector:
    matchLabels:
      app: wiki
  template:
    metadata:
      labels:
        app: wiki
    spec:
      containers:
        - name: wiki
          image: ghcr.io/linuxserver/wikijs
          ports:
            - containerPort: 3000
          envFrom:
            - configMapRef:
                name: wiki-config
            - secretRef:
                name: wiki-secrets
```

`wiki-service.yml` – Wiki.js ClusterIP Service

```
apiVersion: v1
kind: Service
metadata:
  name: wiki
spec:
  selector:
    app: wiki
  ports:
    - port: 80
      targetPort: 3000
  type: ClusterIP
```

`wiki-configmap.yml` – ConfigMap for DB Settings

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: wiki-config
data:
  DB_TYPE: mysql
  DB_HOST: mysql
  DB_PORT: "3306"
  DB_USER: root
```

`wiki-secrets.yml` – Secret for DB Password

```
apiVersion: v1
kind: Secret
metadata:
  name: wiki-secrets
type: Opaque
data:
  DB_PASS: cm9vdHBhc3N3b3Jk
```

> The value `cm9vdHBhc3N3b3Jk` is base64 for `rootpassword`.

`mysql-backup-pv.yml` – Backup PersistentVolume

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: mysql-backup-pv
spec:
  capacity:
    storage: 1Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data/mysql-backup"
```

`mysql-backup-pvc.yml` – Backup PersistentVolumeClaim

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysql-backup-pvc
```

```
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
```

`mysql-backupjob.yml` – Job to Backup MySQL

```
apiVersion: batch/v1
kind: Job
metadata:
  name: mysql-backup
spec:
  template:
    spec:
      containers:
        - name: backup
          image: mysql:5.7
          command: ["/bin/sh", "-c"]
          args: ["mysqldump -h mysql -uroot -prootpassword --all-databases > /backup/all.sql"]
          volumeMounts:
            - name: backup-storage
              mountPath: /backup
      restartPolicy: OnFailure
      volumes:
        - name: backup-storage
          persistentVolumeClaim:
            claimName: mysql-backup-pvc
```

`mysql-restorejob.yml` – Job to Restore MySQL

```
apiVersion: batch/v1
kind: Job
metadata:
  name: mysql-restore
spec:
  template:
    spec:
      containers:
        - name: restore
          image: mysql:5.7
          command: ["/bin/sh", "-c"]
          args: ["mysql -h mysql -uroot -prootpassword < /backup/all.sql"]
          volumeMounts:
            - name: backup-storage
```

```
              mountPath: /backup
      restartPolicy: OnFailure
      volumes:
        - name: backup-storage
          persistentVolumeClaim:
            claimName: mysql-backup-pvc
```

How to Deploy

1. **Create volumes**:

   ```
   kubectl apply -f mysql-pv.yml
   kubectl apply -f mysql-pvc.yml
   kubectl apply -f mysql-backup-pv.yml
   kubectl apply -f mysql-backup-pvc.yml
   ```

2. **Deploy MySQL**:

   ```
   kubectl apply -f mysql.yml
   kubectl apply -f mysql-service.yml
   ```

3. **Deploy Wiki.js**:

   ```
   kubectl apply -f wiki-configmap.yml
   kubectl apply -f wiki-secrets.yml
   kubectl apply -f wiki.yml
   kubectl apply -f wiki-service.yml
   ```

4. **Run Backup Job (optional)**:

   ```
   kubectl apply -f mysql-backupjob.yml
   ```

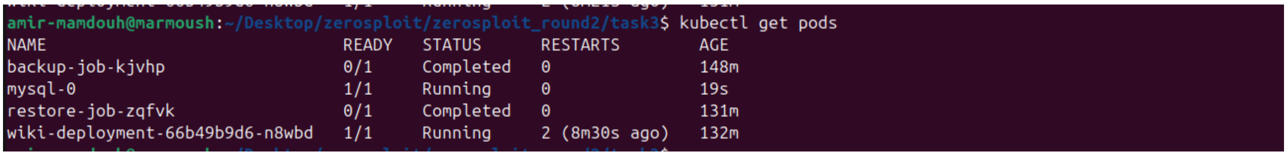
5. **Run Restore Job (optional)**:

   ```
   kubectl apply -f mysql-restorejob.yml
   ```

Test taking backups and restoring

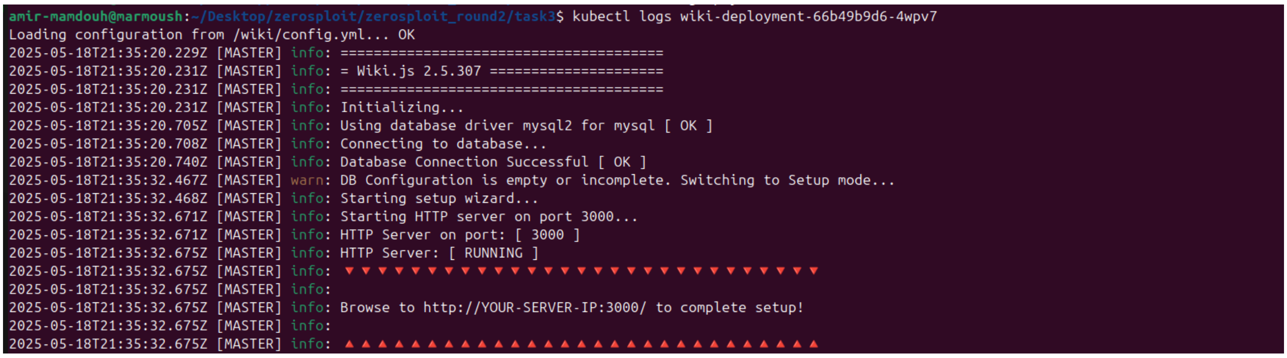
1. **get pods**

- amir pod is for debugging…



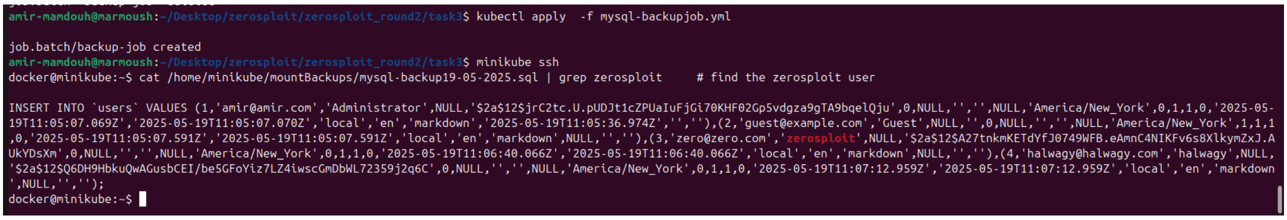
```
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl get pods
NAME                              READY   STATUS      RESTARTS       AGE
backup-job-kjvhp                  0/1     Completed   0              148m
mysql-0                           1/1     Running     0              19s
restore-job-zqfvk                 0/1     Completed   0              131m
wiki-deployment-66b49b9d6-n8wbd   1/1     Running     2 (8m30s ago)  132m
```

2. **open wiki URL on minikube_IP:30001 → pod_ID:3000**



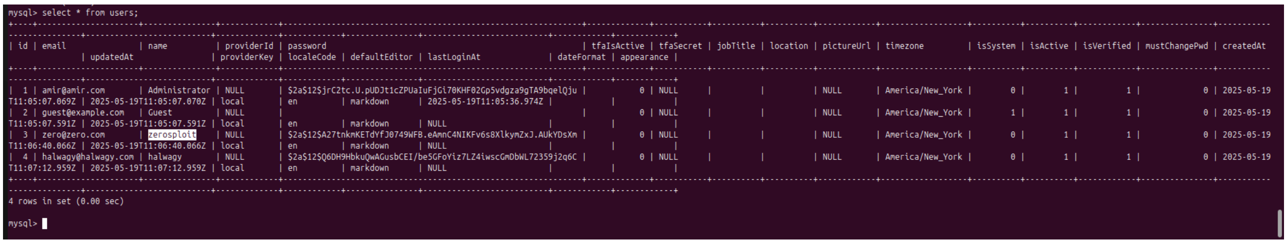
```
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl logs wiki-deployment-66b49b9d6-4wpv7
Loading configuration from /wiki/config.yml... OK
2025-05-18T21:35:20.229Z [MASTER] info: =======================================
2025-05-18T21:35:20.231Z [MASTER] info: = Wiki.js 2.5.307 =====================
2025-05-18T21:35:20.231Z [MASTER] info: =======================================
2025-05-18T21:35:20.231Z [MASTER] info: Initializing...
2025-05-18T21:35:20.705Z [MASTER] info: Using database driver mysql2 for mysql [ OK ]
2025-05-18T21:35:20.708Z [MASTER] info: Connecting to database...
2025-05-18T21:35:20.740Z [MASTER] info: Database Connection Successful [ OK ]
2025-05-18T21:35:32.467Z [MASTER] warn: DB Configuration is empty or incomplete. Switching to Setup mode...
2025-05-18T21:35:32.468Z [MASTER] info: Starting setup wizard...
2025-05-18T21:35:32.671Z [MASTER] info: Starting HTTP server on port 3000...
2025-05-18T21:35:32.671Z [MASTER] info: HTTP Server on port: [ 3000 ]
2025-05-18T21:35:32.675Z [MASTER] info: HTTP Server: [ RUNNING ]
2025-05-18T21:35:32.675Z [MASTER] info: ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼ ▼
2025-05-18T21:35:32.675Z [MASTER] info:
2025-05-18T21:35:32.675Z [MASTER] info: Browse to http://YOUR-SERVER-IP:3000/ to complete setup!
2025-05-18T21:35:32.675Z [MASTER] info:
2025-05-18T21:35:32.675Z [MASTER] info: ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲ ▲
```

3. **Apply the mysql-backjob.yml job and check the backup we took right now**



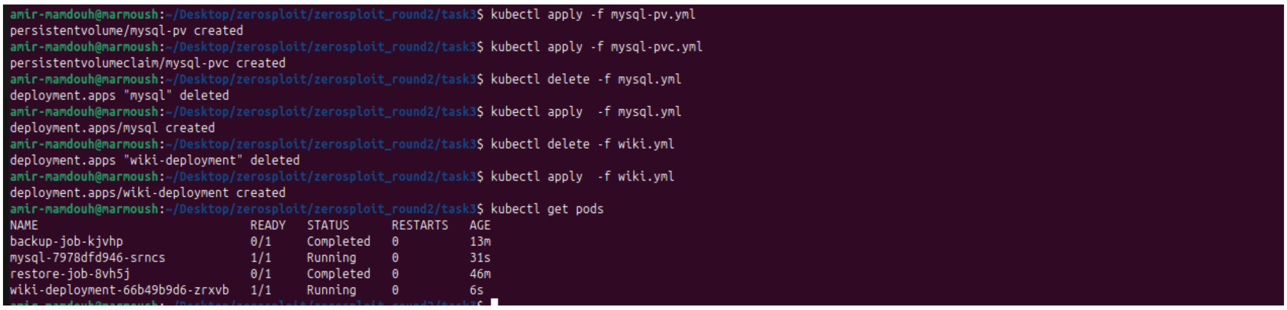
```
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl apply  -f mysql-backupjob.yml

job.batch/backup-job created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ minikube ssh
docker@minikube:~$ cat /home/minikube/mountBackups/mysql-backup19-05-2025.sql | grep zerosploit     # find the zerosploit user

INSERT INTO `users` VALUES (1,'amir@amir.com','Administrator',NULL,'$2a$12$jrC2tc.U.pUDJt1cZPUaIuFjGi70KHF02Gp5vdgza9gTA9bqelQju',0,NULL,'','',NULL,'America/New_York',0,1,1,0,'2025-05-
19T11:05:07.069Z','2025-05-19T11:05:07.070Z','local','en','markdown','2025-05-19T11:05:36.974Z','',''),(2,'guest@example.com','Guest',NULL,'',0,NULL,'','',NULL,'America/New_York',1,1,1
,0,'2025-05-19T11:05:07.591Z','2025-05-19T11:05:07.591Z','local','en','markdown',NULL,'',''),(3,'zero@zero.com','zerosploit',NULL,'$2a$12$A27tnkmKETdYfJ0749WFB.eAmnC4NIKFv6s8XlkymZxJ.A
UkYDsXm',0,NULL,'','',NULL,'America/New_York',0,1,1,0,'2025-05-19T11:06:40.066Z','2025-05-19T11:06:40.066Z','local','en','markdown',NULL,'',''),(4,'halwagy@halwagy.com','halwagy',NULL,
'$2a$12$Q6DH9HbkuQwAGusbCEI/be5GFoYiz7LZ4iwscGmDbWL72359j2q6C',0,NULL,'','',NULL,'America/New_York',0,1,1,0,'2025-05-19T11:07:12.959Z','2025-05-19T11:07:12.959Z','local','en','markdown
',NULL,'','');
docker@minikube:~$ 
```

4. **Check for users inside the mysql database itself**



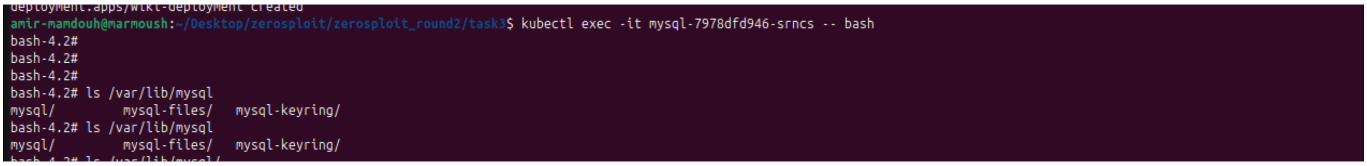
```
mysql> select * from users;
+----+---------------------+---------------+------------+--------------------------------------------------------------+-------------+-----------+----------+----------+------------+-------------------+----------+----------+------------+---------------+-----------
---------------+--------------------------+-------------+------------+---------------+--------------------------+------------+------------+
| id | email               | name          | providerId | password                                                     | tfaIsActive | tfaSecret | jobTitle | location | pictureUrl | timezone          | isSystem | isActive | isVerified | mustChangePwd | createdAt
               | updatedAt                | providerKey | localeCode | defaultEditor | lastLoginAt              | dateFormat | appearance |
+----+---------------------+---------------+------------+--------------------------------------------------------------+-------------+-----------+----------+----------+------------+-------------------+----------+----------+------------+---------------+-----------
---------------+--------------------------+-------------+------------+---------------+--------------------------+------------+------------+
|  1 | amir@amir.com       | Administrator | NULL       | $2a$12$jrC2tc.U.pUDJt1cZPUaIuFjGi70KHF02Gp5vdgza9gTA9bqelQju |           0 | NULL      |          |          | NULL       | America/New_York  |        0 |        1 |          1 |             0 | 2025-05-19
T11:05:07.069Z | 2025-05-19T11:05:07.070Z | local       | en         | markdown      | 2025-05-19T11:05:36.974Z |            |            |
|  2 | guest@example.com   | Guest         | NULL       |                                                              |           0 | NULL      |          |          | NULL       | America/New_York  |        1 |        1 |          1 |             0 | 2025-05-19
T11:05:07.591Z | 2025-05-19T11:05:07.591Z | local       | en         | markdown      | NULL                     |            |            |
|  3 | zero@zero.com       | zerosploit    | NULL       | $2a$12$A27tnkmKETdYfJ0749WFB.eAmnC4NIKFv6s8XlkymZxJ.AUkYDsXm |           0 | NULL      |          |          | NULL       | America/New_York  |        0 |        1 |          1 |             0 | 2025-05-19
T11:06:40.066Z | 2025-05-19T11:06:40.066Z | local       | en         | markdown      | NULL                     |            |            |
|  4 | halwagy@halwagy.com | halwagy       | NULL       | $2a$12$Q6DH9HbkuQwAGusbCEI/be5GFoYiz7LZ4iwscGmDbWL72359j2q6C |           0 | NULL      |          |          | NULL       | America/New_York  |        0 |        1 |          1 |             0 | 2025-05-19
T11:07:12.959Z | 2025-05-19T11:07:12.959Z | local       | en         | markdown      | NULL                     |            |            |
+----+---------------------+---------------+------------+--------------------------------------------------------------+-------------+-----------+----------+----------+------------+-------------------+----------+----------+------------+---------------+-----------
---------------+--------------------------+-------------+------------+---------------+--------------------------+------------+------------+
4 rows in set (0.00 sec)

mysql> 
```

5. **Delete wiki , mysql pods , PVC and PV and recreate them.**



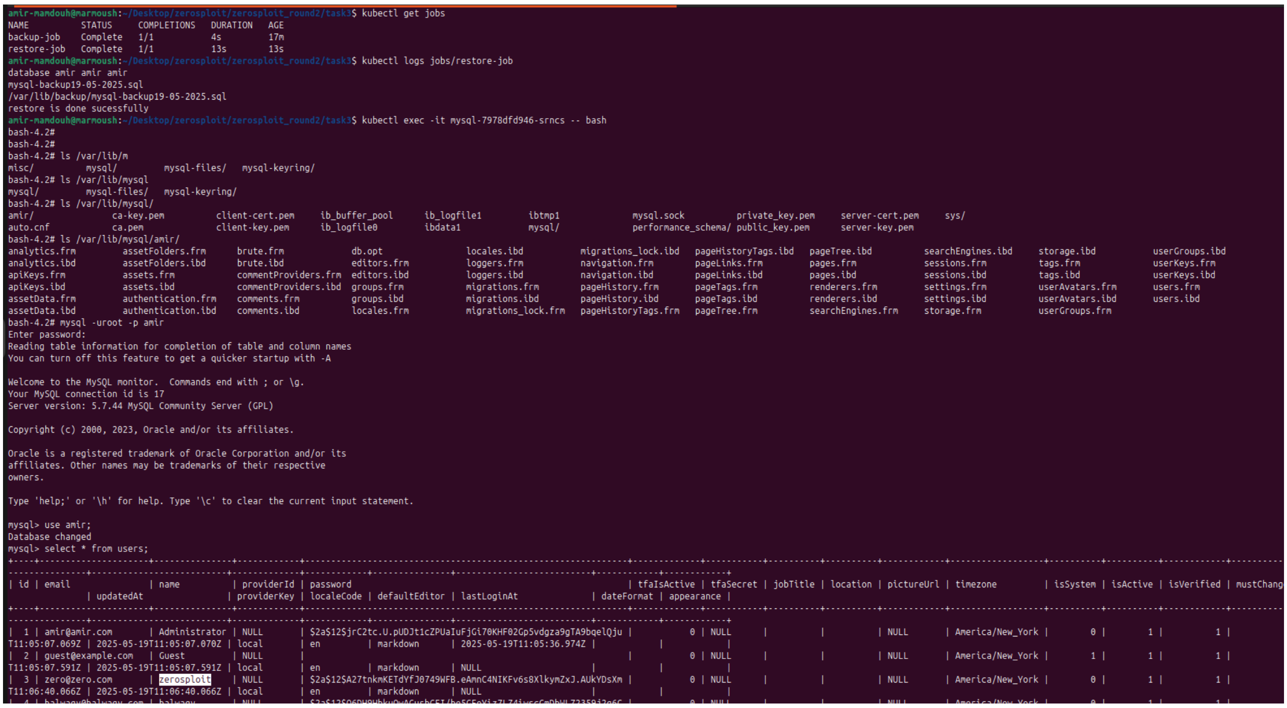
```
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl apply -f mysql-pv.yml
persistentvolume/mysql-pv created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl apply -f mysql-pvc.yml
persistentvolumeclaim/mysql-pvc created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl delete -f mysql.yml
deployment.apps "mysql" deleted
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl apply  -f mysql.yml
deployment.apps/mysql created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl delete -f wiki.yml
deployment.apps "wiki-deployment" deleted
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl apply  -f wiki.yml
deployment.apps/wiki-deployment created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl get pods
NAME                              READY   STATUS      RESTARTS   AGE
backup-job-kjvhp                  0/1     Completed   0          13m
mysql-7978dfd946-srncs            1/1     Running     0          31s
restore-job-8vh5j                 0/1     Completed   0          46m
wiki-deployment-66b49b9d6-zrxvb   1/1     Running     0          6s
```

We deleted everything!!!



```
deployment.apps/wiki-deployment created
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl exec -it mysql-7978dfd946-srncs -- bash
bash-4.2#
bash-4.2#
bash-4.2#
bash-4.2# ls /var/lib/mysql
mysql/          mysql-files/   mysql-keyring/
bash-4.2# ls /var/lib/mysql
mysql/          mysql-files/   mysql-keyring/
bash-4.2# ls /var/lib/mysql/
```

6. **run the restore job and check for data**



```
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl get jobs
NAME          STATUS     COMPLETIONS   DURATION   AGE
backup-job    Complete   1/1           4s         17m
restore-job   Complete   1/1           13s        13s
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl logs jobs/restore-job
database amir amir amir
mysql-backup19-05-2025.sql
/var/lib/backup/mysql-backup19-05-2025.sql
restore is done sucessfully
amir-mamdouh@marmoush:~/Desktop/zerosploit/zerosploit_round2/task3$ kubectl exec -it mysql-7978dfd946-srncs -- bash
bash-4.2#
bash-4.2#
bash-4.2# ls /var/lib/m
misc/         mysql/         mysql-files/   mysql-keyring/
bash-4.2# ls /var/lib/mysql
mysql/         mysql-files/   mysql-keyring/
bash-4.2# ls /var/lib/mysql/
amir/               ca-key.pem          client-cert.pem     ib_buffer_pool      ib_logfile1         ibtmp1              mysql.sock          private_key.pem     server-cert.pem     sys/
auto.cnf            ca.pem              client-key.pem      ib_logfile0         ibdata1             mysql/              performance_schema/ public_key.pem      server-key.pem
bash-4.2# ls /var/lib/mysql/amir/
analytics.frm         assetFolders.frm      brute.frm             db.opt                locales.ibd           migrations_lock.ibd   pageHistoryTags.ibd   pageTree.ibd          searchEngines.ibd     storage.ibd           userGroups.ibd
analytics.ibd         assetFolders.ibd      brute.ibd             editors.frm           loggers.frm           navigation.frm        pageLinks.frm         pages.frm             sessions.frm          tags.frm              userKeys.frm
apiKeys.frm           assets.frm            commentProviders.frm  editors.ibd           loggers.ibd           navigation.ibd        pageLinks.ibd         pages.ibd             sessions.ibd          tags.ibd              userKeys.ibd
apiKeys.ibd           assets.ibd            commentProviders.ibd  groups.frm            migrations.frm        pageHistory.frm       pageTags.frm          renderers.frm         settings.frm          userAvatars.frm       users.frm
assetData.frm         authentication.frm    comments.frm          groups.ibd            migrations.ibd        pageHistory.ibd       pageTags.ibd          renderers.ibd         settings.ibd          userAvatars.ibd       users.ibd
assetData.ibd         authentication.ibd    comments.ibd          locales.frm           migrations_lock.frm   pageHistoryTags.frm   pageTree.frm          searchEngines.frm     storage.frm           userGroups.frm
bash-4.2# mysql -uroot -p amir
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 5.7.44 MySQL Community Server (GPL)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use amir;
Database changed
mysql> select * from users;
+----+---------------------+---------------+------------+--------------------------------------------------------------+-------------+-----------+----------+----------+------------+------------------+----------+----------+------------+-----------
--------------+--------------------------+-------------+------------+---------------+--------------------------+------------+------------+
| id | email               | name          | providerId | password                                                     | tfaIsActive | tfaSecret | jobTitle | location | pictureUrl | timezone         | isSystem | isActive | isVerified | mustChang
              | updatedAt                | providerKey | localeCode | defaultEditor | lastLoginAt              | dateFormat | appearance |
+----+---------------------+---------------+------------+--------------------------------------------------------------+-------------+-----------+----------+----------+------------+------------------+----------+----------+------------+-----------
--------------+--------------------------+-------------+------------+---------------+--------------------------+------------+------------+
|  1 | amir@amir.com       | Administrator | NULL       | $2a$12$jrC2tc.U.pUDJt1cZPUaIuFjGi70KHF02Gp5vdgza9gTA9bqelQju |           0 | NULL      |          |          | NULL       | America/New_York |        0 |        1 |          1 |
T11:05:07.069Z | 2025-05-19T11:05:07.070Z | local       | en         | markdown      | 2025-05-19T11:05:36.974Z |            |            |
|  2 | guest@example.com   | Guest         | NULL       |                                                              |           0 | NULL      |          |          | NULL       | America/New_York |        1 |        1 |          1 |
T11:05:07.591Z | 2025-05-19T11:05:07.591Z | local       | en         | markdown      | NULL                     |            |            |
|  3 | zero@zero.com       | zerosploit    | NULL       | $2a$12$A27tnkmKETdYfJ0749WFB.eAmnC4NIKFv6s8XlkymZxJ.AUkYDsXm |           0 | NULL      |          |          | NULL       | America/New_York |        0 |        1 |          1 |
T11:06:40.066Z | 2025-05-19T11:06:40.066Z | local       | en         | markdown      | NULL                     |            |            |
|  4 | halwagy@halwagy.com | halwagy       | NULL       | [illegible]                                                  |           0 | NULL      |          |          | NULL       | America/New_York |        0 |        1 |          1 |
```

- zerosploit and halwagy users exist..

Notes

- All volumes use `hostPath`, so ensure `/mnt/data/...` directories exist on all cluster nodes.
- Backup job will save `all.sql` inside the backup volume.
- PVC and PV to make them bounded you need to make sure for three things: 1- Same StorageClassName 2- PVC storage <= PV capacity 3- Access Modes are the same
- Secrets must be base64 encoded. Example:

```
echo -n 'rootpassword' | base64
```