Deploy apps

Astra

NetApp
July 12, 2021

Table of Contents

Deploy apps

Deploy Jenkins from a Helm chart

Learn how to deploy Jenkins from the Bitnami Helm chart. After you deploy Jenkins on your cluster, you can register the application with Astra Control.

Jenkins is a validated app for Astra Control. Learn the difference between Validated and Standard apps.

> Applications deployed from Google Marketplace have not been validated. Some users report issues with discovery and/or backup with Google Marketplace deployments of Postgres, MariaDB, and MySQL.

Requirements

- A GKE cluster that has been added to Astra Control Service.
- Updated versions of Helm (version 3.2+) and Kubectl installed.
- Kubeconfig configured using the gcloud tool with a command like `gcloud container clusters get-credentials my-cluster-name`

Astra Control does not currently support the Kubernetes plugin for Jenkins. You can run Jenkins in a Kubernetes cluster without the plugin. The plugin provides scalability to your Jenkins cluster.

Install Jenkins

Two important notes on this process:

- You must deploy your app after the cluster is added to Astra Control, not before.
- You must deploy the Helm chart in a namespace other than the default.

Add the Bitnami chart repo:

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

Create the `jenkins` namespace and deploy Jenkins into it with the command:

```
kubectl create namespace jenkins && helm install jenkins --namespace
jenkins --set persistence.storageClass=netapp-cvs-perf-
premium,persistence.size=100Gi bitnami/jenkins
```

This does the following:

- Creates the `jenkins` namespace.
- Sets the correct storage class.
- Sets the persistent volume storage size to 100Gi.

After the pods are online, you can manage the app with Astra Control. Astra Control allows you to manage an app at the namespace level or using a helm label

Deploy MariaDB from a Helm chart

Learn how to deploy MariaDB from the Bitnami Helm chart. After you deploy MariaDB on your cluster, you can manage the application with Astra Control.

MariaDB is a validated app for Astra Control. Learn the difference between Validated and Standard apps.

Applications deployed from the Google and Azure Marketplaces have not been validated. Some users report issues with discovery and/or backup with Google Marketplace deployments of Postgres, MariaDB, and MySQL.

Requirements

- A GKE or AKS cluster that has been added to Astra Control Service.
- Updated versions of Helm (version 3.2+) and Kubectl installed.
- Kubeconfig configured using the gcloud tool with a command like `gcloud container clusters get-credentials my-cluster-name`
- Kubeconfig configured using the az tool with a command like `az aks get-credentials --resource-group resource-group-name --name aks-cluster-name`

Install MariaDB

Two important notes on this process:

- You must deploy your app after the cluster is added to Astra Control, not before.
- You must deploy the Helm chart in a namespace other than the default.

Add the Bitnami chart repo:

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

Deploy MariaDB with the command:

```
helm install mariadb bitnami/mariadb --namespace testdb --create-namespace
--set db.database=test_db,db.user=test_db_user,db.password=choose-your-
password > /dev/null 2>&1
```

This does the following:

- Creates the `testdb` namespace.
- Deploys MariaDB on the `testdb` namespace.
- Creates a database named `test_db`

- Creates a user `test_db_user` with the password that you provided.

This method of setting the password at deployment is insecure. We do not recommend this for a production environment.

After the pods are online, you can manage the app with Astra Control. Astra Control allows you to manage an app at the namespace level or using a helm label

Deploy MySQL from a Helm chart

Learn how to deploy MySQL from the standard stable chart. After you deploy MySQL on your Kubernetes cluster, you can manage the application with Astra Control.

MySQL is a validated app for Astra Control. Learn the difference between Validated and Standard apps.

Applications deployed from Google Marketplace have not been validated. Some users report issues with discovery and/or backup with Google Marketplace deployments of Postgres, MariaDB, and MySQL.

Requirements

- A GKE cluster that has been added to Astra Control Service.
- Updated versions of Helm (version 3.2+) and Kubectl installed.
- Kubeconfig configured using the gcloud tool with a command like `gcloud container clusters get-credentials my-cluster-name`

Install MySQL

Two important notes on this process:

- You must deploy your app after the cluster is added to Astra Control, not before.
- We recommend that you deploy the Helm chart in a namespace other than the default.

Add the Bitnami chart repo:

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

Deploy MySQL with the command:

```
helm install mysql bitnami/mysql --namespace testdb --create-namespace
--set db.database=test_db,db.user=test_db_user,db.password=NKhjs2wQPt8
```

This does the following:

- Creates the `testdb` namespace.
- Deploys MySQL on the `testdb` namespace.

- Creates a database named `test_db`
- Creates a user `test_db_user` with password `NKhjs2wQPt8`

This method of setting the password at deployment is insecure. We do not recommend this for a production environment.

After the pods are online, you can manage the app with Astra Control. Astra Control allows you to manage an app with its name, at the namespace level, or by using a helm label.

Deploy Postgres from a Helm chart

Learn how to deploy Postgres from the Bitnami Helm chart. After you deploy Postgres on your cluster, you can register the application with Astra Control.

Postgres is a validated app for Astra Control. Learn the difference between Validated and Standard apps.

Applications deployed from Google Marketplace have not been validated. Some users report issues with discovery and/or backup with Google Marketplace deployments of Postgres, MariaDB, and MySQL.

Requirements

- A fresh GKE cluster which has been added to Astra Control Service.
- Updated versions of Helm (version 3.2+) and Kubectl installed.
- Kubeconfig configured using the gcloud tool with a command like `gcloud container clusters get-credentials my-cluster-name`

Install Postgres

Two important notes on this process:

- You must deploy your app after the cluster is added to Astra Control, not before.
- You must deploy the Helm chart in a namespace other than the default.

Add the Bitnami chart repo:

```
helm repo add bitnami https://charts.bitnami.com/bitnami
```

Deploy Postgres with the command:

```
kubectl create namespace testdb && helm install bitnami/postgresql
--namespace testdb --set
postgresqlPassword=U9dH9HT4pWS,postgresqlDatabase=test_db --generate-name
```

This does the following:

- Creates the `testdb` namespace.
- Deploys Postgres on the `testdb` namespace.
- Creates a database named `test_db`
- Creates a user `test_db_user` with password `U9dH9HT4pWS`

This method of setting the password at deployment is insecure. We do not recommend this for a production environment.

After the pods are online, you can manage the app with Astra Control. Astra Control allows you to manage an app at the namespace level or using a helm label.