Experiment – Helm in AKS

We can install Helm, but a fairly current version will exist in our AKS environment. This allows us to work in containerization with Helm Charts immediately after creating an AKS environment.

Redis in AKS with Helm

If we didn't have Helm we could pull the most recent version, but we will not have to do the following steps since Helm is already installed in our Bash Cloud Shell environment.

curl -LO https://get.helm.sh/helm-v3.9.0-linux-amd64.tar.gz

tar -xvzf helm-v3.9.0-linux-amd64.tar.gz

mv linux-amd64/helm /usr/local/bin

Open Cloud Shell with the Bash option

Get our AKS credentials for our AKS environment

student1@Azure:~$ **az aks get-credentials **

**> --resource-group AzureDataEngineering **

> --name adeaks

Check our Helm Help

student1@Azure:~$ **helm -h**

Install Redis in a container in AKS

student1@Azure:~$ **helm repo add bitnami https://charts.bitnami.com/bitnami**

student1@Azure:~$ **helm install my-release bitnami/redis**

NAME: my-release

LAST DEPLOYED: Fri May 20 01:03:22 2022

NAMESPACE: default

STATUS: deployed

REVISION: 1

TEST SUITE: None

NOTES:

CHART NAME: redis

CHART VERSION: 16.9.7

APP VERSION: 6.2.7

** Please be patient while the chart is being deployed **

Redis™ can be accessed on the following DNS names from within your cluster:

    my-release-redis-master.default.svc.cluster.local for read/write operations (port 6379)

    my-release-redis-replicas.default.svc.cluster.local for read-only operations (port 6379)

To get your password run:

    export REDIS_PASSWORD=$(kubectl get secret --namespace default my-release-redis -o jsonpath="{.data.redis-password}" | base64 --decode)

To connect to your Redis™ server:

1. Run a Redis™ pod that you can use as a client:

```
kubectl run --namespace default redis-client --restart='Never'  --env REDIS_PASSWORD=$REDIS_PASSWORD  --image docker.io/bitnami/redis:6.2.7-debian-10-r20 --command -- sleep infinity
```

Use the following command to attach to the pod:

```
kubectl exec --tty -i redis-client \
--namespace default -- bash
```

2. Connect using the Redis™ CLI:

```
REDISCLI_AUTH="$REDIS_PASSWORD" redis-cli -h my-release-redis-master
REDISCLI_AUTH="$REDIS_PASSWORD" redis-cli -h my-release-redis-replicas
```

To connect to your database from outside the cluster execute the following commands:

```
kubectl port-forward --namespace default svc/my-release-redis-master 6379:6379 &
REDISCLI_AUTH="$REDIS_PASSWORD" redis-cli -h 127.0.0.1 -p 6379
```

Examine the Bitnami Repo

student1@Azure:~$ **helm search repo bitnami**

NAME	CHART VERSION	APP VERSION	DESCRIPTION
bitnami/bitnami-common	0.0.9	0.0.9	DEPRECATED Chart with custom templates used in ...

bitnami/airflow 12.3.2 2.3.0 Apache Airflow is a tool to express and execute...

bitnami/apache 9.1.2 2.4.53 Apache HTTP Server is an open-source HTTP serve...

bitnami/argo-cd 3.2.4 2.3.3 Argo CD is a continuous delivery tool for Kuber...

bitnami/argo-workflows 2.1.1 3.3.5 Argo Workflows is meant to orchestrate Kubernet...

bitnami/aspnet-core 3.3.2 6.0.5 ASP.NET Core is an open-source framework for we...

bitnami/cassandra 9.1.19 4.0.3 Apache Cassandra is an open source distributed ...

used for the automated d...

…..

bitnami/kube-prometheus 6.13.0 0.56.2 Prometheus Operator provides easy monitoring de...

bitnami/wordpress 14.2.2 5.9.3 WordPress is the world's most popular blogging ...

bitnami/wordpress-intel 1.2.1 5.9.3 WordPress for Intel is the most popular bloggin...

bitnami/zookeeper 9.1.4 3.8.0 Apache ZooKeeper provides a reliable, centraliz...

Check out what we've installed with Helm

student1@Azure:~$ **helm list**

NAME NAMESPACE REVISION UPDATED STATUS CHART APP VERSION

my-release default 1 2022-05-20 01:03:22.241277304 +0000 UTC deployed redis-16.9.7 6.2.7

student1@Azure:~$ **kubectl run --namespace default redis-client --restart='Never' --env REDIS_PASSWORD=$REDIS_PASSWORD --image docker.io/bitnami/redis:6.2.7-debian-10-r20 --command -- sleep infinity**

pod/redis-client created

student1@Azure:~$ **kubectl get pods**

NAME READY STATUS RESTARTS AGE

azure-vote-back-5f8bd8b-k5zls 1/1 Running 0 45h

azure-vote-front-798779f99d-4p6bf 1/1 Running 0 45h

my-release-redis-master-0 1/1 Running 0 7m47s

my-release-redis-replicas-0 1/1 Running 0 7m47s

my-release-redis-replicas-1 1/1 Running 0 6m13s

my-release-redis-replicas-2 1/1 Running 0 5m25s

redis-client 1/1 Running 0 2m54s

Check the IP for our Redis server container to use with the CLI

student1@Azure:~$ **kubectl get services**

NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE

azure-vote-back ClusterIP 10.0.206.23 <none> 6379/TCP 45h

azure-vote-front LoadBalancer 10.0.127.29 20.236.42.48 80:30651/TCP 45h

kubernetes ClusterIP 10.0.0.1 <none> 443/TCP 46h

my-release-redis-headless ClusterIP None <none> 6379/TCP 9m2s

my-release-redis-master ClusterIP **10.0.122.208** <none> 6379/TCP 9m2s

my-release-redis-replicas ClusterIP 10.0.108.133 <none> 6379/TCP 9m2s

Check the password for our Redis container

student1@Azure:~$ kubectl get secret --namespace default my-release-redis -o jsonpath="{.data.redis-password}" | base64 --decode

tarmG2m7ea

Access our Redis container

student1@Azure:~$ **kubectl exec --tty -i redis-client --namespace default -- bash**

I have no name!@redis-client:/$ **redis-cli --help**

redis-cli 6.2.7

Access the Redis CLI

I have no name!@redis-client:/$ **redis-cli -h 10.0.122.208**

10.0.122.208:6379> **AUTH tarmG2m7ea**

OK

10.0.122.208:6379> **set Name "BringTheRain"**

OK

10.0.122.208:6379> **get Name**

"BringTheRain"

10.0.122.208:6379> **exit**

I have no name!@redis-client:/$ **exit**