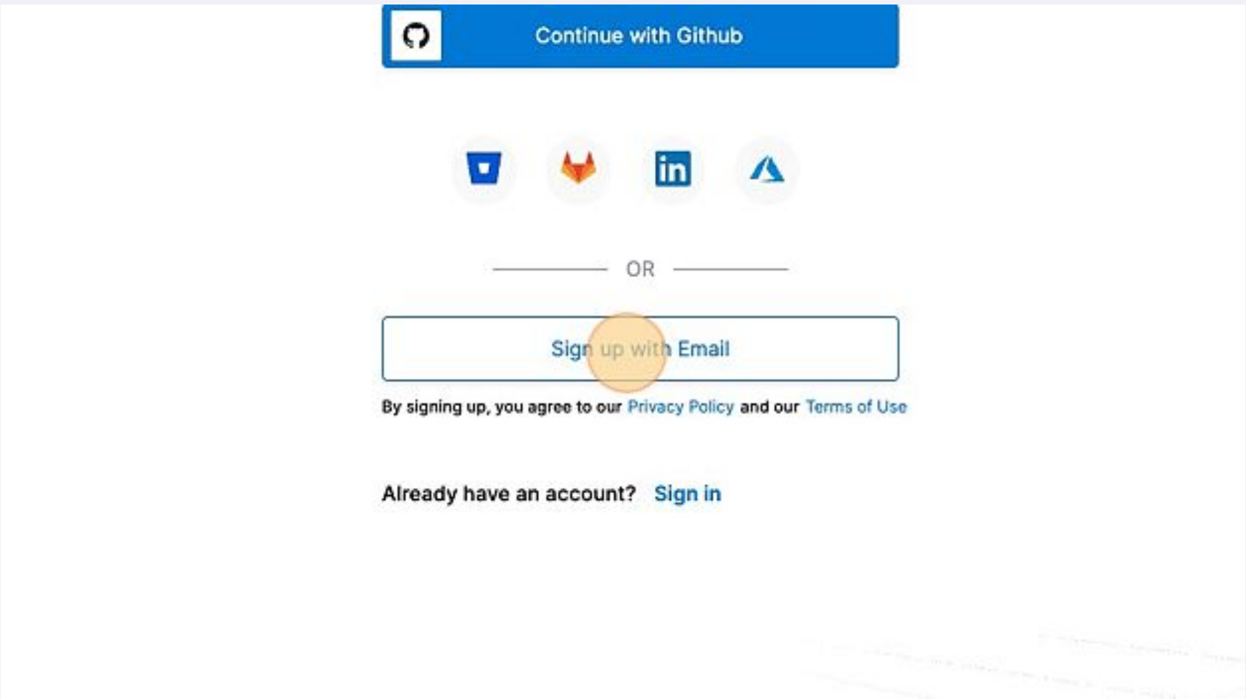
Setting up a Harness Deployment Pipeline: A Visual Step-by-Step Guide

1 Navigate to app.harness.io/auth/#/signup/?module=cd

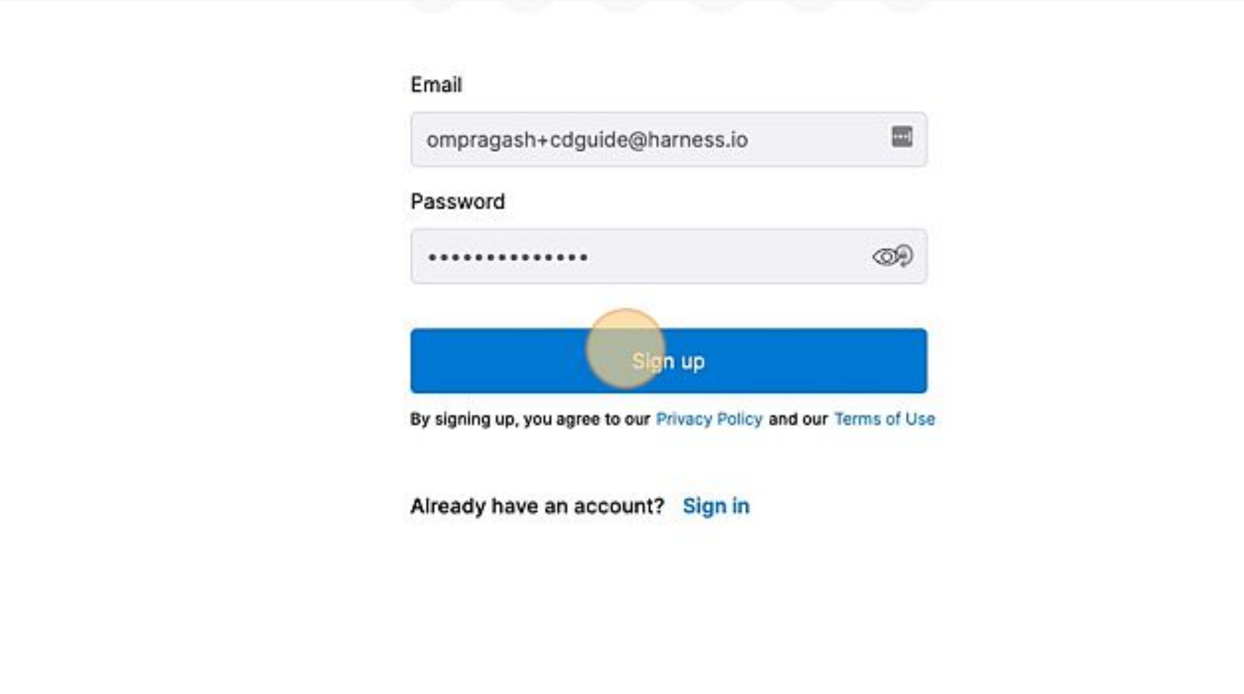
2 Click "Sign up with Email"



By signing up, you agree to our Privacy Policy and our Terms of Use

Already have an account? Sign in

3 Enter Email and Password. Click "Sign up"



Email

ompragash+cdguide@harness.io

Password

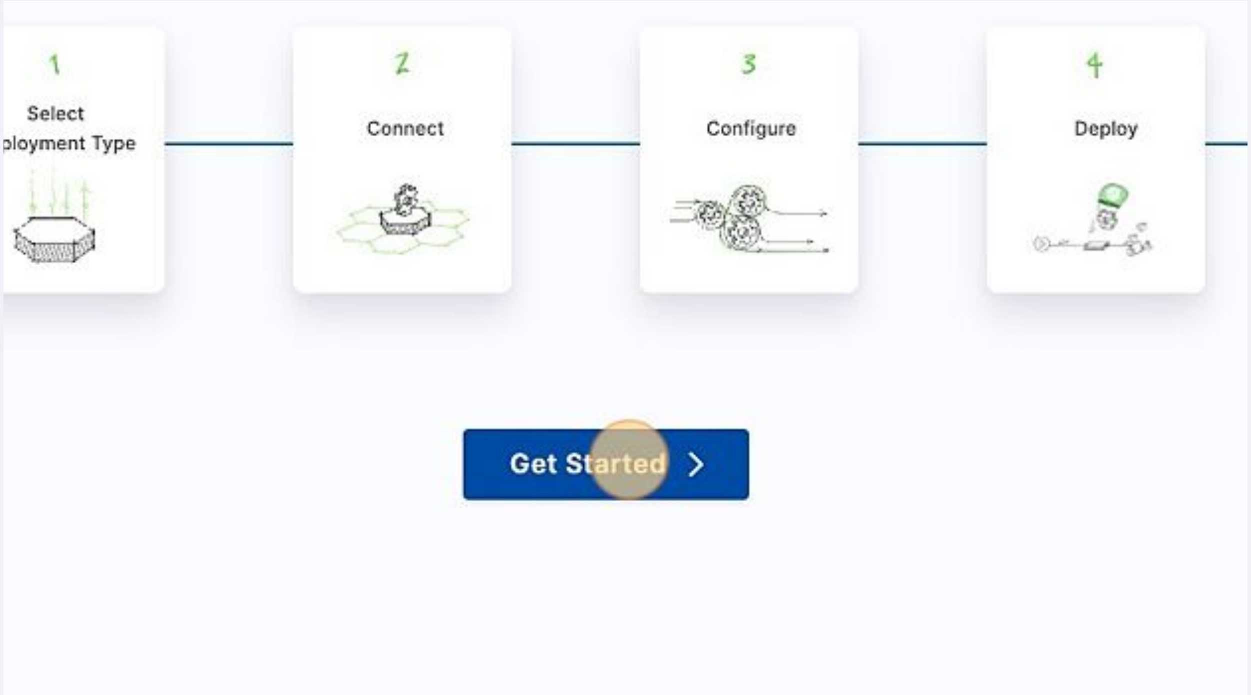
•••••••••••••••

By signing up, you agree to our Privacy Policy and our Terms of Use

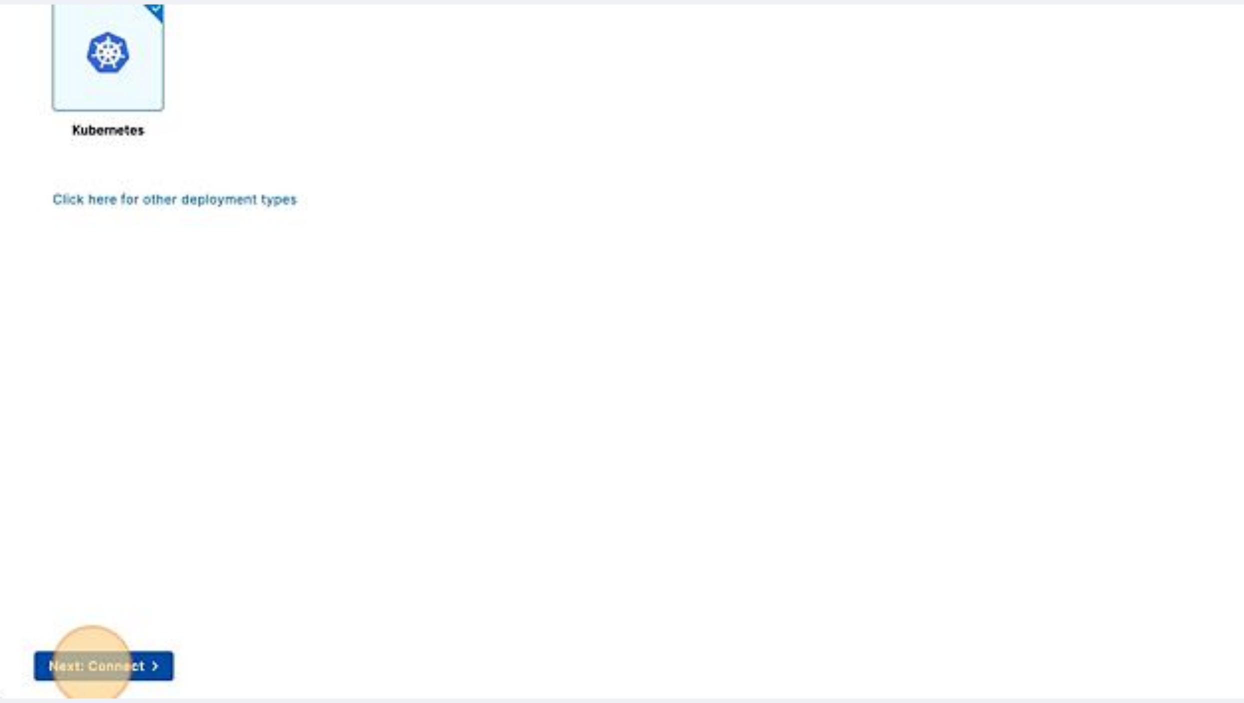
Already have an account? Sign in

4 Verify your Email address and then continue

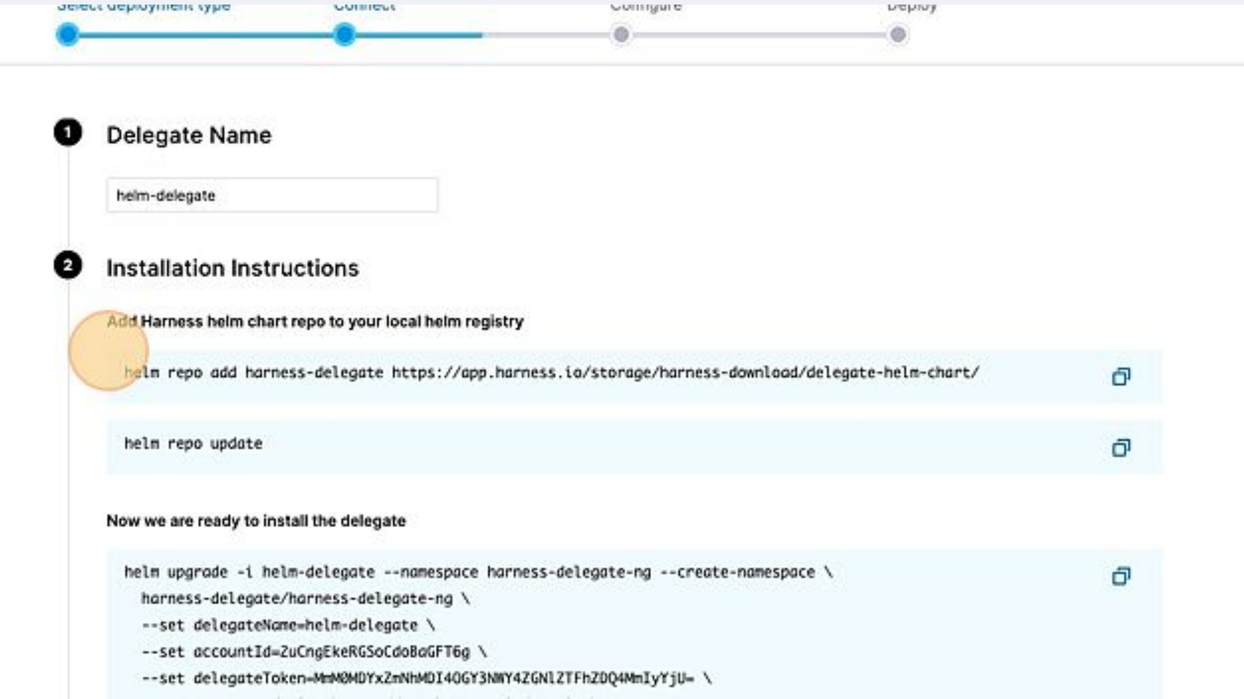
5 Click "Get Started" from CD Onboarding page



6 Kubernetes is great to start with as deployment type!
Click "Next: Connect"



7 Connect Harness to your environment by installing Delegate via helm



1 Delegate Name

helm-delegate

2 Installation Instructions

Add Harness helm chart repo to your local helm registry

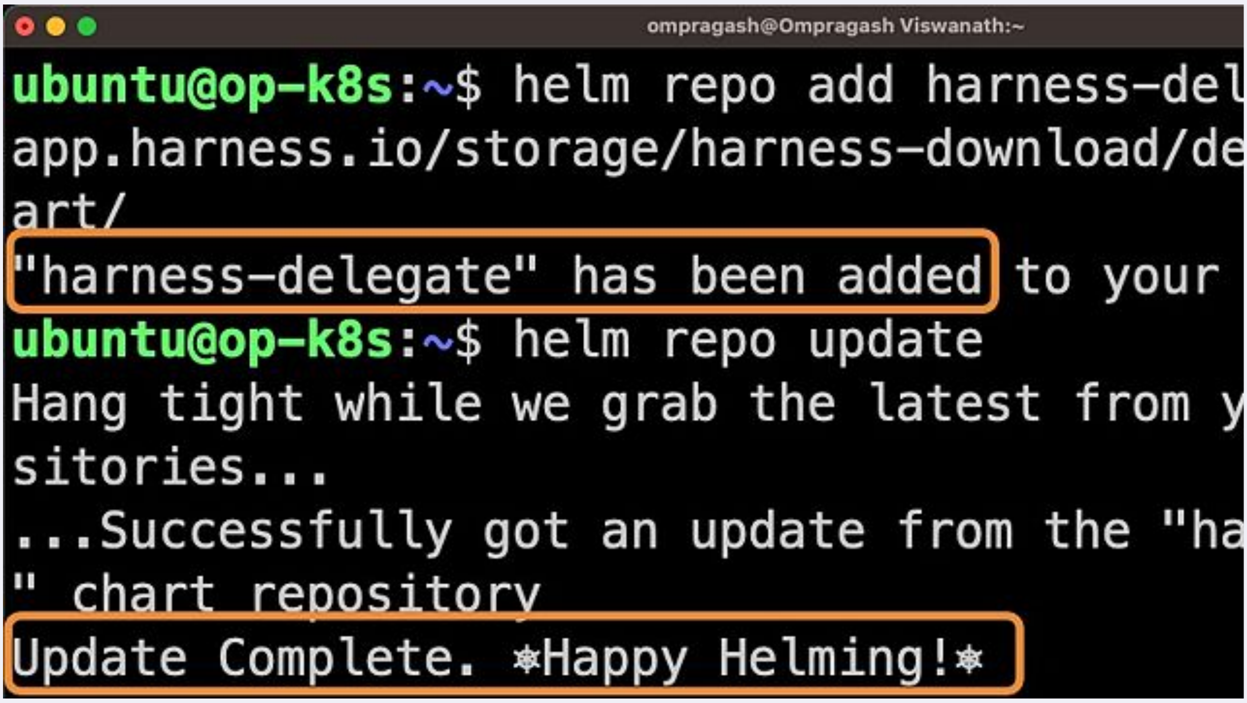
```
helm repo add harness-delegate https://app.harness.io/storage/harness-download/delegate-helm-chart/
```

```
helm repo update
```

Now we are ready to install the delegate

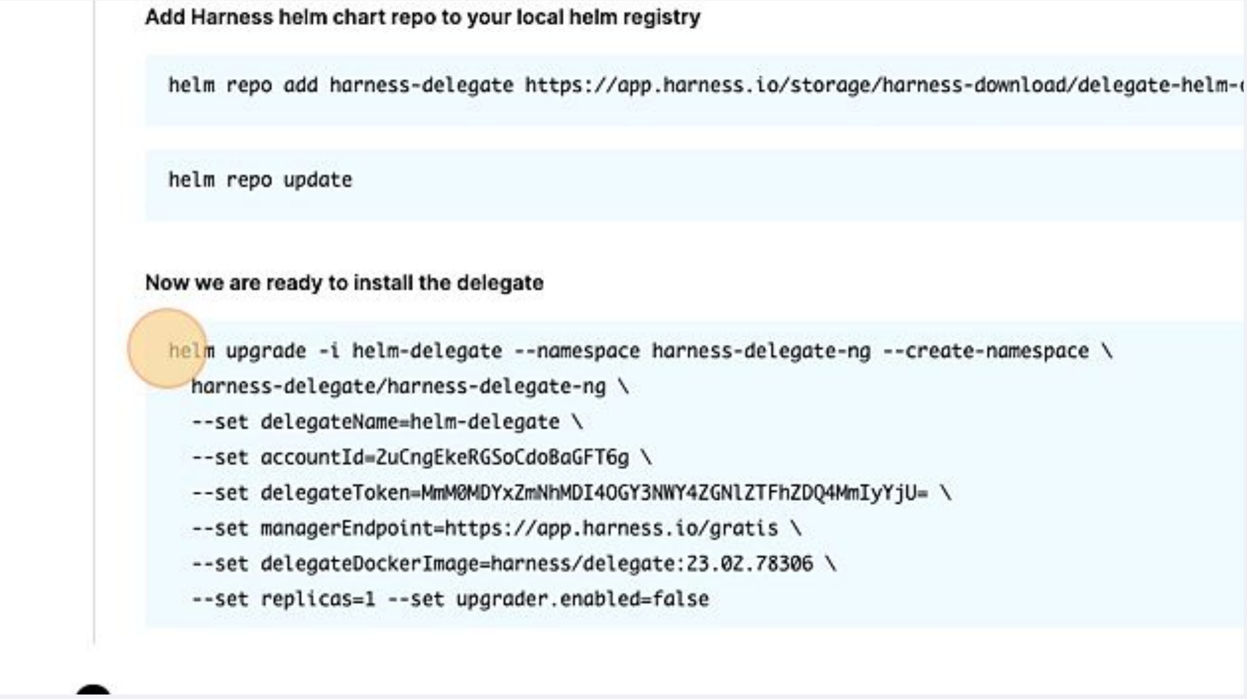
```
helm upgrade -i helm-delegate --namespace harness-delegate-ng --create-namespace \
  harness-delegate/harness-delegate-ng \
  --set delegateName=helm-delegate \
  --set accountId=ZuCngEkeRGSoCdoBaGFT6g \
  --set delegateToken=MmM0MDYxZmNhMDI4OGY3NWY4ZGNlZTFhZDQ4MmIyYjU= \
```

8 Add "delegate-helm-chart" to your local helm registry



```
ubuntu@op-k8s:~$ helm repo add harness-del
app.harness.io/storage/harness-download/de
art/
"harness-delegate" has been added to your
ubuntu@op-k8s:~$ helm repo update
Hang tight while we grab the latest from y
sitories...
...Successfully got an update from the "ha
" chart repository
Update Complete. ⎈Happy Helming!⎈
```

9 Now, copy the "helm upgrade" command



Add Harness helm chart repo to your local helm registry

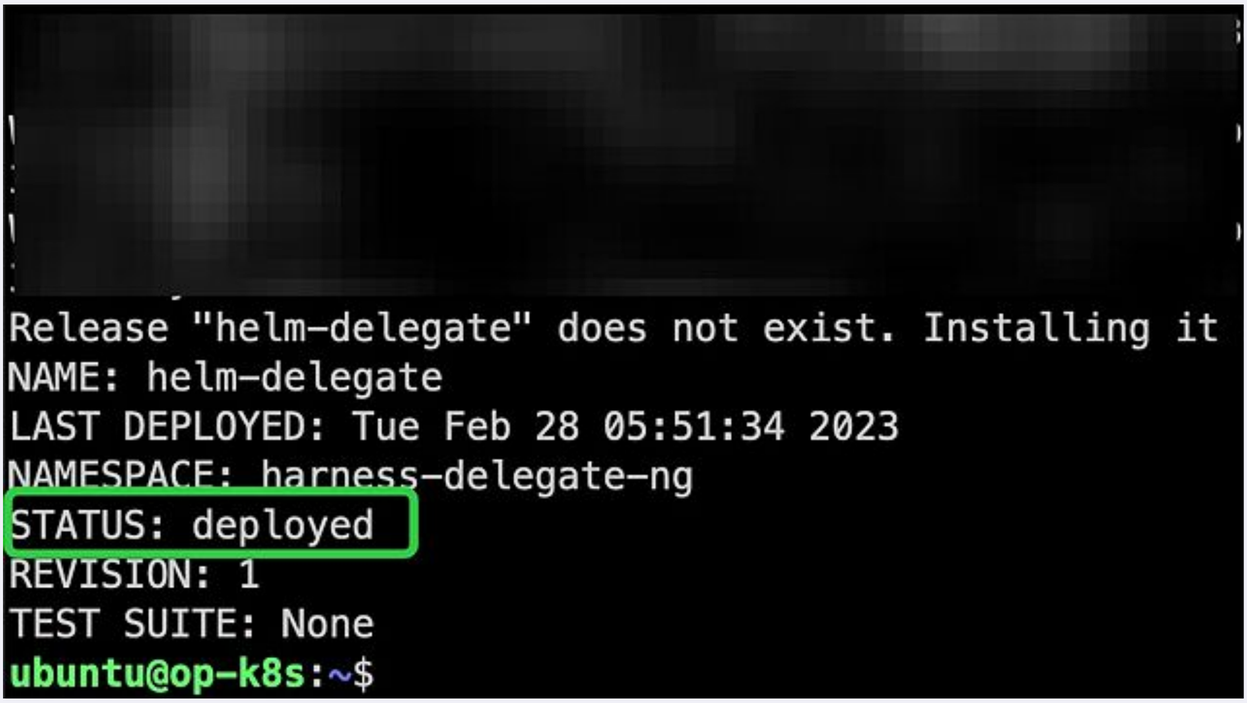
```
helm repo add harness-delegate https://app.harness.io/storage/harness-download/delegate-helm-
```

```
helm repo update
```

Now we are ready to install the delegate

```
helm upgrade -i helm-delegate --namespace harness-delegate-ng --create-namespace \
  harness-delegate/harness-delegate-ng \
  --set delegateName=helm-delegate \
  --set accountId=2uCngEkeRGSoCdoBaGFT6g \
  --set delegateToken=MmM0MDYxZmNhMDI4OGY3NWY4ZGNlZTFhZDQ4MmIyYjU= \
  --set managerEndpoint=https://app.harness.io/gratis \
  --set delegateDockerImage=harness/delegate:23.02.78306 \
  --set replicas=1 --set upgrader.enabled=false
```

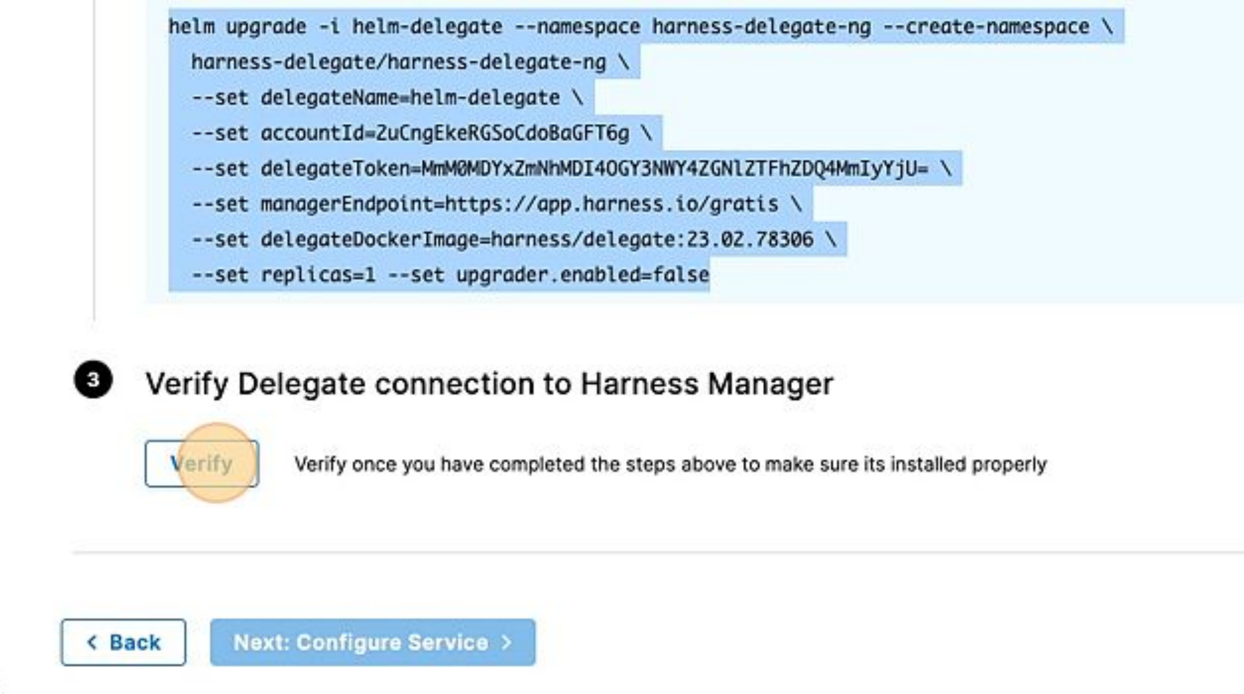
10 On successful install, you should see the status as "deployed"



[redacted]

```
Release "helm-delegate" does not exist. Installing it
NAME: helm-delegate
LAST DEPLOYED: Tue Feb 28 05:51:34 2023
NAMESPACE: harness-delegate-ng
STATUS: deployed
REVISION: 1
TEST SUITE: None
ubuntu@op-k8s:~$
```

11 Click "Verify"

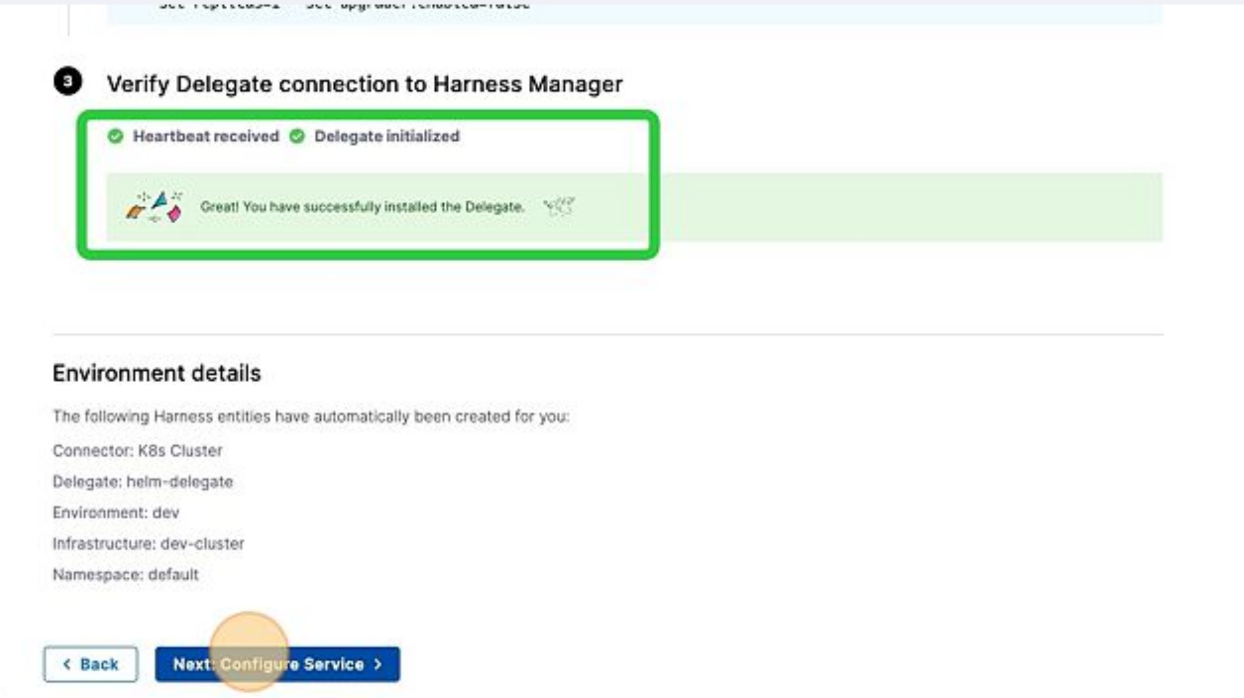


```
helm upgrade -i helm-delegate --namespace harness-delegate-ng --create-namespace \
  harness-delegate/harness-delegate-ng \
  --set delegateName=helm-delegate \
  --set accountId=ZuCngEkeRGSoCdoBaGFT6g \
  --set delegateToken=MmM0MDYxZmNhMDI4OGY3NWY4ZGNlZTFhZDQ4MmIyYjU= \
  --set managerEndpoint=https://app.harness.io/gratis \
  --set delegateDockerImage=harness/delegate:23.02.78306 \
  --set replicas=1 --set upgrader.enabled=false
```

3 Verify Delegate connection to Harness Manager

Verify once you have completed the steps above to make sure its installed properly

12 If the Delegate is installed properly, it will send a heartbeat to the Harness Manager and then click "Next: Configure Service"



3 Verify Delegate connection to Harness Manager

✓ Heartbeat received ✓ Delegate initialized

Great! You have successfully installed the Delegate.

Environment details

The following Harness entities have automatically been created for you:

Connector: K8s Cluster

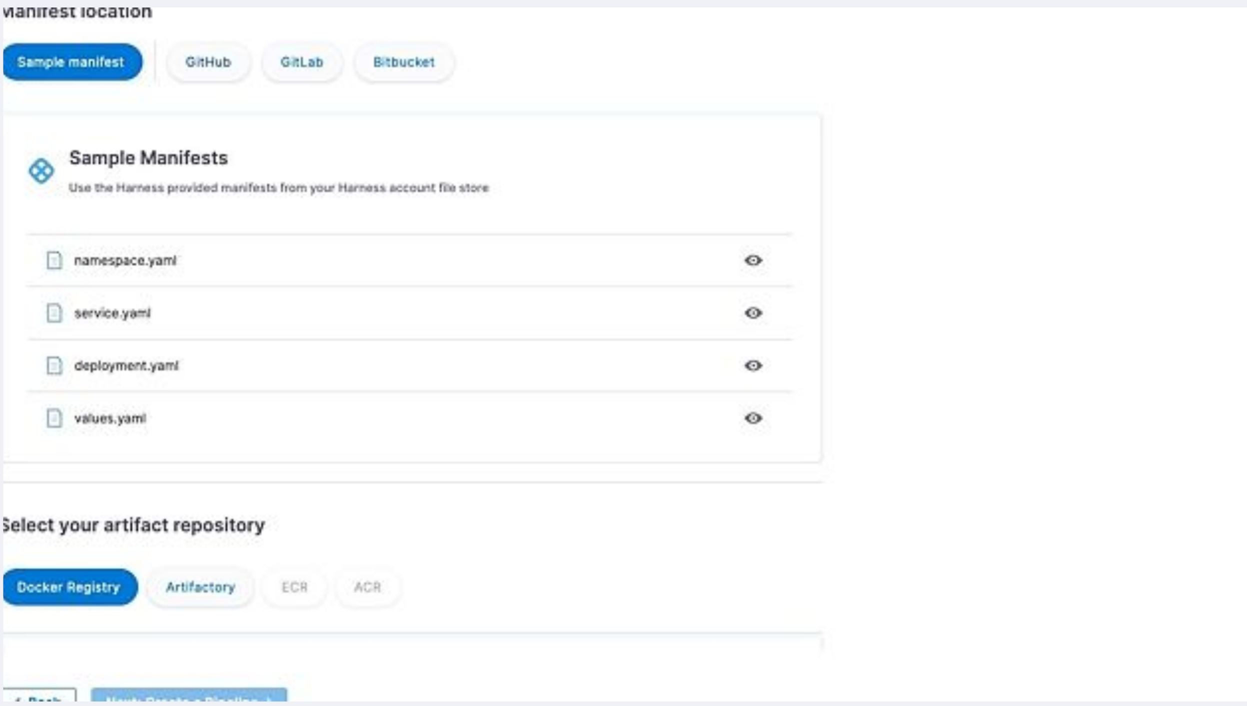
Delegate: helm-delegate

Environment: dev

Infrastructure: dev-cluster

Namespace: default

13 For starters, let's use the Harness provided manifests from Harness File Store. Additionally, click on the 'eye' icon to view the contents of each file



Manifest location

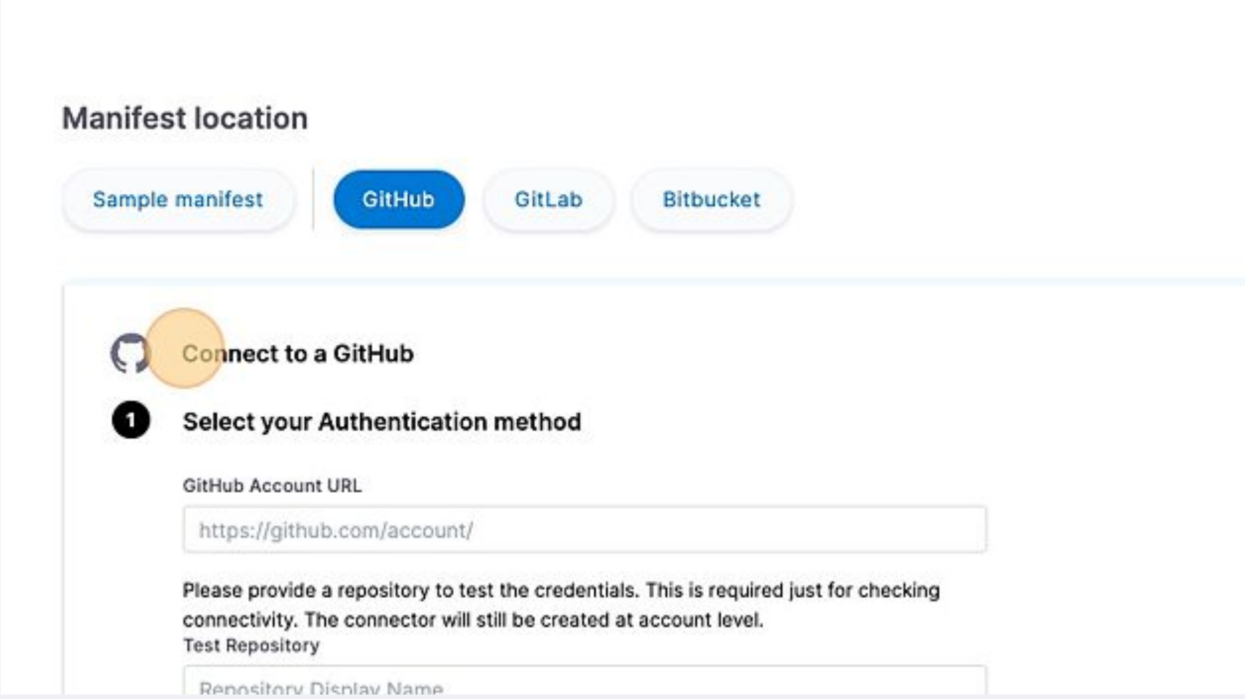
Sample Manifests

Use the Harness provided manifests from your Harness account file store

- namespace.yaml
- service.yaml
- deployment.yaml
- values.yaml

Select your artifact repository

14 Optionally, you can connect your GitHub account and use your own manifests



Manifest location

Connect to a GitHub

1 Select your Authentication method

GitHub Account URL

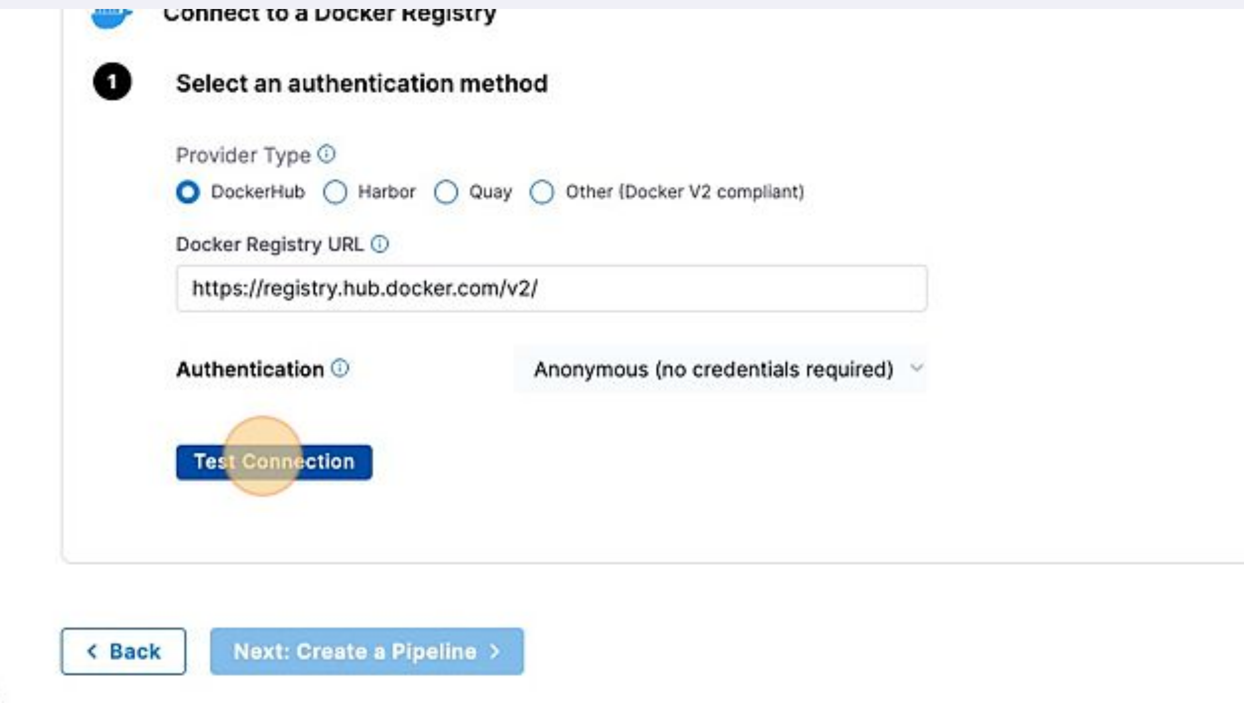
https://github.com/account/

Please provide a repository to test the credentials. This is required just for checking connectivity. The connector will still be created at account level.

Test Repository

Repository Display Name

15 Click "Test Connection" under "Connect to a Docker Registry"



Connect to a Docker Registry

1 Select an authentication method

Provider Type

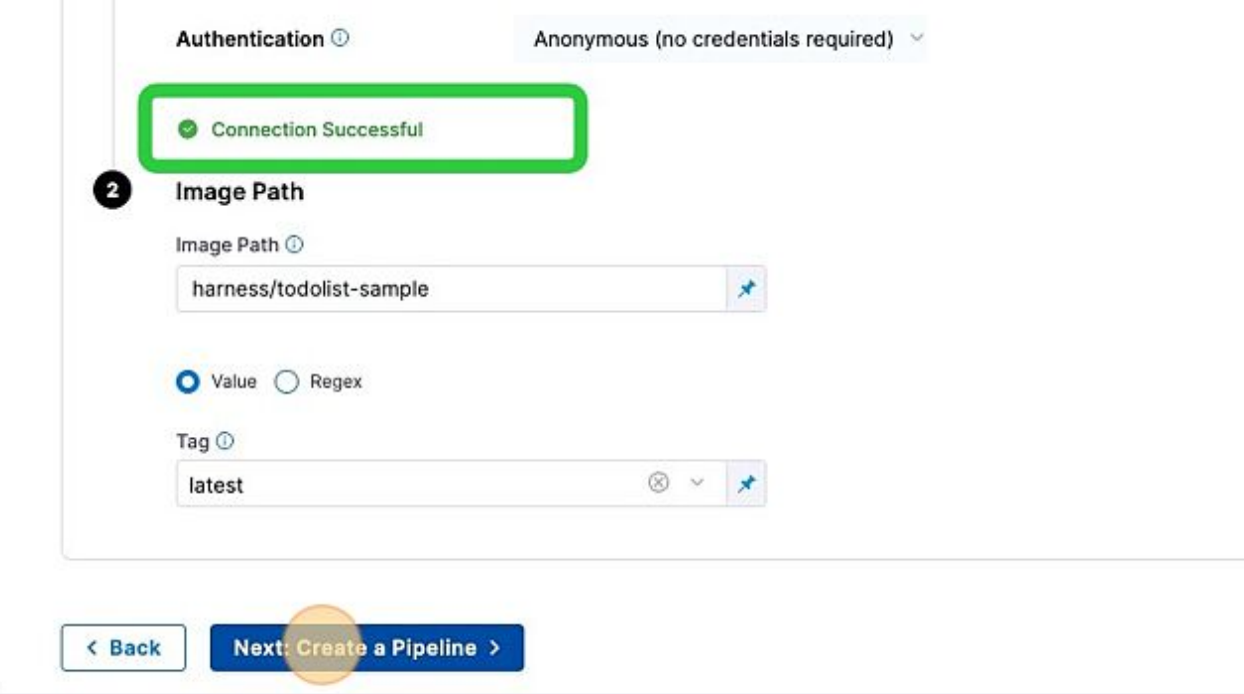
● DockerHub ○ Harbor ○ Quay ○ Other (Docker V2 compliant)

Docker Registry URL

https://registry.hub.docker.com/v2/

Authentication: Anonymous (no credentials required)

16 Check the "Connection Successful" message and the values of "Image Path" & "Tag" fields.
Finally, click "Next: Create a Pipeline"



Authentication: Anonymous (no credentials required)

Connection Successful

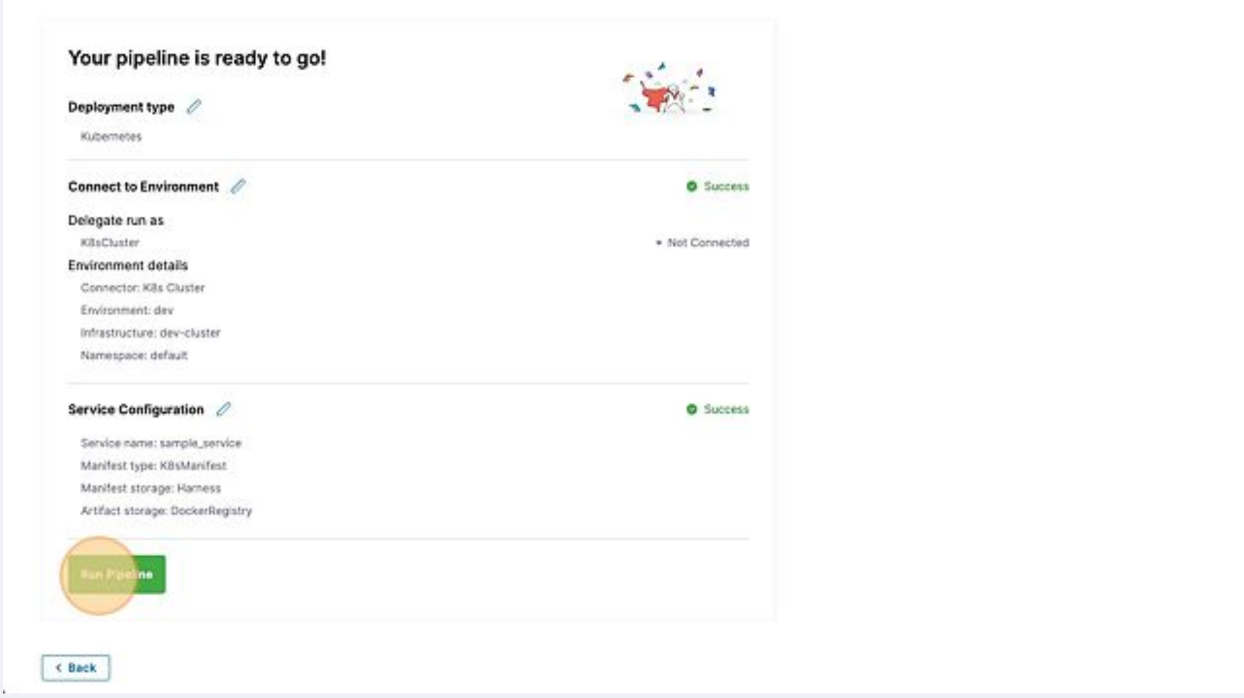
2 Image Path

Image Path: harness/todolist-sample

● Value ○ Regex

Tag: latest

17 Yay! Your Pipeline is ready for launch
Hit "Run Pipeline"



Your pipeline is ready to go!

Deployment type
Kubernetes

Connect to Environment — Success

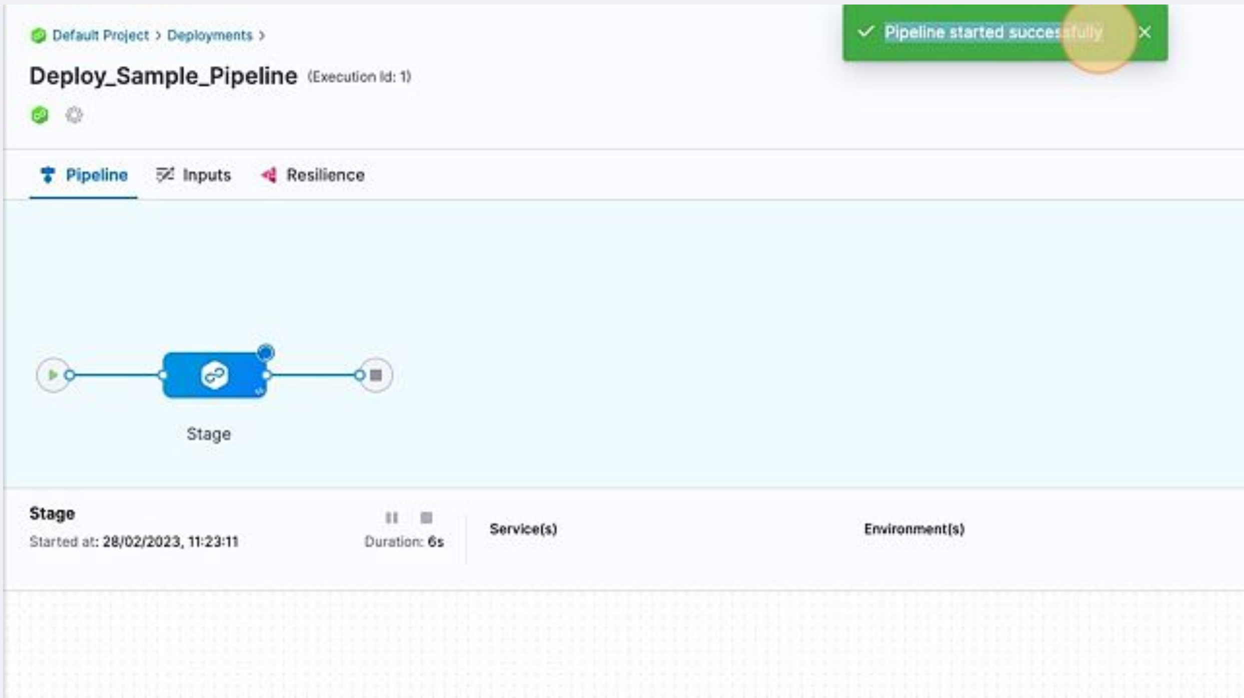
Delegate run as
K8sCluster — • Not Connected

Environment details
Connector: K8s Cluster
Environment: dev
Infrastructure: dev-cluster
Namespace: default

Service Configuration — Success

Service name: sample_service
Manifest type: K8sManifest
Manifest storage: Harness
Artifact storage: DockerRegistry

18 Pipeline started successfully



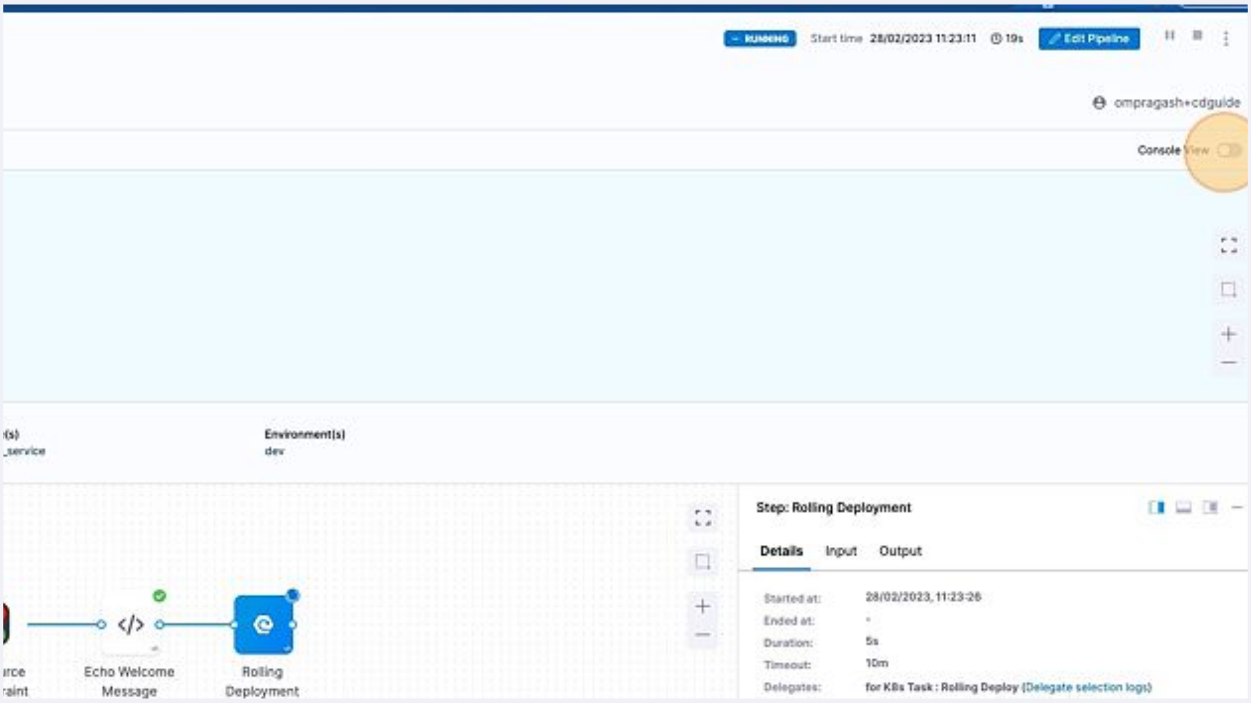
✓ Pipeline started successfully

Default Project › Deployments ›

Deploy_Sample_Pipeline (Execution Id: 1)

Stage
Started at: 28/02/2023, 11:23:11
Duration: 6s
Service(s)
Environment(s)

19 Toggle "Console View" to see the complete execution log



Start time 28/02/2023 11:23:11 19s

ompragash+cdguide

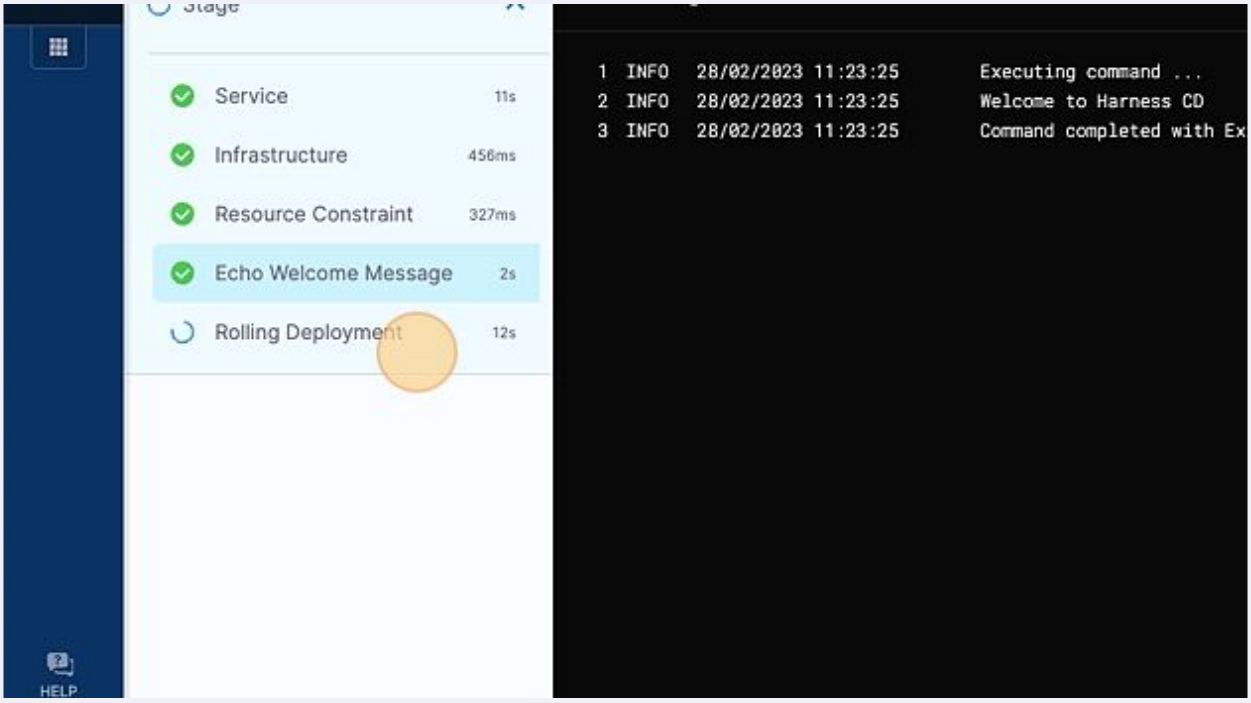
(s) _service

Environment(s) dev

Step: Rolling Deployment

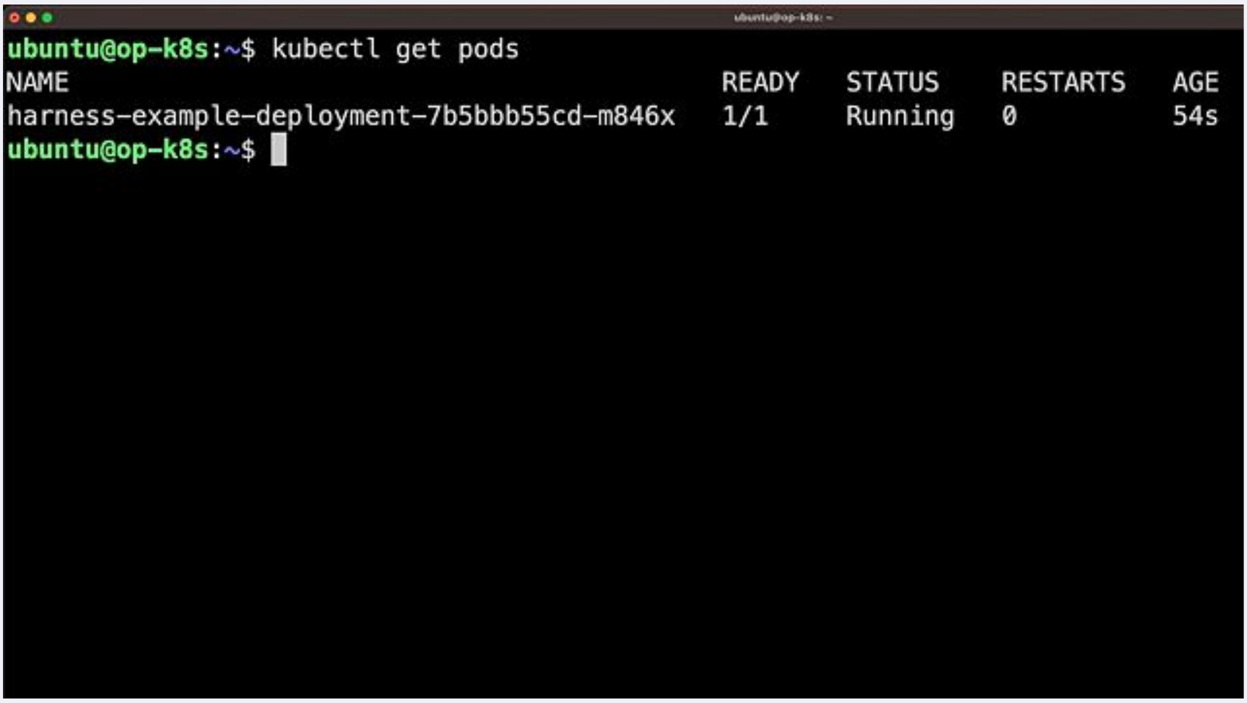
Started at: 28/02/2023, 11:23:26
Ended at: -
Duration: 5s
Timeout: 10m
Delegates: for K8s Task : Rolling Deploy (Delegate selection logs)

20 Console Log view



```
1  INFO  28/02/2023 11:23:25      Executing command ...
2  INFO  28/02/2023 11:23:25      Welcome to Harness CD
3  INFO  28/02/2023 11:23:25      Command completed with Ex
```

21 Run "kubectl get pods" to view the pods deployed by Harness CD



```
ubuntu@op-k8s:~$ kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
harness-example-deployment-7b5bbb55cd-m846x   1/1     Running   0          54s
ubuntu@op-k8s:~$ 
```

22 Congratulation on creating and running your first Deployment Pipeline!