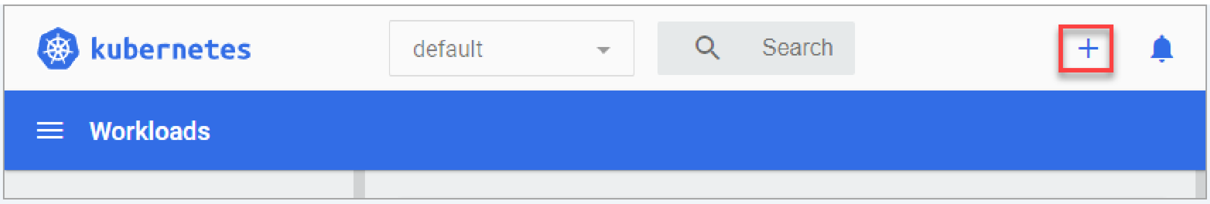
Creating a Deployment in the Minikube Dashboard

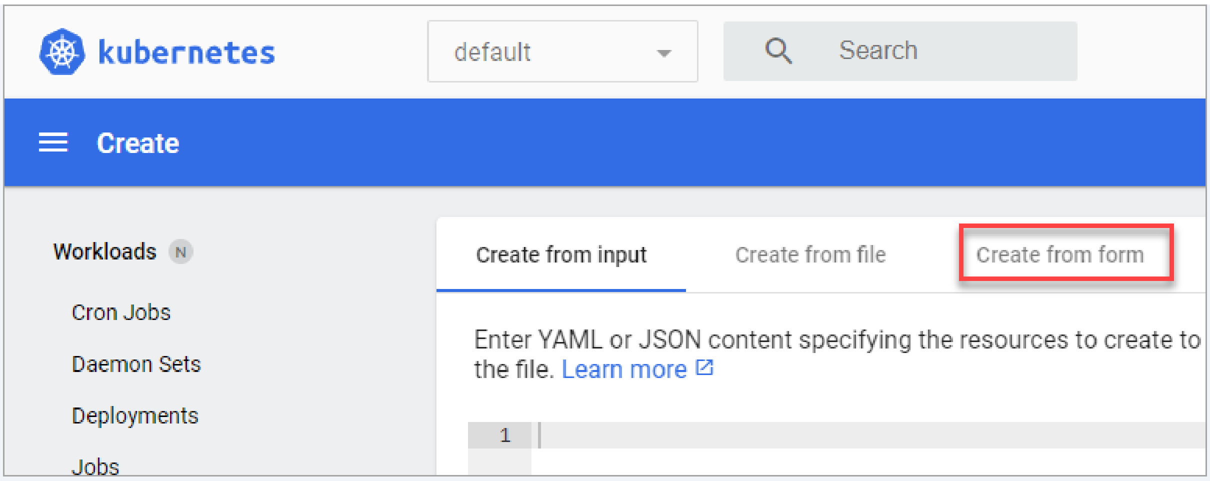
In addition to getting performance data on your cluster, you will, in this section, create and manage a web service deployment via the dashboard.

1. Click the **Create New Resource** button to create a new deployment, as shown below.



Creating a deployment in the Minikube dashboard

2. Click the **Create from form** tab to create the deployment by filling out a form instead of declaratively via [YAML]() or [JSON]() files.



Enter YAML or JSON content specifying the resources to create to the file. Learn more

```
1
```

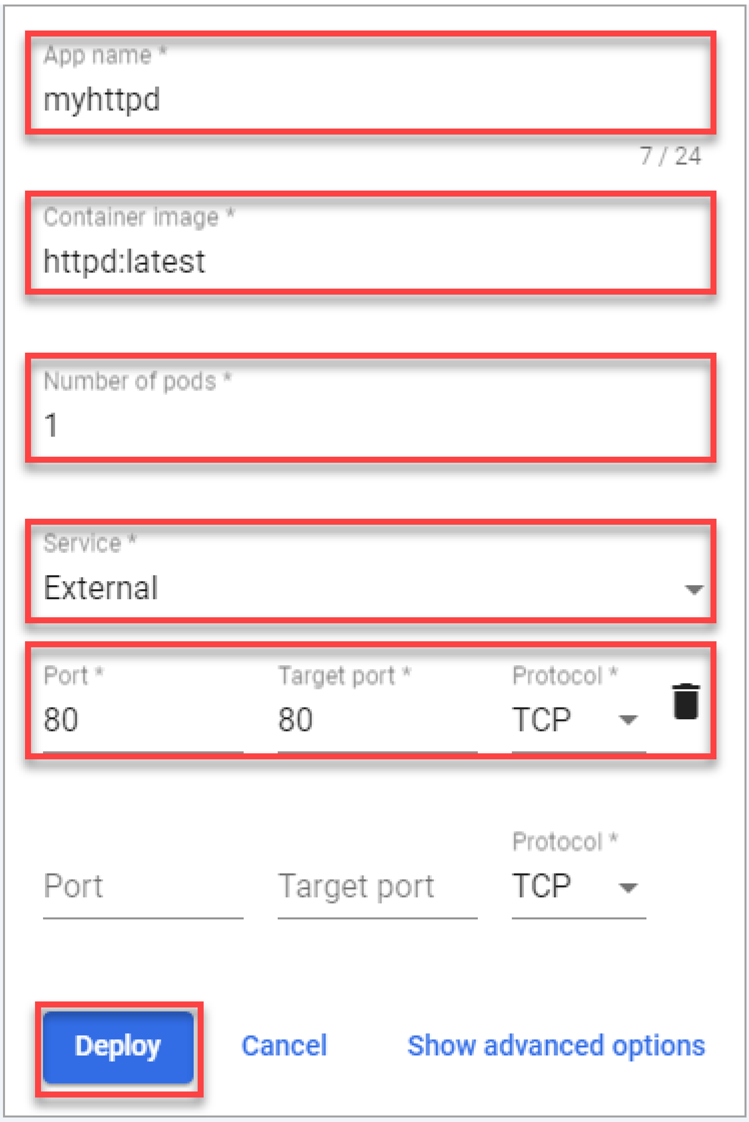
Clicking the Create from form tab

3. Type in the corresponding parameters for the deployment as shown below

- **App name**: The name of the deployment. This tutorial uses `myhttpd`.
- **Container Image**: Defines the base image for the deployment. For this tutorial, enter `httpd:latest`.

- **Number of pods:** Specifies how many pods to include in the deployment. Leave the value to `1` for now.
- **Service:** Select show to expose the deployment. Select `External` to expose the web service outside the cluster using a Loadbalancer service. Doing so allows accessing the service via the web browser.
- Set both the **Port** and **Target port**s values to `80`.

Your form's final entries screen should look like the shot below. Now, click **Deploy** to initiate the deployment.



App name *
myhttpd
7 / 24

Container image *
httpd:latest

Number of pods *
1

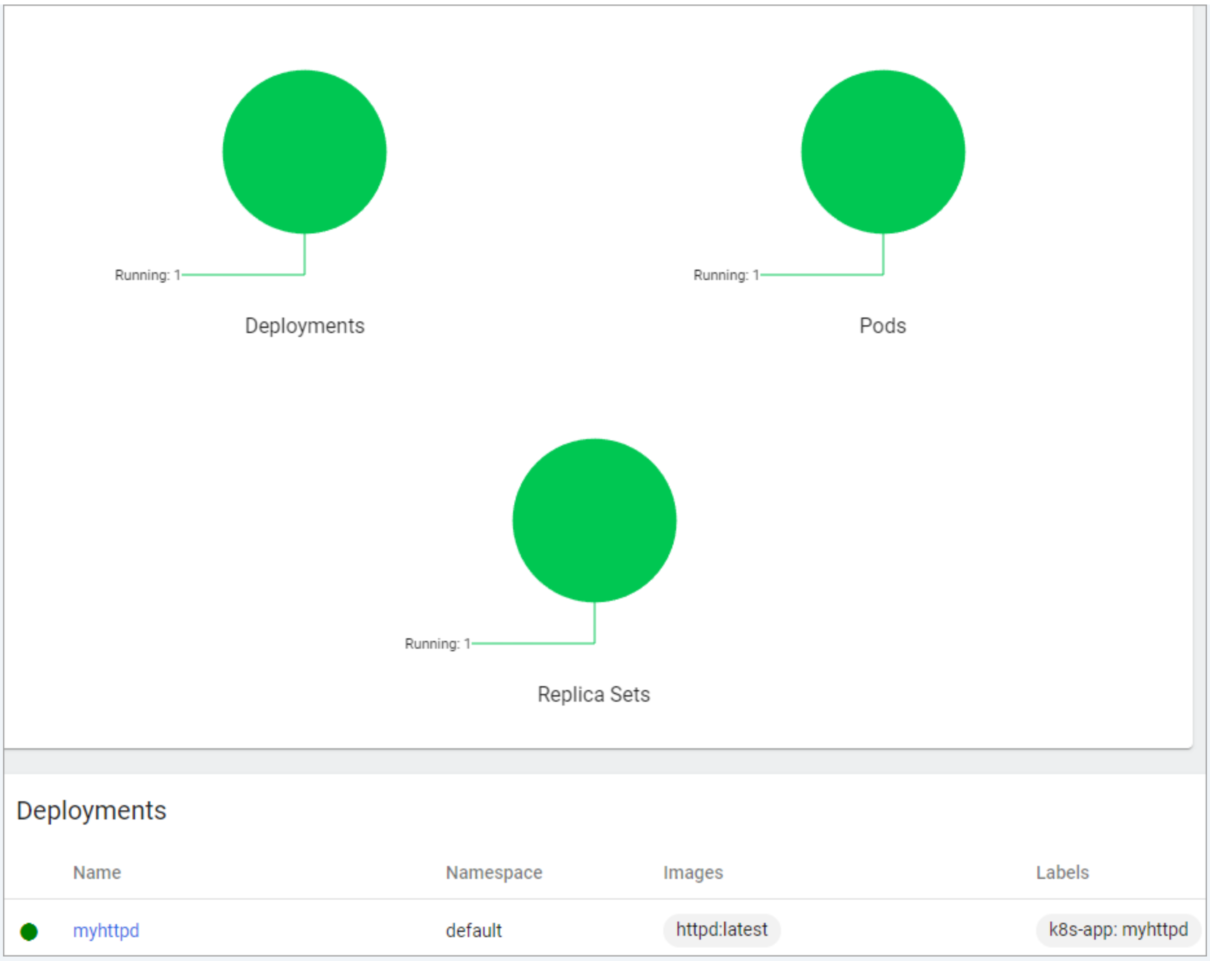
Service *
External

Port * 80 | Target port * 80 | Protocol * TCP

Port | Target port | Protocol * TCP

Creating a deployment in the Minikube Dashboard

Monitor the progress of the deployment on the Workloads dashboard. After a few minutes, the status should transition from an immediate amber to a lively green. Monitoring a successful deployment



Deployments

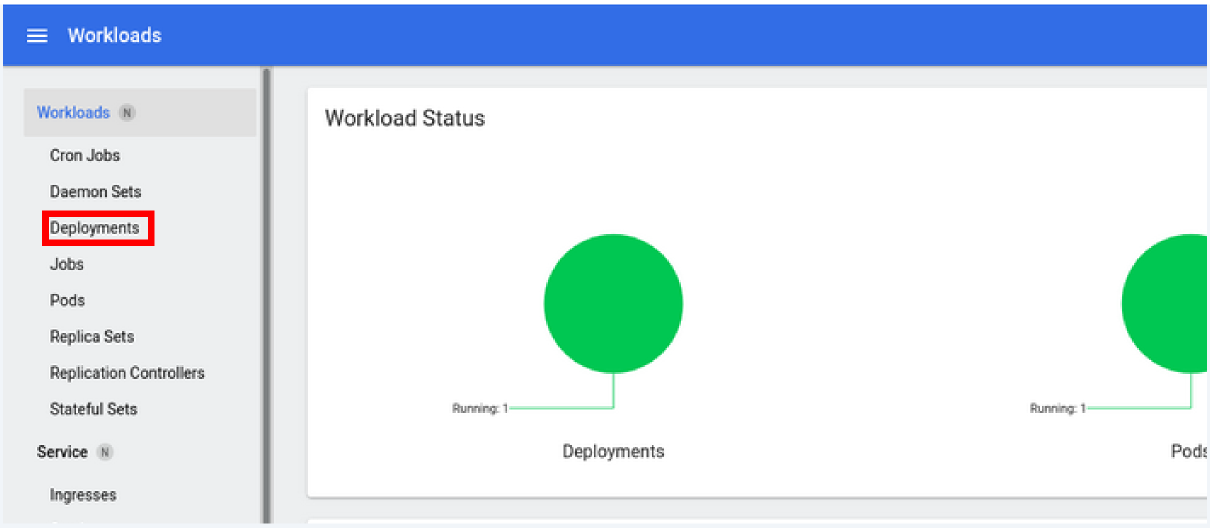
	Name	Namespace	Images	Labels
●	myhttpd	default	httpd:latest	k8s-app: myhttpd

Monitoring a successful deployment

Scaling and Managing Containers in the MiniKube Dashboard

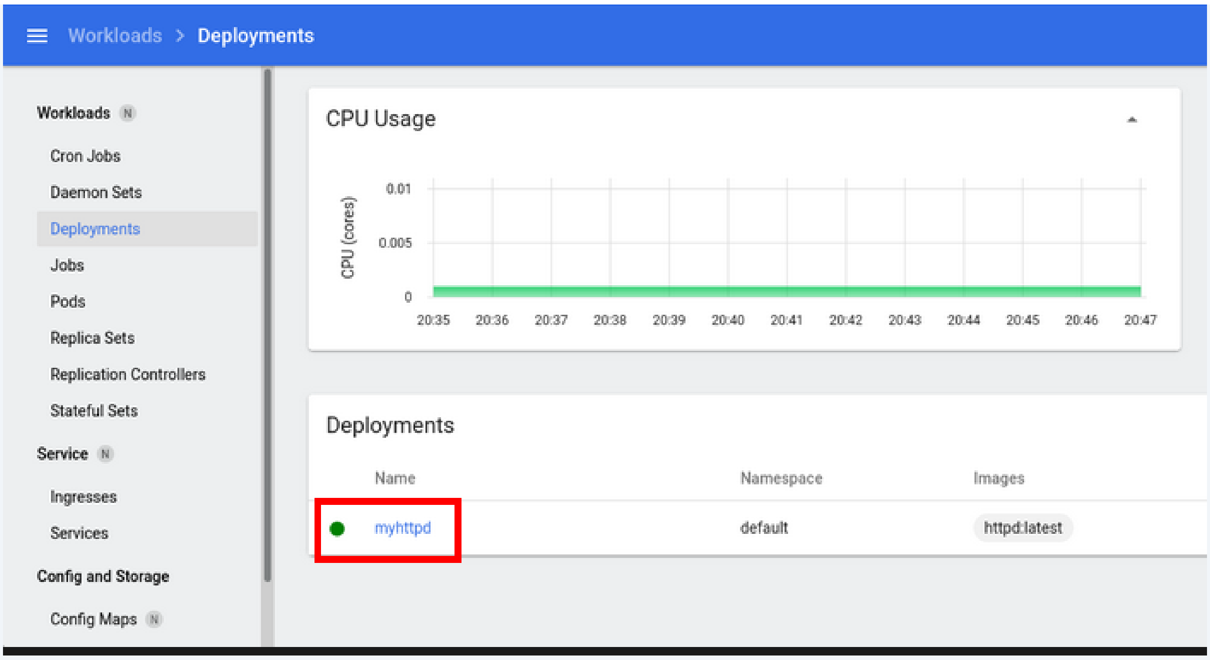
With the deployment now complete, it is time to perform some of the cluster management tasks you would have performed over the command line (if you didn’t have the Mnikube dashboard). First, scale the deployment out with the following steps.

1. Click **Deployments** on the navigation bar on the left-hand side of the browser window.



Clicking Deployments

2. Click **myhttpd** under the **Deployments** section, which is the deployment you set up earlier.

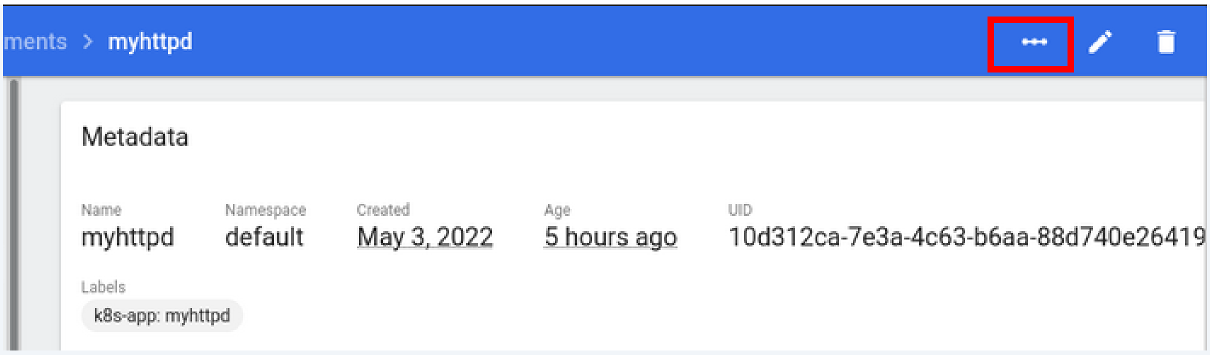


Deployments

	Name	Namespace	Images
●	myhttpd	default	httpd:latest

Scaling a deployment: Selecting the deployment

3. Click the **Scale Resource** icon near the top-right corner, as shown in the following screenshot.



Metadata

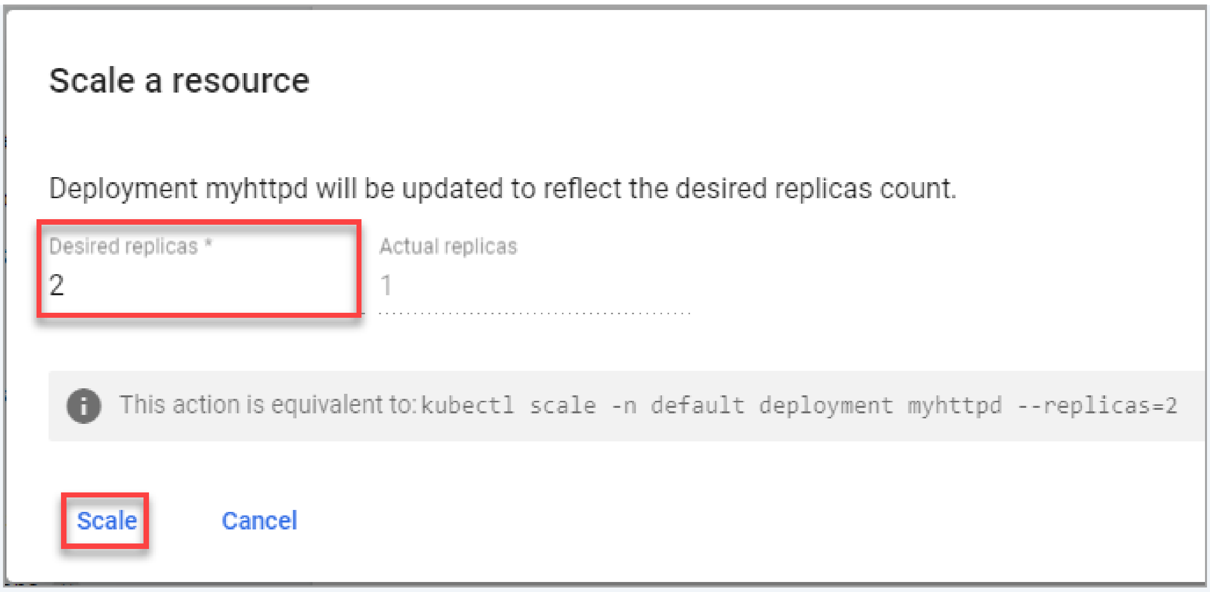
Name	Namespace	Created	Age	UID
myhttpd	default	May 3, 2022	5 hours ago	10d312ca-7e3a-4c63-b6aa-88d740e26419

Labels

k8s-app: myhttpd

Scaling a deployment: Selecting the Scale Resource button

4. Set the **Desired replicas** value to `2` to scale out the deployment. Doing so increases the replica count from 1 (current) to 2. Click **Scale** to effect the changes.



Scale a resource

Deployment myhttpd will be updated to reflect the desired replicas count.

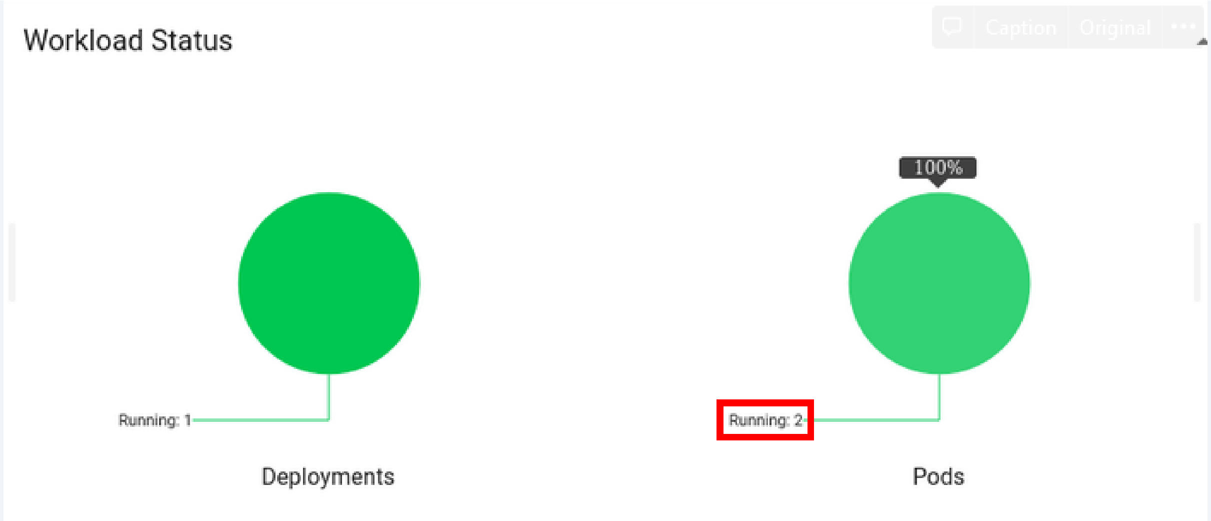
Desired replicas * 2

Actual replicas 1

This action is equivalent to: `kubectl scale -n default deployment myhttpd --replicas=2`

Increasing the replica count

5. Click the **Workload Status** link on the navigation pane and confirm the dashboard reflects the changes. You should see two running pods in the deployment.



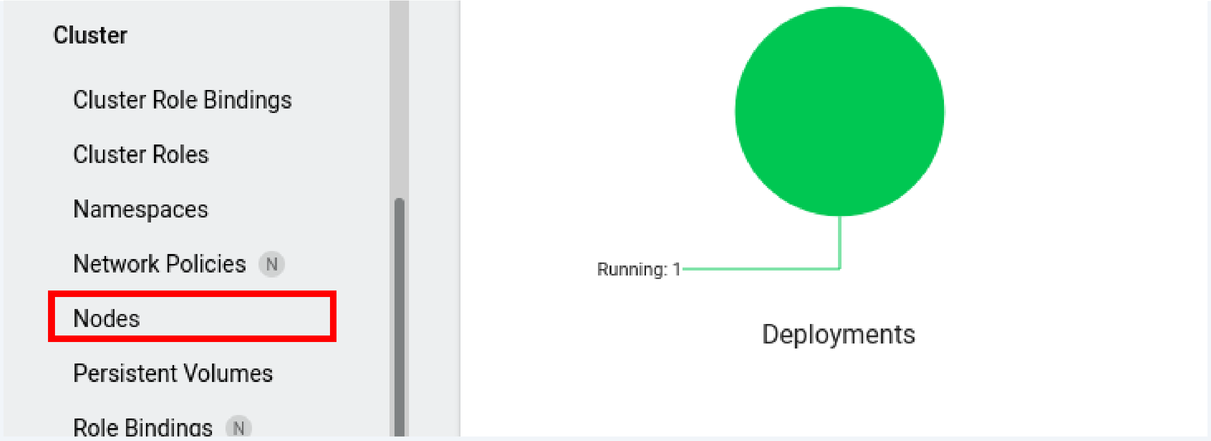
Confirming an increase in running pods

Obtaining the IP Address of the Node from the MiniKube Dashboard

The IP is one way of reaching the services within the cluster. Finding out a node’s IP address typically requires you to run commands. But now that you have the dashboard, you can conveniently view that information.

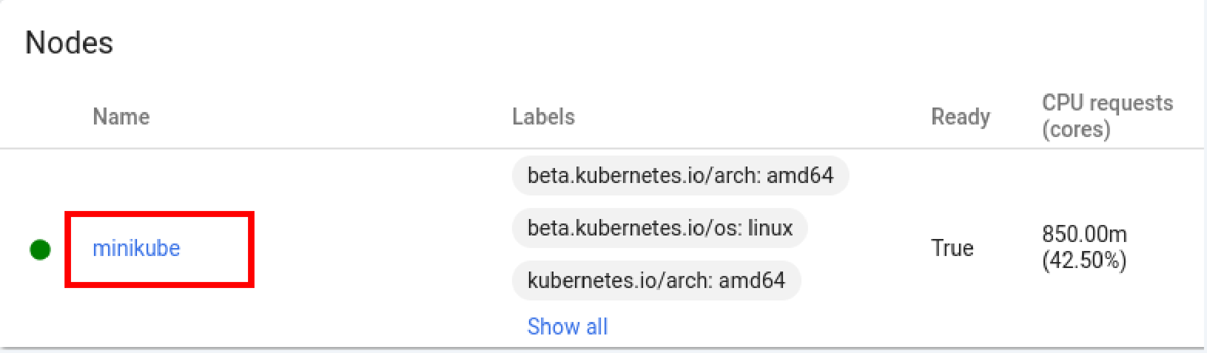
Follow the steps below to get the IP of your cluster.

1. Click **Nodes** from the navigation bar on the right of the dashboard as below.



Opening the Nodes page

2. Click the **minikube** link under the **Nodes** card.

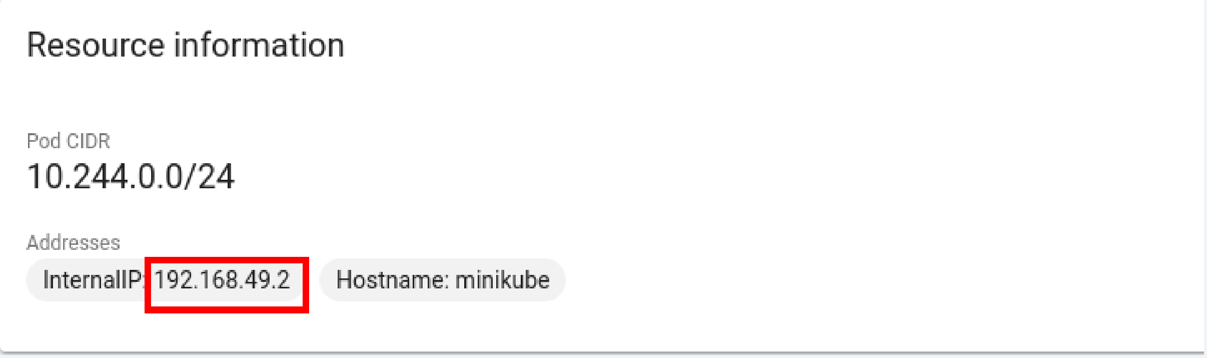


Nodes

Name	Labels	Ready	CPU requests (cores)
● minikube	beta.kubernetes.io/arch: amd64 beta.kubernetes.io/os: linux kubernetes.io/arch: amd64 Show all	True	850.00m (42.50%)

Selecting the Node

3. Scroll down to the **Resource information** card and find the node’s IP address.



Resource information

Pod CIDR
10.244.0.0/24

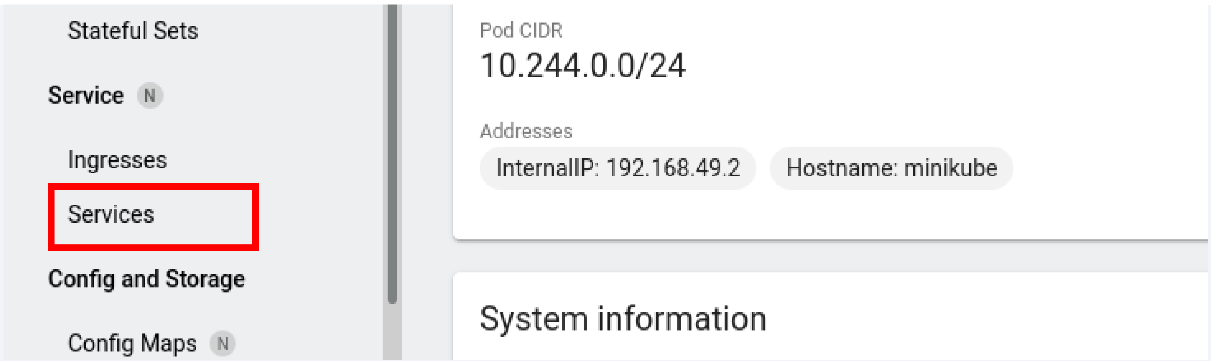
Addresses
InternalIP: 192.168.49.2 Hostname: minikube

Viewing the IP address of the node

Listing Existing Services

You attached a LoadBalancer service to the deployment by selecting External while creating the deployment. You can see the list of services for your deployment by following the steps below.

1. Click **Services** on the navigation bar.



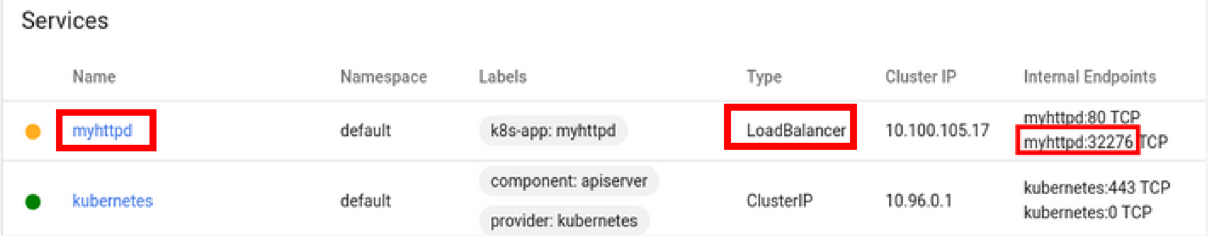
Pod CIDR
10.244.0.0/24

Addresses
InternalIP: 192.168.49.2 Hostname: minikube

System information

Navigating the Services page

2. You should see a list similar to the image below confirming the service type for **myhttpd** is **LoadBalancer.** Take note of the port number that maps the service externally. In this example, the port number is `32276`.

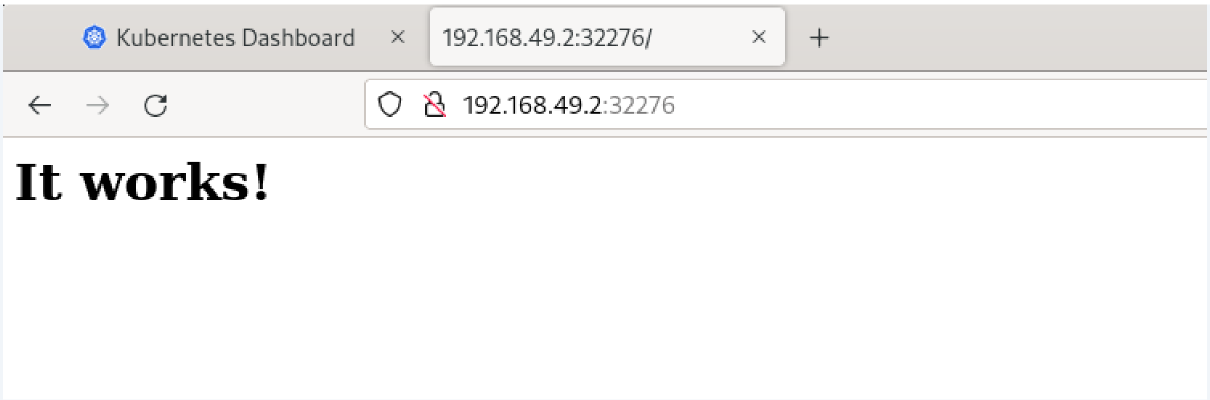


Services

	Name	Namespace	Labels	Type	Cluster IP	Internal Endpoints
●	myhttpd	default	k8s-app: myhttpd	LoadBalancer	10.100.105.17	myhttpd:80 TCP myhttpd:32276 TCP
●	kubernetes	default	component: apiserver provider: kubernetes	ClusterIP	10.96.0.1	kubernetes:443 TCP kubernetes:0 TCP

Viewing service details

Now, open a new browser window or tab and navigate to `HTTP://<node_internal_ip>:external_port`. In this example, the URL is `http://192.168.49.2:32276`.



It works!

Accessing a service externally